| | |
|---|---|
| Project acronym | SIMBAD |
| Project full title | Beyond Features: Similarity-Based Pattern Analysis and Recognition |
| Deliverable Responsible | Dipartimento di Informatica<br>Università Ca' Foscari di Venezia<br>Via Torino 155<br>30172 Venezia Mestre<br>Italy<br>http://www.dsi.unive.it/~pelillo |
| Project web site | http://simbad-fp7.eu |
| EC project officer | Teresa De Martino |
| Document title | Generalizations of game-theoretic notions |
| Deliverable | D5.2 |
| Document type | Report |
| Dissemination level | Public |
| Contractual date of delivery | M 42 |
| Project reference number | 213250 |
| Status & version | Definitive version |
| Work package, deliverable responsible | WP 5, UNIVE |
| Author(s) | Marcello Pelillo, Andrea Torsello, Samuel Rota Bulò, and Nicola Rebagliati |
| Additional contributor(s) | E. R. Hancock, A. Fred |

# 1. Introduction

The objective of workpackage WP5 is to develop novel, general learning models which do not require the (geo)metric assumption, thereby working directly on the original data. Game theory offers an attractive and unexplored perspective that serves well our purpose.

In task WP5.2 we aimed at generalizing the game-theoretic framework developed in WP5.1 in several directions:

- **High-order relations:** extensions to the game-theoretic approach have been studied, allowing k-way interactions among players, which is equivalent to using high-order similarity relations, or payoff functions that are non linear in the population distribution allowing us to deal with context-dependent similarities.
- **Multi-population formulations:** generalizations involving several distinct populations of players. These "multi-population" games allow to learn many related classification functions simultaneously.

Along these lines of research, we further developed some general topics related to the game theoretic approach:

- **Applications:** we studied the applicability of the approach to specific problems
- **Future Work**.

In the following we will report the main results from each line of investigation

# 2. High-order relations

The game-theoretic framework naturally generalizes to allow k-way interactions among players, which is equivalent to using high-order similarity relations (hypergraphs). The main extension in this direction was already anticipated in D5.1 and derived from the work published in (Rota Bulò and Pelillo, 2009). Since that report there has been some further work in the theoretical characterization of the hypergraph clusters (Rota Bulò and Pelillo, submitted).

Let $\mathbf{H}=(\mathbf{V}, s)$ be a $k$-graph modeling a hypergraph clustering problem, where $V=\{1,...,n\}$ is the set of objects to cluster and $s(\{i_1,...,i_k\})$ is the similarity function providing the similarity among $k$ objects $i_1,...,i_k$. We can build a game involving $k$ players, each of them having the same set of (pure) strategies, namely the set of objects to cluster $\mathbf{V}$. Under this setting, a population $\boldsymbol{x} \in \Delta$ of agents playing a clustering game is to all intents and purposes a representation of a cluster, where $x_i$ is the probability for object $i$ to be part of it. Indeed, any cluster can be modelled as a probability distribution over the set of objects to cluster.

The payoff function of the clustering game is defined in a way as to favour the evolution of agents supporting highly coherent objects. Intuitively, this is accomplished by rewarding the $k$ players in proportion to the similarity that the $k$ played objects have.

Hence, assuming $(v_1, ..., v_k) \in \mathbf{V}^k$ to be the tuple of objects selected by $k$ players, the payoff function can be simply defined as

$$\pi(v_1, \ldots, v_k) = \begin{cases} \frac{1}{k!} s(\{v_1, \ldots, v_k\}) & \text{if } \{v_1, \ldots, v_k\} \in \binom{V}{k} \\ 0 & \text{else}, \end{cases}$$

Within this context, the notion of a cluster turns out to be equivalent to a classical equilibrium concept from (evolutionary) game theory, namely Evolutionary Stable Strategies, as the latter reflects both the internal and external cluster conditions of a cluster, i.e., internal coherency condition, which asks that the objects belonging to the cluster have high mutual similarities, and an external incoherency condition, which states that the overall cluster internal coherency decreases by adding to it any external object.

Let $\sigma(x)$ denote the support of a population $x \in \Delta$, that is the set of indices $i \in V$ such that $x_i > 0$. Consider the following function $u : \Delta^k \to \mathbb{R}$ which will be useful in the sequel:

$$u\left(\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(k)}\right) = \sum_{(s_1, \ldots, s_k) \in S^k} \pi(s_1, \ldots, s_k) \prod_{i=1}^{k} y_{s_i}^{(i)}$$

which is invariant under any permutation of its arguments due to the super-symmetry of the payoff function $\pi$. Also, we will use the notations $x^{[k]}$ as a shortcut for a sequence $(x, \ldots, x)$ of k identical states $x$, and $e^j$ to indicate the n-vector with $x_j = 1$ and zero elsewhere. Now, it is easy to see that the expected payoff earned by a j-strategist (an agent playing strategy $j \in S$) in a population $x \in \Delta$ is given by $u(e^j, x^{[k-1]})$, while the expected payoff over the entire population is given by $x^{[k]}$.

A population state $x \in \Delta$ is a *Nash equilibrium* of the clustering game $\Gamma = (P, S, \pi)$ if

$$u\left(\mathbf{e}^j, \mathbf{x}^{[k-1]}\right) \leq u\left(\mathbf{x}^{[k]}\right), \qquad \text{for all } j \in S.$$

An ESS is a refinement of the notion of Nash equilibrium which is stable to some extent under evolutionary pressure. Formally, assume assume that in a population $x \in \Delta$, a small share $\epsilon$ of mutant agents appears, whose distribution of strategies is $y \in \Delta$. The resulting post-entry population is then given by $w_\epsilon = (1 - \epsilon)x + \epsilon y$. Biological intuition suggests that evolutionary forces select against mutant individuals if and only if the expected payoff of a mutant agent in the postentry population is lower than that of an individual from the original population, i.e.,

$$u\left(\mathbf{y}, \mathbf{w}_\epsilon^{[k-1]}\right) < u\left(\mathbf{x}, \mathbf{w}_\epsilon^{[k-1]}\right).$$

Hence, a population $x \in \Delta$ is said to be *evolutionary stable* if the previous inequality holds for any distribution of mutant agents $y \in \Delta \setminus \{x\}$, granted the population share of mutants $\epsilon$ is sufficiently small.

Within our framework , a cluster of a hypergraph clustering problem instance is, by definition, an ESS of the corresponding clustering game.

***Definition*** *(ESS-cluster): Given an instance of a hypergraph clustering problem* $H=(V,E,\omega)$ *, an ESS-cluster of* $H$ *is an ESS of the corresponding hypergraph clustering game.*

Note that given an ESS-cluster $x\in\Delta$ the expected population payoff $u(x^{[k]})$ can be regarded as a measure of the cluster's internal coherency in terms of the average similarity of the objects forming the cluster, whereas the expected payoff $u(e^j, x^{[k-1]})$ of a player selecting object $j\in V$ in $x$ measures the average similarity of object $j$ with respect to the cluster.

The internal coherency of an ESS-cluster is a direct consequence of the Nash condition, which is satisfied by any ESS. Indeed, if $x\in\Delta$ is an ESS of a clustering game, then from the Nash condition it follows that every object belonging to the cluster, i.e., every object in $\sigma(x)$ , has the same average similarity with respect to the cluster, which in turn corresponds to the cluster's overall average similarity. This is formally stated in the following theorem.

***Theorem****: Let* $H=(V,E,\omega)$ *be an instance of a hypergraph clustering problem, and* $\Gamma=(P,V,\pi)$ *the corresponding clustering game. If* $x\in\Delta$ *is an ESS-cluster of* $H$ *, with support* $\sigma(x)=C$ *, then*

$$u\left(\mathbf{e}^j,\mathbf{x}^{[k-1]}\right)=u\left(\mathbf{x}^{[k]}\right), \qquad \textit{for all } j\in C.$$

We provide also in the following theorem a characterization of the ESS-equilibria in terms of two-covers, i.e. subgraphs such that each pair of vertices belongs to at least one (hyper-)edge.

**Theorem**: Let $H=(V,E,\omega)$ be an instance of a hypergraph clustering problem, and $\Gamma=(P,V,\pi)$ the corresponding clustering game. If $x\in\Delta$ is an ESS-cluster of $H$ , then its support $\sigma(x)$ is a two-cover of $H$ .

Intuitively, the previous result shows that two objects cannot belong to (the support of) an ESS-cluster if there is no similarity relationship between them within the cluster. This is a minimal property that a cluster should satisfy in order to guarantee some form of internal coherency.

As for the external incoherency, we show in the next theorem that any deviation from the equilibrium leads to a drop in the cluster internal similarity.

***Theorem****: Let* $H=(V,E,\omega)$ *be an instance of a hypergraph clustering problem, and* $\Gamma=(P,V,\pi)$ *the corresponding clustering game. Then,* $x\in\Delta$ *is an ESS-cluster of* $H$ *if and only if for any* $y\in\Delta\setminus\{x\}$ *and all sufficiently small positive values of* $\epsilon$ *the following inequality holds:*

$$u\left(\mathbf{w}_\epsilon^{[k]}\right)<u\left(\mathbf{x}^{[k]}\right),$$

*where* $w_\epsilon=(1-\epsilon)x+\epsilon y$ .

We additionally show that there exists a correspondence between the ESS-clusters and the local solutions of a polynomial, linearly-constrained, optimization problem.

***Theorem**: Let* $H=(V,E,\omega)$ *be a hypergraph clustering problem,* $\Gamma=(P,V,\pi)$ *the corresponding clustering game, and* $f(x)$ *a function defined as*

$$f(\mathbf{x}) = u\left(\mathbf{x}^{[k]}\right) = \sum_{e\in E}\omega(e)\prod_{j\in e}x_j\,.$$

*Nash equilibria of* $\Gamma$ *are in one-to-one correspondence with the critical points of* $f(x)$ *over* $\Delta$ *, while ESS's of* $\Gamma$ *are in one-to-one correspondence with strict local maximizers of* $f(x)$ *over* $\Delta$ *.*

We also provide the following dynamics as an algorithm for finding ESS-clusters, which derives straightforwardly from the Baum-Eagon inequality (L. E. Baum and J. A. Eagon, 1967), which can be regarded to as a generalization of the replicator dynamics to multi-population games :

$$x_j(t+1) = x_j(t)\frac{u\left(\mathbf{e}^j,\mathbf{x}(t)^{[k-1]}\right)}{u\left(\mathbf{x}(t)^{[k]}\right)}, \qquad j=1\ldots n\,.$$

### 2.1 Hypergraph matching

In (Ren, Wilson, Hancock, 2011) an approach to hypergraph matching is proposed, which is based on the factorization of the compatibility tensor, derived from two hypergraphs to match, in terms of a matrix defined in probability domain. This factorization is accomplished by using the game-theoretic framework described in the previous section.

Let $H=(V,E,\omega)$ be a k-graph and let $H$ denote also the adjacency tensor of hypergraph $H$ . The adjacency tensor is indexed by k-tuples of vertices in $V$ and provides for each tuple $(i_{1,\ldots},i_k)$ corresponding to an edge $\{i_{1,\ldots},i_k\}\in E$ of the hypergraph the similarity $\omega(\{i_{1,..}..,i_k\})$ and 0 otherwise.

Consider two k-graphs $H=(V,E,\omega)$ and $H'=(V',E',\omega')$ to be matched and construct a *compatibility tensor* as follows

$$C_{i_1 i'_1,\cdots,i_K i'_K} = \begin{cases} 0 & \text{if } H_{i_1,\cdots,i_K}=0 \ \text{ or } H'_{i'_1,\cdots,i'_K}=0; \\ s(H_{i_1,\cdots,i_K},H'_{i'_1,\cdots,i'_K}) & \text{otherwise;} \end{cases}$$

where $s(\cdot,\cdot)$ is a function measuring hyperedge similarity. This function can be defined using a Gaussian kernel $s(H_{i_{1,\ldots,i_k}},H'_{i'_{1,..},i'_k})=\exp(-\|H_{i_{1,\ldots,i_k}}-H'_{i'_{1,..},i'_k}\|_2^2/\sigma_1)$ where $\sigma_1$ is a scaling parameter. The hyperedge pair $\{i_{1,\ldots},i_k\}$ and $\{i'_{1,\ldots},i'_k\}$ with a large similarity measure has a large probability $Pr(\{i_{1,\ldots},i_k\}\Leftrightarrow\{i'_{1,\ldots},i'_k\}|H,H')$ for matching. Here $\Leftrightarrow$ denotes a possible matching between the two corresponding hyperedges or a pair of vertices. Under the conditional independence assumption of the matching process, the hyperedge matching probability can be factorized over the associated vertices of the hypergraphs as

$$Pr(\{i_{1,\dots},i_k\}\Leftrightarrow\{i'_{1,\dots},i'_k\}|H,H')=\prod_{n=1}^{k}Pr(i_n\Leftrightarrow i'_n|H,H')$$

where $Pr(i_n\Leftrightarrow i'_n|H,H')$ denotes the probability for the possible matching $i_n\Leftrightarrow i'_n$ to be correct.

Let $P$ be a marix in probability domain, where $P_{ii'}=Pr(i\Leftrightarrow i'|H,H')$ . High order matching problems can be formulated as locating the matching probability that most closely accords with the elements of the compatibility tensor, i.e., seeking the optimal $P$ by maximizing the objective function

$$\begin{aligned}f(\mathbf{P})&=\sum_{i_1=1}^{N}\sum_{i'_1=1}^{N'}\cdots\sum_{i_K=1}^{N}\sum_{i'_K=1}^{N'}C_{i_1i'_1,\cdots,i_Ki'_K}\Pr(\{i_1,\cdots,i_K\}\leftrightarrow\{i'_1,\cdots,i'_K\}|HG,HG')\\&=\sum_{i_1=1}^{N}\sum_{i'_1=1}^{N'}\cdots\sum_{i_K=1}^{N}\sum_{i'_K=1}^{N'}C_{i_1i'_1,\cdots,i_Ki'_K}\prod_{n=1}^{K}P_{i_ni'_n}\end{aligned}$$

subject to $P_{ij}\geq 0$ and $\sum_{i,j}P_{ij}=1$ .

By applying the Baum-Eagon inequality, we obtain the following iterative scheme,:

$$p_\alpha^{\text{new}}=\frac{p_\alpha\sum_{\alpha_2=1}^{N_\times}\cdots\sum_{\alpha_K=1}^{N_\times}C_{\alpha,\alpha_2,\cdots,\alpha_K}\prod_{n=2}^{K}p_{\alpha_n}}{\sum_{\beta=1}^{N_\times}p_\beta\sum_{\beta_2=1}^{N_\times}\cdots\sum_{\beta_K=1}^{N_\times}C_{\beta,\beta_2,\cdots,\beta_K}\prod_{n=2}^{K}p_{\beta_n}}$$

which allows to locally optimize $f(P)$ over the set of matrices $P$ defined over the probability domain. Finally, the matching vertices can be determined from matrix $P$ .

### 2.2. Feature Selection

In (Z. Zhang and E. R. Hancock, 2011a; Z. Zhang and E. R. Hancock, 2011b; Z. Zhang and E. R. Hancock, 2011c) a game-theoretic approach for feature selection is proposed, which is based on the framework presented in (Pavan and Pelillo, 2007). The idea is to cluster features based on the mutual information of the feature vectors and select the features that best describe each cluster. With this view, the process is threefold: First, a relevance matrix must be computed from the feature vectors, then clusters are extracted using the dominant sets framework, and finally the features are selected based on multidimensional interaction information $I(F;C)$ between features $F=\{f_1,\dots,f_m\}$ and class $C$:

$$I(F;C)=I(f_1,\dots,f_m;C)=\sum_{f_1,\dots,f_m}\sum_{c\in C}P(f_1,\dots,f_m;c)\ \times\log\frac{P(f_1,\dots,f_m;c)}{P(f_1,\dots,f_m)P(c)}$$

In (Z. Zhang and E. R. Hancock, 2011a; Z. Zhang and E. R. Hancock, 2011c) the relevance measure based on the (pairwise) mutual information $I(F_{k1},F_{k2})$ of feature vectors $F_{k1}$, $F_{k2}$: and their Entropies $H(F_{k1})$, $H(F_{k2})$:

$$\mathbf{W}(F_{k1},F_{k2})=\frac{2I(F_{k1},F_{k2})}{H(F_{k1})+H(F_{k2})}$$

In (Z. Zhang and E. R. Hancock, 2011b) higher order relations are used. Here, the relevance between feature vectors $F_{k1}$, $F_{k2}$, $F_{k3}$ is

$$\mathbf{S}(F_{k1}, F_{k2}, F_{k3}) = \frac{3I(F_{k1}, F_{k2}, F_{k3})}{H(F_{k1}) + H(F_{k2}) + H(F_{k3})}$$

where $I(X,Y,Z)$ is the interaction information, defined in terms of the conditional mutual information.

$$I(X;Y;Z) = I(X;Y|Z) - I(X;Y)$$

With this high order relation to hand, the game-theoretic hyper-clustering framework is used to extract the clusters and then the multidimensional interaction information framework is used to extract the relevant features for each cluster.

These approaches provide competitive results on both standard datasets from the machine learning archive, and on the problem of gender determination.

## 3. Multi-population formulations

A further generalization involves the existence of distinct populations of players, each characterized by its own strategies and payoffs. In this context the search for equilibria is over the multi-simplex

$$\Delta^{m} = \{x \in \mathbf{R}^{nm} \mid \quad x_i^j \geq 0 \quad \text{for all } i=1,...,n,\ j=1,...,m, \quad \sum_{i=1}^{n} x_i^j = 1 \quad \text{for all } j=1,...,m\},$$

allowing us to jointly estimate several distributions. To this end, we start from a generalization of the link between Evolutionary Stable Strategies and the local maxima of problem (1), where in the general context of super-symmetric multi-population games, equilibria correspond to local maxima of a multi-simplex constrained polynomial optimization problem. This generalization has been at the basis of several developments where the dynamics derived from the Baum-Eagon inequality have been applied to several different Machine Learning problems.

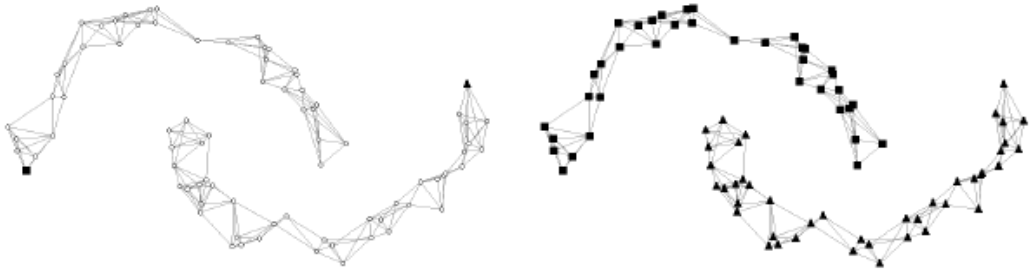
### 3.1. Graph transduction

Multi-population approaches have been used also for semi-supervised learning. In (A. Erdem and M. Pelillo, 2011; A. Erdem and M. Pelillo, to appear) a multi-population game-theoretic approach for graph transduction was developed. Graph transduction is a popular class of semi-supervised learning techniques which aims to estimate a classification function defined over a graph of labeled and unlabeled data points. The general idea is to propagate the provided label information to unlabeled nodes in a consistent way. In contrast to the traditional view, in which the process of label propagation is defined as a graph Laplacian regularization, within the proposed game-theoretic framework, the t consistent labelings of the data correspond to equilibria of the game. An attractive feature of this formulation is that it is inherently a multi-class approach and imposes no constraint whatsoever on the structure of

the pairwise similarity matrix, being able to naturally deal with asymmetric and negative similarities alike.

### *3.1.1 Transductive learning on unweighted undirected graphs*

The theoretical motivation for the proposed approach stems from the analysis of the simplest case of graph transduction where the graph expressing the similarity relationships among the data points is an unweighted undirected graph. To give an example, the graph can be seen as a k-nearest neighbor ($k$-NN) graph with 0/1 weights over points in which the presence of an edge simply denotes the perfect similarity between a pair of two data points, otherwise the points are completely dissimilar. To illustrate this toy problem, consider the graph shown the following Figure in which edges are unweighted.



Initial label assignment Final label assignment

The classification task is to estimate the labels of the unlabeled points based exclusively on the information available at the two labeled points, each of which is marked with a different label. Recall the cluster assumption of semi-supervised learning that neighboring objects and objects in the same cluster (or on the same manifold structure) tend to belong to the same class. Clearly, in the unweighted graph setting, the cluster assumption is also valid and can be expressed as the hypothesis that every node in a connected component of a binary similarity graph has the same class label as each connected component describes a manifold.

Following this observation, we can formulate this toy version of graph transduction as a (binary) constraint satisfaction problem (CSP) (Tsang, 1993;Marriott and Stuckey, 1998). CSPs are widely used to solve combinatorial problems in a variety of application domains, such as artificial intelligence and computer vision. In the computer vision literature, the problem is often known as the consistent labeling problem (Waltz, 1975; Haralick and Shapiro, 1979).

A binary CSP is defined by a set of variables representing the elements of the problem being modeled and a set of binary constraints representing the relationships among variables. A solution of the problem is simply an assignment of values to the variables which satisfies all the constraints. If there is no such assignment, then the problem is unsatisfiable. When each

variable can take a value from a finite domain, a binary CSP can be described in a formal manner as a triple $(\boldsymbol{V},\boldsymbol{D},\boldsymbol{R})$, where $\boldsymbol{V} = \{v_1,..., v_n\}$ is a set of variables, $\mathbf{D} = \{\boldsymbol{D}v_1,...,\boldsymbol{D}v_n\}$ is a set of domains of the variables, each $\boldsymbol{D}v_i$ denoting a finite set of possible values for variable $v_i$, and $\boldsymbol{R} = \{\boldsymbol{R}_{ij} \mid \boldsymbol{R}ij \subseteq \boldsymbol{D}v_i \times \boldsymbol{D}v_j\}$ is a set of binary constraints, each $\boldsymbol{R}_{ij}$ describing compatible pairs of values for the variables $v_i$ and $v_j$. If the cardinality of the domains of variables are $p$ and $q$, respectively, then $\boldsymbol{R}_{ij}$ can be expressed by a 0/1 matrix of size $p \times q$, where $\boldsymbol{R}_{ij}(\lambda, \lambda') = 1$ if the assignment $v_i = \lambda$ is compatible with the assignment $v_j = \lambda'$. For a general CSP on a finite domain, the problem of finding a solution is known to be NP-complete (Haralick et al., 1978). The simplest way to obtain an assignment satisfying all the given constraints or to report non-existence of such a solution is to perform backtracking. However, it is time consuming, so in practice, either a constraint propagation technique or a local search method is used to solve the problem (Tsang, 1993; Marriott and Stuckey, 1998).

Returning back to the motivating problem of transductive learning on an unweighted undirected graph, suppose that we are given a data set $\boldsymbol{D} = \{\boldsymbol{D}_\ell,\boldsymbol{D}_u\}$ consisting of labeled points $\boldsymbol{D}_\ell = \{d_1, ..., d_\ell\}$ and unlabeled points $\boldsymbol{D}_u = \{d_{\ell+1}, ... , d_n\}$ and a set of labels $\Phi = \{1,... , c\}$ such that the labels provided for the first $\boldsymbol{\ell}$ labeled points are given by $\{\varphi_1,... , \varphi_\ell\} \in \Phi$. The task of transductive learning is to estimate the unknown labels $\{\varphi_{\ell+1}, ..., \varphi_n\}$ of unlabeled points $\{d_{\ell+1}, ..., d_n\}$. Now further suppose that the relationships among the data points are given by an unweighted undirected graph $\boldsymbol{G} = (\boldsymbol{D}, \boldsymbol{E})$, where $\boldsymbol{D}$ is the set of nodes and $\mathcal{E}$ is the set of edges such that an edge $e_{ij} \in \mathcal{E}$ shows that points $i$ and $j$ are perfectly similar to each other. Let $\boldsymbol{A} = (a_{ij})$ denote the 0/1 adjacency matrix of $\boldsymbol{G}$. Reflecting the constraints imposed by the cluster assumption of SSL, the problem of graph transduction on an unweighted graph can be formalized as a binary CSP as follows:

- The set of variables: $\boldsymbol{V} = \{v_1, ..., v_n\}$
- Domains: $D_{v_i} = \begin{cases} \{\phi_i\}, & \text{for all } 1 \le i \le \ell \\ \Phi & \text{for all } \ell + 1 \le i \le n \end{cases}$
- Binary constraints: $\forall$ij: if aij = 1, then vi = vj.

Each assignment of values to the variables satisfying all the constraints is a solution of the CSP, and provides a consistent labeling for the unlabeled points.

Classical CSPs such as the one given in this section assume crisp constraints, in the sense that constraints are either completely satisfied or completely violated. However, for many real-world applications, such a formulation is too restrictive to be practical. A classical generalization to deal with soft constraints is described in (Hummel and Zucker, 1983), in which each constraint is assigned a weight representing a level of confidence. Later, it was shown that the notion of consistency proposed in (Hummel and Zucker, 1983) is related to the Nash equilibrium concept in non-cooperative game theory (Miller and Zucker, 1991). In this

study, we build on this connection to devise a graph transduction game which serves as a generalization of the binary CSP for the motivating problem.

### *3.1.2 The graph transduction game*

Let the geometry of the data be modeled with a weighted graph $\boldsymbol{G} = (\boldsymbol{D}, \boldsymbol{E}, w)$ in which $\boldsymbol{D}$ is the set of nodes representing both labeled and unlabeled points, and $w : \boldsymbol{E} \rightarrow \mathbb{R}$ is a weight function assigning a similarity value to each edge $e \in \boldsymbol{E}$. Representing the graph with its weighted adjacency matrix $\boldsymbol{W} = (w_{ij})$, the partial payoff matrix between two players $i$ and $j$ is set as $\boldsymbol{A}_{ij} = w_{ij} \times \boldsymbol{I}_c$, where $\boldsymbol{I}_c$ is the identity matrix of size $c$. Note that when partial payoff matrices are represented in block form as $\boldsymbol{A} = (\boldsymbol{A}_{ij})$, the matrix $\boldsymbol{A}$ is given by the Kronecker product $\boldsymbol{A} = \boldsymbol{I}_c \otimes \boldsymbol{W}$.

The transduction is defined in terms of the following graph transduction game. Assume each player $i \in \boldsymbol{I}$ participating in the game corresponds to a particular point in a data set $\boldsymbol{D} = \{d_1, \ldots, d_n\}$ and can choose a strategy among the set of strategies $\boldsymbol{S}_i = \{1, \ldots, c\}$, each expressing a certain hypothesis about its membership to a class and $c$ being the total number of classes. Hence, the mixed strategy profile of each player $i \in \boldsymbol{I}$ lies in the $c$-dimensional simplex $\Delta_i$. By problem definition, the players of the game can be categorized into two disjoint groups: those which already have knowledge of their membership, referred to as labeled players and denoted with the symbol $\mathbf{I}_\ell$, and those which do not have any idea about this at the beginning of the game, which are hence called unlabeled players and correspondingly denoted with $\boldsymbol{I}_u$. In this approach we assume that the proposed graph transduction game is an instance of a special subclass of multi-player games, known as polymatrix games (Janovskaya, 1968; Howson, 1972), in which players are nodes of a graph and every edge denote a two-player game between corresponding pair of players. In other words, we suppose that only pairwise interactions are allowed in the game and the payoffs associated to each player are additively separable so that the payoff of each player is given by the sum of the payoffs gained from each game played with one of its neighbor. Formally speaking, for a pure strategy profile $s = (s_1, \ldots, s_n) \in \boldsymbol{S}$, the payoff function of every player $i \in \boldsymbol{I}$ is in the form:

$$\pi_i(s) = \sum_{j=1}^{n} A_{ij}(s_i, s_j)$$

In an instance of the transduction game, since each labeled player is restricted to play a definite strategy of its own, all of these fixed choices can be reflected directly in the payoff function of a unlabeled player $i \in \boldsymbol{I}_u$ as follows:

$$u_i(e_i^h) = \sum_{j \in \mathcal{I}_{\mathcal{U}}} (A_{ij} x_j)_h + \sum_{k=1}^{c} \sum_{j \in \mathcal{I}_{\mathcal{D}|k}} A_{ij}(h, k)$$

$$u_i(x) = \sum_{j \in \mathcal{I}_{\mathcal{U}}} x_i^T A_{ij} x_j + \sum_{k=1}^{c} \sum_{j \in \mathcal{I}_{\mathcal{D}|k}} x_i^T (A_{ij})_k$$

The multi-population replicator dynamics are then used to find an equilibrium.

Empirically, we observed that specifying payoffs in terms of normalized similarity matrix

$$\overline{\boldsymbol{W}} = \boldsymbol{D}^{-1/2}\boldsymbol{W}\boldsymbol{D}^{-1/2}$$

with $\boldsymbol{D} = (d_{ii})$ being the diagonal degree matrix of $\boldsymbol{W}$ whose elements are given by $d_{ii} = \sum_j w_{ij}$ performs better than the case with the original similarities. In that regard, we add that the use of normalization is a common practice in graph-based approaches because it can typically achieve a better performance. To give an example, while the GFHF method (Zhu et al., 2003) uses original (unnormalized) similarities, the LGC method (Zhou et al., 2004) employs the normalized similarity matrix in its formulation. Moreover, while not directly related to graph transduction, it has been shown that the use of normalization has nice convergence properties in spectral clustering (von Luxburg et al., 2004). In terms of game theory, however, it is interesting to note that both versions of the transduction game (with and without normalizing input similarities) belong to the so-called class of normalized games, i.e., games with payoffs in a unit-length interval (Daskalakis, 2011), but the gap in their classification performance requires further investigation.

### *3.1.3 Connection to graph-based approaches*

Contrary to our derivation, the vast majority of graph-based SSL studies starts with writing down an objective function that casts the transductive learning problem as an energy minimization problem and most of the focus is on how to compute the optima of corresponding objective function. In general, all these methods attempt to estimate an optimal classification function which is defined on the nodes of the graph by minimizing an objective function with two terms. One term penalizes the mismatch between the initial label assignments and the labels estimated by the classifier. The second term is a regularization term that enforces the smoothness of the classification function. Although the game-theoretic perspective shifts the focus from optima of objective functions to equilibria of the non-cooperative games, we can shed some light on the connection between the proposed transduction game to the energy-based graph transduction methods in the special case in which the pairwise similarities are assumed to be symmetric. In this situation, computing ESS is equivalent to computing the optimal values of the following quadratic program (Hummel and Zucker, 1983; Miller and Zucker, 1991):

$$E(x) = \sum_{i=1}^{n} x_i^T \left( \sum_{j=1}^{n} A_{ij} x_j \right) = x^T A x$$

The Nash equilibria of a symmetric non-normalized transduction game is, thus, equivalent to solving the following optimization problem:

$$\begin{aligned} &\text{maximize } E(X) = \text{tr}\{X^T W X\} \\ &\text{subject to } x_i \in \Delta_i \; \forall i \in \mathcal{I}_{\mathcal{U}} \\ &\qquad\qquad x_i = e_i^k \; \forall i \in \mathcal{I}_{\mathcal{D}|k} \end{aligned}$$

where $\boldsymbol{X} = [x_1 \ldots x_n]^T$ is the $n \times c$ matrix of mixed strategies. This functional resembles the continuous relaxation of the $k$-way normalized cut criterion (Yu and Shi, 2003). However, note that there is a key difference in the game-theoretic formulation in that each mixed strategy $x_i$ is constrained to lie in the $c$-dimensional standard simplex $\Delta_i$. This subtle difference is very important since it provides robustness against noise and outliers (Pavan and Pelillo, 2007). Moreover, unlike the proposed approach, hard labeling constraints cannot be embedded into the Normalized Cuts framework in an explicit way, such that partial grouping constraints could be enforced by introducing extra linear equality constraints (Eriksson et al., 2007; Xu et al., 2009; Yu and Shi, 2004). The framework suggested recently in (Ghanem and Ahuja, 2010) is an exception but it is inherently a two-class clustering approach and requires a recursive strategy to solve multi-class problems.

In the case of the symmetric normalized game, the optimization problem is in fact equivalent to that of the graph Laplacian regularization used in (Zhu et al., 2003), which is known to be a special case of the regularization in (Zhou et al., 2004) with the parameter $\mu = \infty$ and the graph Laplacian being unnormalized. In this way, one can argue that the method in (Zhu et al., 2003) solves a special case of transduction games in which the pairwise similarities are symmetric and the partial payoffs are specified in terms of negative graph Laplacian.

### 3.2. Consensus and probabilistic clustering

In (Rota Bulò, Lourenco, Fred and Pelillo, 2010) a multi-population approach has been applied to the problem of consensus clustering, i.e, the problem of finding a proper consensual aggregation of data objects from a number of different input clusterings which have been obtained for a particular dataset. The proposed approach is built upon the evidence accumulation framework: For each pair of data objects, the probability of them to be clustered together (co-occurrence probability) is estimated from the ensemble of clusterings. This yields a co-occurrence matrix $\boldsymbol{C}$. The idea behind the consensus clustering approach is that the co-occurrence matrix can be factorized in terms of hidden probabilistic assignments of data points to $K$ classes. These assignments reside in a $K$-dimensional simplex $\Delta_K$ and the union of all these assignments then lays in a multi-simplex $\Delta^n_K$, where $n$ is the number of objects to be clustered. If we consider $\boldsymbol{Y}$ to be a $K \times n$ matrix representing such assignments, then the co-occurrence matrix $\mathbf{C}$ can be seen as an estimate of the product $\boldsymbol{Y}^T\boldsymbol{Y}$. By exploiting this fact, we find a consensus clustering by solving the following polynomial optimization problem in the multi-simplex $\Delta^n_K$

$$\begin{aligned} Y^* = \arg\min \quad & \|C - Y^\top Y\|_F^2 \\ \text{s.t.} \quad & Y \in \Delta_K^n \,. \end{aligned}$$

Note that a local solution of this optimization problem is equivalent to an equilibrium of a super-symmetric multi-population game. In order to use the Baum-Eagon theorem we need to meet the requirement of having a polynomial to maximize with nonnegative coefficients in

$$\begin{aligned} \max \quad & 2Tr(CY^\top Y) + \|Y^\top E_K Y\|^2 - \|Y^\top Y\|^2 \\ \text{s.t.} \quad & Y \in \Delta_K^n \,, \end{aligned}$$

the simplex-constrained variables. To this end, we consider the following equivalent optimization program

where $\boldsymbol{E}_K$ is the $K \times K$ matrix of all 1's. By applying the Baum-Eagon inequality we derive a growth transformation in $\Delta^n_K$, which allows us to find a local solution to the consensus clustering problem.

In (S. Rota Bulo' and M. Pelillo, 2010) a similar approach is applied in a context where the similarity matrix $\boldsymbol{S}$ is provided directly and is assumed to represent the likelihood that two objects are clustered together. In this case, our approach tries to factorize $\boldsymbol{S}$ as $\alpha \boldsymbol{Y}^T \boldsymbol{Y}$, where the additional parameter $\alpha$ is a nonnegative real value. The cluster assignments $\boldsymbol{Y}$ are computed by means of a 2-step optimization approach, where a step of the Baum-Eagon dynamics is interleaved with an update of the parameter $\alpha$, until convergence.

### 3.3. Semantic image labelling

A similar multi-population approach has been applied to the problem of semantic image labelling in (P. Kontschieder, S. Rota Bulò, M. Donoser, M. Pelillo and H. Bischof, 2011a) and (P. Kontschieder, S. Rota Bulò, H. Bischof and M. Pelillo, 2011s). Semantic image labelling problem is the task of assigning object class labels to all pixels in a test image. We provide an interpretation in terms of a label puzzle game by considering semantic image labelling as the task of assembling possibly overlapping label puzzle pieces, where the pieces are label configurations obtained by means of a modified random forest classifier.

Each puzzle piece $p \in P$ is a function $p : \mathbb{Z}^2 \to Y \cup \perp$ mapping two-dimensional points to labels $Y = \{1, \ldots, k\}$ or to void ( $\perp$ ), a special symbol indicating the absence of a label. A puzzle piece represents a topological and semantically plausible label configuration stemming from pixel-wise annotated training data. A *puzzle configuration* is a function $z : D \to P$ associating each pixel in $D \subseteq \mathbb{Z}^2$ with exactly one puzzle piece in $P$ . The set of puzzle configurations is denoted as $Z$ . A *labelling* for an image is a function $\ell : D \to Y$ mapping pixels in $D$ to labels in $Y$. The sets of images, labellings and puzzle pieces are denoted by $I$, $L$ and $P$, respectively.

We define the *agreement* of a puzzle piece $p \in P$ located in $(i, j) \in D$ with a labelling $\ell \in L$ as the number of corresponding pixels sharing the same label, i.e.,

$$\phi^{(i,j)}(p, \ell) = \sum_{(u,v) \in D} [p(u - i, v - j) = \ell(u, v)]$$

where $[Q]$ are the Iverson brackets yielding 1 if proposition $Q$ is true and 0 otherwise.

Given a puzzle configuration $z \in Z$ and a labelling $\ell \in L$, the *total agreement* $\Phi(z, \ell)$ of the image labelling puzzle is the sum of the agreements of each puzzle piece in $z$ with the labelling $\ell$ according to

$$\Phi(z,\ell) = \sum_{(i,j)\in D} \phi^{(i,j)}\left(z_{i,j},\ell\right)$$

A *label puzzle game* for an image $f \in I$ is a function $\pi_f$ mapping each pixel $(i, j) \in D$ to a non-empty set of puzzle pieces $\pi_f(i, j) \subseteq P$. This function restricts the possible choices of puzzle pieces per pixel and hence, also the set of admissible puzzle configurations to

$$z_{i,j}^{(t+1)} \in \arg\max_{p} \left\{ \phi^{(i,j)}\left(p, \ell^{(t)}\right) \middle| p \in \pi_f(i,j) \right\}$$

A solution of a label puzzle game $\pi_f$ is a pair $(z^*, \ell^*) \in Z|_{\pi_f} \times L$ consisting of an admissible puzzle configuration and a labelling for $f$ yielding the maximum total agreement and can be obtained by iteratively updating the puzzle configuration $z^{(t+1)}$ at time $(t+1)$ for a labelling $\ell^{(t)}$ by

$$(z^*,\ell^*) \in \arg\max_{(z,\ell)} \left\{ \Phi(z,\ell) \middle| (z,\ell) \in \mathcal{Z}|_{\pi_f} \times \mathcal{L} \right\}$$

and producing the new labelling $\ell^{(t+1)}$ by taking a majority vote from all overlapping puzzle pieces according to

$$\ell^{(t+1)}(u,v) \in \arg\max_{y} \left\{ \sum_{(i,j)\in D} \left[ z_{i,j}^{(t+1)}(u-i, v-j) = y \right] \middle| y \in Y \right\}$$

Note that from a multi-population game perspective, this approach is equivalent to having a game involving for each pixel $(i, j) \in D$ two types of players: One player plays by selecting a class label in Y, whereas the second one plays by selecting a puzzle piece in the set $\pi_f(i, j)$. Both of them have a payoff function which is proportional to the agreement of their choice with respect to the actual puzzle configuration and labelling. The dynamics adopted to find an equilibrium are best-response dynamcs, i.e. at each iteration each player takes a strategy in such a way as to maximizes his own payoff. Note also that an alternative solution would be to allow players to take mixed strategies. This would lead to a multi-simplex optimization, which could be addressed by means of the Baum-Eagon dynamics.

### 3.4. Correlation Clustering

In (N. Rebagliati, S. Rota Bulò e M. Pelillo, 2011) we applied the multi-population formulation in the context of Correlation Clustering. Correlation clustering is the problem of finding a crisp partition of the vertices of a correlation graph in such a way as to minimize the disagreements in the cluster assignments. In the standard formulation a correlation graph is a

fully connected graph with edge weights in {-1, +1} where a -1 implies a disagreement between two data and a +1 implies an agreement between two data. However, in a more general setting we can relax edge weights with the probability that two vertices agrees $p(i, j)$ and the solution clustering is not a crisp partition but is relaxed to be an assignment matrix $\boldsymbol{Y}$ in $\Delta^n_K$. The main motivation of our work was to use the most general concept of multi-population to capture those situations where a vertex belongs to more than one cluster. This is a common situation arising, for example, in the context of Consensus Clustering. If a vertex belongs to more than one cluster the resulting probabilistic assignment should weight its ownership equally among those clusters.

We relaxed the disagreement minimization of Correlation Clustering formulation into:

$$\phi_{\mathcal{G}}(\mathbf{Y}) = \sum_{(i,j)\in E} p_{ij} + \mathbf{y}_i^\top \mathbf{y}_j (1 - 2p_{ij}) \,.$$

and we developed a heuristic algorithm, based on the Baum-Eagon inequality, for the Correlation Clustering problem, returning a partition solution in the form of an assignment matrix $Y$ *in* $\Delta^n_K$.

$$y_{i\ell}^{(t+1)} = y_{i\ell}^{(t)} \frac{\left[\sum_{j|(i,j)\in E} 1 - (1 - 2p_{ij}) y_{j\ell}^{(t)}\right]}{\sum_{\ell\in L_k} y_{i\ell}^{(t)} \left[\sum_{j|(i,j)\in E} 1 - (1 - 2p_{ij}) y_{j\ell}^{(t)}\right]} \,, \qquad \text{(Alg-Q1)}$$

We made two key observations on the solution $\boldsymbol{Y}$. Firstly $K$, the number of classes, does not need to be fixed in advance, but it is automatically selected by the algorithm (consistently with the properties Correlation Clustering). Secondly we proved that the returned matrix $\boldsymbol{Y}$ is *deterministic,* that is each vertex is assigned to a single class. This actually means that relaxing the crisp partition constraint in the Correlation Clustering formulation is not sufficient for capturing overlapping clusters.

In order to overcome these limitations we adapted the functional of (Rota Bulò, Lourenco, Fred and Pelillo, 2010) in the Correlation Clustering context with the following relaxed formulation (Q2):

$$\varphi_{\mathcal{G}}(\mathbf{Y}) = \sum_{(i,j)\in E} p_{ij} + \mathbf{y}_i^\top \mathbf{y}_j (\mathbf{y}_i^\top \mathbf{y}_j - 2p_{ij}) \,. \qquad \text{(Q2)}$$

This relaxed formulation, differently from the one in (Rota Bulò, Lourenco, Fred and Pelillo, 2010), does not impose a fixed number of clusters. Both (Q2) and (Rota Bulò, Lourenco, Fred and Pelillo, 2010) can be considered in a general framework of Matrix Factorization with multi-population matrices.

Additionally, we propose a simple way for building an ensemble of weak hyperplane classifiers sampled from a reproducing kernel Hilbert space, which allows to apply Correlation Clustering without the empirical estimation of pairwise correlation values. Experiments on datasets from the UCI repository, using this ensemble, show that formulation (Q2), solved with a scheme similar to (Rota Bulò, Lourenco, Fred and Pelillo, 2010), can actually perform a

significative selection of the number of clusters and that the probabilistic assignments of **Y** found by (Q2) can be used to build a precision/recall like function of the misclassification error. As explained before this could not be achieved within the Correlation Clustering framework.

## 4. Applications

One of the most interesting property of the single-population game-theoretic learning approach is its selectivity, i.e., its tendency to select small sets of highly compatible elements. This property derives directly from the simplex constraint of the game-theoretic estimation approach, and finds a theoretical justification in the fact that the Lagrangian of the simplex is equivalent to an $L_1$ regularization. However, since the constraint is enforced in a hard way, the game-theoretic competition drives for an even stronger sparsity of the solution than what is obtained with the lasso constraint.

### 4.1. Matching and inlier selection

This is particularly evident in the graph-matching framework, where in contrast to the traditional quadratic assignment-based formulation which favors the largest possible set of consistent matches, the game-theoretic matcher selects a sparse set of very good correspondences, thus favoring low false positive rather than low false negative matches. This property made the approach particularly attractive for inlier selection problems, where the presence of outliers severely affect the final estimation. In these situations a common approach is to use ex post filtering approaches like RANSAC, which fail with a very strong or structured noise. In contrast, with the use of a game-theoretic approach, the evolution of the population drives the selection towards a highly cohesive subset of inliers. In (Albarelli, Rodolà, and Torsello, 2010a) this approach have been used in the selection of matching features from wide baseline stereo images for pose estimation. Here similarity is measured by the adherence to a weak affine camera model, extracting small sets of local matches that are highly reliable, each small set being a distinct equilibrium in a matching game.

Central to this framework is the definition of a matching game, or, specifically, the definition of the strategies available to the players and of the payoffs related to these strategies. Given a set $\boldsymbol{M}$ (model) of feature points in a source image and a set $\boldsymbol{D}$ (data) of features in a target image, we call a matching strategy any pair $(a_1, a_2)$ with $a_1 \in \boldsymbol{M}$ and $a_2 \in \boldsymbol{D}$. We call the set of all the matching strategies $\boldsymbol{S} \subseteq \boldsymbol{M} \times \boldsymbol{D}$. The total number of matching strategies in $\boldsymbol{S}$ can, in theory, be as large as the Cartesian product of the sets of features detected in the images. Since most interest point detectors extract thousands of features from an image, a suitable selection should be made in order to keep its size limited. To this end we can exploit unary information such as the distance between descriptors or the photo-consistency of local image patches to select only feasible pairs. Specifically, for each source

feature we can generate $k$ matching strategies that connect it to the $k$ destination features that are nearest in terms of descriptor distance. Since our game-theoretic approach operates inlier selection regardless of the descriptor, we do not need to set any threshold with respect to the absolute descriptor distance or the distinctiveness between the rest and the second nearest point. In this sense, the only constraint that we need to impose over $k$ is that it should be large enough that we can expect the correct correspondence to be among the candidates for a significant proportion of the source features.

Once $\mathbf{S}$ has been selected, our goal becomes to extract from it a large subset of correspondences that includes only correctly matched features: that is, strategies that associate a physical point in the source image with the same physical point (if visible) in the destination image. To this end, it is necessary to define a payoff function $\prod : \boldsymbol{S} \times \boldsymbol{S} \rightarrow \mathbb{R}^{+}$ that exploits some pairwise information available at this early stage (i.e., before estimating camera and scene parameters) and that can be used to impose consistency globally. Since location, scale, and rotation are associated to each feature, we can associate to each correspondence $(a, b)$ between feature $a$ in the source image and feature $b$ in the target image a similarity transform $\boldsymbol{T}(a, b)$ that maps the neighborhood of $a$ into the neighborhood of $b$, transforming the location, orientation, and scale measured in the source image into the location, orientation, and scale observed in the target image. Under small motion assumptions, we can expect these similarity transforms to be very similar locally. Thus, imposing the conservation of the similarity transform, we aim to extract clusters of feature matches that belong to the same region of the object and that tend to lie in the same level of depth. While this could seem to be an unsound assumption for general camera motion, we show experimentally that it holds well with the typical disparity found in standard multiple view and stereo data sets. Further, it should be noted that with large camera motion, most, if not all, commonly used feature detectors fail, thus any inlier selection attempt becomes meaningless.

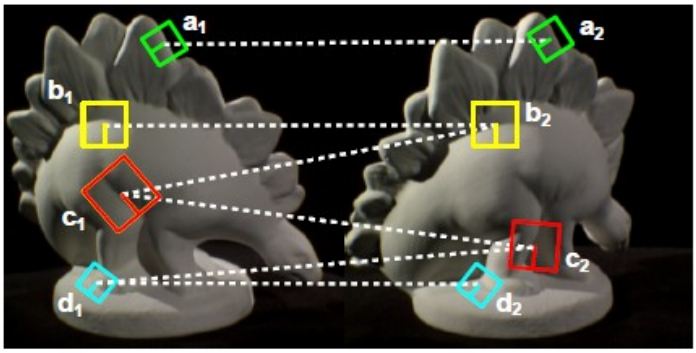
In order to define the payoff function $\prod$, we need a way to measure the distance between similarity transforms. In order to avoid the problem of mixing incommensurable quantities, we compute the distance in terms of the reprojection error expressed in pixels. Specifically, given two matching strategies $(a_1, a_2)$ and $(b_1, b_2)$ and their respective associated similarities $\boldsymbol{T}(a_1, a_2)$ and $\boldsymbol{T}(b_1, b_2)$, we calculate virtual points $a'_2$ and $b'_2$ by applying the other strategy'sS. Rota Bulò and M. Pelillo (Submitted) transformation to the source features $a_1$ and $b_1$. Given virtual points $a'_2$ and $b'_2$, we can measure the similarity between $(a_1, a_2)$ and $(b_1, b_2)$ as:

$$\mathrm{sim}((a_1, a_2), (b_1, b_2)) = e^{-\lambda max(|a_2 - a'_2|, |b_2 - b'_2|)}$$

where $\lambda$ is a selectivity parameter: If is small, then the similarity function (and thus the matching) is more tolerant with respect to deviation in the similarity trans forms, becoming more selective as grows.

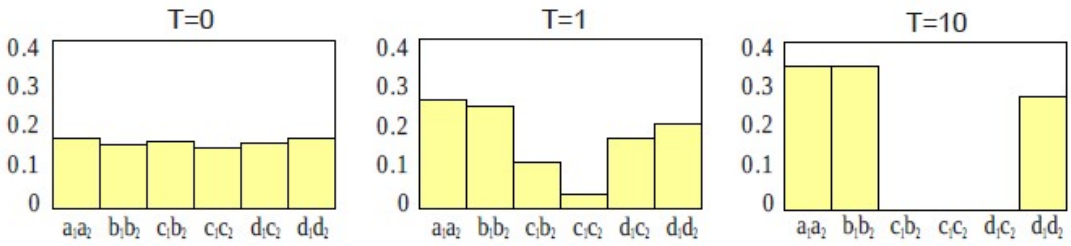
The main idea of the proposed approach is that by playing a matching game driven by a similarity enforcing payoff function the strategies (i.e., correspondence candidates) that share

a similar locally affine transformation are advantaged from an evolutionary point of view and shall emerge in the surviving population. In the following Figure we illustrate a simplified example of this process. Once the population has reached a local maximum, all the non-extinct mating strategies can be considered valid.




Π

| | $a_1a_2$ | $b_1b_2$ | $c_1b_2$ | $c_1c_2$ | $d_1c_2$ | $d_1d_2$ |
|---|---|---|---|---|---|---|
| $a_1a_2$ | 0 | 1 | 0.1 | 0.1 | 0.7 | 0.9 |
| $b_1b_2$ | 1 | 0 | 0 | 0.1 | 0.7 | 0.9 |
| $c_1b_2$ | 0.1 | 0 | 0 | 0 | 0.6 | 0.4 |
| $c_1c_2$ | 0.1 | 0.1 | 0 | 0 | 0 | 0.1 |
| $d_1c_2$ | 0.7 | 0.7 | 0.6 | 0 | 0 | 0 |
| $d_1d_1$ | 0.9 | 0.9 | 0.4 | 0.1 | 0 | 0 |



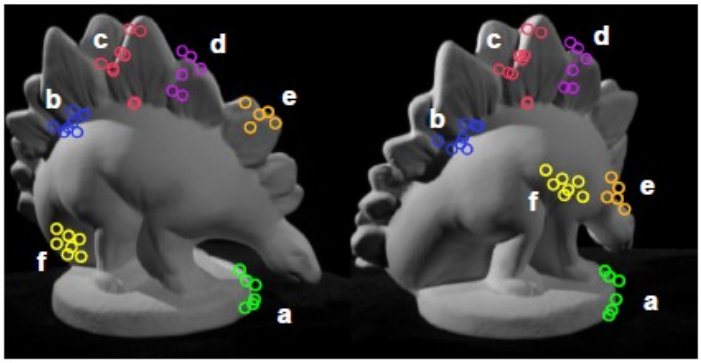

In (Albarelli, Rodolà, and Torsello, 2011b) the approach has been extended to obtain a single global similarity model by letting the various local groups play a hierarchical consistency game.

The game formulation introduced shifts the matching problem to a more global scope by producing a set of correspondences between groups of features. While the affine camera model extract very coherent groups, making such macro features more robust and descriptive than single points, in principle there is nothing that prevents the system to still produce wrong or weak matches. To reduce this chance we propose a different game setup that allows for a further refinement. In this game the strategies set $\boldsymbol{S}$ corresponds to the set of paired features groups extracted from the affine matching game and the payoff between them is related to the features' agreement to a common epipolar geometry. More specifically, given two pairs of matching groups $a \subseteq \boldsymbol{M}\times\boldsymbol{D}$ and $b \subseteq \boldsymbol{M}\times\boldsymbol{D}$, each one made up of model and data features, we estimate the epipolar geometry from $a \cup b$ and define the payoff among them as:

$$\Pi(a,b) = e^{-\lambda \sum_{(s,t)\in a\cup b} d(t,l(s))}$$

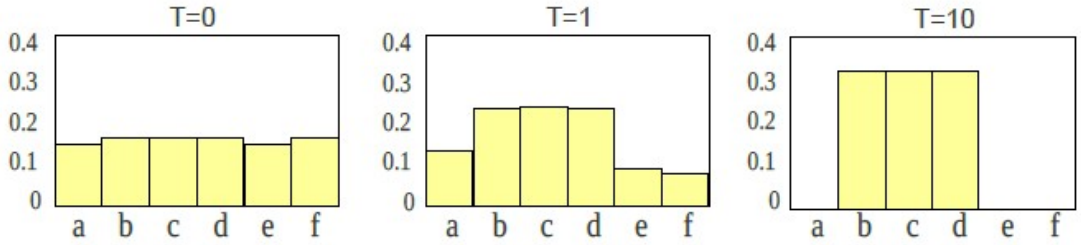
where $l(p)$ is a function that gives the epipolar line in the data image from the feature point p in the model image, according to the estimated epipolar geometry, and $d(p,l)$ calculates the distance between point $p$ and the epipolar line $l$. It is clear that this distance is low (and thus the payoff high) if the two groups share a common projective interpretation and high otherwise. Of course, different pairs of groups can agree on different epipolar geometry, but the transitive closure induced by the selection process ensures that the strategies in the

surviving population will agree on the same (or very similar) projective transformation (See the Figure below for an example of the evolution of this refinement game)




$\pi$

| | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| a | 0 | 0.3 | 0.27 | 0.24 | 0.3 | 0.1 |
| b | 0.3 | 0 | 0.95 | 0.9 | 0.2 | 0.2 |
| c | 0.27 | 0.95 | 0 | 0.95 | 0.19 | 0.21 |
| d | 0.24 | 0.9 | 0.95 | 0 | 0.18 | 0.22 |
| e | 0.3 | 0.2 | 0.19 | 0.18 | 0 | 0.17 |
| f | 0.1 | 0.2 | 0.21 | 0.22 | 0.17 | 0 |




The approach results in a correspondence selection that is more robust with respect to outliers, even when structured or repeated patterns are present, resulting in a much lower error in pose estimation and surface reconstruction. Other semi-local geometric models, more versatile than the local affine camera model, have been tested in (Rodolà, Albarelli, and Torsello, 2010a) and (Rodolà, Albarelli, and Torsello, 2010b).

Another interesting application for the sparse matcher is the robust alignment of 3D surfaces. In (Albarelli, Rodolà, and Torsello, 2010b) and (Albarelli, Rodolà, and Torsello, Submitted) surface registration is obtained by matching points consistent with an isometric transformation. The approach was shown to obtain a level of precision typical of fine registration approaches, without requiring an initial pose estimate. As in the inlier selection application, we are given a set $\boldsymbol{M}$ of *model* points, a set $\boldsymbol{D}$ of data points, and a set $\boldsymbol{S}\subseteq\boldsymbol{M}\times\boldsymbol{D}$ of candidate matches obtained by matching some surface descriptor. Our goal is of course to extract from $\boldsymbol{S}$ a subset of correct matches, that is, strategies that associate a point in the model surface with the same point in the data surface.

The game theoretic matching process requires that a sufficient amount of good matches be present in the initial set $\boldsymbol{S}$. Clearly, the distinctiveness of the descriptor used to characterize the data points has a big influence in fixing a reasonable number of candidates for each match. This observation, in turn, raises the quandary between the repeatability and the distinctiveness of feature descriptors. In fact, while a high level of distinctiveness is always desirable, this often comes at the price of much more instability with respect to noise and thus can lead to a poor repeatability.

Fortunately, with our approach, the descriptor itself is only used during the building step of set S and has no role in the evolutionary selection, which is purely driven by the payoff function. For this reason we find it reasonable to resort to a feature characterization that is little distinctive and to allow for several candidate matches letting the game-theoretic selection to operate a severe culling.

With these matches to hand, we define a matching game where the payoff enforces an isometric constrain between corresponding model and data points.

*Definition 1: Given a function* $\pi : S \times S \to \mathbb{R}^+$ *we call it a* isometry-enforcing payoff function *if for any* $((a_1, a_2), (b_1, b_2))$ *and* $((c_1, c_2), (d_1, d_2)) \in \boldsymbol{S}\times\boldsymbol{S}$ *we have that* $||a_1-b_1|-|a_2-b_2|| > ||c_1-d_1|-|c_2-d_2||$ *implies* $\pi((a_1, a_2), (b_1, b_2)) < \pi((c_1, c_2), (d_1, d_2))$. *In addition, if* $\pi((a_1, a_2), (b_1, b_2)) < \pi((b_1, b_2), (a_1, a_2))$ *for all* $(a_1, a_2), (b_1, b_2) \in \boldsymbol{S}$, $\pi$ *is said to be* symmetric.

A isometry-enforcing payoff function is a function that is monotonically decreasing with the absolute difference of the Euclidean distances between respectively the model and data points of the matching strategies compared. In other words, given two matching strategies, their payoff should be high if the distance between the model points is equal to the distance between the data points and it should decrease as the difference between such distances increases. This isometry-enforcing matching game exhibits some interesting properties.

*Theorem 1: Given a set of model points* $\boldsymbol{M}$, *a set of data points* $\boldsymbol{D} = \boldsymbol{TM}$ *that are exact rigid transformations of the points in* $\boldsymbol{M}$, *a set of matching strategies* $\boldsymbol{S}\subseteq\boldsymbol{M}\times\boldsymbol{D}$ *with* $(m, \boldsymbol{T}m) \in \boldsymbol{S}$ *for all* $m \in \boldsymbol{M}$, *and a matching game over them with a payoff function* $\pi$, *the vector* $\bar{x} \in \Delta_{|S|}$ *defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M; \\ 0 & \text{otherwise,} \end{cases}$$

*is an ESS and obtains the global maximum average payoff.*

This theorem states that when matching a surface with a rigidly transformed copy of itself the optimal solution (i.e., the population configuration that selects all the matching strategies assigning each point to its copy) is the stable state of maximum payoff. While the quality of the solution in presence of noise can only be assessed experimentally, we can give some theoretical results regarding occlusions.

*Theorem 2: Let* $\boldsymbol{M}$ *be a set of points with* $\boldsymbol{M}_a\subseteq\boldsymbol{M}$ *and* $\boldsymbol{D} = \boldsymbol{TM}_b$ *a rigid transformation of* $\boldsymbol{M}_b\subseteq\boldsymbol{M}$ *such that* $|\boldsymbol{M}_a \cap \boldsymbol{M}_b| > 3$, *and* $\mathbf{S}\subseteq\boldsymbol{M}_a\times\boldsymbol{D}$ *be a set of matching strategies over* $\boldsymbol{M}_a$ *and* $\boldsymbol{D}$ *with* $(m, \boldsymbol{T}m) \in \boldsymbol{S}$ *for all* $m \in \boldsymbol{M}_a \cap \boldsymbol{M}_b$. *Further, assume that the points that are not in the overlap, that is the points in* $\boldsymbol{E}_a = \boldsymbol{M}_a\backslash(\boldsymbol{M}_a \cap \boldsymbol{M}_b)$ *and* $\boldsymbol{E}_b = \boldsymbol{M}_b\backslash(\boldsymbol{M}_a \cap \boldsymbol{M}_b)$, *are sufficiently far away such that for every* $s \in \boldsymbol{S}$, $s = (m, \boldsymbol{T}m)$ $\pi(q, s) < \frac{|M_a\cap M_b|-1}{|M_a\cap M_b|}$

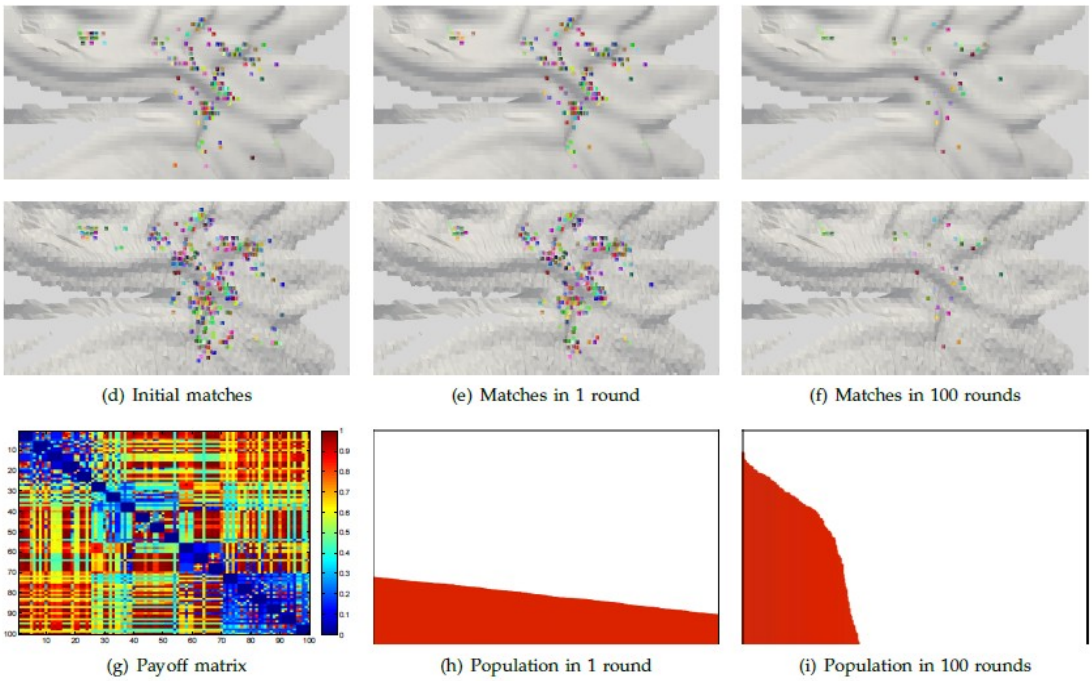
*with* $m \in \boldsymbol{M}_a \cap \boldsymbol{M}_b$ *and every* $q\in\boldsymbol{S}$, $q=(m_a, \boldsymbol{T}m_b)$ *with* $m_a\in\boldsymbol{E}_a$ *and* $m_b\in\boldsymbol{E}_b$, *we have then, the vector* $\bar{x} \in \Delta_{|S|}$ *defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M_a \cap M_b\,; \\ 0 & \text{otherwise,} \end{cases}$$

*is an ESS.*

The following Figure illustrates the evolution of the correspondences through the matching process.

The approach also proved to be more robust and less susceptible to local minima than other coarse registration approaches at the state of the art. In(Albarelli, Rodolà, and Torsello, 2011a) and (Rodolà, Albarelli, and Torsello, Work in progress) the approach was extended to recognize objects from cluttered 3D scenes, providing better recognition rate and matching accuracy than the approaches at the state of the art.



(d) Initial matches (e) Matches in 1 round (f) Matches in 100 rounds

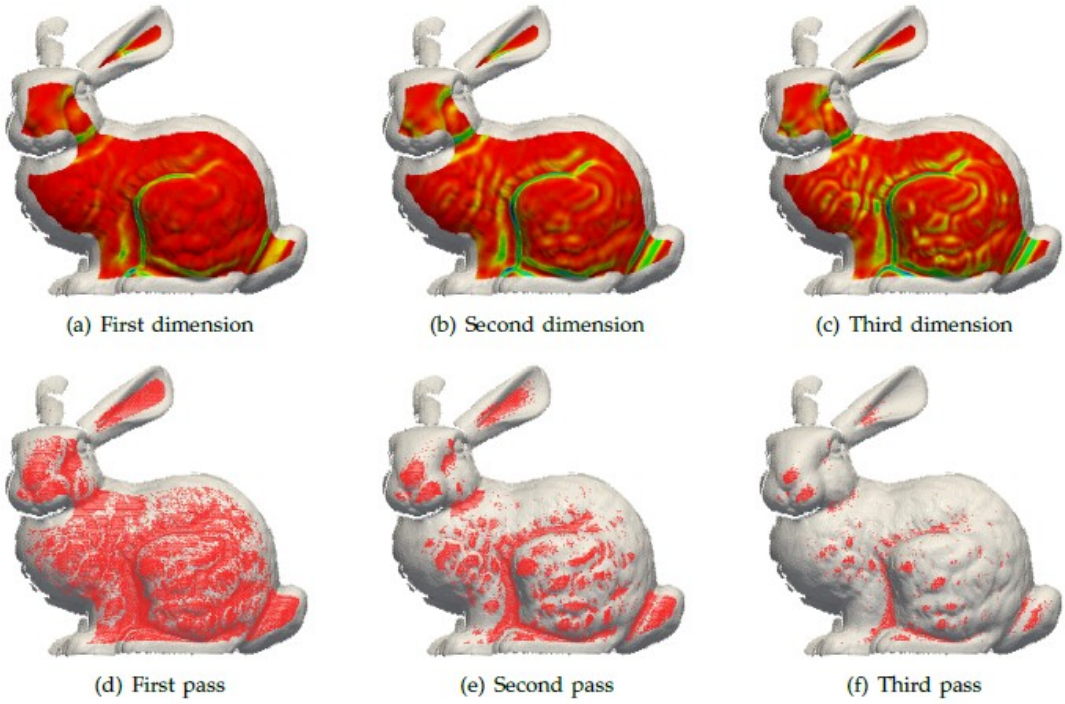
(g) Payoff matrix (h) Population in 1 round (i) Population in 100 rounds

### 4.2. Feature selection

The selectivity of the game-theoretic approach and its characteristic of being a one-class clustering framework can be used to perform relevant feature selection. The idea is that relevant features are the least common and self similar. Thus, relevant features are extracted by iteratively removing cluster of similar features. In (Albarelli, Rodolà, Cavallarin, and Torsello, 2010) this approach was applied to the problem of extracting and matching a signature from a cluttered and textured background, while in (Albarelli, Rodolà, and Torsello 2010c; Albarelli, Rodolà, and Torsello, Submitted) the idea was applied to the selection of robust surface descriptors for 3D registration. In this context, the feature selector approach differ from the more common interest point detection approach in that the latter are only local: points are deemed interesting if there are enough local high frequency features. This makes the point localizable, but not distinctive. By contrast our approach performs a global selection, where distinctiveness is the most important feature. In (Albarelli, Rodolà, and Torsello 2010c;

Albarelli, Rodolà, and Torsello, Submitted) we start from a set of very loose integral features designed to be robust to noise, and apply a selection game where the similarity between features is an exponentially decaying function of the Euclidean distance of the features.

The following Figure illustrates the loose figures and the progressive elimination of non distinctive points operated by the game-theoretic selection process.



(a) First dimension (b) Second dimension (c) Third dimension

(d) First pass (e) Second pass (f) Third pass

### 4.3. Feature combination

Feature combination is an effective method in improving object recognition and classification performance. Feature combination methods can be categorized into two types according to the level at which they operate. The first one use features of all individual classifiers to form a joint feature vector, which is then used in later classification. In the case of SVM classification, feature combination translates to combining a set of kernel functions into one final kernel function. The second type operates at the decision or the score level, namely, the outputs of all individual classifiers are used in combination. This approach is attractive as different types of classifiers, e.g., SVM and kNN, can be combined together. In this paper we focus on kernel combination with application in SVM classification.

In (Hou and Pelillo, 2011) we developed a simple yet effective weighting scheme for feature combination based on the dominant set concept. Specifically, we use the dominant set clustering method to evaluate how difficult a kernel matrix is for a SVM classifier. This degree of difficulty is found to be related to the classification performance and thus is used as the weight in feature combination.

A SVM classifier partitions training examples of different classes with as large a margin as possible. From this mechanism, we see that if the training examples of the same class are highly similar and those of different classes are dissimilar, it's likely that the SVM separates

different classes with a large margin resulting in high recognition rate. In other words, the possibility of a kernel matrix producing a high recognition rate is determined by the degree with which it satisfies the high intra-class and low inter-class similarity constraint. We measure the accuracy of a kernel matrix by comparing the label partition on the training data with the dominant set clusters obtained from the kernel matrix. More precisely, we define the kernel accuracy to be inversely related to the (scaled) amount of entropy within each dominant set:

$$w_{dset} = \frac{1}{N}\sum_{i=1}^{N}(1 - \frac{H_i}{\log C})$$

where $n_{ij}$ is the number of elements in dominant set $i$ which belong to class $j$, and $N_i$ is the overall number of elements in dominant set $i$, and $C$ is the number of classes. Clearly, $w_{\text{dset}}$ reaches 1 with the ideal case where all dominant sets are of single-class, and becomes 0 when all dominant sets are shared equally by all classes.

For each kernel, we calculate $w_{\text{dset}}$ and use it as the weight in kernel matrix combination. In implementation we use $w^3_{\text{dset}}$ instead of $w_{\text{dset}}$ as the weight to highlight the difference between different kernels. Our weighting scheme is intuitive, in that it provides a meaningful weight to feature combination, and simple, in that the weights of features are computed separately. In case of a large set of kernel matrices for combination, the latter property implies a much smaller memory requirement than joint optimization methods. Experiments have been conducted on different object classification tasks and, in particular, on the biomedical applications of WP6-7. The results show that the approach consistently outperforms the state of the art.

## 5. Stability of dominant sets

The concept of Nash Equilibrium (NE) is very general and has many refinements, those are usually introduced as adaptations to some practical situation. Here we focus on the possible noise on payoff matrices arising from applications. Clearly, we are interested in NE robust to noise, so that a NE $x$ of a game $G$ is said *essential* is for every $\varepsilon$ there exist a $\delta$ such that every $G'$ that is distant $\delta$ from *G* has a NE closer than $\varepsilon$ to $x$. The essentiality notion is due to (Wu and Jiang, 1962). It is not difficult to show that Evolutionary Stable Strategy, and thus Dominant Sets, are essential (Weibull, 1995).

We are interested in extending the essentiality concept into a quantified *robustness* of the support of Dominant Sets w.r.t. noise. For this reason we introduced (Rebagliati and Pelillo, work in progress) the concept of $\eta$-essentiality as those dominant sets whose support does not change with respect to a small variation of the payoffs, quantified by $\eta$. Then we tried to relate this concept with two different quantities: the spectrum of the payoff matrix (related to the support of the Dominant Set), in particular to the second smallest eigenvalue and the values of the indicator vector of a Dominant Set.

Finally, we studied a regularization operation on the payoff matrix that can increase the robustness of a payoff matrix. This regularization operation is as follows:

$$(1-\mu)A + \mu(E-I), \mu \geq 0$$

with E the matrix of all ones and I the identity.

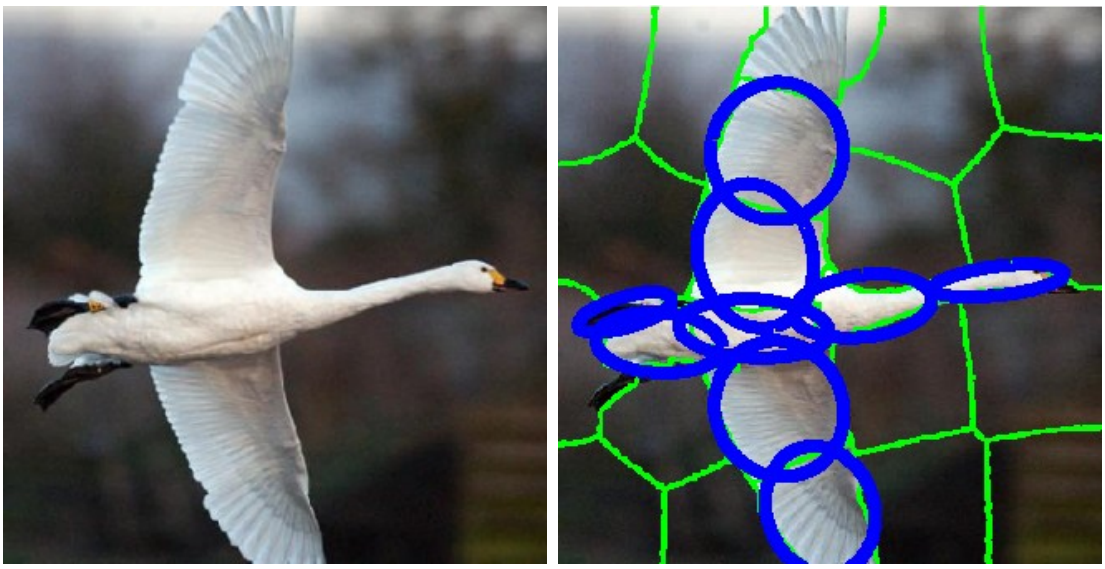
This operation was previously used by (Pavan, Pelillo 2003) to create a hierarchy of Dominat Sets solutions. Here we studied its use to increase the robustness of a Dominant Set. The most important empirical phenomenon of this regularization is that the indicator vector of a Dominant Set strongly converges to the constant vector, where for strong convergence we mean that the minimum value increases up to 1/n (n is the cardinality of the Dominant Set). At the same time two opposite effects occur: the stability of the payoff matrix increases but the information on the payoff matrix decreases. For this reason we have to set a parameter accounting for a trade-off between robustness and the quantity of information we retain.

## 6. On-going and future work

There is ongoing work on further generalizations of the game-theoretic framework. We plan to generalize the InImDyn algorithm to polymatrix games, in order to have a faster and more accurate alternative to the Baum-Eagon dynamics. Given a mixed strategy profile, the idea is to iteratively select a player violating most the Nash conditions and perform an InImDyn-like update of his playing strategy. Similar theoretical properties, holding for InImDyn, should hold also for this generalization.

We also plan to study the relation between Markov Random Fields (MRF) / Conditional Random Fields (CRF), which are widely used in Machine Learning and Computer Vision, and multi-population games, in order to propose alternative relaxation models based on game-theory, and consequently alternative algorithms for learning the parameters characterizing the new models.

In a previous work (Levinshtein, Sminchisescu and Dickinson 2009) developed a method for detecting and assembling medial parts from a multiscale superpixel segmentation of an image. In a current work (Rebagliati, Fidler, Dickinson and Pelillo) we extend this method by means of a modified superpixel definition, based upon an ellipse inscribed into the superpixel. These ellipses can be visually seen in the images of the swan below (superpixels in green, ellipses in blue):



Indeed, this visual feature allows the creation of dense similarity matrices by pairwise comparison among the ellipses. The final matrix includes the similarity of their alignment, their direction, their eccentricity or their texture. Given this matrix we can apply Dominant Set Clustering and tune parameters according to the dataset at hand. Preliminary results show improvements over the previous method of (Levinshtein, Sminchisescu and Dickinson 2009).

## 7. Publications resulting from WP5.2

A. Albarelli, E. Rodolà, A. Cavallarin, and A. Torsello (2010), "Robust Figure Extraction on Textured Background: a Game-Theoretic Approach." In20th International Conference on Pattern Recognition. (ICPR2010), ISBN 978-1-424-47542-1, doi:10.1109/ICPR.2010.97.

A. Albarelli, E. Rodolà, and A. Torsello (2010a), "Robust Game-Theoretic Inlier Selection for Bundle Adjustment." In 3D Data Processing, Visualization and Transmission (3DPVT).

A. Albarelli, E. Rodolà, and A. Torsello (2010b), "A Game-Theoretic Approach to Fine Surface Registration without Initial Motion Estimation." In IEEE International Conference on Computer Vision and Pattern Recognition (CVPR2010), ISBN 978-1-424-46984-0, doi:10.1109/CVPR.2010.5540183, IEEE Computer Society.

A. Albarelli, E. Rodolà, and A. Torsello (2010c), "Loosely Distinctive Features for Robust Surface Alignment." In 11th European Conference on Computer Vision (ECCV2010), pp. 519-532, ISBN 978-3-642-15554-3, doi:10.1007/978-3-642-15555-0_38.

A. Albarelli, E. Rodola, and A. Torsello (Submitted), “Fast and Accurate Surface Alignment Through an Isometry-Enforcing Game.” Submitted to IEEE Trans. Pattern Anal. Mach. Intell.

A. Albarelli, E. Rodolà, A. Torsello (2011a), "A Non-Cooperative Game for 3D Object Recognition in Cluttered Scenes." In International Conference on 3D Imaging, Modeling,

Processing, Visualization and Transmission, pp. 252-259, ISBN 978-0-769-54369-7, doi:10.1109/3DIMPVT.2011.39, IEEE Computer Society.

A. Albarelli, E. Rodolà, A. Torsello (2011b), "Imposing Semi-Local Geometric Constraints for Accurate Correspondence Selection in Structure from Motion: A Game-Theoretic Perspective."*International Journal of Computer Vision*, doi:10.1007/s11263-011-0432-4, published online 24 March 2011.

A. Erdem and M. Pelillo (2011), “Graph Transduction as a Non-Cooperative Game.” In Proc. 8th IAPR-TC-15 International Workshop on Graph-Based Representations in Pattern Recognition (GbR2011), pp. 195-204, Springer, LNCS 6658.

A. Erdem, and M. Pelillo (to appear), “Graph Transduction as a Non-Cooperative Game.” Neural Computation.

J. Hou and M. Pelillo (2011), “Feature Combination Based on Dominant Set Clustering .”, SIMBAD TR Series, 2011_58.

P. Kontschieder, S. Rota Bulò, M. Donoser, M. Pelillo and H. Bischof (2011a), “Semantic Image Labelling as a Label Puzzle Game.” The 22nd British Machine Vision Conference (BMVC2011), Dundee, UK.

P. Kontschieder, S. Rota Bulò, H. Bischof and M. Pelillo (2011b), “Structured Class-Labels in Random Forests for Semantic Image Labelling.” International Conference on Computer Vision (ICCV2011), Barcelona, Spain. (oral presentation)

P. Ren, R. C. Wilson, E. R. Hancock (2011), “High Order Structural Matching Using Dominant Cluster Analysis”. In 16th *International Conference on Image Analysis and Processing, Ravenna, Italy*.

N. Rebagliati and M. Pelillo (work in progress) “On Regularization of Dominant Set Clusters”.

E. Rodolà, A. Albarelli, and A. Torsello (2010a), "A Game-Theoretic Approach to Robust Selection of Multi-View Point Correspondence." In 20th International Conference on Pattern Recognition. (ICPR2010), ISBN 978-1-424-47542-1, doi:10.1109/ICPR.2010.23.

E. Rodolà, A. Albarelli, and A. Torsello (2010b), "A Game-Theoretic Approach to the Enforcement of Global Consistency in Multi-View Feature Matching." In Joint IAPR International Workshops on Structural and Syntactic Pattern Recognition (SSPR 2010) and Statistical Techniques in Pattern Recognition (SPR 2010), ISBN 978-3-642-14979-5, doi:10.1007/978-3-642-14980-1_34.

E. Rodolà, A. Albarelli, A. Torsello (Work in progress), "A multiscale Game for 3D Object Recognition in Cluttered Scenes."

N. Rebagliati, S. Rota Bulò e M. Pelillo (2011), “Correlation Clustering with Stochastic Labelings.”, SIMBAD TR Series, 2011_58.

S. Rota Bulò, I. M. Bomze and M. Pelillo (2010), “Fast Population Game Dynamics for Dominant Sets and Other Quadratic Optimization Problems.” In International Workshop on Structural and Syntactic Pattern Recognition (SSPR2010), Cesme, Izmir, Turkey.

S. Rota Bulò, A. Lourenco, A. Fred and M. Pelillo (2010), “Pairwise Probabilistic Clustering Using Evidence Accumulation.” In International Workshop on Statistical Techniques in Pattern Recognition (SPR2010), Cesme, Izmir, Turkey.

S. Rota Bulò and M. Pelillo (2009), “A Game-Theoretic Approach to Hypergraph Clustering.” In Advances in Neural Information Processing Conference (NIPS2009), vol. 22, pages 1571-1579.

S. Rota Bulò and M. Pelillo (2010), “Probabilistic Clustering using the Baum-Eagon Inequality.” In International Conference on Pattern Recognition (ICPR2010), Istanbul, Turkey.

S. Rota Bulò and M. Pelillo (Submitted), “A Game-Theoretic Approach to Hypergraph Clustering.” Submitted to IEEE Trans. Pattern Anal. Mach. Intell.

S. Rota Bulò, M. Pelillo and I. M. Bomze (2011), “Graph-Based Quadratic Optimization: A Fast Evolutionary Approach.” Computer Vision and Image Understanding, vol. 115, pp.984-995.

Z. Zhang and E. R. Hancock (2011a), “A Graph-Based Approach to Feature Selection.” In Proc. 8th IAPR-TC-15 International Workshop on Graph-Based Representations in Pattern Recognition (GbR2011).

Z. Zhang and E. R. Hancock (2011b), “A Hypergraph-Based Approach to Feature Selection.” In Proc. 14th International Conference on Computer Analysis of Images and Patterns (CAIP2011).

Z. Zhang and E. R. Hancock (2011c), “Mutual Information Criteria for Feature Selection.” In Proc. 1st International Workshop on Similarity-Based Pattern Analysis and Recognition, to appear.

## 8. Other references cited in the report

L. E. Baum and J. A. Eagon (1967), “An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology.” Bull. Amer. Math. Soc., 73:360–363.

W. van Damme (1991), “Stability and Perfection of Nash Equilibria”, Berlin: Springer Verlag 2nd Edition.

Daskalakis, C. (2011). On the complexity of approximating a Nash equilibrium. In SODA.

Eriksson, A. P., Olsson, C., & Kahl, F. (2007). Normalized cuts revisited: A reformulation for segmentation with linear grouping constraints. In ICCV.

Ghanem, B., & Ahuja, N. (2010). Dinkelbach NCUT: An efficient framework for solving normalized cuts problems with priors and convex constraints. Int. J. Comput. Vis., 89(1), 40–55.

Haralick, R.M., & Shapiro, L. G. (1979). The consistent labeling problem: Part I. IEEE Trans. Pattern Anal. Mach. Intell., 1(2), 173–184.

Haralick, R.M., Davis, L.S., Rosenfeld, A., & Milgram, D.L. (1978). Reduction operations for constraint satisfaction. Inf. Sci., 199–219.

Howson, J. T. (1972). Equilibria of polymatrix games. Management Science, 18(5), 312–318.

Hummel, R. A., & Zucker, S. W. (1983). On the foundations of relaxation labeling processes. IEEE Trans. Pattern Anal. Mach. Intell., 5(3), 267–287.

Janovskaya, E. B. (1968). Equilibrium points in polymatrix games. (in Russian) Litovskii Matematicheskii Sbornik, 8, 381–384 (Math. Reviews 39 #3831.).

A. Levinshtein, C. Sminchisescu, and S. Dickinson, (2009) “Multiscale Symmetric Part Detection and Grouping”. Proceedings, International Conference on Computer Vision (ICCV), Kyoto, Japan, September 2009.

von Luxburg, U., Bousquet, O., & Belkin, M. (2004). On the convergence of spectral clustering on random samples: The normalized case. In COLT, 457–471.

Marriott, K., & Stuckey, P. J. (1998). Programming with Constraints: An Introduction. MIT Press, Cambridge, MA.

Miller, D. A., & Zucker, S. W. (1991). Copositive-plus Lemke algorithm solves polymatrix games. Operations research letters, 10(5), 285–290.

M. Pavan and M. Pelillo (2003), "Dominant sets and hierarchical clustering" In ICCV 2003 - 9th IEEE International Conference on Computer vision, vol I, pp 362-369.

Pavan, M., & Pelillo, M. (2007). Dominant sets and pairwise clustering. IEEE Trans. on Pattern Anal. and Mach. Intell., 29(1), 167–172.

M. Pelillo (1999), “Replicator equations, naximal cliques, and graph isomorphism.” NeuralS. Rota Bulò and M. Pelillo (Submitted) Computation, 11(8), 1933–1955.

M. Pelillo, K. Siddiqi, and S. W. Zucker (1999), “Matching hierarchical structures using association graphs.” IEEE Trans. on Pattern Anal. and Mach. Intell., 21(11), 1105–1120.

Tsang, E. (1993). Foundations of Constraint Satisfaction. Academic Press, London and San Diego.

Waltz, D. L. (1975). Understanding line drawings of scenes with shadows. In Winston, P. H., editor, The Psychology of Computer Vision, 19–92. McGraw-Hill.

J. Weibull (1995), “Evolutionary Game Theory”, The MIT Press, Cambridge.

W. Wu and J. Jiang (1962), “Essential equilibrium points of n-person non-cooperative games”. Sci.Sinica 11:1307-22

Xu, L., Li, W., & Schuurmans, D. (2009). Fast normalized cut with linear constraints. In CVPR, 2866–2873.

Yu, S. X., & Shi, J. (2003). Multiclass spectral clustering. In ICCV, 313–319.

Yu, S. X., & Shi, J. (2004). Segmentation given partial grouping constraints. IEEE Trans. Pattern Anal. Mach. Intell., 26(2), 173–183.

Zhu, X., Ghahramani, Z., & Lafferty, J. (2003). Semi-supervised learning using Gaussian fields and harmonic functions. In ICML, 912–919.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & Schölkopf, B. (2004). Learning with local and global consistency. In NIPS, 321–328.

# Robust Figure Extraction on Textured Background: a Game-Theoretic Approach

Andrea Albarelli, Emanuele Rodolà, Alberto Cavallarin, and Andrea Torsello
Dipartimento di Informatica - Università Ca' Foscari
via Torino, 155 - 30172 Venice Italy
http://www.dsi.unive.it

## Abstract

*Feature-based image matching relies on the assumption that the features contained in the model are distinctive enough. When both model and data present a sizeable amount of clutter, the signal-to-noise ratio falls and the detection becomes more challenging. If such clutter exhibits a coherent structure, as it is the case for textured background, matching becomes even harder. In fact, the large amount of repeatable features extracted from the texture dims the strength of the relatively few interesting points of the object itself. In this paper we introduce a game-theoretic approach that allows to distinguish foreground features from background ones. In addition the same technique can be used to deal with the object matching itself. The whole procedure is validated by applying it to a practical scenario and by comparing it with a standard point-pattern matching technique.*

## 1. Introduction

Given its central role in many computer vision tasks, image matching and registration is a widely investigated topic in literature. Several approaches exploit global properties of the images, ranging from the many techniques based on cross-correlation [6] to those that work in the frequency domain [4] or adopt the mutual information as a similarity measure [10]. While successful in many scenarios, the global nature of those techniques makes them little robust to changes in illumination and to the presence of clutter. Feature-based approaches partially solve those problems. Attributed feature points are extracted from images using detectors [8, 9, 7] and descriptors [5, 2] that are locally invariant to illumination, scale and rotation. Usually, the model features are matched with those obtained from the target image by means of some RANSAC-based approach that can exploit the prior given by the descriptors [3]. Critical to the success of this kind of technique is of course the distinctiveness of the extracted features. Unfortunately, when dealing with textured clutter, this distinctiveness comes short and the number of very repeatable but irrelevant features overshadows those coming from the foreground object. To avoid false matches it is mandatory to recognize and ignore the background. In this paper we cope with both the filtering of the background features and the recognition task by tailoring the matching framework introduced in [1]. Specifically we model the filtering step as a self-matching game, where features that show high mutual similarity in the same image are deemed not distinctive enough and thus screened away. By converse, the recognition step is performed as a matching game between the model and a data image, where a set of highly coherent pairs of corresponding features is seeked.

## 2. The Matching Game

Evolutionary game theory [11] considers an idealized scenario where pairs of individuals are repeatedly drawn at random from a large population to play a two-player game. Each player obtains a payoff that depends only on the strategies played by him and its opponent. Players are not supposed to behave rationally, but rather they act according to a pre-programmed behavior, or mixed strategy. It is supposed that some selection process operates over time on the distribution of behaviors favoring players that receive larger payoffs. More formally, let $O = \{1, \cdots, n\}$ be the set of available strategies (*pure strategies* in the language of game theory) and $C = (c_{ij})$ be a matrix specifying the payoff that an individual playing strategy $i$ receives against someone playing strategy $j$. A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, \ldots, x_n)^T$ over the available strategies $O$.

Being probability distributions, mixed strategies are constrained to lie in the n-dimensional standard simplex

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.97

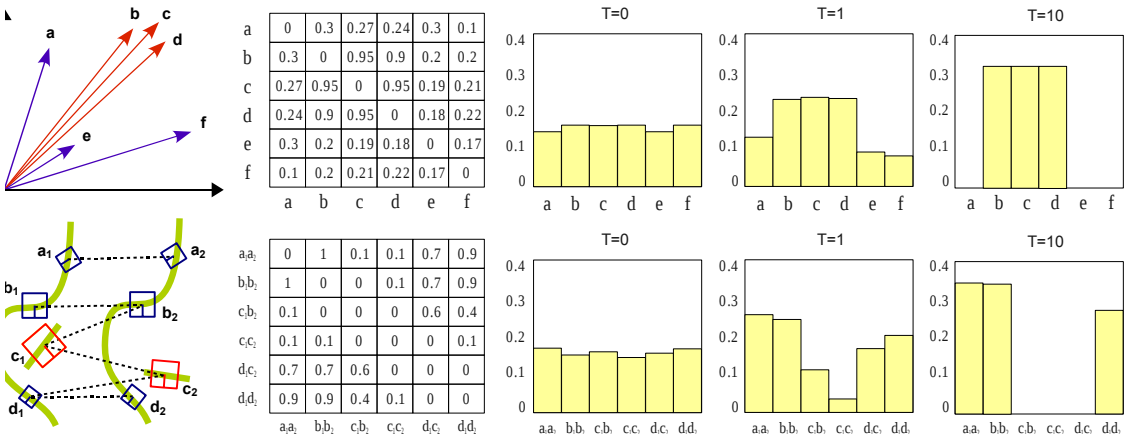



**Figure 1. Examples of the two evolutionary matching games proposed**

$\Delta^n = \{\mathbf{x} \in \mathbb{R}^n : \forall i \in 1 \dots n\ x_i \geq 0,\ \sum_{i=1}^n x_i = 1\}$. The *support* of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, is defined as the set of elements chosen with non-zero probability: $\sigma(\mathbf{x}) = \{i \in O \mid x_i > 0\}$. The expected payoff received by a player choosing element $i$ when playing against a player adopting a mixed strategy $\mathbf{x}$ is $(C\mathbf{x})_i = \sum_j c_{ij}x_j$, hence the expected payoff received by adopting the mixed strategy $\mathbf{y}$ against $\mathbf{x}$ is $\mathbf{y}^T C\mathbf{x}$. The *best replies* against mixed strategy $\mathbf{x}$ is the set of mixed strategies

$$\beta(\mathbf{x}) = \{\mathbf{y} \in \Delta \mid \mathbf{y}^T C\mathbf{x} = \max_{\mathbf{z}}(\mathbf{z}^T C\mathbf{x})\}.$$

A strategy $\mathbf{x}$ is said to be a *Nash equilibrium* if it is the best reply to itself, i.e., $\forall \mathbf{y} \in \Delta,\ \mathbf{x}^T C\mathbf{x} \geq \mathbf{y}^T C\mathbf{x}$. This implies that $\forall i \in \sigma(\mathbf{x})$ we have $(C\mathbf{x})_i = \mathbf{x}^T C\mathbf{x}$; that is, the payoff of every strategy in the support of $\mathbf{x}$ is constant. A strategy $\mathbf{x}$ is said to be an *evolutionary stable strategy* (ESS) if it is a Nash equilibrium and

$$\forall \mathbf{y} \in \Delta \quad \mathbf{x}^T C\mathbf{x} = \mathbf{y}^T C\mathbf{x} \Rightarrow \mathbf{x}^T C\mathbf{y} > \mathbf{y}^T C\mathbf{y}.$$

This condition guarantees that any deviation from the stable strategies does not pay. The search for a stable state is performed by simulating the evolution of a natural selection process. Under very loose conditions, any dynamics that respect the payoffs is guaranteed to converge to Nash equilibria [11] and (hopefully) to ESS's; for this reason, the choice of an actual selection process is not crucial and can be driven mostly by considerations of efficiency and simplicity. We chose to use the replicator dynamics, a well-known formalization of the selection process governed by the following equation

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t)\frac{(C\mathbf{x}(t))_i}{\mathbf{x}(t)^T C\mathbf{x}(t)}$$

where $\mathbf{x}_i$ is the $i$-th element of the population and $C$ the payoff matrix. Once the population has reached a local maximum, all the non-extincted pure strategies (i.e., $\sigma(\mathbf{x})$) can be considered selected by the game.

## 2.1. Filtering a Textured Background

When dealing with textures, we can expect a large number of features that exhibit very similar descriptors. This is a very unfortunate condition for matching: in fact, this high level of conguence can easily distract any matcher from the foreground object. Paradoxically we use this property to screen out background features. Following [1], we model each feature as a strategy in a matching game where the payoff matrix is defined by:

$$C(i,j) = e^{-\alpha|d_i - d_j|} \tag{1}$$

where $d_i$ and $d_j$ are the descriptor vectors associated to features $i$ and $j$, and $\alpha$ is a parameter that controls the level of selectivity. Clearly, features that are similar will get a large mutual payoff and thus are more likely to be selected by the evolutive process. A simplified (but numerically correct) example of such evolution is shown in the first row of Fig. 1. Here, six descriptors of dimensionality 2 are labeled from $a$ to $f$. Vectors $b$,$c$ and $d$ get high values in the payoff matrix since they are close in the descriptor space. Other descriptors get lower mutual payoffs, according to their respective distances. We start the replicator dynamics ($T = 0$) near the barycenter of $\Delta^6$, which is sligthly perturbed to help avoiding local minima. After just one iteration ($T = 1$), strategies $b$,$c$ and $d$ get a significant evolutionary boost over the others, and after ten iterations ($T = 10$) they are the only strategies left in the support. We can then classify those features as background and filter them out.

## 2.2. Matching Model and Data

In order to match model and data points we need to define a slightly different matching game. In this con-

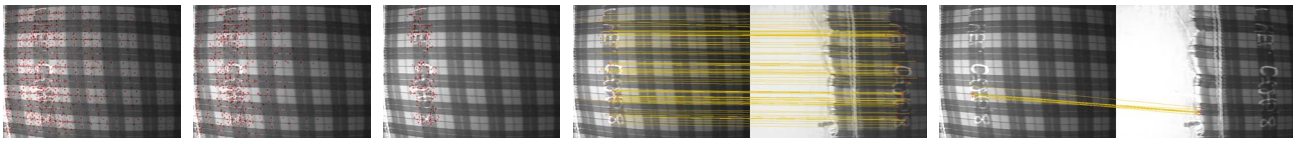
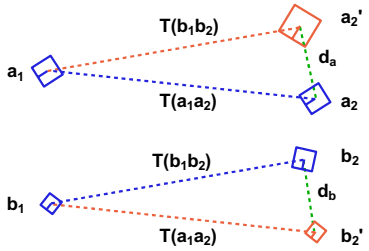


**Figure 2. Background fi ltering and feature matching (best viewed in color)**

text, each strategy models a pair of features $(a_1, a_2)$ that belong respectively to the model and the data. We defi ne a payoff among strategies that is proportional to the compatibility of the affine transformation estimated by the descriptor used (for instance, SIFT [5] or SURF [2]). Specifi cally, we are able to associate to each strategy $(a_1, a_2)$ an affi ne transformation, which we call $T(a_1, a_2)$.




When this is applied to $a_1$ it produces the point $a_2$, but when it is applied to the model point $b_1$ it will give a point $b'_2$ that is near to $b_2$ if $T(a_1, a_2)$ is similar to $T(b_1, b_2)$. Given two strategies $(a_1, a_2)$ and $(b_1, b_2)$ and their associated transformations $T(a_1, a_2)$ and $T(b_1, b_2)$ we calculate their reciprocal reprojected virtual points as: $a'_2 = T(b_1, b_2)a_1$ and $b'_2 = T(a_1, a_2)b_1$. Given virtual points $a'_2$ and $b'_2$ we are finally able to defi ne the payoff between$(a_1, a_2)$ and $(b_1, b_2)$ as:

$$C((a_1, a_2), (b_1, b_2)) = e^{-\beta max(|a_2 - a'_2|, |b_2 - b'_2|)} \quad (2)$$

where $\beta$ is a selectivity parameter that allows to operate a more or less selective matching game. Clearly, large groups of point pairs that are coherent with respect to an affi ne transformation will receive a large payoff and thus an evolutive advantage. In the second row of Fig. 1 we show an example of this matching game. Here, coherent strategies exhibit high payoff values (i.e., $C((a_1, a_2), (b_1, b_2)) = 1$), while less compatible pairs get lower scores (i.e., $C((a_1, a_2), (c_1, c_2)) = 0.1$). Note that strategies that share the same model or data point get payoff 0 to avoid one-to-many matching. Initially, the population is set to a slightly perturbed barycenter of $\Delta^6$. After one iteration, $(c_1, b_2)$ and $(c_1, c_2)$ have lost a signifi cant amount of support, while $(d_1, c_2)$ and $(d_1, d_2)$ are still played by a sizeable amount of population, despite being mutually exclusive. After ten iterations, $(d_1, d_2)$ has fi nally prevailed over $(d_1, c_2)$ and the fi nal support has emerged.

## 3. Experimental Evaluation

We tested our game-theoretic approach by applying it to the detection of hand-written markers placed on textured fabric. This is a typical scenario for batch tracking in the textile industry, where barcodes or RFID tags are not viable solutions due to the harsh cloth processing conditions that would destroy them. The fi rst three frames of Fig. 2 show the background filtering performance of our method. The fi rst frame contains all the original SIFT features extracted, the second one shows those survived after applying our filter with selectivity parameter $\alpha = 10^{-4}$. By using $\alpha = 10^{-3}$ all the background is screened in the third frame. We observed that a larger value of $\alpha$ does not affect much the result, as foreground features are quite disjointed. The matcher performance has been evaluated by comparing its precision-recall curve with those obtained by using an optimized RANSAC-based technique. Specifically, we implemented a PROSAC [3] variant by using descriptor vectors as hints for the selection of transformation candidates in an affine point-pattern matching. In order to assess the effect of the background elimination step, we applied this RANSAC schema to both fi ltered and unfi ltered frames.

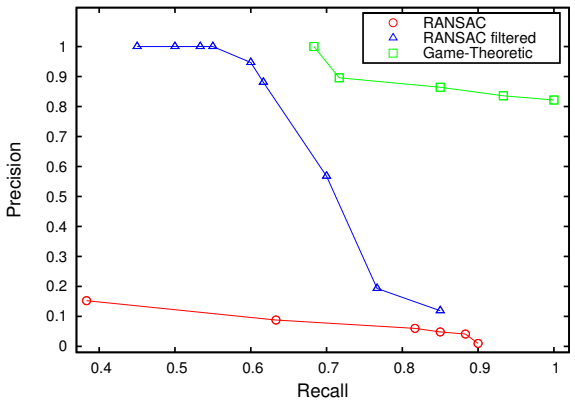



**Figure 3. Comparison with RANSAC**

The trade-off between precision and recall was adjusted respectively by means of parameter $\beta$ and by using different thresholds for the consensus. Tests were performed with 20 markers and 15 different fabric patterns. The markers were present in 59 frames of a

30.000 frames long video sequence. Given the constant presence of a textured background, the poor results obtained with RANSAC and the unfiltered video were expected. Indeed, we were unable to reach a full recall without a complete loss of precision, and even when accepting a low recall most of the detected frames were false positives due to background matching. RANSAC performance increases dramatically after application of the filter. Nevertheless, it is not possible to obtain a high level of recall without losing precision. This is due to the presence of features that do not belong to the foreground marker and neither are part of a texture. This happens, for instance, with sewings, seams or dirt present in the fabric. In the right half of Fig. 2 we show an instance where our method obtains the correct match, while RANSAC is distracted by a junction in the fabric. The game-theoretic matcher (applied over filtered frames) obtains by far the best results. In fact, a perfect recall is obtained with a precision value above 0.8 ($\beta = 10^{-3}$) and, by using a more selective parameter ($\beta = 10^{-2}$) all the false positives are avoided while still obtaining a recall just slightly below 0.7. In some practical applications it is more important to guarantee a recall of 1 since a moderate number of false positives can be tolerated (and filtered bottomward in the pipeline), while a miss in the detection is not allowed. To measure the loss in precision with respect to noise, we corrupted both data and model with additive Gaussian noise. At each noise level (expressed with the standard deviation in Fig. 4) we tuned $\beta$ to maintain a recall of 1 and measured the precision. While it was always possible to obtain a complete recall, we observed a linear decay of the precision. This is not a failure of the matcher itself, but an impaired effectiveness of the background filter due to the reduced similarity among the extracted descriptors. It should be noted, however, that in this experimental setup a precision of 0.3 with a recall of 1 corresponds to a fall-out of 0.006 (about 180 false positives over 30.000 tests).

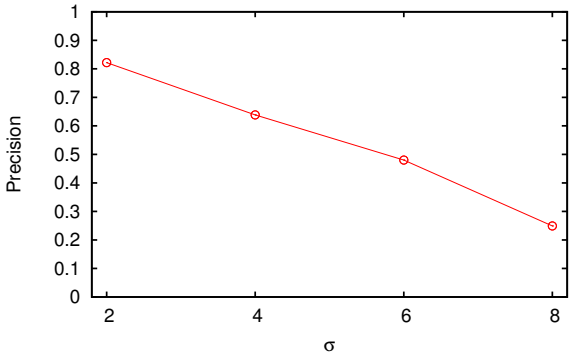



**Figure 4. Effect of image noise**

## 4. Conclusions

We presented a game-theoretic approach that allows to perform a robust feature-based matching even when the foreground is absorbed in a highly textured background. This is done by playing two different non-cooperative games: a filter game, that separates foreground from background, and a matching game, that performs the actual point-pattern matching. An experimental validation shows that both the steps concur to the improvement of the whole matching task and the obtained results outperform in terms of precision and recall an optimized RANSAC-based approach.


### Acknowledgments

We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.


## References


[1] A. Albarelli, S. Rota Bulò, A. Torsello, and M. Pelillo. Matching as a non-cooperative game. In *ICCV 2009: Proceedings of the 2009 IEEE International Conference on Computer Vision*. IEEE Computer Society, 2009.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. J. V. Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[3] O. Chum and J. Matas. Matching with prosac - progressive sample consensus. In *CVPR 05: Proceedings of the 2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR05) - Volume 1*, pages 220–226, Washington, DC, USA, 2005. IEEE Computer Society.

[4] E. De Castro and C. Morandi. Registration of translated and rotated images using finite fourier transforms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(5):700–703, 1987.

[5] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003.

[6] W. K. Pratt. *Digital image processing (2nd ed.)*. John Wiley & Sons, Inc., New York, NY, USA, 1991.

[7] E. Rosten, R. Porter, and T. Drummond. Faster and better: a machine learning approach to corner detection. *CoRR*, abs/0810.2434, 2008.

[8] J. Shi and C. Tomasi. Good features to track. In *1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 593 – 600, 1994.

[9] S. M. Smith and J. M. Brady. Susan—a new approach to low level image processing. *Int. J. Comput. Vision*, 23(1):45–78, 1997.

[10] P. Viola and W. M. Wells, III. Alignment by maximization of mutual information. *Int. J. Comput. Vision*, 24(2):137–154, 1997.

[11] J. Weibull. *Evolutionary Game Theory*. MIT Press, 1995.

# Robust Game-Theoretic Inlier Selection for Bundle Adjustment

Andrea Albarelli, Emanuele Rodolà and Andrea Torsello
Dipartimento di Informatica - Università Ca' Foscari
via Torino, 155 - 30172 Venice Italy
albarelli@unive.it rodola@dsi.unive.it torsello@dsi.unive.it

## Abstract

*Bundle Adjustment is a widely adopted self-calibration technique that allows to estimate scene structure and camera parameters at once. Typically this happens by iteratively minimizing the reprojection error between a set of 2D stereo correspondences and their predicted 3D positions. This optimization is almost invariantly carried out by means of the Levenberg-Marquardt algorithm, which is very sensitive to the presence of outliers in the input data. For this reason many structure-from-motion techniques adopt some inlier selection algorithm. This usually happens both in the initial feature matching step and by pruning matches with larger reprojection error after an initial optimization. While this works well in many scenarios, outliers that are not filtered before the optimization can still lead to wrong parameter estimation or even prevent convergence. In this paper we introduce a novel stereo correspondences selection schema that exploits Game Theory in order to perform a robust inlier selection before any optimization step. The practical effectiveness of the proposed approach is confirmed by an extensive set of experiments and comparisons with state-of-the-art techniques.*

## 1. Introduction

The selection of 2D point correspondences is arguably the most important step in image based multi-view reconstruction. As a matter of fact, differently from techniques augmented by structured light or known markers, wrong initial correspondences can lead to sub-optimal parameter estimation or, in the worst cases, to the inability of the optimization algorithm to obtain a feasible solution. For this reason reconstruction approaches adopt several specially crafted expedients to avoid as much as possible the inclusion of outliers. In the first place correspondences are not searched throughout all the image plane, but only points that are both repeatable and well characterized are considered. This selection is carried out by means of interest point detectors and feature descriptors. Salient points are localized with sub-pixel accuracy by general detectors, such as Harris Operator [2] and Difference of Gaussians [7], or by using techniques that are able to locate affine invariant regions, such as Maximally Stable Extremal Regions (MSER) [8] and Hessian-Affine [9]. The affine invariance property is desirable since the change in appearance of a scene region after a small camera motion can be locally approximated with an affine transformation. Once interesting points are found, they must be matched to form the candidate pairs to be fed to the bundle adjustment algorithm. Most of the currently used techniques for point matching are based on the computation of some affine invariant feature descriptor. Specifically, to each point is assigned a descriptor vector with tens to hundreds of dimensions, a scale and a rotation value. Among the most used feature descriptor algorithms are the Scale-Invariant Feature Transform (SIFT) [6, 5], the Speeded Up Robust Features (SURF) [3], the Gradient Location and Orientation Histogram (GLOH) [10] and more recently the Local Energy based Shape Histogram (LESH) [11]. In all of these techniques, the descriptor vector itself is robust with respect to affine trasformations: i.e., similar image regions exhibit descriptor vectors with small mutual Euclidean distance. This property is used to match each point with the candidate that is associated to the nearest descriptor vector. If the descriptor is not distinctive enough this approach is prone to select many outliers. A common optimization involves the definition of a maximum threshold over the distance ratio between the first and the second nearest neighbors. In addition, points that are matched multiple times are deemed as ambiguous and discarded (i.e., one-to-one matching is enforced). Another common heuristic for the elimination of erroneous matches is to exclude points that exhibit a large reprojection error after a first round of Levenberg-Marquardt optimization [4] (see for instance [14]). Unfortunately this afterthought is based upon an error estimation that depends on the point pairs chosen beforehand; this leads to a quandary that can only be solved by avoiding wrong matches from the start. In this paper we introduce a robust matching technique that allows to operate a very accurate inlier selection at an early stage of the process and without any need to rely on 3D reprojection. In the experimental section, to assess the advantages of our approach, we present a comprehensive set of comparisons between the results delivered by our technique and those

obtained with a reference implementation of the structure-from-motion system presented in [13] and [14].

## 2. Game-Theoretic Point Pairs Selection

The selection of matching points on behalf of the feature descriptor is only able to exploit local information. This limitation conflicts with the richness of information that is embedded in the scene structure. For instance, under the assumption of rigidity and small camera motion, intuition suggests that features that are close in one view cannot be too far apart in the other one. Further, if a pair of features exhibit a certain difference of angles or ratio of scales, this relation should be maintained among their respective matches. Our basic idea is to formalize this intuitive notion of consistency between pairs of feature matches into a real-valued utility function and to find a large set of matches that express a high level of mutual compatibility. Of course, the ability to define a meaningful pairwise utility function and a reliable technique for finding a consistent set as large as possible is paramount for the effectiveness of the approach. Following [15, 1], we model the matching process in a game-theoretic framework, where two players extracted from a large population select a pair of matching points from two images. The player then receives a payoff from the other players proportional to how compatible his match is with respect to the other player's choice, where the compatibility derives from some utility function that rewards pairs of matches that are consistent. In Section 2.2 such a function will be proposed, but in practice many different choices can be made: for instance it is possible to assign a high payoff to pairs of matches that preserve the distance between source and destination points and a low payoff otherwise. Clearly, it is in each player's interest to pick matches that are compatible with those the other players are likely to choose. In general, as the game is repeated, players will adapt their behavior to prefer matchings that yield larger payoffs, driving all inconsistent hypotheses to extinction, and settling for an equilibrium where the pool of matches from which the players are still actively selecting their associations forms a cohesive set with high mutual support. Within this formulation, the solutions of the matching problem correspond to evolutionary stable states (ESS's), a robust population-based generalization of the notion of a Nash equilibrium. In a sense, this matching process can be seen as a contextual voting system, where each time the game is repeated the previous selections of the other players affect the future vote of each player in an attempt to reach consensus. This way the evolving context brings global information into the selection process.

### 2.1. Non-cooperative Games

Originated in the early 40's, Game Theory was an attempt to formalize a system characterized by the actions of entities with competing objectives, which is thus hard to characterize with a single objective function [16]. According to this view, the emphasis shifts from the search of a local optimum to the definition of equilibria between opposing forces. In this setting multiple players have at their disposal a set of strategies and their goal is to maximize a payoff that depends also on the strategies adopted by other players. Evolutionary game theory originated in the early 70's as an attempt to apply the principles and tools of game theory to biological contexts. Evolutionary game theory considers an idealized scenario where pairs of individuals are repeatedly drawn at random from a large population to play a two-player game. In contrast to traditional game-theoretic models, players are not supposed to behave rationally, but rather they act according to a pre-programmed behavior, or mixed strategy. It is supposed that some selection process operates over time on the distribution of behaviors favoring players that receive higher payoffs.

More formally, let $O = \{1, \cdots, n\}$ be the set of available strategies (*pure strategies* in the language of game theory), and $C = (c_{ij})$ be a matrix specifying the payoff that an individual playing strategy $i$ receives against someone playing strategy $j$. A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, \ldots, x_n)^T$ over the available strategies $O$. Clearly, mixed strategies are constrained to lie in the n-dimensional standard simplex

$$\Delta^n = \left\{ \mathbf{x} \in \mathbb{R}^n \ : \ x_i \geq 0 \text{ for all } i \in 1 \ldots n, \ \sum_{i=1}^{n} x_i = 1 \right\} .$$

The *support* of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, is defined as the set of elements chosen with non-zero probability: $\sigma(\mathbf{x}) = \{i \in O \mid x_i > 0\}$. The expected payoff received by a player choosing element $i$ when playing against a player adopting a mixed strategy $\mathbf{x}$ is $(C\mathbf{x})_i = \sum_j c_{ij} x_j$, hence the expected payoff received by adopting the mixed strategy $\mathbf{y}$ against $\mathbf{x}$ is $\mathbf{y}^T C \mathbf{x}$. The *best replies* against mixed strategy $\mathbf{x}$ is the set of mixed strategies

$$\beta(\mathbf{x}) = \{\mathbf{y} \in \Delta \mid \mathbf{y}^T C \mathbf{x} = \max_{\mathbf{z}} (\mathbf{z}^T C \mathbf{x})\} .$$

A strategy $\mathbf{x}$ is said to be a *Nash equilibrium* if it is the best reply to itself, i.e., $\forall \mathbf{y} \in \Delta, \ \mathbf{x}^T C \mathbf{x} \geq \mathbf{y}^T C \mathbf{x}$. This implies that $\forall i \in \sigma(\mathbf{x})$ we have $(C\mathbf{x})_i = \mathbf{x}^T C \mathbf{x}$; that is, the payoff of every strategy in the support of $\mathbf{x}$ is constant.

A strategy $\mathbf{x}$ is said to be an *evolutionary stable strategy* (ESS) if it is a Nash equilibrium and

$$\forall \mathbf{y} \in \Delta \quad \mathbf{x}^T C \mathbf{x} = \mathbf{y}^T C \mathbf{x} \Rightarrow \mathbf{x}^T C \mathbf{y} > \mathbf{y}^T C \mathbf{y} . \qquad (1)$$

This condition guarantees that any deviation from the stable strategies does not pay.

### 2.2. Matching Strategies and Payoffs

Central to this framework is the definition of a *matching game*, which implies the definition of the strategies avail-

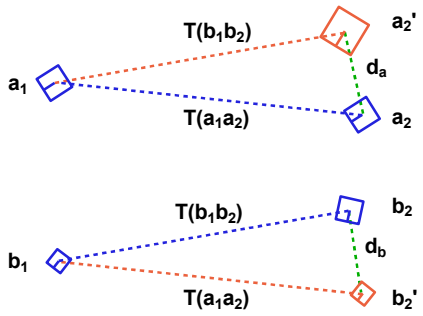



Figure 1. The payoff between two matching strategies is inversely proportional to the maximum reprojection error obtained by applying the affine transformation estimated by a match to the other.

able to the players and of the payoffs related to these strategies. Given a set $M$ of feature points in a source image and a set $D$ of potentially corresponding features in a destination image, we call a *matching strategy* any pair $(a_1, a_2)$ with $a_1 \in M$ and $a_2 \in D$. We call the set of all the matching strategies $S$. In principle, all the features extracted by an interest point detector could be used to build the matching strategies set, thus leading to a size of the set $S$ that grows quadratically with the average number of features detected in an image. In practice, however, in Section 2.3 we adopt some heuristics that allow us to obtain good overall results with a much smaller set. Once $S$ has been selected, our goal becomes to extract from it the largest subset that includes only correctly matched points: that is, strategies that associate a feature in the source image with the same feature in the destination image. To this extent, it is necessary to define a payoff function $\Pi : S \times S \rightarrow \mathbb{R}^+$ that exploits some pairwise information available at this early stage (i.e. before estimating camera and scene parameters). Since scale and rotation are associated to each feature, it seems natural to try to use this information to enforce coherence between matching strategies. Specifically, we are able to associate to each matching strategy $(a_1, a_2)$ one and only one similarity transformation, that we call $T(a_1, a_2)$. When this transformation is applied to $a_1$ it produces the point $a_2$, but when applied to the source point $b_1$ of the matching strategy $(b_1, b_2)$ it does not need to produce $b_2$. In fact it will produce $b_2$ if and only if $T(a_1, a_2) = T(b_1, b_2)$, otherwise it will give a point $b'_2$ that is as near to $b_2$ as the transformation $T(a_1, a_2)$ is similar to $T(b_1, b_2)$. Given two matching strategies $(a_1, a_2)$ and $(b_1, b_2)$ and their respective associated similarities $T(a_1, a_2)$ and $T(b_1, b_2)$, we calculate their reciprocal reprojected points as:

$$\begin{aligned} a'_2 &= T(b_1, b_2)a_1 \\ b'_2 &= T(a_1, a_2)b_1 \end{aligned}$$

That is, the virtual points obtained by applying to each source point the similarity transformation associated to the other match (see Fig. 1). Thus, given virtual points $a'_2$ and $b'_2$, the payoff between $(a_1, a_2)$ and $(b_1, b_2)$ is:

$$\Pi((a_1, a_2), (b_1, b_2)) = e^{-\lambda max(|a_2 - a'_2|, |b_2 - b'_2|)} \quad (2)$$

where $\lambda$ is a selectivity parameter that allows to operate a more or less strict inlier selection. If $\lambda$ is small, then the payoff function (and thus the matching) is more tolerant, otherwise the evolutionary process becomes more selective as $\lambda$ grows. We define 2 as a *similarity enforcing payoff function* and we call a *matching game* any symmetric non-cooperative game that involves a matching strategies set $S$ and a similarity enforcing payoff function $\Pi$.

The rationale of the payoff function proposed in equation 2 is that, while by changing point of view the similarity relationship between features is not maintained (as the object is not planar and the transformation is projective), we can expect the transformation to be a similarity at least "locally". This means that we aim to extract clusters of feature matches that belong to the same region of the object and that tend to lie in the same level of depth. While this could seem to be an unsound assumption for general camera motion, in the experimental section we will show that it holds well with the typical disparity found in standard multiple view and stereo data sets. Further it should be noted that with large camera motion most, if not all, commonly used feature detectors fail, thus any inlier selection attempt becomes meaningless. One final note should be made about one-to-one matching. Since each source feature can correspond with at most one destination point, it is desirable to avoid any kind of multiple match. It is easy to show that a pair of strategies with zero mutual payoff cannot belong to the support of an ESS (see [1]), thus any payoff function $\Pi$ can be easily adapted to enforce one-to-one matching by defining:

$$\Pi' = \begin{cases} \Pi((a_1, a_2), (b_1, b_2)) & \text{if } a_1 \neq b_1 \text{ and } a_2 \neq b_2 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

We can define 3 as a *one-to-one similarity enforcing payoff function*.

### 2.3. Building the Matching Strategies Set

From a theoretical point of view the total number of matching strategies can be as large as the Cartesian product of the sets of features detected in the images. Since most interest point detectors extract thousands of features from an image and the size of the payoff matrix grows quadratically with the number of matching strategies, this leads to problems too large to be managed in an efficient way. While the feature descriptor has not been used to define the payoff function $\Pi$, it could be useful to reduce the number of matching strategies considered. Specifically, for each source feature we can generate $k$ matching strategies that connect it to the $k$ destination features that are the nearest

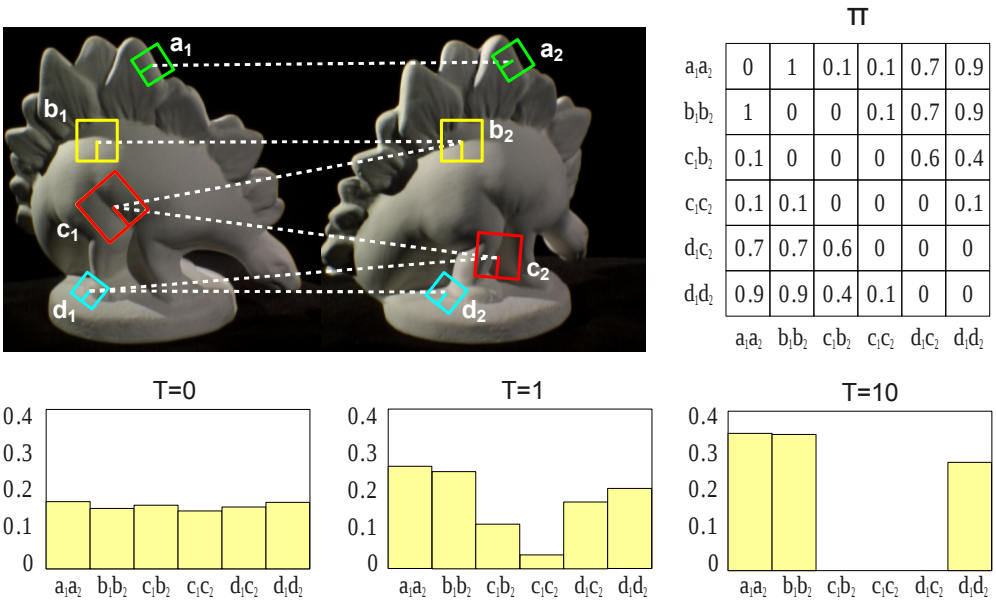



Figure 2. An example of the evolutionary process. Four feature points are extracted from two images and a total of six matching strategies are selected as initial hypotheses. The matrix $\Pi$ shows the compatibilities between pairs of matching strategies according to a one-to-one similarity-enforcing payoff function. Each matching strategy got zero payoff with itself and with strategies that share the same source or destination point (i.e., $\Pi((b_1,b_2),(c_1,b_2)) = 0$). Strategies that are coherent with respect to similarity transformation exhibit high payoff values (i.e., $\Pi((a_1,a_2),(b_1,b_2)) = 1$ and $\pi((a_1,a_2),(d_1,d_2)) = 0.9$)), while less compatible pairs get lower scores (i.e., $\pi((a_1,a_2),(c_1,c_2)) = 0.1$). Initially (at T=0) the population is set to the barycenter of the simplex and slightly perturbed. After just one iteration, $(c_1,b_2)$ and $(c_1,c_2)$ have lost a significant amount of support, while $(d_1,c_2)$ and $(d_1,d_2)$ are still played by a sizable amount of population. After ten iterations (T=10) $(d_1,d_2)$ has finally prevailed over $(d_1,c_2)$ (note that the two are mutually exclusive). Note that in the final population $((a_1,a_2),(b_1,b_2))$ have a higher support than $(d_1,d_2)$ since they are a little more coherent with respect to similarity.

in terms of descriptor distance. Since our game-theoretic approach operates inlier selection regardless of the descriptor, we do not need to set any threshold with respect to the absolute descriptor distance or the distinctiveness between the first and the second nearest point. In this sense, the only constraint that we need to impose over $k$ is that it should be high enough to allow the correct correspondence to be among the candidates a significative percentage of the times. In the experimental section we will analyze the influence of $k$ over the quality of the matches obtained.

### 2.4. Evolving to an Optimal Solution

The search for a stable state is performed by simulating the evolution of a natural selection process. Under very loose conditions, any dynamics that respect the payoffs is guaranteed to converge to Nash equilibria [16] and (hopefully) to ESS's; for this reason, the choice of an actual selection process is not crucial and can be driven mostly by considerations of efficiency and simplicity. We chose to use the replicator dynamics, a well-known formalization of the selection process governed by the following equation

$$\mathbf{x}_i(t+1) = x_i(t)\frac{(C\mathbf{x}(t))_i}{\mathbf{x}(t)^T C\mathbf{x}(t)} \qquad (4)$$

where $\mathbf{x}_i$ is the $i$-th element of the population and $C$ the payoff matrix. Once the population has reached a local maximum, all the non-extincted mating strategies can be considered valid (see Fig. 2). In practice strategies are extincted only after an infinite number of iterations. Since we halt the evolution when the population ceases to change significantly, it is necessary to introduce some criteria to distinguish correct from non-correct matches. To avoid a hard threshold we chose to keep as valid all the strategies played by a population amount exceeding a percentage of the most popular strategy. We call this percentage *quality threshold*. As mentioned in Section 2.2, each evolution process selects a group of matching strategies that are coherent with respect to a local similarity transformation. This means that if we want to cover a large portion of the subject we need to iterate many times and prune the previously selected matches at each new start. Obviously, after all the depth levels have been swept, small and not significant residual groups start to emerge from the evolution. To avoid the selection of these spurious matches we fixed a minimum cardinality for each valid group. We call this cardinality *group size*.

## 3. Experimental Results

We conducted different sets of experiments. Our first goal was to analyze the impact of the algorithm parameters, namely $\lambda$, $k$, *quality threshold* and *group size*, over the quality of the results obtained. For this purpose we used a pair of

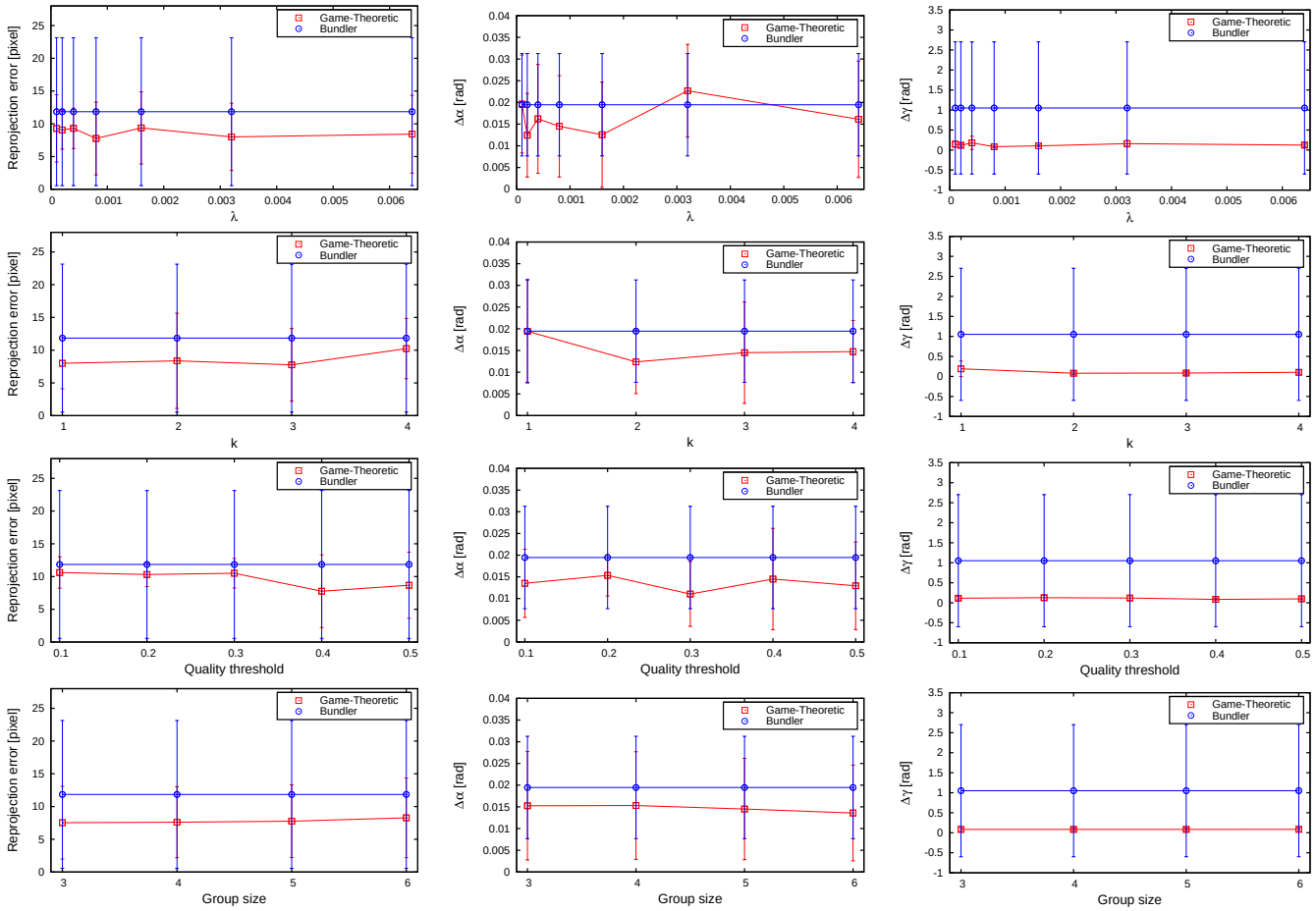
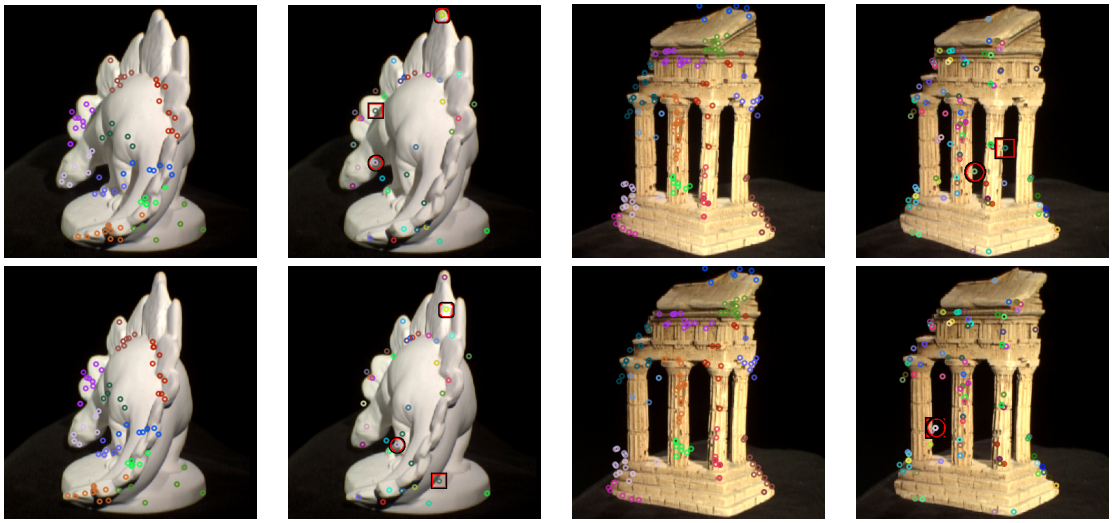



Figure 3. Analysis of the performance of the approach with respect to variation of the parameters of the algorithm.

cameras previously calibrated through a standard procedure and took stereo pictures of 20 different, isolated objects; then, we investigated the influence of the four parameters separately. For each test we evaluated three quality measures: the average reprojection error in pixels ($\epsilon$) and the differences in radians between the (calibrated) ground-truth and respectively the estimated rotation angle ($\Delta\alpha$) and rotation axis ($\Delta\gamma$). In addition, each stereo pair was processed with the keymatcher included in the structure-from-motion suite Bundler [13, 14]. Finally, the correspondences produced by both the Bundler keymatcher and our technique were given as an input to the bundle adjustment procedure included in the suite. This allows to obtain a fair comparison of the two approaches, whose quality parameters can be directly compared, being the result of running the same optimizer on different inputs. In Fig. 3 we reported the results of these experiments. The first row shows the effect of the selectivity parameter $\lambda$. As expected both a too low and a too high value lead to less satisfactory results, mainly with respect to the estimation of the angle between the two cameras. This is probably due respectively to a too tight and a too relaxed enforcement of local coherence. The three rows below show the impact of the number of candidate matches for each source point, the quality threshold that a match must exceed to be considered feasible and the minimum size of a valid group. Overall, these experiments suggest that those parameters have little influence over the quality of the result, notwithstanding the Game-Theoretic approach achieves better results in nearly every case.

For the purpose of exploring further the differences between our technique and the Bundler keymatcher, we investigated in depth four cases. We will describe them here in two separate sets. The first set of unordered images comes from the "DinoRing" and "TempleRing" sequences from the Middlebury Multi-View Stereo dataset [12]; for these models, camera parameters are provided and used as a ground-truth. The second set is composed of two calibrated stereo scenes selected from the previously acquired collection, specifically a statue of Ganesha and a handful of screws placed on a table. It should be noted however that Bundler did not find a feasible matching for many stereo pairs in the collection. Again, for all the sets of experiments we evaluated both the rotation error of all the cameras in terms of angular distance and axis discrepancy, and



**Dino sequence** **Temple sequence**

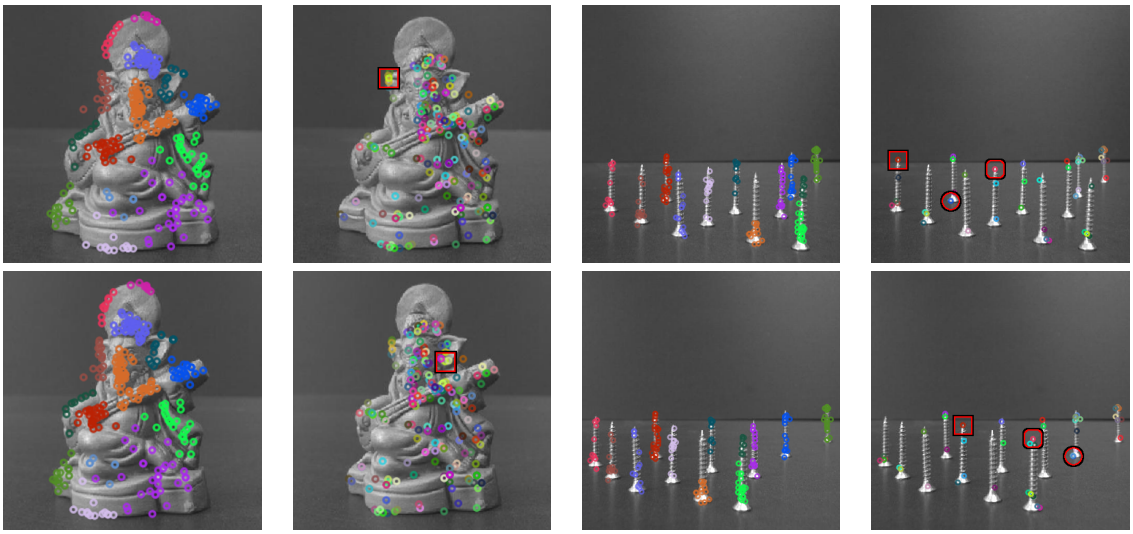
| | | Game-Theoretic | Bundler Keymatcher | Game-Theoretic | Bundler Keymatcher |
|---|---|---|---|---|---|
| Matches | | 14573 | 9245 | 25785 | 22317 |
| $\epsilon$ | $\leq 1$ pix | 24.83 | 6.49406 | 22.6049 | 24.6729 |
| | $\leq 5$ pix | 54.94 | 48.3659 | 62.7737 | 61.8957 |
| | $\geq 5$ pix | 20.21 | 45.1401 | 14.6214 | 13.4314 |
| | Avg. | 2.3086 | 4.5255 | 2.3577 | 2.3732 |
| $\Delta\alpha$ | Avg. | 0.005751 | 0.005561 | 0.010514 | 0.009376 |
| | S. dev. | 0.003242 | 0.003184 | 0.005282 | 0.004646 |
| | Max | 0.012057 | 0.011475 | 0.021527 | 0.017016 |
| $\Delta\gamma$ | Avg. | 0.008313 | 0.009561 | 0.014050 | 0.014079 |
| | S. dev. | 0.002948 | 0.006738 | 0.000511 | 0.000825 |
| | Max | 0.013449 | 0.030661 | 0.014692 | 0.015442 |
| Avg. levels | | 8.42 | - | 9.27 | - |

Figure 4. Results obtained with two multiple view data sets (image best viewed in color).

the reprojection error of the detected keypoints. The average number of matching groups is also given for the Game-Theoretic method.

The "Dino" model is a difficult case in general, as it embodies very few features; the upper part of Fig. 4 shows the correspondences produced by our method (left column) in comparison with the other matcher (right column). A set of optimal parameters detected in the previous experiments was used for configuring our matcher. This resulted as expected in the detection of many correct matches organized in groups, each corresponding to a different level of depth, and visualized with a unique color in the figure. As can be seen, different levels of depth are properly estimated; this is particularly evident throughout the arched back going from the tail (in foreground) to the head of the model (in background), where clustered sets of keypoints follow one after the other. Furthermore, these sets of interest points maintain the right correspondences within the pair of images. The Bundler keymatcher on the other hand, while still achieving good results in the whole process, also outputs erroneous correspondences (marked in the figure).

The quality of reconstruction following the application of both methods can be visually compared by looking at the distribution of the reprojection error in the left half of Fig. 6. While most reprojections fall within 1-3 pixels of distance for the Game-Theoretic approach, the Bundler keymatcher exhibits a long-tail trend, reaching an error spread of 20 pixels. Differently from "Dino", the "Temple" model is quite rich of features; for visualization purposes we only show a subset of the detected matches for both the techniques. While the effectiveness of our approach is not negatively impacted by the model characteristics, mismatches are revealed with Bundler. In particular, the symmetric parts of the object (mainly represented by the pillars) result in very similar features and this causes the matcher to establish one-to-many pairings over them. However, it should be noted that for both the "Dino" and "Temple" models the two matchers deliver comparably good results when fed with a



**Ganesha stereo** **Screws stereo**

| | | Game-Theoretic | Bundler Keymatcher | Game-Theoretic | Bundler Keymatcher |
|---|---|---|---|---|---|
| Matches | | 280 | 200 | 211 | 46 |
| $\epsilon$ | $\leq 1$ pix | 98.2824 | 20 | 0 | 0 |
| | $\leq 5$ pix | 1.7175 | 80 | 34.7716 | 6.75676 |
| | $\geq 5$ pix | 0 | 0 | 65.2284 | 93.2432 |
| | Avg. | 0.321248 | 1.67583 | 5.86237 | 10.2208 |
| $\Delta\alpha$ | | 0.001014 | 0.007424 | 0.020822 | 0.030995 |
| $\Delta\gamma$ | | 0.048076 | 0.078715 | 0.106485 | 0.117885 |
| Levels | | 14 | - | 12 | - |

Figure 5. Results obtained with two stereo view data sets (image best viewed in color).

whole set of views of the object.

In the calibrated stereo scenario, "Ganesha stereo" images are rich of distinctive features and should pose no difficulty to any of the methods. The Bundler keymatcher provides very good results, with only one evident false match out of a total of 200 matches (see Fig. 5). The resulting bundle adjustment is quite accurate, giving very small rotation errors and reprojection distances. Nevertheless, our method performs considerably better: reprojection errors dramatically decrease, with around 98 percent of the keypoints falling below one pixel of reprojection distance.

The second calibrated stereo scene, "Screws stereo", is an emblematic case and provides some meaningful insight. The images depict a dozen of screws standing on a table, placed by hand at different levels of depth. This configuration, together with the abundance of features in the objects themselves, should provide enough information for the two algorithms to extract significant matches. Indeed, the scene proves to be a difficult one due to the very nature of the objects depicted, which are all identical and highly symmetric, and diverse false matches are established by the Bundler keymatcher (see the last column of Fig. 5). This matching results nevertheless in a good estimation of the rigid transformation linking the two cameras, since erroneous pairings are removed *a posteriori* during the subsequent phases of bundle adjustment. By contrast, the Game-Theoretic approach outputs large and accurate sets of matches, roughly one per object, each corresponding to a level of depth; even moderately difficult cases, such as the left-right "swaps" due to the change of viewpoint taking place at the borders, are correctly dealt with. Again, a histogram of the reprojection error for this object is shown in Fig. 6.

Execution times for the matching steps of our technique are plotted in Fig. 7; the scatter plot shows a substantially linear growth of convergence time as the number of matching strategies increases, staying below half a second even with a large number of players.

## 4. Conclusions

In this paper we introduced a novel game-theoretic technique that performs an accurate feature matching between multiple views of the same subject as a preliminary step for bundle adjustment. Differently from other approaches, we do not rely on a first estimation of scene and camera param-

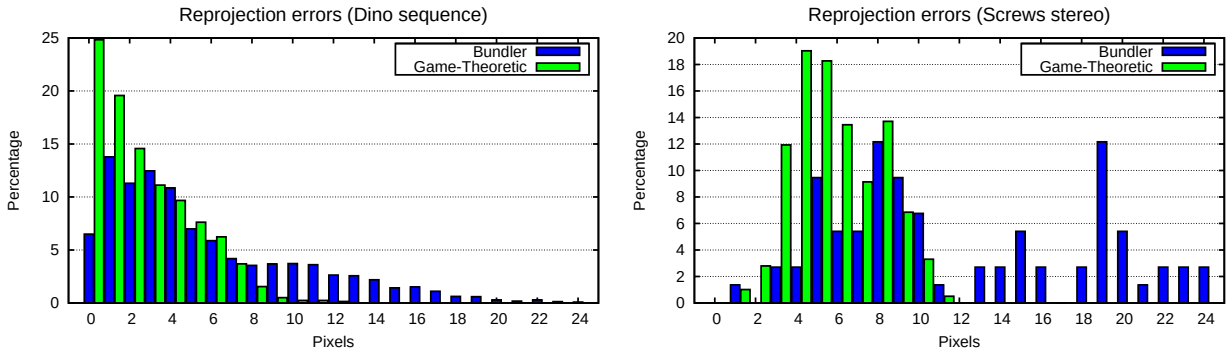



Figure 6. Distribution of the reprojection error on one multiple view (left) and one stereo pair (right) example.

eters in order to obtain a robust inlier selection. Rather, we enforce local compatibility of groups of features with respect to a common similarity transformation. By extracting one group at a time by means of an evolutive process, we are able to cover the entire subject. Experimental comparisons with a widely used technique show the ability of our approach to obtain a tighter inlier selection and thus a more accurate estimation of the scene parameters.

## Acknowledgments

We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.

## References

[1] A. Albarelli, S. Rota Bulò, A. Torsello, and M. Pelillo. Matching as a non-cooperative game. In *ICCV 2009: Proceedings of the 2009 IEEE International Conference on Computer Vision*. IEEE Computer Society, 2009. 2, 3

[2] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Fourth Alvey Vision Conference*, pages 147–151, 1988. 1

[3] T. T. Herbert Bay and L. V. Gool. Surf: Speeded up robust features. In *9th European Conference on Computer Vision*, volume 3951, pages 404–417, 2006. 1

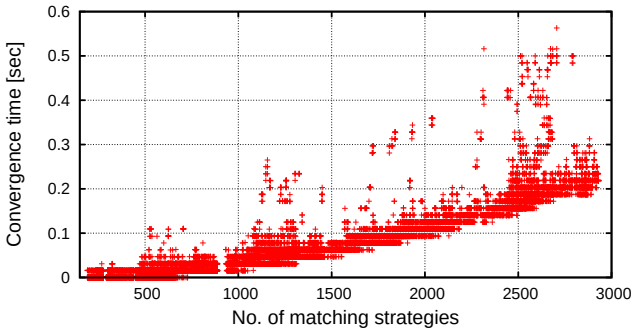



Figure 7. Plot of the convergence time of the replicator dynamics with respect to the number of matching strategies.

[4] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathmatics*, II(2):164–168, 1944. 1

[5] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003. 1

[6] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Conference on Computer Vision ICCV, Corfu*, pages 1150–1157, 1999. 1

[7] D. Marr and E. Hildreth. Theory of Edge Detection. *Royal Soc. of London Proc. Series B*, 207:187–217, Feb. 1980. 1

[8] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004. British Machine Vision Computing 2002. 1

[9] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I*, pages 128–142, London, UK, 2002. Springer-Verlag. 1

[10] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630, 2005. 1

[11] M. S. Sarfraz and O. Hellwich. Head pose estimation in face recognition across pose scenarios. In *VISAPP (1)*, pages 235–242, 2008. 1

[12] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR '06*, pages 519–528, Washington, USA, 2006. IEEE Computer Society. 5

[13] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, pages 835–846, New York, NY, USA, 2006. ACM. 2, 5

[14] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *Int. J. Comput. Vision*, 80(2):189–210, 2008. 1, 2, 5

[15] A. Torsello, S. Rota Bulò, and M. Pelillo. Grouping with asymmetric affinities: A game-theoretic perspective. In *CVPR '06*, pages 292–299, Washington, USA, 2006. IEEE Computer Society. 2

[16] J. Weibull. *Evolutionary Game Theory*. MIT P., 1995. 2, 4

# A Game-Theoretic Approach to Fine Surface Registration without Initial Motion Estimation

Andrea Albarelli, Emanuele Rodolà, and Andrea Torsello
Dipartimento di Informatica - Università Ca' Foscari - via Torino, 155 - 30172 Venice Italy
albarelli@unive.it rodola@dsi.unive.it torsello@dsi.unive.it

## Abstract

*Surface registration is a fundamental step in the reconstruction of three-dimensional objects. This is typically a two step process where an initial coarse motion estimation is followed by a refinement. Most coarse registration algorithms exploit some local point descriptor that is intrinsic to the shape and does not depend on the relative position of the surfaces. By contrast, refinement techniques iteratively minimize a distance function measured between pairs of selected neighboring points and are thus strongly dependent on initial alignment. In this paper we propose a novel technique that allows to obtain a fine surface registration in a single step, without the need of an initial motion estimation. The main idea of our approach is to cast the selection of correspondences between points on the surfaces in a game-theoretic framework, where a natural selection process allows mating points that satisfy a mutual rigidity constraint to thrive, eliminating all the other correspondences. This process yields a very robust inlier selection scheme that does not depend on any particular technique for selecting the initial strategies as it relies only on the global geometric compatibility between correspondences. The practical effectiveness of the proposed approach is confirmed by an extensive set of experiments and comparisons with state-of-the-art techniques.*

## 1. Introduction

The distinction between coarse and fine surface registration techniques is mainly related to the different strategies adopted to find pairs of mating points to be used for the estimation of the rigid transformation. Almost invariably, fine registration algorithms exploit an initial guess in order to constrain the search area for compatible mates and minimize the risk of selecting outliers. On the other hand, coarse techniques, which cannot rely on any motion estimation, must adopt a mating strategy based on the similarity between surface-point descriptors or resort to random selection schemes. The tension between the precision required for fine alignment versus the recall needed for an initial motion estimation stands as the main hurdle to the unification of such approaches.

The large majority of currently used fine alignment methods are modifications to the original ICP proposed by Zhang [23] and Besl and McKay [3]. These variants generally differ in the strategies used to sample points from the surfaces, reject incompatible pairs, or measure error. In general, the precision and convergence speed of these techniques is highly data-dependent and very sensitive to the fine-tuning of the model parameters. Several approaches that combine these variants have been proposed in the literature in order to overcome these limitations (see [16] for a comparative review). Some recent variants avoid hard culling by assigning a probability to each candidate pair by means of evolutionary techniques [14] or Expectation Maximization [10]. ICP variants, being iterative algorithms based on local, step-by-step decisions, are very susceptible to the presence of local minima. Other fine registration methods include the well-known method by Chen [6] and signed distance fields matching [15].

Coarse registration techniques can be roughly classified into methods that exploit some global property of the surface, such as PCA [8] or Algebraic Surface Model [18], and methods that use some 3D feature descriptor to find plausible candidates pairs over the model and data surfaces. Global techniques are generally very sensitive to occlusion. Feature-based approaches are more precise and can align surfaces that exhibit only partial overlap. Nevertheless, the unavoidable localization error of the feature points prevents them from obtaining accuracies on par with fine registration methods. Among the most successful descriptors are Point Signatures [7] and Spin Images [12]. A completely different coarse registration approach is the RANSAC-based DARCES [5], which is based on the random extraction of sets of mates from the surfaces and their validation based on the accuracy of the estimated transformation. Other recent methods include [1]; a recent and extensive review of all the different methods can be found in [17].

Regardless of the criteria used to obtain pairs of mating points, the subsequent step in surface registration is to search for the rigid transformation that minimizes the squared distance between them. Since many mature techniques are available to do this (for instance [11]), in this paper our effort is toward the matching step itself: specifically by proposing a novel game-theoretic approach that is able to deal equally well with both coarse and fine registration scenarios.

## 2. Game-Theoretic Surface Registration

We are looking for a robust set of inliers for correspondence selection from which we can estimate the rigid transformation. Most of the currently adopted matching schemes operate on a local level, and global information comes only as an afterthought by checking the quality of the candidate matches with respect to the registration error obtained. The approach we are proposing, on the other hand, brings global information into the matching process by favoring sets of point-associations that are mutually compatible with a single rigid transformation. Fundamental to our approach is the fact that requiring the compatibility to a single transformation is equivalent to requiring that there exists a compatible transformation for each pair of mates. Following [19, 2], we model the mating process in a game-theoretic framework, where two players extracted from a large population select a pair of corresponding points from two surfaces to be registered with one another. The player then receives a payoff from the other players proportional to how compatible his pairings are with respect to the other player's choice, where the compatibility derives from the existence of a common rigid transformation. More explicitly, if there exists a rigid transformation that moves both his point and the other player's point close to the corresponding mates, then both players receive a high payoff, otherwise the payoff will be low. Clearly, it is in each player's interest to pick correspondences that are compatible with the mates the other players are likely to choose. In general, as the game is repeated, players will adapt their behavior to prefer matings that yield larger payoffs, driving all inconsistent hypotheses to extinction, and settling for an equilibrium where the pool of mates from which the players are still actively selecting their associations forms a cohesive set with high mutual support. Within this formulation, the solutions of the matching problem correspond to evolutionary stable states (ESS's), a robust population-based generalization of the notion of a Nash equilibrium.

In a sense, this mating process can be seen as a contextual voting system, where each time the game is repeated the previous selections of the other players affect the future vote of each player in an attempt to reach consensus. This way the evolving context brings global information into the selection process.

### 2.1. Non-cooperative Games

Originated in the early 40's, Game Theory was an attempt to formalize a system characterized by the actions of entities with competing objectives, which is thus hard to characterize with a single objective function [21]. According to this view, the emphasis shifts from the search of a local optimum to the definition of equilibria between opposing forces. In this setting multiple players have at their disposal a set of strategies and their goal is to maximize a payoff that depends also on the strategies adopted by other players. Evolutionary game theory originated in the early 70's as an attempt to apply the principles and tools of game theory to biological contexts. Evolutionary game theory considers an idealized scenario where pairs of individuals are repeatedly drawn at random from a large population to play a two-player game. In contrast to traditional game-theoretic models, players are not supposed to behave rationally, but rather they act according to a pre-programmed behavior, or mixed strategy. It is supposed that some selection process operates over time on the distribution of behaviors favoring players that receive higher payoffs.

More formally, let $O = \{1, \cdots, n\}$ be the set of available strategies (*pure strategies* in the language of game theory), and $C = (c_{ij})$ be a matrix specifying the payoff that an individual playing strategy $i$ receives against someone playing strategy $j$. A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, \ldots, x_n)^T$ over the available strategies $O$. Clearly, mixed strategies are constrained to lie in the n-dimensional standard simplex

$$\Delta^n = \left\{ \mathbf{x} \in \mathbb{R}^n \ : \ x_i \geq 0 \text{ for all } i \in 1 \ldots n, \ \sum_{i=1}^{n} x_i = 1 \right\} .$$

The *support* of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, is defined as the set of elements chosen with non-zero probability: $\sigma(\mathbf{x}) = \{i \in O \mid x_i > 0\}$. The expected payoff received by a player choosing element $i$ when playing against a player adopting a mixed strategy $\mathbf{x}$ is $(C\mathbf{x})_i = \sum_j c_{ij} x_j$, hence the expected payoff received by adopting the mixed strategy $\mathbf{y}$ against $\mathbf{x}$ is $\mathbf{y}^T C \mathbf{x}$. The *best replies* against mixed strategy $\mathbf{x}$ is the set of mixed strategies

$$\beta(\mathbf{x}) = \{\mathbf{y} \in \Delta \mid \mathbf{y}^T C \mathbf{x} = \max_{\mathbf{z}} (\mathbf{z}^T C \mathbf{x})\} .$$

A strategy $\mathbf{x}$ is said to be a *Nash equilibrium* if it is the best reply to itself, i.e., $\forall \mathbf{y} \in \Delta, \ \mathbf{x}^T C \mathbf{x} \geq \mathbf{y}^T C \mathbf{x}$ . This implies that $\forall i \in \sigma(\mathbf{x})$ we have $(C\mathbf{x})_i = \mathbf{x}^T C \mathbf{x}$; that is, the payoff of every strategy in the support of $\mathbf{x}$ is constant.

A strategy $\mathbf{x}$ is said to be an *evolutionary stable strategy* (ESS) if it is a Nash equilibrium and

$$\forall \mathbf{y} \in \Delta \quad \mathbf{x}^T C \mathbf{x} = \mathbf{y}^T C \mathbf{x} \Rightarrow \mathbf{x}^T C \mathbf{y} > \mathbf{y}^T C \mathbf{y} . \qquad (1)$$

This condition guarantees that any deviation from the stable strategies does not pay.

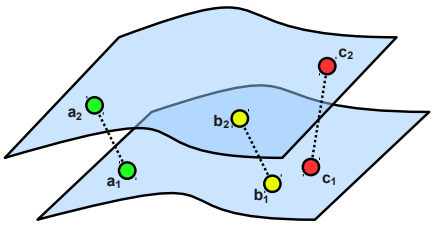


Figure 1. Example of mating strategies.

## 2.2. Mating Strategies and Payoffs

Central to this framework is the definition of a *mating game*, which implies the definition of the strategies available to the players and of the payoffs related to these strategies. Given a set of model points $M$ and a set of data points $D$ we call a *mating strategy* any pair $(a_1, a_2)$ with $a_1 \in M$ and $a_2 \in D$. We call the set of all the mating strategies $S$. In principle, all the model and data points could be used to build the mating strategies set, thus giving $S = M \times D$. In practice, however, we adopt some heuristics that allow us to obtain good alignments with a much smaller set. Once $S$ has been selected, our goal becomes to extract from it the largest subset that includes only correctly matched points: that is, strategies that associate a point in the model surface with the same point in the data surface. To enforce this we assign to each pair of mating strategies a payoff that is inversely proportional to a measure of violation of the rigidity constraint. This violation can be expressed in several ways, but since all the rigid transformations preserve Euclidean distances, we choose this property to express the coherence between mating strategies.

**Definition 1.** *Given a function* $\pi : S \times S \rightarrow \mathbb{R}^+$, *we call it a* rigidity-enforcing payoff function *if for any* $((a_1, a_2), (b_1, b_2))$ *and* $((c_1, c_2), (d_1, d_2)) \in S \times S$ *we have that* $||a_1 - b_1| - |a_2 - b_2|| > ||c_1 - d_1| - |c_2 - d_2||$ *implies* $\pi((a_1, a_2), (b_1, b_2)) < \pi((c_1, c_2), (d_1, d_2))$. *In addition, if* $\pi((a_1, a_2), (b_1, b_2)) = \pi((b_1, b_2), (a_1, a_2))$, $\pi$ *is said to be* symmetric.

A rigidity-enforcing payoff function is a function that is monotonically decreasing with the absolute difference of the Euclidean distances between respectively the model and data points of the mating strategies compared. In other words, given two mating strategies, their payoff should be high if the distance between the model points is equal to the distance between the data points and it should decrease as the difference between such distances increases. In the example of Figure 1, mating strategies $(a_1, a_2)$ and $(b_1, b_2)$ are coherent with respect to the rigidity constraint, whereas $(b_1, b_2)$ and $(c_1, c_2)$ are not, thus it is expected that $\pi((a_1, a_2), (b_1, b_2)) > \pi((b_1, b_2), (c_1, c_2))$.

Further, if we want mating to be one-to-one, we must put an additional constraint on the payoffs, namely that mates sharing a point are incompatible.

**Definition 2.** *A rigidity-enforcing payoff function* $\pi$ *is said to be* one-to-one *if* $a_1 = b_1$ *or* $a_2 = b_2$ *implies* $\pi((a_1, a_2), (b_1, b_2)) = 0$.

Given a set of mating strategies $S$ and an enumeration $O = \{1, ..., |S|\}$ over it, a *mating game* is a non-cooperative game where the population is defined as a vector $\mathbf{x} \in \Delta^{|S|}$ and the payoff matrix $C = (c_{ij})$ is defined as $c_{ij} = \pi(s_i, s_j)$, where $s_i, s_j \in S$ are enumerated by $O$ and $\pi$ is a symmetric one-to-one rigidity-enforcing payoff function. Intuitively, $\mathbf{x}_i$ accounts for the percentage of the population that plays the $i$-th mating strategy. By using a symmetric one-to-one payoff function in a mating game we are guaranteed that ESS's will not include mates sharing either model or data nodes. In fact, given a non-negative payoff function, a stable state cannot have in its support a pairs of strategies with payoff 0 [2]. Moreover, a mating game exhibits some additional interesting properties.

**Theorem 1.** *Given a set of model points* $M$, *a set of data points* $D = TM$ *that are exact rigid transformations of the points in* $M$, *and a set of mating strategies* $S \subseteq M \times D$ *with* $(m, Tm) \in S$ *for all* $m \in M$, *and a mating game over them with a payoff function* $\pi$, *the vector* $\hat{\mathbf{x}} \in \Delta^{|S|}$ *defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M; \\ 0 & \text{otherwise,} \end{cases}$$

*is an ESS and obtains the global maximum average payoff.*

*Sketch of proof.* Let $\hat{\mathbf{S}} \subseteq S$ be the set of mates that match a point to its copy, clearly for all $s, q \in \hat{\mathbf{S}}, s \neq q$ we have $\pi(s, q) = 1$, while for $s \in \hat{\mathbf{S}}$ and $q \in S \setminus \hat{\mathbf{S}}$, we have $\pi(s, q) < 1$. For all $s \in \hat{\mathbf{S}}$ we have that $\pi(\hat{\mathbf{x}}, \hat{\mathbf{x}}) = \frac{|M|-1}{|M|}$ while, since $\pi$ is one-to-one, for any $q \in S \setminus \hat{\mathbf{S}}$ there must be at least one $s_q \in \hat{\mathbf{S}}$ with $\pi(q, s_q) = 0$, thus $\pi(q, \hat{\mathbf{x}}) < \frac{|M|-1}{|M|}$, thus $\hat{\mathbf{x}}$ is a Nash equilibrium. Further, since the inequality is strict, it is an ESS. Finally, $\hat{\mathbf{x}}$ is a global maximizer of $\pi$ since $\frac{|M|-1}{|M|}$ is the maximum value that a one-to-one normalized payoff function over $|M|$ points can attain. □

This theorem states that when matching a surface with a rigidly transformed copy of itself the optimal solution (i.e., the population configuration that selects all the mating strategies assigning each point to its copy) is the stable state of maximum payoff. Since well established algorithms to evolve a population to such a state exist, this provides us with an effective mating approach. Clearly, aligning a surface to an identical copy is not very useful in practical scenarios, where occlusion and measurement noise come into play. While the quality of the solution in presence of noise will be assessed experimentally, we can give some theoretical results regarding occlusions.

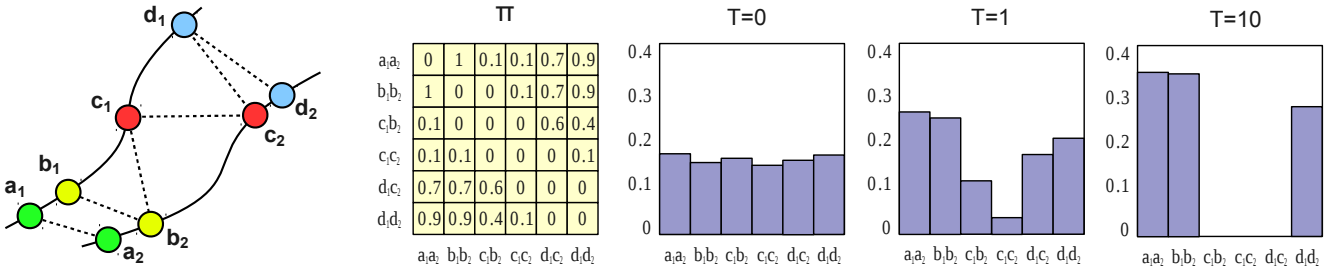



Figure 2. An example of the evolutionary process. Four points are sampled from the two surfaces and a total of six mating strategies are selected as initial hypotheses. The matrix $\Pi$ shows the compatibilities between pairs of mating strategies according to a one-to-one rigidity-enforcing payoff function. Each mating strategy got zero payoff with itself and with strategies that share the same source or destination point (i.e., $\pi((b_1, b_2), (c_1, b_2)) = 0$). Strategies that are coherent with respect to rigid transformation exhibit high payoff values (i.e., $\pi((a_1, a_2), (b_1, b_2)) = 1$ and $\pi((a_1, a_2), (d_1, d_2)) = 0.9$)), while less compatible pairs get lower scores (i.e., $\pi((a_1, a_2), (c_1, c_2)) = 0.1$). Initially (at T=0) the population is set to the barycenter of the simplex and slightly perturbed (3-5%). After just one iteration, $(c_1, b_2)$ and $(c_1, c_2)$ have lost a significant amount of support, while $(d_1, c_2)$ and $(d_1, d_2)$ are still played by a sizable amount of population. After ten iterations (T=10), $(d_1, d_2)$ has finally prevailed over $(d_1, c_2)$ (note that the two are mutually exclusive). Note that in the final population $((a_1, a_2), (b_1, b_2))$ have a larger support than $(d_1, d_2)$ since they are a little more coherent with respect to rigidity.

**Theorem 2.** *Let $M$ be a set of points with $M_a \subseteq M$ and $D = TM_b$ a rigid transformation of $M_b \subseteq M$ such that $|M_a \cap M_b| \geq 3$, and $S \subseteq M_a \times D$ be a set of mating strategies over $M_a$ and $D$ with $(m, Tm) \in S$ for all $m \in M_a \cap M_b$. Further, assume that the points that are not in the overlap, that is the points in $E_a = M_a \setminus (M_a \cap M_b)$ and $E_b = M_b \setminus (M_a \cap M_b)$, are sufficiently far away such that for every $s \in S, s = (m, Tm)$ with $m \in M_a \cap M_b$ and every $q \in S, q = (m_a, Tm_b)$ with $m_a \in E_a$ and $m_b \in E_b$, we have $\pi(q, s) < \frac{|M_a \cap M_b| - 1}{|M_a \cap M_b|}$, then, the vector $\hat{\mathbf{x}} \in \Delta^{|S|}$ defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M_a \cap M_b\,; \\ 0 & \text{otherwise,} \end{cases}$$

*is an ESS.*

*Sketch of proof.* We have $\pi(\hat{\mathbf{x}}, \hat{\mathbf{x}}) = \frac{|M_a \cap M_b| - 1}{|M_a \cap M_b|}$. Let $q \in S$ be a strategy not in the support of $\hat{\mathbf{x}}$, then, either it maps a point in $M_a$ or $M_b$, thus receiving payoff $\pi(q, \hat{\mathbf{x}}) < \frac{|M_a \cap M_b| - 1}{|M_a \cap M_b|}$ because of the one-to-one condition, or it maps a point in $E_a$ to a point in $E_b$, receiving, by hypothesis, a payoff $\pi(q, \hat{\mathbf{x}}) < \frac{|M_a \cap M_b| - 1}{|M_a \cap M_b|}$. Hence, $\hat{\mathbf{x}}$ is an ESS. □

The result of theorem 2 is slightly weaker than theorem 1, as the face of the simplex corresponding to the "correct" overlap, while being an evolutionary stable state, is not guaranteed to obtain the overall highest average payoff. This is not a limitation of the framework as this weakening is actually due to the very nature of the alignment problem itself. The inability to guarantee the maximality of the average payoff is due to the fact that the original object ($M$) could contain large areas outside the overlapping subset that are perfectly identical. Further, objects that are able to slide (for instance a plane or a sphere) could allow to move between different mixed strategies without penalty. These situations cannot be addressed by any algorithm without relying on supplementary information. However, in practice, they are quite unlikely, exceptional cases. In the experimental section we will show that our approach can effectively register even quasi-planar surfaces.

### 2.3. Building the Mating Strategies Set

From a theoretical point of view the total number of mating strategies in a registration problem is $|M \times D|$, which can be very large even with medium-sized surfaces. In practice, it is possible to apply several heuristics to select a lower number of candidates while still achieving good alignment results. Since the proposed approach is very selective it is not necessary to use all the model points: even a highly aggressive subsampling does not affect the registration quality, provided that some points in the overlapping region between model and data are retained. In fact, our approach does not try to find a good registration by means of a vote counting validation; instead it takes quite the opposite route, by self-validating the selected mixed strategy exploiting its internal coherence. Once the model points have been subsampled, the mating strategies set could be created by pairing each one of them with all the data points. Again, while this approach would work, it is somewhat wasteful since most of the mating strategies could be dropped on the basis of some local property of the surface surrounding the model and data point. For instance, the mean or Gaussian curvatures can be compared or some surface feature can be calculated in order to select only meaningful pairs. In the experimental section we will suggest an effective selection strategy. Once a proper set of mates has been chosen, a payoff function is needed. In principle, any proper one-to-one symmetric rigidity-enforcing payoff function could be used to capture the coherence between pairs of mating strategies. From a practical point of view it is often advisable to use bounded functions, usually in the interval $(0, 1]$. Very good candidates are the negative exponentiation of the difference

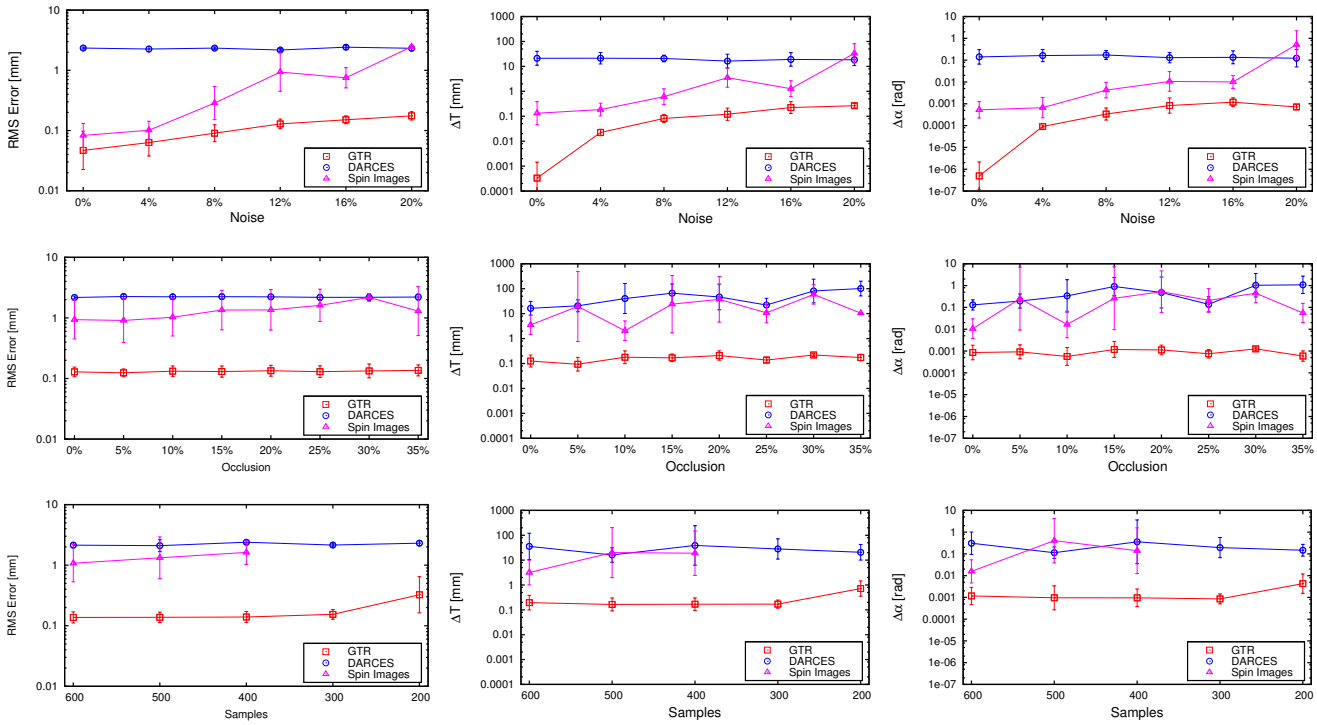



Figure 3. Comparison of coarse registration techniques using real range data, measuring ground RMS (column 1), translation (column 2) and rotation (column 3) errors as a function of noise (row 1), occlusion (row 2) and number of samples (row 3).

between the distances of the model and data points, or the ratio between the min and the max distance. In general, the steeper is the function, the more selective is the choice of the inlier mating strategies.

### 2.4. Evolving to an Optimal Solution

The search for a stable state is performed by simulating the evolution of a natural selection process. Under very loose conditions, any dynamics that respect the payoffs is guaranteed to converge to Nash equilibria [21] and (hopefully) to ESS's; for this reason, the choice of an actual selection process is not crucial and can be driven mostly by considerations of efficiency and simplicity. In this paper we chose to use the replicator dynamics, a well-known formalization of the selection process governed by the following equation

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t)\frac{(C\mathbf{x}(t))_i}{\mathbf{x}(t)^T C\mathbf{x}(t)} \qquad (2)$$

where $\mathbf{x}_i$ is the $i$-th element of the population and $C$ the payoff matrix. A simple but complete example of the evolution process is shown in Figure 2.

Once the population has reached a local maximum, all the non-extincted mating strategies can be used to calculate the rigid transformation between data and model surfaces. A clear advantage of our approach is that in the final mixed strategy each pair of points is weighted proportionally to its degree of participation in the equilibrium (see Figure 2). This is similar in spirit to the concept of compatibility between mates adopted by a number of fine registration algorithms, yet it does not depend at all on supplementary information such as surface normals or texture color. This compatibility can be used to weigh each pair when calculating the best surface alignment by using a weighted least squares fitting technique [11].

## 3. Experimental Results

Since the proposed technique can be used independently for coarse and fine registration, we evaluated its performance with respect to state-of-the-art algorithms of both fields. All the experiments have been executed on two sets of data: range images obtained from real-world scanners and synthetically-generated surfaces. For the first set of experiments we selected models from publicly available databases; specifically the Bunny [20], the Armadillo [13] and the Dragon [9] from the Stanford 3D scanning repository. To further assess the shortcomings of the various approaches, we used three synthetic surfaces representative of as many classes of objects: a wave surface, a fractal landscape and an incised plane (see Figure 4).

In all the experiments the set of mating strategies was obtained using the same selection technique. We used the MeshDOG [22] 3D feature detector to find interesting points in both the model and the data range images. A descriptor was associated to each point of interest; after considering both the MeshHOG and the Spin Image descrip-

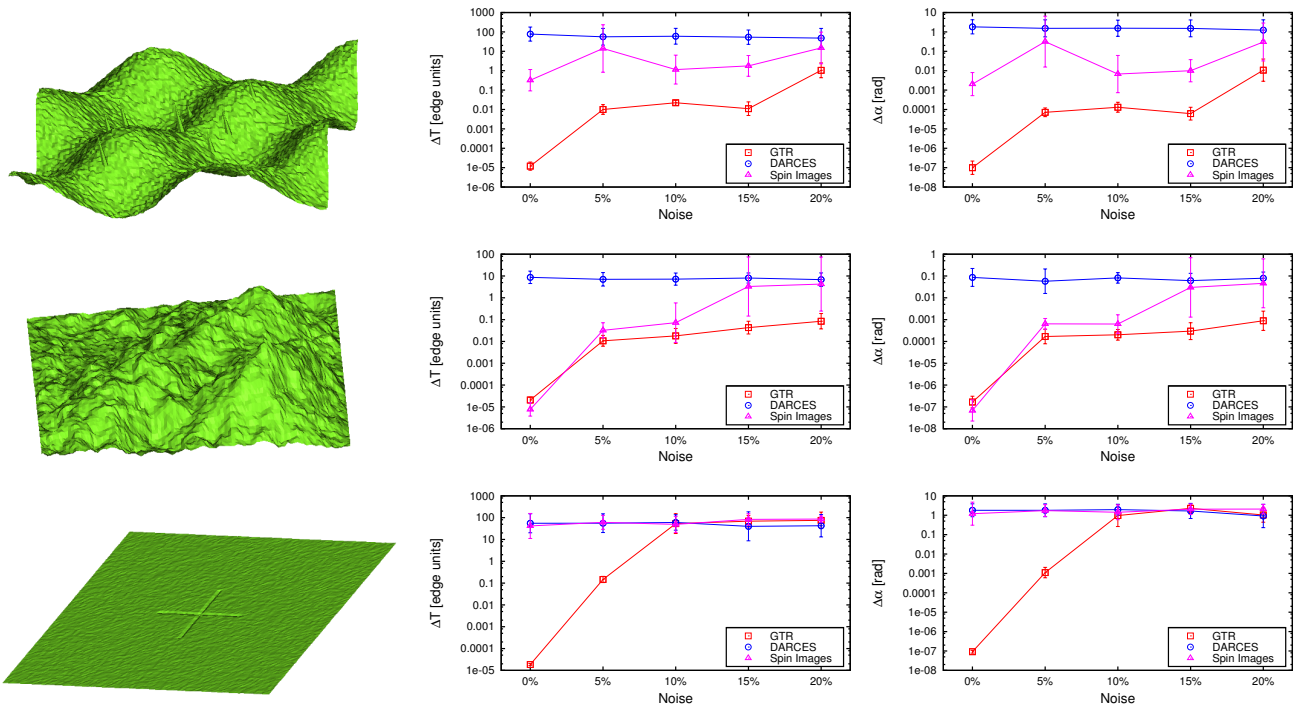



Figure 4. Comparison of coarse registration techniques using synthetic objects, measuring motion error as a function of noise.

tors, we preferred the latter as we found it to be more distinctive. Then a set of candidate source points was subsampled from the model and for each source point we created 5 mating strategies by connecting it to the 5 points with the most compatible descriptors. The rigidity-enforcing payoff function chosen was

$$\pi((a_1, b_1), (a_2, b_2)) = \frac{min(|a_1 - a_2|, |b_1 - b_2|)}{max(|a_1 - a_2|, |b_1 - b_2|)} \quad (3)$$

where $a_1$, $a_2$, $b_1$ and $b_2$ are respectively the two model (source) and data (destination) points in the compared mating strategies.

### 3.1. Coarse Registration

We compared our method with two coarse registration methods: RANSAC-based DARCES [5] and Spin Images [12]. DARCES has been implemented according to the original paper, while we used the Spin Images variant suggested in [4] to obtain a higher accuracy. In Figure 3 we show the results obtained using the set of surfaces from the Stanford repository. Each test was made under different conditions of noise, occlusion and subsampling and was run for a total of 12 times over the set of range images. For each set of experiments we plot the RMS distance for the actual point correspondences in the two meshes, and the estimation errors of the translation vector and rotation angle. In order to obtain a ground-truth for precise error measurement we generated the data points by adding Gaussian noise, random occlusion and motion to the model points. In these experiments the surfaces were obtained from laser scans of objects of hundreds of millimeters in size, with a resolution of about one tenth of millimeter. The first row of Figure 3 plots the sensitivity to Gaussian noise exhibited by the different techniques. The noise level is expressed as the ratio between the standard deviation of the noise and the average edge length. While DARCES is not very sensitive to noise, it delivers by far the worst overall results. By contrast, Spin Images give fairly good results at low noise levels, but their performance worsens quickly as noise is increased. The proposed approach (GTR), on the other hand, exhibits errors that are consistently an order of magnitude below Spin Images. In the second row we show the effect of occlusion under a constant level of Gaussian noise with standard deviation equal to 12% of the average edge length. The results show that the tested techniques are substantially insensitive to occlusions, our technique constantly outperforming the other approaches. Finally, the third row shows the effect of subsampling. Our game-theoretic method outperforms the other approaches. Note that the Spin Images based technique was never able to find a correct transformation when provided with less than 300 samples.

Figure 4 plots the alignment results on the three synthetic surfaces. Each set of experiments was conducted over a single type of surface (displayed at the beginning of the row) with 12 runs for each technique and noise level. Since these objects are synthetic, errors on translation are expressed

in edge units. The "wave" test object (first row) offers a regular surface with few outstanding features and high redundancy of the pattern; in this scenario the Spin Images technique is affected by the inability to discern among a large amount of similar descriptors, thus it performs poorly at all noise levels. Conversely, the geometric-based consensus exploited by our registration approach allows for a more precise selection and thus a more accurate registration. The "fractal landscape" test object (second row) is an irregular surface that allows to produce very distinctive feature descriptors. In fact, with low levels of noise both Spin Images and our technique perform very well, albeit as noise increases we achieve better results. Finally, the "incised plane" object (third row) is a big flat domain with a small cross just half an edge deep. This represents a very difficult target for most registration techniques, since very few and faint features are available, while a large planar surface dominates the landscape. Despite the lack of good detectable points, our technique is able to register the surface as long as noise is minimal. With higher noise levels the bumped cross fades and becomes almost indistinguishable from the plane itself. Note that DARCES achieves mediocre results under all tested conditions.

### 3.2. Fine Registration

The performance of our approach with respect to fine registration has been studied in a separate batch of experiments. The goal of this test is two-fold: we want to evaluate our quality as a complete alignment tool and, at the same time, find the breaking point for traditional fine registration techniques. The method we used for comparison is a best-of-breed ICP variant, similar to the one proposed in [20]. Point selection is based on Normal Space Sampling [16], and point-surface normal shooting is adopted for finding correspondences; distant mates or candidates with back-facing normals are rejected. To minimize the influence of incorrect normal estimates, matings established on the boundary of the mesh are also removed. The resulting pairings are weighted with a coefficient based on compatibility of normals, and finally a 5%-trimming is used. Each test was performed by applying a random rotation and translation to different range images selected from the Stanford 3D scanning repository. Additionally, each range image was perturbed with a constant level of Gaussian noise with standard deviation equal to 12% of the average edge length. We completed 100 independent tests and for each of them we measured the initial RMS error between the ground-truth corresponding points and the resulting error after performing a full round of ICP (ICP) and a single run of our registration method (GTR). In addition, we applied a step of ICP to the registration obtained with our method (GTR + ICP) in order to assess how much the solution extracted using our approach was further refinable. A scatter plot of the obtained errors before and after registration is shown in Figure 6. We observe that ICP reaches its breaking point quite early; in fact with an initial error above the threshold of about 20mm it is unable to find a correct registration. By contrast, GTR is able to obtain excellent alignment regardless of the initial motion perturbation. Finally, applying ICP to GTR decreases the RMS only by a very small amount.

While we did not carry out any formal benchmark of the execution time required by our technique, we always observed a very fast convergence of the replicator dynamics, even with several thousands of mating strategies. In the

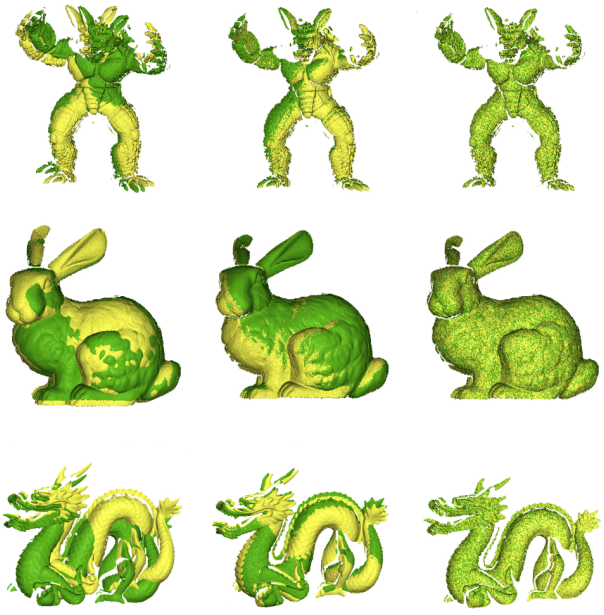


Figure 5. Examples of surface registration obtained respectively with RANSAC-based DARCES (first column), Spin Images (second column), and our game-theoretic registration technique (third column).

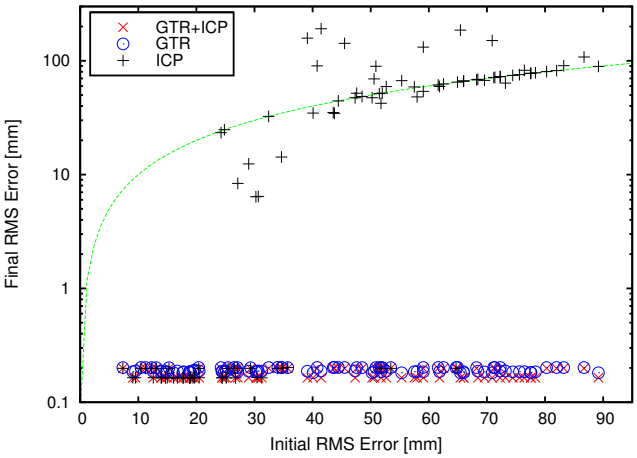



Figure 6. Comparison of fine registration accuracies (the green dashed line represents y=x). Graph best viewed in color.

worst scenarios our unoptimized C++ implementation[1] of the framework required less than 2 seconds (on a typical desktop PC) to evolve a population of 4000 to a stable state.

## 4. Conclusions

In this paper we introduced a novel game-theoretic technique that solves both the coarse and fine surface registration problems at once. Our approach has several advantages over the state-of-the-art: it does not require any kind of initial motion estimation, as it does not rely on spatial relationships between model and data points, it does not need any threshold as it produces a continuous compatibility weight on the selected point matches that can be used directly for alignment estimation, and, differently from most inlier selection techniques, it is not affected by a large number of outliers since it operates an explicit search for good inliers rather than using random selection or vote counting for validation. From a theoretical point of view, a sound correspondence between optimal alignments and evolutionary equilibria has been presented and a wide range of experiments validated both the robustness of the approach with respect to noise and its performance in comparison with other well-known techniques.

## Acknowledgments

We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.

## References

[1] D. Aiger, N. J. Mitra, and D. Cohen-Or. 4-points congruent sets for robust surface registration. *ACM Transactions on Graphics*, 27(3):#85, 1–10, 2008. 1

[2] A. Albarelli, S. R. Bulò, A. Torsello, and M. Pelillo. Matching as a non-cooperative game. In *ICCV 2009: Proc. of the 2009 IEEE Intl. Conf. on Comput. Vis.* IEEE Computer Society, 2009. 2, 3

[3] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992. 1

[4] O. T. Carmichael, D. F. Huber, and M. Hebert. Large data sets and confusing scenes in 3-d surface matching and recognition. In *3DIM*, pages 358–367, 1999. 6

[5] C.-S. Chen, Y.-P. Hung, and J.-B. Cheng. Ransac-based darces: A new approach to fast automatic registration of partially overlapping range images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(11):1229–1234, 1999. 1, 6

[6] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *Proc. 1991 IEEE Intl. Conf. on Robotics and Automat.*, pages 2724–2729 vol.3, 1991. 1

[7] C. S. Chua and R. Jarvis. Point signatures: A new representation for 3d object recognition. *Intl. J. of Comput. Vis.*, 25(1):63–85, October 1997. 1

[8] D. H. Chung, I. D. Yun, and S. U. Lee. Registration of multiple-range views using the reverse-calibration technique. *Pattern Recognition*, 31(4):457–464, 1998. 1

[9] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proc. of SIGGRAPH 96*, pages 303–312, New York, NY, USA, 1996. ACM. 5

[10] S. Granger, X. Pennec, and A. Roche. Rigid point-surface registration using an em variant of icp for computer guided oral implantology. In *Proc. of the 4th Intl. Conf. on Medical Image Comput. and Computer-Assisted Interv.*, pages 752–761, London, UK, 2001. Springer-Verlag. 1

[11] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *J. of the Optical Society of America. A*, 4(4):629–642, Apr 1987. 2, 5

[12] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(5):433–449, 1999. 1, 6

[13] V. Krishnamurthy and M. Levoy. Fitting smooth surfaces to dense polygon meshes. In *Proc. of SIGGRAPH 96*, pages 313–324, 1996. 5

[14] Y. Liu. Replicator dynamics in the iterative process for accurate range image matching. *Intl. J. Comput. Vis.*, 83(1):30–56, 2009. 1

[15] T. Masuda. Registration and integration of multiple range images by matching signed distance fields for object shape modeling. *Comput. Vis. Image Underst.*, 87(1-3):51–65, 2002. 1

[16] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proc. of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pages 145–152, 2001. 1, 7

[17] J. Salvi, C. Matabosch, D. Fofi, and J. Forest. A review of recent range image registration methods with accuracy evaluation. *Image Vis. Comput.*, 25(5):578–596, 2007. 1

[18] J.-P. Tarel, H. Civi, and D. B. Cooper. Pose estimation of free-form 3d objects without point matching using algebraic surface models. In *Proc. of IEEE Workshop Model Based 3D Image Analysis*, pages 13–21, Mumbai, India, 1998. 1

[19] A. Torsello, S. R. Bulò, and M. Pelillo. Grouping with asymmetric affinities: A game-theoretic perspective. In *CVPR '06: Proc. of the 2006 IEEE Conf. on Comput. Vis. and Pattern Recognit.*, pages 292–299, Washington, DC, USA, 2006. IEEE Computer Society. 2

[20] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *Proc. of SIGGRAPH 94*, pages 311–318, New York, NY, USA, 1994. ACM. 5, 7

[21] J. Weibull. *Evolutionary Game Theory*. MIT Press, 1995. 2, 5

[22] A. Zaharescu, E. Boyer, K. Varanasi, and R. P. Horaud. Surface feature detection and description with applications to mesh matching. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recognit.*, Miami Beach, Florida, June 2009. 5

[23] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *Intl. J. Comput. Vis.*, 13(2):119–152, 1994. 1

[1] A reference implementation of the framework will be made available for downloading at http://www.dsi.unive.it/~rodola/sw.html

# Loosely Distinctive Features for Robust Surface Alignment

Andrea Albarelli, Emanuele Rodolà and Andrea Torsello

Dipartimento di Informatica - Università Ca' Foscari di Venezia

**Abstract.** Many successful feature detectors and descriptors exist for 2D intensity images. However, obtaining the same effectiveness in the domain of 3D objects has proven to be a more elusive goal. In fact, the smoothness often found in surfaces and the lack of texture information on the range images produced by conventional 3D scanners hinder both the localization of interesting points and the distinctiveness of their characterization in terms of descriptors. To overcome these limitations several approaches have been suggested, ranging from the simple enlargement of the area over which the descriptors are computed to the reliance on external texture information. In this paper we offer a change in perspective, where a game-theoretic matching technique that exploits global geometric consistency allows to obtain an extremely robust surface registration even when coupled with simple surface features exhibiting very low distinctiveness. In order to assess the performance of the whole approach we compare it with state-of-the-art alignment pipelines. Furthermore, we show that using the novel feature points with well-known alternative non-global matching techniques leads to poorer results.

## 1 Introduction

Feature detection and characterization is a key step in many tasks involving the recognition, registration or database search of 2D and 3D data. Specifically, when suitable interest points are available, all these problems can be tackled by working with the set of extracted features, rather than dealing with the less stable and noisier information carried by the whole data. Of course, for an interest point to be reliable it must exhibit two properties: repeatability and distinctiveness. A feature is highly repeatable if it can be detected with good positional accuracy over a wide range of noise levels and sampling conditions as well as different scales and transformations of the data itself. Further, description vectors calculated over interesting points are said to be distinctive if they are well apart when related to different features, yet coherent when associated to multiple instances of the same point. These properties are somewhat difficult to attain since they are subject to antithetical goals. In fact, to achieve good repeatability despite of noise, larger patches of data must be considered. Unfortunately this leads to a lower positional precision and a less sharp culling of uninteresting points. Moreover, for descriptor vectors to be distinctive among different features, they

need to adopt a large enough basis, which, owing to the well known "dimensionality curse", also affects their coherence over perturbed versions of the same feature. In the last two decades these quandaries have been addressed with great success in the domain of 2D images where salient points are localized with subpixel accuracy by detectors exploiting strong local variation in intensity, such as Harris Operator [1] and Difference of Gaussians [2], or by using techniques that are able to locate affine invariant regions, such as Maximally stable extremal regions (MSER) [3] and Hessian-Affine [4]. Among the most used descriptors are the Scale-invariant feature transform (SIFT) [5], the Speeded Up Robust Features (SURF) [6] and Gradient Location and Orientation Histogram (GLOH) [7]. While these approaches work well with 2D intensity images, they cannot be easily extended to handle 3D surfaces since no intensity information is directly available. Of course several efforts have been made to use other local measures, such as curvature or normals. One of the first descriptor to capture the structural neighborhood of a surface point was described by Chua and Jarvis that with their Point Signatures [8] suggest both a rotation and translation invariant descriptor and a matching technique. Later, Johnson and Hebert introduced Spin Images [9], a rich characterization obtained by a binning of the radial and planar distances of the surface samples respectively from the feature point and from the plane fitting its neighborhood. Given their ability to perform well with both surface registration and object recognition, Spin Images have become one of the most used 3D descriptors. More recently, Pottmann et al. proposed the use of Integral Invariants [10], stable multi-scale geometric measures related to the curvature of the surface and the properties of its intersection with spheres centered on the feature point. Finally, Zaharescu et al. [11] presented a comprehensive approach for interest point detection (MeshDOG) and description (MeshHOG), based on the value of any scalar function defined over the surface (i.e. curvature or texture, if available). MeshDOG localizes feature points by searching for scale-space extrema over progressive Gaussian convolutions of the scalar function and thus by applying proper thresholding and corner detection. MeshHOG calculates a histogram descriptor by binning gradient vectors with respect to a rotational invariant local coordinate system.

In this paper we introduce a novel technique to detect and describe 3D interest points and to use them for robust surface registration. Unlike previous approaches we do not aim to obtain a very distinctive characterization. Instead, we settle for very simple descriptors, named *Surface Hashes*, that span only 3 to 5 dimensions. As their name suggests, we expect Surface Hashes to be repeatable through the same feature point, yet to suffer a high level of clashing due to their limited distinctiveness. In order to overcome this liability we avoid the use of classical RANSAC-based matchers; rather we adopt a robust game-theoretic inlier selector which exploits rigidity constraints among surfaces to guarantee a global geometric consistency. The combination of these loosely distinctive features and our robust matcher leads to an effective surface alignment approach. In the experimental section we point out this symbiosis by showing that standard matching techniques are not able to make the most of our descriptors.

## 2 Game-Theoretic Matching

Before describing in detail the Surface Hashes features we need to introduce some basic concepts about Evolutionary Game Theory and to present the idea of a Matching Game, originally presented in [12] and exploited by our technique both as an inlier selector and a robust matcher.

Evolutionary Game Theory [13] considers an idealized scenario where pairs of individuals are repeatedly drawn at random from a large population to play a two-player game. Each player obtains a payoff that depends only on the strategies played by him and its opponent. Players are not supposed to behave rationally, but rather they act according to a pre-programmed behavior, or mixed strategy. It is supposed that some selection process operates over time on the distribution of behaviors favoring players that receive larger payoffs. More formally, let $S = \{1, \cdots, n\}$ be the set of available strategies (*pure strategies* in the language of game theory) and $C = (c_{ij})$ be a matrix specifying the payoff that an individual playing strategy $i$ receives against someone playing strategy $j$. A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, \ldots, x_n)^T$ over the available strategies $S$; being probability distributions, mixed strategies lie in the n-dimensional standard simplex $\Delta^n = \{\mathbf{x} \in \mathbb{R}^n \ : \forall i \in 1 \ldots n \ x_i \geq 0, \ \sum_{i=1}^n x_i = 1\}$. The *support* of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, is defined as the set of elements chosen with non-zero probability: $\sigma(\mathbf{x}) = \{i \in S \mid x_i > 0\}$. The expected payoff received by a player choosing element $i$ when playing against a player adopting a mixed strategy $\mathbf{x}$ is $(C\mathbf{x})_i = \sum_j c_{ij} x_j$, hence the expected payoff received by adopting the mixed strategy $\mathbf{y}$ against $\mathbf{x}$ is $\mathbf{y}^T C\mathbf{x}$. The *best replies* against mixed strategy $\mathbf{x}$ is the set of mixed strategies

$$\beta(\mathbf{x}) = \{\mathbf{y} \in \Delta \mid \mathbf{y}^T C\mathbf{x} = \max_{\mathbf{z}}(\mathbf{z}^T C\mathbf{x})\}.$$

A strategy $\mathbf{x}$ is said to be a *Nash equilibrium* if it is the best reply to itself, i.e., $\forall \mathbf{y} \in \Delta, \ \mathbf{x}^T C\mathbf{x} \geq \mathbf{y}^T C\mathbf{x}$. This implies that $\forall i \in \sigma(\mathbf{x})$ we have $(C\mathbf{x})_i = \mathbf{x}^T C\mathbf{x}$ that is, the payoff of every strategy in the support of $\mathbf{x}$ is constant. A strategy $\mathbf{x}$ is said to be an *evolutionary stable strategy* (ESS) if it is a Nash equilibrium and $\forall \mathbf{y} \in \Delta \quad \mathbf{x}^T C\mathbf{x} = \mathbf{y}^T C\mathbf{x} \Rightarrow \mathbf{x}^T C\mathbf{y} > \mathbf{y}^T C\mathbf{y}$. This condition guarantees that any deviation from the stable strategies does not pay. The search for a stable state is performed by simulating the evolution of a natural selection process. Under very loose conditions, any dynamics that respect the payoffs is guaranteed to converge to Nash equilibria [13] and (hopefully) to ESS's; for this reason, the choice of an actual selection process is not crucial and can be driven mostly by considerations of efficiency and simplicity. We chose to use the replicator dynamics, a well-known formalization of the selection process governed by the following equation

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) \frac{(C\mathbf{x}(t))_i}{\mathbf{x}(t)^T C\mathbf{x}(t)}$$

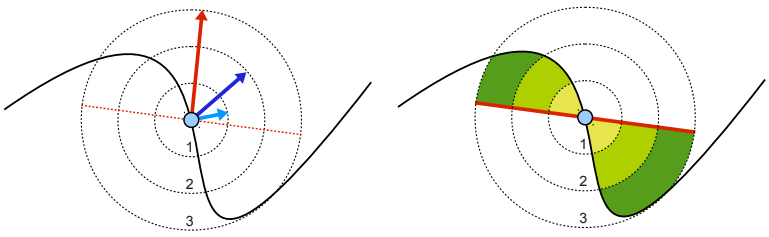
where $\mathbf{x}_i$ is the $i$-th element of the population and $C$ the payoff matrix.

Once the population has reached a local maximum, all the non-extinct pure strategies (i.e. $\sigma(\mathbf{x})$) can be considered selected by the process.

Following [12] and [14], we define a *Matching Game* as a non-cooperative game where the set of strategies $S$ is a subset of all the possible correspondences, and the payoff $c_{ij}$ between two strategies is proportional to some notion of compatibility between correspondences. By using different sets to be matched and alternative payoff functions, we are able to define games specially crafted to solve specific problems. In the following section we will define more formally two Matching Games. Respectively the first game will be dedicated to the localization of interest points over a surface described by Surface Hashes, while the second one will address the search for reliable correspondences between feature points extracted from two different meshes.

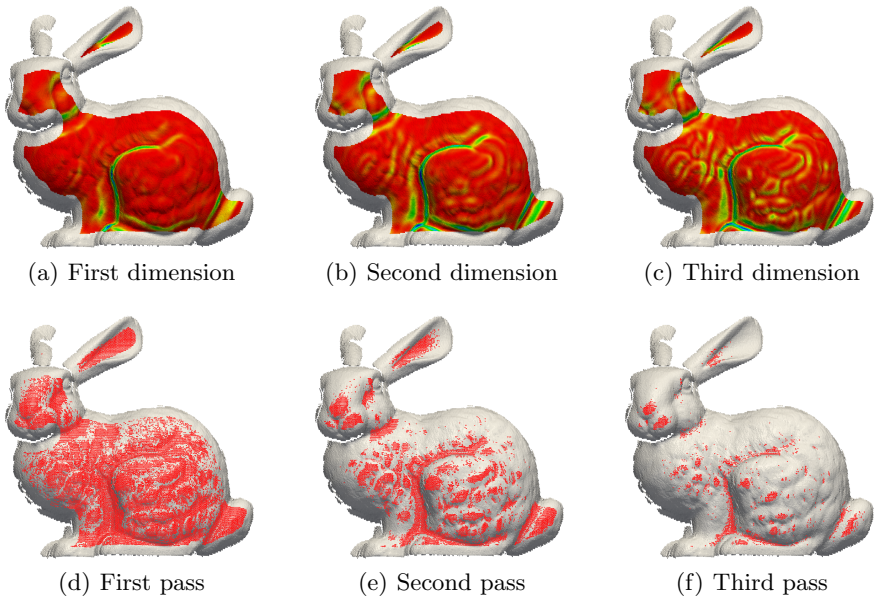
## 3 Surface Hashes

Intuitively, a Surface Hash is a concise point feature descriptor which exhibit the property of being highly repeatable at the cost of a relatively high probability of clashing. In practice this happens with any low-dimensional descriptor, such as the Gaussian or Mean Curvature (1 dimension), the first two Principal Components of a patch (2 dimensions), or the normal vector associated to a point (2 dimensions). While those descriptors could be used with our registration pipeline, we prefer to introduce some multiscale Surface Hashes based respectively on the dot product between normals and a local surface integral. Each of our descriptors corresponds to a vector of scalar measures evaluated at different scales. By increasing or reducing the number of scales, we are able to obtain vectors of different length, thus being more or less distinctive. The *Normal Hash* (see Fig. 1(a)) is obtained by setting as a reference the average surface normal over a patch that extends to the largest scale (red arrow in figure) and then, for each smaller scale, calculate the dot product between the reference and the average normal over the reduced patches (blue arrows in figure). This measure finds its rationale in the observation that at the largest scale the average normal is more stable with respect to noise and that the dot product offers a concise representation of the relation between the vectors obtained at various scales. The *Integral Hash* (see Fig. 1(b)) is similar in spirit to the Normal Hash. In this case we search for the best fitting plane (in the least squares sense) with respect




(a) Normal Hash (b) Integral Hash

**Fig. 1.** Example of the two basic Surface Hashes proposed in this paper



(a) First dimension (b) Second dimension (c) Third dimension

(d) First pass (e) Second pass (f) Third pass

**Fig. 2.** Example of a 3-dimensional Normal Hash and the related detection process

to the surface patch associated to the largest scale. We calculate the volume enclosed between the surface and such a plane. In practice, it is not necessary to evaluate this volume accurately: even naive approximations, such as the sum of the distances of the surface points from the plane, have shown to provide a reasonable approximation in all the empirical tests. Note that Normal Hashes evaluated over $n$ scales yield descriptor vectors of length $n-1$ (since the larger scale is used only to calculate the reference normal), while Integral Hashes provide $n$-dimensional vectors. In Fig. 2 a Normal Hash of dimension 3 (respectively from (a) to (c)) evaluated over 4 scales is shown. Note that the descriptor is not defined on the points for which the larger support is not fully contained in the surface, i.e., points close to the surface boundary.

### 3.1 Interest Points Detection

Given the large number of points contained in typical 3D objects, it is not practical for any matching algorithm to deal with all of them. In addition, the isolation of a relatively small number of interest points can enhance dramatically the ability of the matcher to avoid false correspondences, usually due to a large number of features with very common characterizations. This is particularly true when using Surface Hashes, which are loosely distinctive by design. Paradoxically, we use exactly this property to screen out features exhibiting descriptors that are too common over the surface. This happens by defining a Matching Game where the strategy set $S$ corresponds to the set of all the surface points and the payoff

matrix is defined by:

$$C(ij) = e^{-\alpha|d_i - d_j|}, \tag{1}$$

where $d_i$ and $d_j$ are the descriptor vectors associated to surface point $i$ and $j$, and $\alpha$ is a parameter that controls the level of selectivity. Clearly, features that are similar in terms of Surface Hashes will get a large mutual payoff and thus are more likely to be selected by the evolutive process. In this sense, our goal is to let the population evolve to an ESS and then remove from the set of interest points the features that survived the evolutive process. At the beginning we can initialize the set of retained features to the whole surface and run a sequence of Matching Games until the desired number of points are left. At this point, the remaining features are those characterized by less-common descriptors which are more likely to represent good cues for the matching. It should be noted that by choosing high values for $\alpha$ the payoff function decreases more rapidly with the growth of the distance between the Surface Hashes, thus the Matching Game becomes more selective and less points survive after reaching an ESS. In the end this results in a blander decimation and thus in a larger ratio of retained interest points. By converse, a low value for $\alpha$ leads to a more greedy filtering and thus to a more selective interest point detector. In Fig. 2 (from (d) to (f)) we show three steps of the evolutive interest point selection with respect to the 3-dimensional Normal Hash shown from (a) to (c). In Fig. 2(d) we see that after a single pass of the Matching Game most of the surface points are still considered interesting, while after respectively two and three passes only very distinctive points (belonging to areas with less common curvatures) are left.

### 3.2 Matching Surface Hashes

After obtaining a reduced set of interest points from the two surfaces, we could proceed to align them using some robust algorithm such as a basic RANSAC [15], that would use just the point locations and some initial match hypotheses, or PROSAC [16], that could better exploit the prior expressed by the descriptors. Unfortunately, Surface Hashes, despite the proposed filtering technique, are still not distinctive enough to be used directly by such methods. For this reason we define another Matching Game that ignores the information given by the descriptors and takes advantage of the rigidity constraint to be enforced in the surface registration problem. While this can sound counterintuitive, the main idea of this approach is to limit the use of the weak features to the selection of interest points and to use a more reliable global approach (that does not depend on descriptors) for the registration itself.

Given a set of model interest points $M$ and a set of data interest points $D$ we define the set of strategies for our Matching Game as all the possible correspondences between them: $S = \{(a_1, a_2) | a_1 \in M \text{ and } a_2 \in D\}$. Of course for practical reasons it is perfectly reasonable to limit the size of $S$ by including only pairs that show similar descriptors.

Once $S$ has been selected, our goal becomes to extract from it the largest subset that includes only correctly matched points: that is, strategies that associate a point in the model surface with the same point in the data surface.

To enforce this we assign to each pair of strategies a payoff that is inversely proportional to a measure of violation of the rigidity constraint. This violation can be expressed in several ways, but since all the rigid transformations preserve Euclidean distances, we choose this property to express the coherence between strategies.

**Definition 1.** *Given a function* $\pi : S \times S \to \mathbb{R}^+$*, we call it a* rigidity-enforcing payoff function *if for any* $((a_1, a_2), (b_1, b_2))$ *and* $((c_1, c_2), (d_1, d_2)) \in S \times S$ *we have that* $||a_1 - b_1| - |a_2 - b_2|| > ||c_1 - d_1| - |c_2 - d_2||$ *implies* $\pi((a_1, a_2), (b_1, b_2)) < \pi((c_1, c_2), (d_1, d_2))$*. In addition, if* $\pi((a_1, a_2), (b_1, b_2)) = \pi((b_1, b_2), (a_1, a_2))$*,* $\pi$ *is said to be* symmetric*.*

A rigidity-enforcing payoff function is a function that is monotonically decreasing with the absolute difference of the Euclidean distances between respectively the model and data points of the strategies compared. In other words, given two strategies, their payoff should be high if the distance between the model points is equal to the distance between the data points and it should decrease as the difference between such distances increases.

Further, if we want matching to be one-to-one, we must put an additional constraint on the payoffs, namely that mates sharing a point are incompatible.

**Definition 2.** *A rigidity-enforcing payoff function* $\pi$ *is said to be* one-to-one *if* $a_1 = b_1$ *or* $a_2 = b_2$ *implies* $\pi((a_1, a_2), (b_1, b_2)) = 0$*.*

Given a set of strategies $S$ and an enumeration $O = \{1, ..., |S|\}$ over it, a *mating game* is a non-cooperative game where the population is defined as a vector $\mathbf{x} \in \Delta^{|S|}$ and the payoff matrix $C = (c_{ij})$ is defined as $c_{ij} = \pi(s_i, s_j)$, where $s_i, s_j \in S$ are enumerated by $O$ and $\pi$ is a symmetric one-to-one rigidity-enforcing payoff function. Intuitively, $\mathbf{x}_i$ accounts for the percentage of the population that plays the $i$-th strategy. By using a symmetric one-to-one payoff function in a mating game we are guaranteed that ESS's will not include mates sharing either model or data nodes (see [12]). Moreover, a mating game exhibits some additional interesting properties.

**Theorem 1.** *Given a set of model points* $M$*, a set of data points* $D = TM$ *that are exact rigid transformations of the points in* $M$*, and a set of strategies* $S \subseteq M \times D$ *with* $(m, Tm) \in S$ *for all* $m \in M$*, and a mating game over them with a payoff function* $\pi$*, the vector* $\hat{\mathbf{x}} \in \Delta^{|S|}$ *defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M\,; \\ 0 & \text{otherwise,} \end{cases}$$

*is an ESS and obtains the global maximum average payoff.*

This theorem states that when matching a surface with a rigidly transformed copy of itself the optimal solution (i.e., the population configuration that selects all the strategies assigning each point to its copy) is the stable state of maximum

payoff. Clearly, aligning a surface to an identical copy is not very useful in practical scenarios, where occlusion and measurement noise come into play. While the quality of the solution in presence of noise will be assessed experimentally, we can give some theoretical results regarding occlusions.

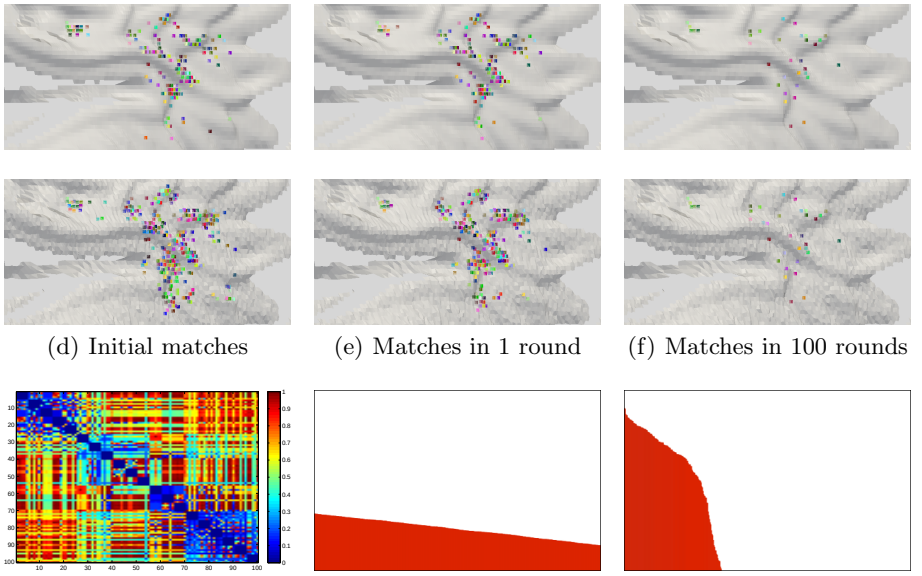
**Theorem 2.** *Let $M$ be a set of points with $M_a \subseteq M$ and $D = TM_b$ a rigid transformation of $M_b \subseteq M$ such that $|M_a \cap M_b| \geq 3$, and $S \subseteq M_a \times D$ be a set of strategies over $M_a$ and $D$ with $(m, Tm) \in S$ for all $m \in M_a \cap M_b$. Further, assume that the points that are not in the overlap, that is the points in $E_a = M_a \backslash (M_a \cap M_b)$ and $E_b = M_b \backslash (M_a \cap M_b)$, are sufficiently far away such that for every $s \in S, s = (m, Tm)$ with $m \in M_a \cap M_b$ and every $q \in S, q = (m_a, Tm_b)$ with $m_a \in E_a$ and $m_b \in E_b$, we have $\pi(q, s) < \frac{|M_a \cap M_b| - 1}{|M_a \cap M_b|}$, then, the vector $\hat{\mathbf{x}} \in \Delta^{|S|}$ defined as*

$$\hat{\mathbf{x}}_i = \begin{cases} 1/|M| & \text{if } s_i = (m, Tm) \text{ for some } m \in M_a \cap M_b\,; \\ 0 & \text{otherwise,} \end{cases}$$

*is an ESS.*

The result of theorem 2 is slightly weaker than theorem 1, as the face of the simplex corresponding to the "correct" overlap, while being an evolutionary stable state, is not guaranteed to obtain the overall highest average payoff. This is not a limitation of the framework as this weakening is actually due to the very nature of the alignment problem itself. The inability to guarantee the maximality of the average payoff is due to the fact that the original object ($M$) could contain large areas outside the overlapping subset that are perfectly identical. Further, objects that are able to slide (for instance a plane or a sphere) could be allowed to move between different mixed strategies without penalty. These situations cannot be addressed by any algorithm without relying on supplementary information. However, in practice, they are quite unlikely extreme cases. In the experimental section we will show that our approach can effectively register a wide range of surface types.

In Fig. 3 we show a complete example of the evolutionary matching process. In order to make the example easy to understand we restricted our focus to a detail of a range scan of the Stanford "dragon". In this example (and throughout all the experimental section) $S$ is built by including all the strategy pairs composed by a feature point in the model and the 5 nearest feature points in the data in terms of Surface Hash (in this example we used an Integral Hash with 3 scales). In Fig. 3(g) we show, on a colored scale from 0 to 1, the payoff matrix of the rigid enforcing function used (which is discussed in the experimental section). Note that in the diagonal area of the matrix blocks of five strategies with reciprocal 0 payoff can be found: this is related to the way we built $S$. In fact we chose to include for each model point 5 candidates in the data and they are mutually non compatible as they share the same source point and we are looking for a one-to-one match. In the top and bottom half of Fig. 3(d) we can see respectively model and data feature points at the beginning of the matching



(d) Initial matches (e) Matches in 1 round (f) Matches in 100 rounds

(g) Payoff matrix (h) Population in 1 round (i) Population in 100 rounds

**Fig. 3.** Example of a rigid enforcing payoff and of the evolution of the matching process

process. After just one round of replicator dynamics we see that many outliers have been peeled off from the initial set $S$, but still some wrong matches are present. After 100 iterations only a few matches have been retained, but it is easy to see that they are extremely coherent. Finally, in Fig. 3(h) and Fig. 3(i) we show the (sorted) population histogram respectively after 1 and 100 iterations. The first histogram shows that all the strategies are still played by a sizeable amount of the population, while after 100 iterations most of the consensus is held by the few surviving matches.

## 4 Experimental Results

In this section we study the behavior of the proposed surface registration technique with respect to different Surface Hashes and scales. In addition we evaluate both the performance of the proposed feature descriptor with other matches and the quality of the alignment obtained by comparison with other pipelines. The rigidity-enforcing payoff function used throughout the experiments is defined as

$$\pi((a_1, b_1), (a_2, b_2)) = \frac{min(|a_1 - a_2|, |b_1 - b_2|)}{max(|a_1 - a_2|, |b_1 - b_2|)} \tag{2}$$

where $a_1$, $a_2$, $b_1$ and $b_2$ are respectively the two model (source) and data (destination) points in the compared mating strategies. The initial set of strategies $S$ was built by including all the pairs composed by a feature point in the model and the 5 feature points in the data with the nearest descriptor.

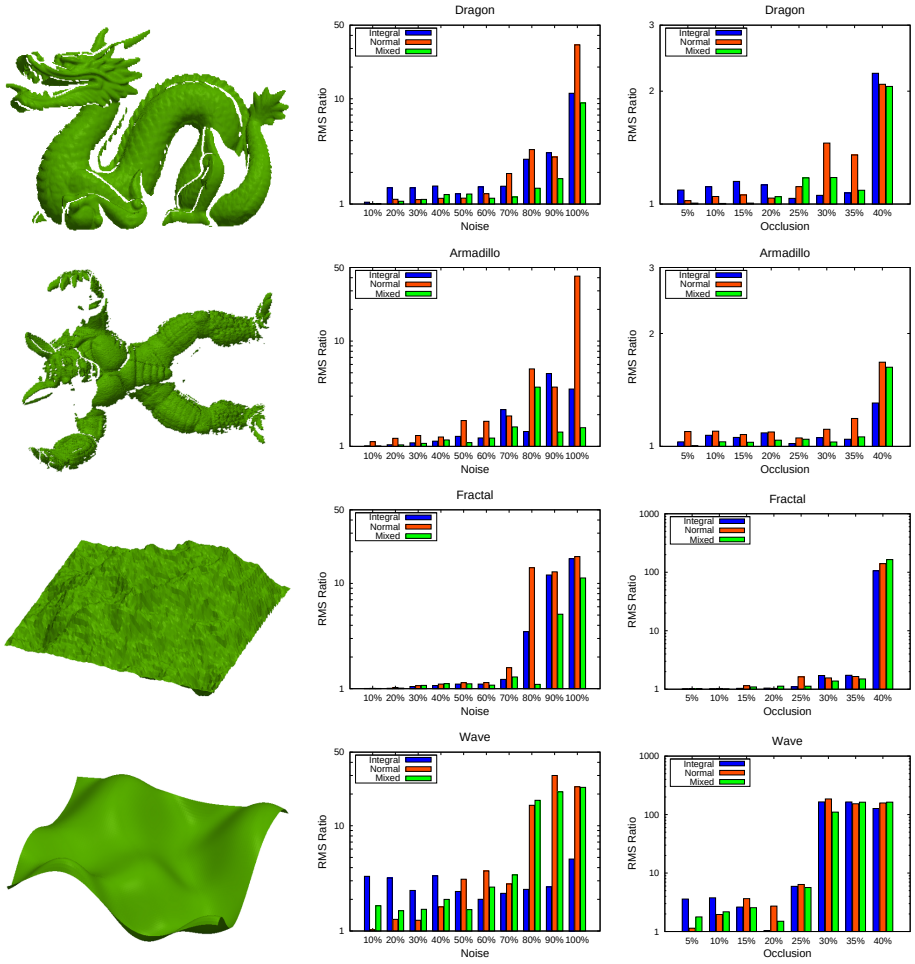



**Fig. 4.** Comparison of different descriptors using real and synthetic objects.

### 4.1 Sensitivity to Noise, Occlusion, and Scale of the Descriptor

The performance of different descriptors was tested for various levels of noise and occlusion applied to two surfaces obtained from real range scans ("armadillo" and "dragon" from Stanford) and two synthetic surfaces designed to be challenging for coarse registration techniques ("fractal" and "wave"). The noise is a positional Gaussian perturbation on the point coordinates with its level ($\sigma$) expressed in terms of the percentage of the average edge length, while the occlusion denotes the percentage of data and model surfaces removed. The RMS Ratio in the charts is the ratio of the root mean square error (RMS) obtained after registration and the RMS of the ground truth alignment. The Normal and Integral Hashes were calculated over 3 levels of scale and the "Mixed" Hash

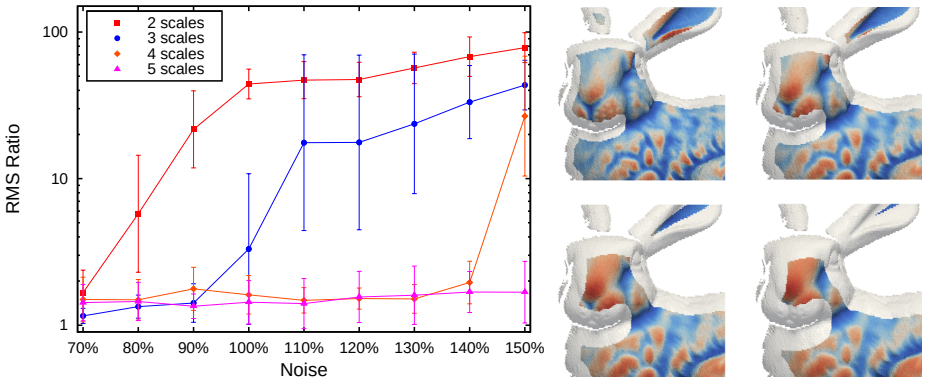



**Fig. 5.** Effect of scale on the matching accuracy

is simply the juxtaposition of the previous two. In Fig. 4 we see that all the descriptors obtain good results with real ranges and the registration "breaks" only with very high levels of noise (on the same order of magnitude of the edge length). It is interesting to observe that the Mixed Hash always obtains the best performance, even with high level of noise: This higher robustness is probably due to the orthogonality between the Normal and Integral Hashes. The behavior with the "fractal" synthetic surface is quite similar, by contrast all the descriptors seem to perform less well with the "wave" surface. This is due to the lack of distinctive features on the model itself, which indeed represents a challenge for any feature based registration technique. The performance obtained with respect to occlusion is similar: all the descriptors achieve fairly good results and are resilient to high levels of occlusion (note that 40 percent occlusion is applied both to data and model). Overall the Mixed Hash appears to be consistently more robust. Since we found that the descriptors calculated over 3 levels of scale break at a certain level of noise, we were interested in evaluating if their performance can be improved by increasing their dimension. In Fig. 5 we present the results obtained with different levels of scale for the Mixed Hash. The graphs show the average over all the surfaces and the associated RMS. It is interesting to observe that by reducing the scale level the technique becomes less robust, whereas its performance increases dramatically when the number of scales increases. With a scale level of 5 our approach can deal even with surfaces subject to Gaussian positional noise of $\sigma$ greater than the edge length. Unfortunately this enhanced reliability comes with a drawback: by using larger levels of scale the portion of boundary that cannot be characterized grows. In the right half of Fig. 5 the shrinking effect is shown for scale levels from 2 to 5.

### 4.2 Comparisons with other matchers

Our goal in this set of experiments is to study if Surfaces Hashes can be used successfully with matchers alternative to the Matching Game described in Sec-

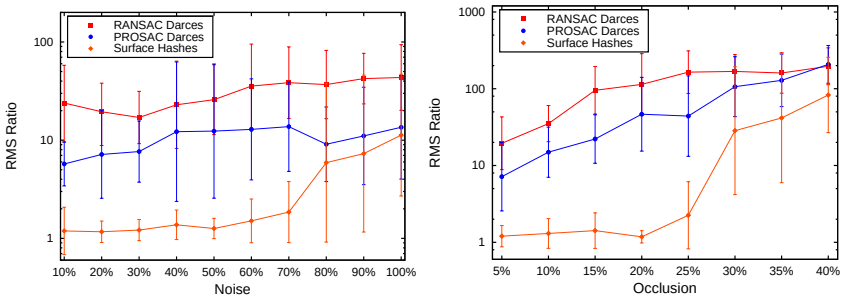



**Fig. 6.** Comparison of the performance obtained with different matchers

tion 3.2. Specifically, we compared our full pipeline with standard DARCES [17] and with a DARCES variant that adopts PROSAC instead of plain RANSAC to take advantage of our descriptors as prior. To this end, we sorted the initial correspondence hypotheses by descriptor similarity and operated a PROSAC-like selection starting from an initial set of high-ranked matches and enlarging it progressively. In Fig. 6 we show the results of this test. As expected, RANSAC-based DARCES yields the worst results. Our PROSAC based variant obtains slightly better average registrations, but, the additional information provided by the descriptors is not distinctive enough to boost this technique to performance levels of the Matching Game that relies only on the global rigidity constraints.

### 4.3 System-level Comparisons

Since our alignment approach does not need any initial estimate of the motion between surfaces, it can be classified as a coarse registration technique. For this reason we found appropriate to compare it with other widely used coarse registration methods. To this extent, we chose to use the Spin Images based approach proposed by Johnson [9] and the MeshDOG/MeshHOG combination suggested by Zaharescu [11]. The latter was selected because it adopts short descriptors very similar to the one proposed in this paper. In Fig. 7 we see that both techniques perform worse than the one based on Surface Hashes, even at low noise and occlusion levels. Surprisingly MeshDOG/MeshHOG obtains the worst results, probably because of the combination of a weak descriptor with a greedy matcher. Finally, we used the coarse registrations obtained with each approach to initialize a fine registration made with a best-of-breed ICP variant similar to the one proposed in [18]. Point selection is based on Normal Space Sampling [19], and point-surface normal shooting is adopted for finding correspondences, distant mates, candidates with back-facing normals, or matings established on the boundary of the mesh are rejected. In the leftmost plot of Fig. 8 we histogram the frequency of RMS ratio intervals obtained after the coarse registration. The histogram is based on bins of exponentially increasing size. In the rightmost chart the distribution change after a full round of ICP refinement can be seen. We can observe that while ICP is able to correct some

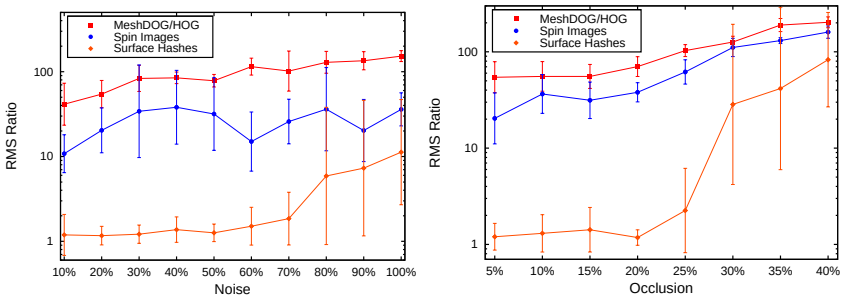



**Fig. 7.** Comparison of the performance obtained with different coarse techniques

wrong registrations with lower RMS Ratio, our approach still reaches the optimal alignment with a frequency that is almost double of the one obtained by the closest competitor. Regarding the computational complexity, it should be noted that the algorithm is quadratic in the number of strategies and thus the number of feature correspondences. Nevertheless, the initial interest points selection and the correspondences filtering by means of the descriptors, allow us to keep the computational time within a few seconds in all of our experiments.

## 5 Conclusions

In this paper we introduced a novel surface registration technique that uses very simple descriptors to create several weak correspondence hypotheses that are further optimized by a robust game-theoretic matcher. A theoretical result exposed the correspondence between optimal alignments and evolutionary equilibria, and the approach was validated on a wide range of experiments showing its greater robustness with respect to noise and occlusion in comparison with other well-known techniques.

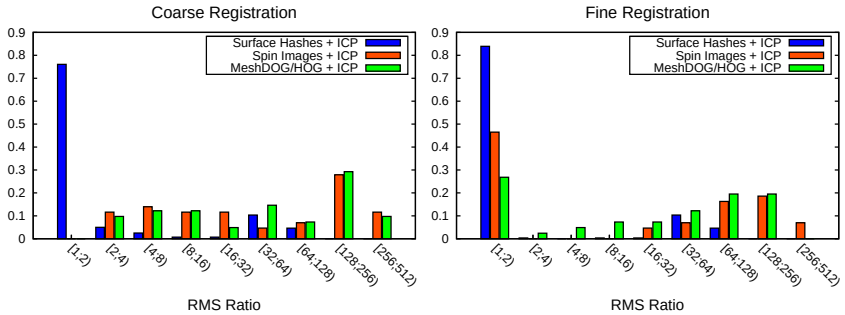



**Fig. 8.** Comparison of the performance between complete pipelines

## Acknowledgement

We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.

## References

1. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proc. Fourth Alvey Vision Conference. (1988) 147–151
2. Marr, D., Hildreth, E.: Theory of edge detection. Royal Soc. of London Proc. Series B **207** (1980) 187–217
3. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing **22** (2004) 761–767 British Machine Vision Computing 2002.
4. Mikolajczyk, K., Schmid, C.: An affine invariant interest point detector. In: ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I, London, UK, Springer-Verlag (2002) 128–142
5. Lowe, D.: Distinctive image features from scale-invariant keypoints. In: International Journal of Computer Vision. Volume 20. (2003) 91–110
6. Herbert Bay, T.T., Gool, L.V.: Surf: Speeded up robust features. In: 9th European Conference on Computer Vision. Volume 3951. (2006) 404–417
7. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. Pattern Analysis and Machine Intelligence, IEEE Transactions on **27** (2005) 1615–1630
8. Chua, C.S., Jarvis, R.: Point signatures: A new representation for 3d object recognition. International Journal of Computer Vision **25** (1997) 63–85
9. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3d scenes. IEEE Trans. Pattern Anal. Mach. Intell. **21** (1999) 433–449
10. Pottmann, H., Wallner, J., Huang, Q.X., Yang, Y.L.: Integral invariants for robust geometry processing. Comput. Aided Geom. Des. **26** (2009) 37–60
11. Albarelli, A., Rota Bulò, S., Torsello, A., Pelillo, M.: Matching as a non-cooperative game. In: ICCV, IEEE Computer Society (2009)
12. Weibull, J.: Evolutionary Game Theory. MIT P. (1995)
13. Albarelli, A., Rodolà, E., Torsello, A.: A game-theoretic approach to fine surface registration without initial motion estimation. In: CVPR. (2010)
14. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24** (1981) 381–395
15. Chum, O., Matas, J.: Matching with prosac - progressive sample consensus. In: CVPR, Washington, DC, USA, IEEE Computer Society (2005) 220–226
16. Chen, C.S., Hung, Y.P., Cheng, J.B.: Ransac-based darces: A new approach to fast automatic registration of partially overlapping range images. IEEE Trans. Pattern Anal. Mach. Intell. **21** (1999) 1229–1234
17. Zaharescu, A., Boyer, E., Varanasi, K., Horaud, R.P.: Surface feature detection and description with applications to mesh matching. In: CVPR. (2009)
18. Turk, G., Levoy, M.: Zippered polygon meshes from range images. In: SIGGRAPH '94: Proc. of the 21st annual conference on Computer graphics and interactive techniques, New York, NY, USA, ACM (1994) 311–318
19. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: Proceedings of the Third Intl. Conf. on 3D Digital Imaging and Modeling. (2001) 145–152

# A Non-Cooperative Game for 3D Object Recognition in Cluttered Scenes

Andrea Albarelli, Emanuele Rodolà, Filippo Bergamasco, and Andrea Torsello
*Dipartimento di Scienze Ambientali, Informatica e Statistica*
*Università Ca' Foscari Venezia*
*Venice, Italy*
*Email: albarelli@unive.it rodola@dsi.unive.it fbergama@dsi.unive.it torsello@dsi.unive.it*

***Abstract*—During the last few years a wide range of algorithms and devices have been made available to easily acquire range images. To this extent, the increasing abundance of depth data boosts the need for reliable and unsupervised analysis techniques, spanning from part registration to automated segmentation. In this context, we focus on the recognition of known objects in cluttered and incomplete 3D scans. Fitting a model to a scene is a very important task in many scenarios such as industrial inspection, scene understanding and even gaming. For this reason, this problem has been extensively tackled in literature. Nevertheless, while many descriptor-based approaches have been proposed, a number of hurdles still hinder the use of global techniques. In this paper we try to offer a different perspective on the topic. Specifically, we adopt an evolutionary selection algorithm in order to extend the scope of local descriptors to satisfy global pairwise constraints. In addition, the very same technique is also used to shift from an initial sparse correspondence to a dense matching. This leads to a novel pipeline for 3D object recognition, which is validated with an extensive set of experiments and comparisons with recent well-known feature-based approaches.**

***Keywords*-Object Recognition; Rigid Alignment; Game Theory; Object in Clutter;**

## I. Introduction

In the recent past, the acquisition of 3D data was only viable for research labs or professionals that could afford to invest in expensive and difficult to handle high-end hardware. However, due to both technological advances and increased market demand, this scenario has been altered significantly: Semi-professional range scanners can be found at the same price level of a standard workstation, widely available software stacks can be used to obtain reasonable results even with cheap webcams, and, finally, range imaging capabilities have been introduced even in very low-end devices such as game controllers. Given this trend, it is safe to forecast that range scans will be so easy to acquire that they will complement or even replace traditional intensity based imaging in many computer vision applications. The added benefit of depth information can indeed enhance the reliability of most inspection and recognition tasks, as well as providing robust cues for scene understanding or pose estimation. Many of these activities include fitting a known model to a scene as a fundamental step. For instance, a setup for in-line quality control within a production line, could need to locate the manufactured objects that are meant to be measured [1]. Moreover, a range-based SLAM system [2], can exploit the position of known 3D reference objects to achieve a more precise and robust robot localization. Finally, non-rigid fitting could be used to recognize hand or whole-body gestures in next generation interactive games or novel man-machine interfaces [3]. The matching problem in 3D scenes shares many aspects with object recognition and location in 2D images: The common goal is to find the relation between a model and its transformed instance (if any) in the scene. In both cases, transformations could include uniform and non-uniform scaling, differences in pose or partial modification of the shape. They also share common hurdles, such as measurement errors on intensities or point positions, and indirect changes in the appearance due to occlusion or the simultaneous presence in the scene of extraneous objects that can act as distractions. Feature-based approaches, both in 2D and in 3D, adopt descriptors that are associated to single points respectively on the image or on the object surface. In principle, each feature can be matched individually by comparing the descriptors, which of course decouples the effect of partial occlusion. In the 2D domain, intensity based descriptors such as SIFT [4] have proven to be very distinctive and be able to perform very well even with naive matching methods that do not include any global information [5]. However, the problem of balancing local and global robustness is more binding

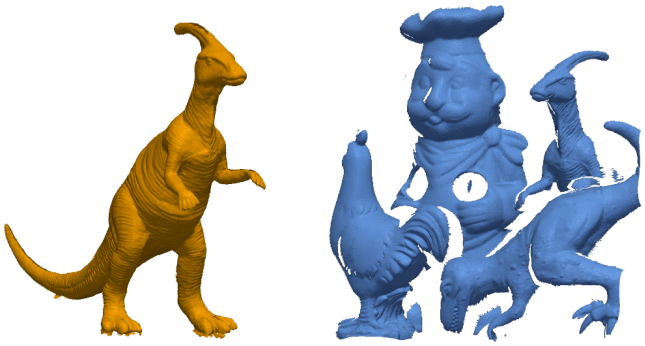


Figure 1. A typical 3D object recognition scenario. Clutter of the scene and occlusion due to the geometry of the ranging sensor seriously hinder the ability of both global and feature-based techniques to spot the model.

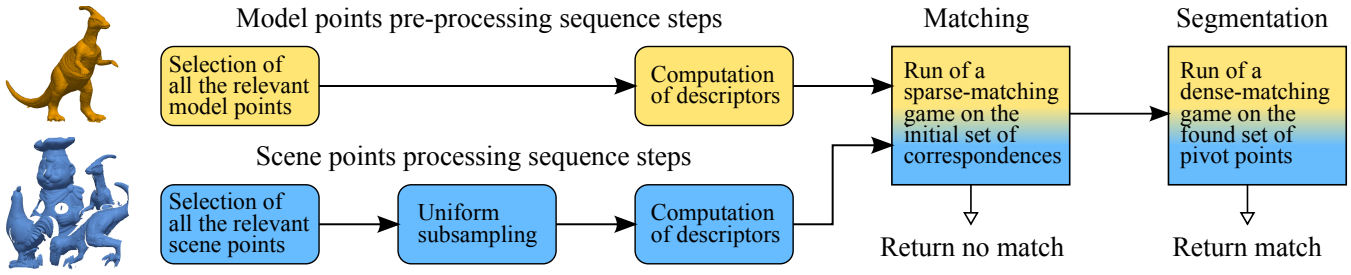



Figure 2. An overview of the object recognition pipeline presented (see text for description).

with 3D scenes than with images, as no natural scalar field is available on surfaces and thus feature descriptors tend to be less distinctive. In practice, global or semi-global inlier selection techniques are often used to avoid wrong correspondences. This, while making the whole process more robust to a moderate number of outliers, could introduce additional weaknesses. For instance, if a RANSAC-like inlier selection is applied, occlusion coupled with the presence of clutter (*i.e.*, unrelated objects in the scene) can easily lower the probability for the process to find the correct match. The limited distinctiveness of surface features can be tackled by introducing scalar quantities computed over the local surface area. This is the case, for instance, with values such as mean curvature, Gaussian curvature or shape index and curvedness, which can be constructed in order to classify surface patches into types such as pits, peaks or saddles [6]. Unfortunately, this kind of characterization has proven to be not very selective for matching purposes, since it is frequent to obtain similar values in many different locations. Another approach is to augment the point data with additional scalar values that can be obtained during the acquisition process. To this extent, the use of natural textures coming from the scanned object have shown to allow good performance since they show high variability and can be used to compute descriptors similar to those usually adopted in the 2D domain [7]. Still, textures cannot be obtained from all the surface digitizing techniques and, even when available, their usability for descriptor extraction strongly depends on the appearance of the scanned object. To overcome the limitations of scalar descriptors, methods that gather information from the whole neighborhood of each point to characterize have been introduced. Such methods can be roughly classified in approaches that define a full reference frame for each point (for instance, by using PCA) and techniques that only need a reference axis (usually some kind of normal direction for the point). When a full reference frame is available it is possible to build very discriminative descriptors [8], [9]. Unfortunately, noise and differences in the mesh could lead to instabilities in the reference frame, and thus to a brittle descriptor. By converse, methods that just require a reference axis (and are thus invariant to the rotation of the frame) trade some descriptiveness to gain greater robustness. These latter techniques almost invariably build histograms based on some properties of points falling in a cylindrical volume centered and aligned to the reference axis. The most popular histogram-based approach is certainly Spin Images [10], but many others have been proposed in literature [11], [12]. Lately, an approach that aims to retain the advantages of both full reference frames and histograms has been introduced [13]. Other recent contributions include scale invariant detectors [14], [15] and tensor-based descriptors [16]. Any of these interest point descriptors can be used to find correspondences between a model and a 3D scene that could possibly contain it. Most of the cited papers, in addition to introducing the descriptor itself, propose some matching technique. These span from very naive approaches, such as associating each point in the model with the point in the scene having the most similar descriptor, to more advanced techniques such as customized flavors of PROSAC and specialized keypoint matchers that exploit locally fitted surfaces for computing depth values to use as feature components [17].

In this paper we introduce a novel feature-based 3D object recognition pipeline crafted to deal in a robust manner with both strong occlusion and clutter. This happens by adopting a histogram-based local surface descriptor to find a set of matching candidates among a selection of relevant points on the model and the scene. Such candidates are then let to compete in a non-cooperative game where payoffs are proportional to the degree of Euclidean compatibility between them. This leads to a smaller set of sparse but reliable surviving matches which, in turn, will be used as the seeds for an additional game aimed at the selection of a denser population. While the use of Game Theory for matching has already been explored [18], the contribution of this paper is threefold. It introduces a novel pipeline that outperforms the state-of-the-art for 3D object recognition in clutter. Further, it suggests a simple but general rule for samples selection for the purpose of recognition. Finally, it defines a new kind of game for building a dense surface correspondence starting from a sparse set of pivot points, which can be useful also for other matching techniques.

## II. A Game-Theoretic Pipeline for Recognition

Following [19], we base our matching framework on the recently introduced Game-Theoretic techniques for inlier selection. The complete pipeline we are proposing is made up of a preprocessing step and two non-cooperative games (see Fig. 2). The preprocessing is performed both on the model and on the scene. This step involves an initial selection of relevant points on the respective surfaces. The relevance criteria will be explained in the next section, however, in this context the general meaning of the culling is to avoid surface patches that are not significant from a matching standing point, such as flat areas. All the interest points on the model are kept while those on the scene are uniformly subsampled. This makes sense for many reasons. In many applications the set of models does not change in time, and thus descriptors must be computed just once. In addition, as explained in the following sections, the direction of the matching will be from the scene to the model and having less source than target points allows the game to proceed faster without compromising accuracy. Finally, the model tends to be measured with greater accuracy (either because more time can be spent on it or because it comes from a CAD model). A descriptor is computed for all the retained points, and these are used to build the initial candidates that will be fed to two matching games. The games are played respectively to build a coarse initial set of fiducial correspondences and to make those into dense matches by exploiting neighborhood relationships.

In general, a matching game [19] can be built by defining just four basic entities: a set of model points $M$, a set of data points $D$, a set of candidate correspondences $S \subseteq M \times D$ and a pairwise compatibility function between them $\Pi : S \times S \rightarrow \mathbb{R}^+$. The goal of the gameplay is to operate a (natural) selection among the elements in the initial set $S$. This happens by setting up a non-cooperative game where the set $S$ represents the available strategies and $\Pi$ the payoffs between them. In this game, a real-valued vector $\mathbf{x} = (x_1, \ldots, x_{|S|})^T$ that lies in the $|S|$-dimensional standard simplex

$$\Delta^{|S|} = \left\{ \mathbf{x} \in \mathbb{R}^{|S|} \ : \ x_i \geq 0, i = 1 \ldots |S|, \ \sum_{i=1}^{|S|} x_i = 1 \right\}$$

represents the amount of population that plays each strategy $i$ at a given time. The game starts by setting the initial population around the barycenter (to be fair with respect to each strategy). Then, the population can be evolved at discrete steps by applying the replicator dynamics equation:

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) \frac{(\Pi \mathbf{x}(t))_i}{\mathbf{x}(t)^T \Pi \mathbf{x}(t)} \tag{1}$$

where $\Pi$ is a matrix that assigns to row $i$ and column $j$ the payoff (compatibility) between strategies (correspondences) $i$ and $j$. Under very weak assumptions it can be shown that such dynamics must converge (in an infinite time) to a *Nash equilibrium*, *i.e.*, a point in the simplex where the average payoff obtained by the population is a local maximum constant for each strategy. In addition, the values of the elements of $\mathbf{x}$ are proportional to the degree of compatibility of each strategy with the equilibrium [19]. In practice, a much faster convergence to the equilibrium can be obtained by replacing the iteration in equation (1) with the adaptive exponential replicator dynamics introduced in [20]. Since we defined the payoff as the compatibility between candidates, these are all desirable properties from a selection standpoint. In our context, $M$ and $D$ always correspond to the retained model and scene points, while $S$ and $\Pi$ will be defined differently for the sparse and dense matching game. Specifically, for the sparse game the construction of $S$ will be driven by descriptor similarity, whereas positional information can be used in the segmentation game. Likewise, the payoff $\Pi$ will be proportional to the different notions of compatibility.

### A. Feature Detection and Description

For both efficiency and robustness reasons, the proposed matching technique works on a subset of the model and scene data. First, a culling of all the vertices is performed. This happens by computing for each point a single-component *Integral Hash* [21] at a given support scale $\sigma$, and thus retaining only those samples that obtain a negative value (*i.e.*, that belongs to a concave surface patch). In practice, this means that we are avoiding flat and convex areas which we experimented to be less distinctive. By modulating the value of $\sigma$ a more or less selective sample selection can be made (see Fig. 3). All the model relevant points are kept. By contrast, an optional uniform subsampling can be performed on the relevant points in the scene. Finally, a descriptor vector must be computed for each vertex to be matched. To this extent, any of the descriptors discussed in the introduction could be used; however, after an initial round of tests, SHOT [13] was chosen as it obtains the best performance over the whole pipeline. In the experimental section both the influence of the relevant point selection and of the adopted descriptor are studied.

### B. Sparse Matching Game

In this matching game the set of candidates $S$ is built by associating each reference point in the scene with the $k$ nearest points in the model in terms of the descriptor:

$$S = \{(a, b) \in D \times M | b \in dn_k(a)\}, \tag{2}$$

where $dn_k(a)$ is the set of the $k$ model vertices with the nearest descriptor with respect to the descriptor of $a$. In practice, this means that each sample in the scene is considered to be a possible match with samples in the model that exhibit similar surface characteristics, and we limit the

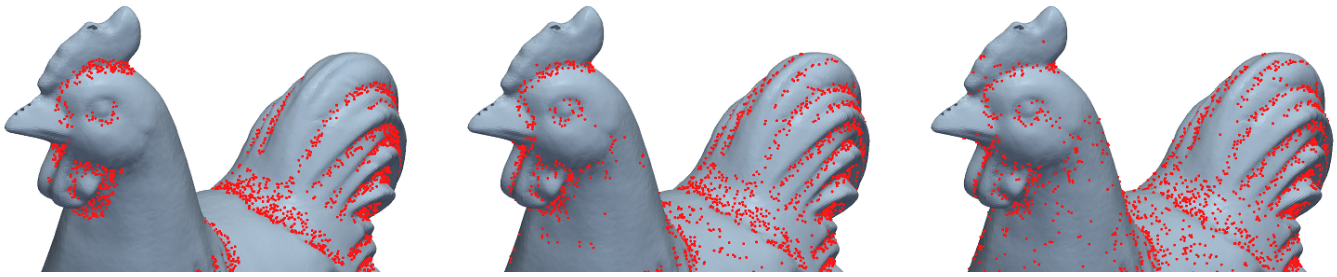


Figure 3. In order to avoid mismatches and reduce the convergence time it is important to use only relevant points. Model vertices selected with a $\sigma$ respectively equal to 8, 5 and 2 times the median model edge are shown from left to right.

number of "attempts" to $k$. It should be noted that candidates are built from scene to model. Since we are interested in finding a correspondence between the model and part of the surface in the scene, we are looking for a subset of candidates that enforce the Euclidean rigidity constraint. Such candidates are likely to lay on the same surface both in the scene and in the model and thus to be a viable solution. To this extent, we define this distance measure between pairs of strategies in $S$ as

$$\delta((a_1, b_1), (a_2, b_2)) = \frac{min(|a_1 - a_2|, |b_1 - b_2|)}{max(|a_1 - a_2|, |b_1 - b_2|)} \quad (3)$$

where $a_1$, $a_2$, $b_1$ and $b_2$ are respectively the two model and scene vertices in the compared strategies. The value of $\delta$ will be 1 if the corresponding source and destination points are separated by exactly the same Euclidean distance. By contrast, $\delta$ will be small when the two pairs exhibit very different distances. This kind of check will succeed with correct pairs and will give false positives only for a small amount of cases, those preserving the rigid constraint by chance. However, since our game is seeking for a large group of candidates with large mutual payoff, such sneaky outliers will be filtered out with high probability by the other strategies that participate to the Nash equilibrium. Finally, we also want to avoid many-to-many matches, since we do not expect any point in the scene to correspond to more than one point in the model. This can be done easily by forcing to 0 the compatibility between candidates that share the same source or destination vertex [19]. Thus, the final payoff for the sparse matching game that we are defining will be

$$\Pi = \begin{cases} \delta((a_1, b_1), (a_2, b_2)) & \text{if } a_1 \neq a_2 \text{ and } b_1 \neq b_2 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

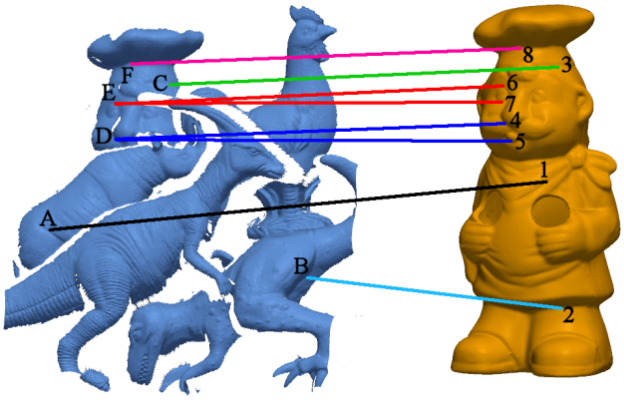
Once the candidate set and the payoff matrix are built, the game is started from the barycenter of the simplex: when a stable state is reached, all the strategies supported by a large percentage of the population (above a threshold based on the most played strategy) are considered non-extinct and retained as correct matches (see Fig. 4). If the surviving matches are more than a fixed minimum (set to 8 in our experiments), then the object is recognized and its pose can be computed.

### C. Dense Matching Game

If the matching game succeeds, then a fiducial set of correspondences has been found; it would be interesting to use these matches as a seed for segmenting the surface belonging to the model from the scene. In most cases, growing from the fiducial points to the connected part of the range surface would be enough. However, in cluttered range images the unintentional merging between surfaces of different objects is quite frequent, thus a better selection mechanism could be useful. In order to demonstrate the flexibility of the Game-Theoretic framework we define another type of game to solve these problems (albeit other more direct solutions are also possible). We start by using the initial correspondences to estimate the rigid transformation between the model and the scene using the closed form method proposed in [22]. The computed transformation is then used to register the model within the scene coordinate system. At this point, if the initial matches are correct, the model vertex corresponding to each scene point should be in its neighborhood. For this reason we define the set of candidates $S$ as

$$S' = \{(a, b) \in D \times M | b \in en_k(a)\} \quad (5)$$

where $en_k(a)$ is the set of the $k$ nearest model vertices with respect to the Euclidean distance from $a$. Note that since we trust the alignment to be good (even if it is not perfect) we do not need point descriptors anymore. We want to enforce the rigidity constraint for this game as well, thus the compatibility $\delta$ defined in the previous game could still be used. However we would like to apply two modifications to the payoff function. The first one is the introduction of an exponent $\alpha$ to the measured compatibility. This is needed because within this game all the points are very close to each other and small variations in the position of a point in the model and its scene correspondence can easily lead to low compatibility. The second modification is related to the observation that we are interested in operating a model-driven segmentation of the scene, thus we are not really




| | A1 | B2 | C3 | D4 | D5 | E6 | E7 | F8 |
|---|---|---|---|---|---|---|---|---|
| A1 | 0 | 0.12 | 0.77 | 0.83 | 0.98 | 0.77 | 0.66 | 0.75 |
| B2 | 0.12 | 0 | 0.05 | 0.21 | 0.37 | 0.07 | 0.32 | 0.17 |
| C3 | 0.77 | 0.05 | 0 | 0.99 | 0.6 | 0.99 | 0.99 | 0.7 |
| D4 | 0.83 | 0.21 | 0.99 | 0 | 0 | 0.99 | 0.99 | 0.96 |
| D5 | 0.98 | 0.37 | 0.6 | 0 | 0 | 0.9 | 0.88 | 0.69 |
| E6 | 0.77 | 0.07 | 0.99 | 0.99 | 0.9 | 0 | 0 | 0.98 |
| E7 | 0.66 | 0.32 | 0.99 | 0.99 | 0.88 | 0 | 0 | 0.99 |
| F8 | 0.75 | 0.17 | 0.7 | 0.96 | 0.69 | 0.98 | 0.99 | 0 |

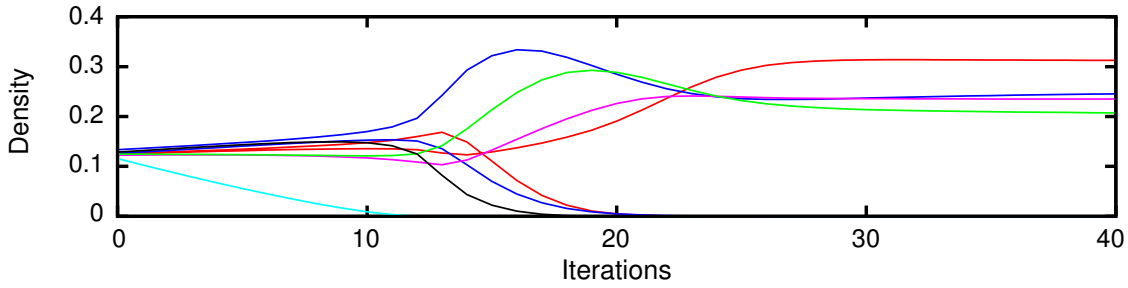



Figure 4. An example of the evolutionary process (with real data). A set of 8 matching candidates is chosen (upper left), a payoff matrix is built to enforce their respective Euclidean constraints (upper right, note that cells associated to many-to-many matches are set to 0) and the replicator dynamics are executed (bottom graph). At the start of the process the population is set around the barycenter (at 0 iterations). This means that initially the vector $\mathbf{x}$ represents a quasi-uniform probability distribution. After a few evolutionary iterations the matching candidate B2 (cyan) is extinct. This is to be expected since it is a clearly wrong correspondence and its payoff with respect to the other strategies is very low (see the payoff matrix). After a few more iterations, strategy A1 vanishes as well. It should be noted that strategies D4/D5 and E6/E7 are mutually exclusive, since they share the same scene vertex. In fact, after an initial plateau, the demise of A1 breaks the tie and finally E6 prevails over E7 and D4 over D5. After just 30 iterations the process stabilizes and only 4 strategies (corresponding to the correct matches) survive.

looking for a one-to-one correspondence between points, but rather we are trying to match each vertex in the scene to at least one reasonable vertex in the model to which it belongs. To this extent, the one-to-one constraint enforced in the previous game can be relaxed to a many-to-one constraint and the payoff function can be defined as

$$\Pi' = \begin{cases} \delta((a_1, b_1), (a_2, b_2))^\alpha & \text{if } a_1 \neq a_2 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Again, this segmentation game can be played by starting from the barycenter of the standard simplex and letting the population evolve by means of appropriate dynamics.

## III. Experimental Results

In order to evaluate the performance of the proposed pipeline we performed a wide range of tests and comparisons with recent techniques. To offer a fair comparison we used the model/scene dataset adopted in [14], [16], [17]. This dataset is composed of five high resolution models scanned from real objects (chef, dino1, dino2, chicken and rhino), plus about two hundred range scans of these objects under various conditions of occlusion (due to the overlap of objects and limits on the field of view of the sensor) and clutter (due to the presence of many objects in the scene). All the tests were performed on a standard desktop PC equipped with a Core Duo processor clocked at 1.6Ghz. The evolutionary process makes use of the adaptive exponential replicator dynamics [20]. The minimum number of matches to assume the model as recognized in the scene was 8. The value of $\alpha$ for the segmentation game was 0.2. For the sparse matching game, the SHOT descriptor [13] was used.

### A. Comparison with the State-of-the-art

In Fig. 5 we compare our results in terms of recognition rate with recent state-of-the-art algorithms (respectively [14], [16], [17]) and with the well-known 3D Spin Image matching technique [10], which is often used as a baseline. Looking at the recognition rate with respect to model occlusion, the proposed pipeline outperforms even the most recent techniques. Regarding the evaluation of the effects of clutter we could compare our algorithm only to [14], since an implementation for the other approaches and the data they used were not available. Still, it is apparent that the Game-Theoretic approach obtains good recognition with uniform performance. Some examples of critical scenes

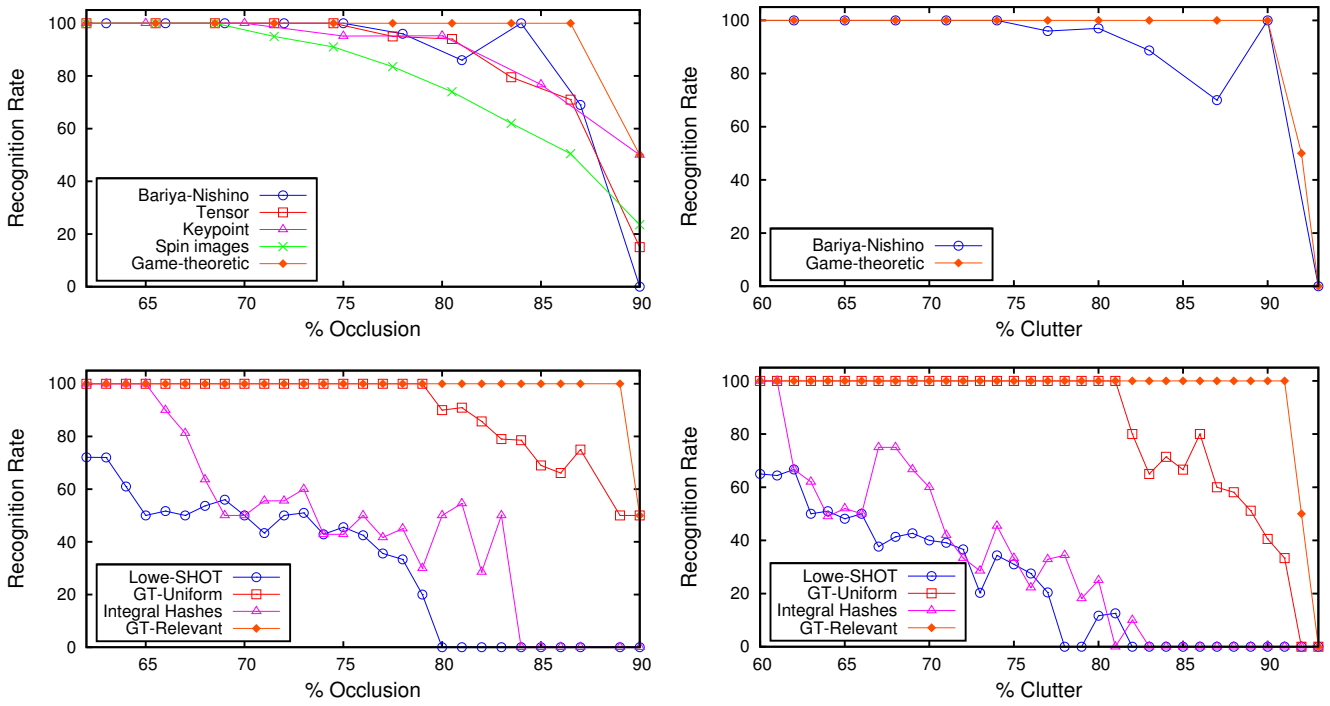



Figure 5. In the top row the recognition rate of our pipeline is compared with state-of-the-art techniques, which are outperformed with respect to both occlusion and clutter. In the bottom row the contribution of each part of the overall approach is tested separately (see text for details).

where the proposed technique fixes matches missed by the other methods are shown in Fig. 7. The behavior with respect to false positives has not been plotted since the proposed pipeline does not get any throughout the whole dataset. In the second row of Fig. 5 several combinations of components of the pipeline are evaluated one at a time in order to shape their respective contribution. Specifically, we show the results obtained using the same descriptor [13] with the classical matcher proposed by Lowe [5] (Lowe-SHOT), the Game-Theoretic matcher without operating the initial relevance-based sampling (GT-Uniform), the descriptors and matching proposed in [21] (Integral-Hashes) and finally the full proposed pipeline (GT-Relevant). It is apparent that the proposed pipeline only works with all the components in place (note that with these latter experiments the sampling of the plots is more dense).

### B. Resilience to Noise

All the experiments so far have been done using a dense model and slightly less dense scenes produced with a range scanner. Although there is not an exact correspondence between model and scenes, they are still very similar by construction. It would be interesting to study the performance of the proposed method in presence of positional noise. To do so, we added Gaussian displacement of varying intensity to each vertex in the scene. In Fig. 6 the results obtained with two different SHOT parameterizations are shown. As expected, performance gets lower as the noise level increases; still, reasonable recognition rates are maintained also with a moderate amount of noise (with standard deviation equal to 30% the median edge length).

### C. Sparse to Dense Matching

An example of the dense matching game used to segment the parts of the scene belonging to the model is shown in the last row of Fig. 7. Segmented points are highlighted both on the model and on the scene. In this case the naive growing approach would have failed since in the range scene the chef's foot is partially merged with the hind foot boundaries of dino1.

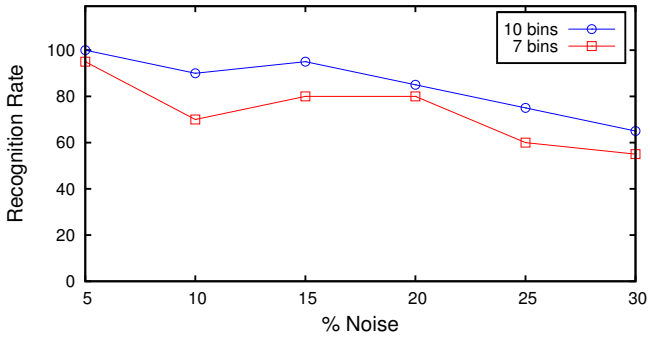



Figure 6. Evaluation of the robustness of the proposed pipeline with respect to increasing positional noise applied to the scene.

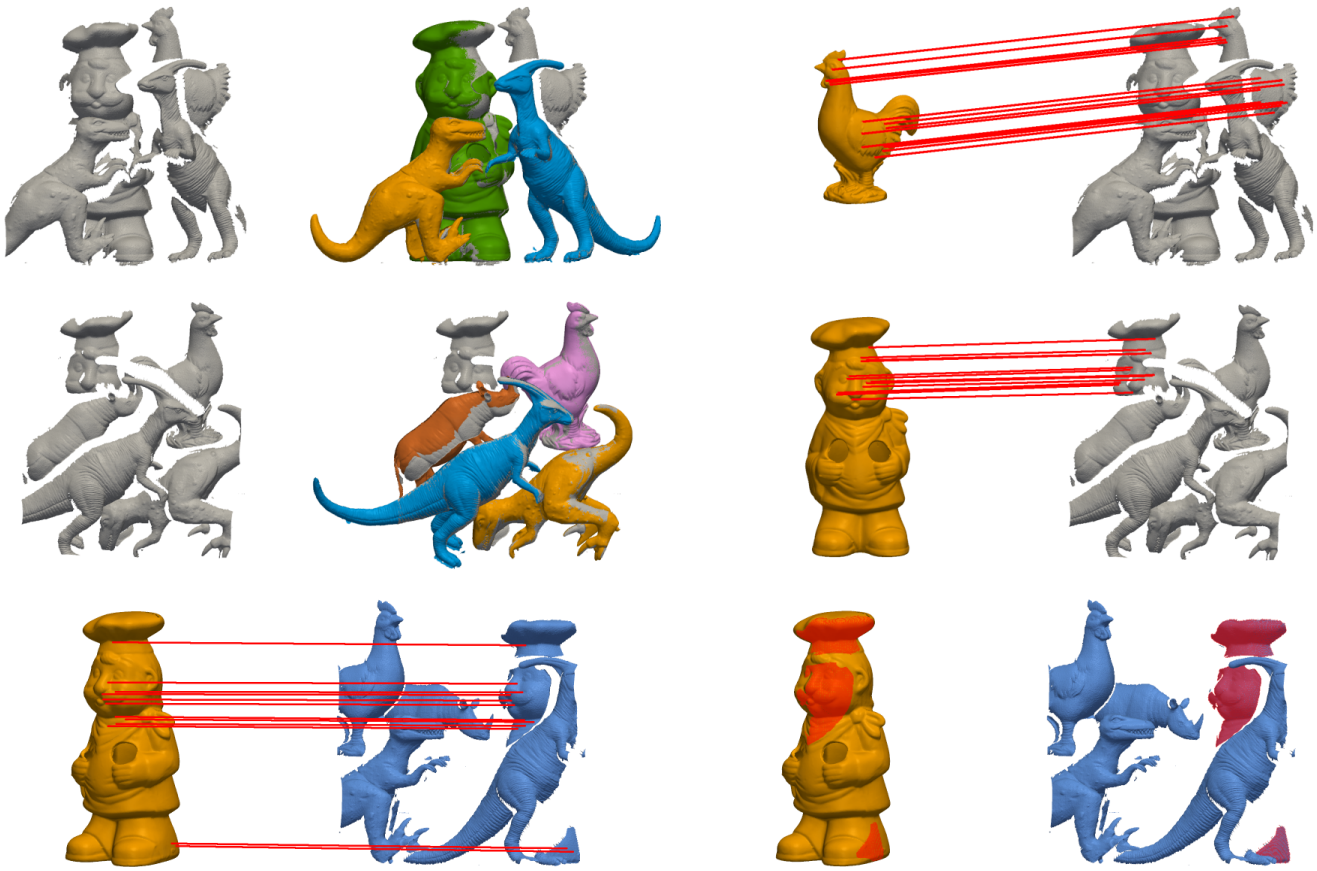


Figure 7. In the first and second rows we show an example of models correctly matched in scenes that break the method by Mian *et al.* (the chicken in the first row has been missed) and the method by Bariya-Nishino (the chef in the second row has been missed). In the third row we show the results obtained by playing a segmentation game (right) starting with the matches produced by a sparse game (left).

### *D. Performance Considerations*

We did not systematically evaluate the performance of the proposed pipeline. In practice, the most demanding step from a computational point of view is the evolutionary process. Since this is an iterative process, it is difficult to give an upper bound for its convergence time. However, the time required for each iteration is roughly proportional to the square of the number of strategies, which in turn means that the overall complexity could reach $O(n^4)$ with respect to the number of mesh points. However, since only the initial subset of strategies is used, the actual complexity is much lower and can be controlled by the parameter $k$ (described in section 2). Empirically we always observed convergence of the process within 50 iterations (tens of seconds).

## IV. Conclusions and Future Work

We described and empirically evaluated a novel pipeline for model-based 3D object recognition and segmentation in cluttered range scans. The pipeline starts with the detection of distinctive keypoints in the scene, which in turn is composed of a relevance filter, a subsampling step and the calculation of a descriptor for each sample kept. Such keypoints are then pairwise matched with all the relevant points of the model and a set of candidate pairings is obtained. Finally, two non-cooperative games are played: a rigid-matching game and a dense-growing game. The first one performs the actual recognition step and returns a sparse set of reliable matches. The second game expands these matches to segment all the surface patches in the scene that are compatible with the model. The overall approach combines a simple but effective relevance sampling schema with a recent local surface descriptor and with techniques borrowed from the emerging field of Game-Theoretic inlier selection. An extensive experimental evaluation shows that the proposed method outperforms recently proposed state-of-the art techniques on the same dataset. The contribution of the sampling schema is highlighted by testing the performance of the same pipeline leaving out this step; moreover, different keypoints descriptors are shown to give worse results. Finally, resilience to noise and the ability to obtain dense correspondences are evaluated individually obtaining encouraging results. The running time of the

matching algorithm is in line with the techniques currently found in literature. In the immediate future we are aiming at the extension of the proposed framework to both non-rigid and scale-invariant object recognition. We believe that such an extension could take place by introducing in the payoff function of the selection game a measure taking into account the geodesic path between the pairs of matching candidates, rather than attempting to preserve their Euclidean distance.

## ACKNOWLEDGMENTS

We wish to thank Dr. Samuele Salti for contributing code to compute SHOT descriptors, Prof. Ajmal S. Mian and Dr. Prabin Bariya for providing us with the experimental results used to compare our approach with their methods. We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.

## REFERENCES

[1] T. S. Newman and A. K. Jain, "A system for 3d cad-based inspection using range images," *Pattern Recognition*, vol. 28, no. 10, pp. 1555 – 1574, 1995.

[2] D. Borrmann, J. Elseberg, K. Lingemann, A. Nüchter, and J. Hertzberg, "Globally consistent 3d mapping with scan matching," *Robot. Auton. Syst.*, vol. 56, pp. 130–142, February 2008.

[3] Y.-K. Ahn, Y.-C. Park, K.-S. Choi, W.-C. Park, H.-M. Seo, and K.-M. Jung, "3d spatial touch system based on time-of-flight camera," *WSEAS Trans. Info. Sci. and App.*, vol. 6, pp. 1433–1442, 2009.

[4] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. of the International Conference on Computer Vision*, 1999, pp. 1150–1157.

[5] ——, "Distinctive image features from scale-invariant keypoints," in *Int. J. Comput. Vis.*, vol. 20, 2003, pp. 91–110.

[6] E. Akagündüz, O. Eskizara, and I. Ulusoy, "Scale-space approach for the comparison of hk and sc curvature descriptions as applied to object recognition," in *Proc. of the 16th IEEE International Conference on Image processing*, ser. ICIP'09, Piscataway, NJ, USA, 2009, pp. 413–416.

[7] A. Zaharescu, E. Boyer, K. Varanasi, and R. P. Horaud, "Surface feature detection and description with applications to mesh matching," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2009.

[8] C. S. Chua and R. Jarvis, "Point signatures: A new representation for 3d object recognition," *Int. J. Comput. Vision*, vol. 25, pp. 63–85, October 1997.

[9] Y. Sun, J. Paik, A. Koschan, and M. A. Abidi, "Point fingerprint: A new 3-d object representation scheme," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 33, pp. 712–717, 2003.

[10] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, 1999.

[11] H. Chen and B. Bhanu, "3d free-form object recognition in range images using local surface patches," *Pattern Recognition Letters*, vol. 28, pp. 1252–1262, July 2007.

[12] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *ECCV 2004, 8th European Conference on Computer Vision*, 2004, pp. 224–237.

[13] F. Tombari, S. Salti, and L. di Stefano, "Unique signatures of histograms for local surface description," in *ECCV 2010 - 11th European Conference on Computer Vision*, 2010, pp. 356–369.

[14] P. Bariya and K. Nishino, "Scale-hierarchical 3d object recognition in cluttered scenes," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010*, 2010, pp. 1657–1664.

[15] J. Novatnack and K. Nishino, "Scale-dependent/invariant local 3d shape descriptors for fully automatic registration of multiple sets of range images," in *Proc. of the 10th European Conference on Computer Vision*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 440–453.

[16] A. S. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, pp. 1584–1601, October 2006.

[17] ——, "On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes," *Int. J. Comput. Vision*, vol. 89, pp. 348–361, September 2010.

[18] A. Albarelli, S. R. Bulò, A. Torsello, and M. Pelillo, "Matching as a non-cooperative game," in *ICCV 2009: Proc. of the 2009 IEEE International Conference on Computer Vision*. IEEE Computer Society, 2009.

[19] A. Albarelli, E. Rodolà, and A. Torsello, "Robust game-theoretic inlier selection for bundle adjustment," in *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT2010)*, 2010.

[20] M. Pelillo and A. Torsello, "Payoff-monotonic game dynamics and the maximum clique problem," *Neural Computing*, vol. 18, pp. 1215–1258, May 2006.

[21] A. Albarelli, E. Rodolà, and A. Torsello, "Loosely distinctive features for robust surface alignment," in *ECCV 2010 - 11th European Conference on Computer Vision*, 2010, pp. 519–532.

[22] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. of the Optical Society of America. A*, vol. 4, no. 4, pp. 629–642, Apr 1987.

Int J Comput Vis
DOI 10.1007/s11263-011-0432-4

# Imposing Semi-Local Geometric Constraints for Accurate Correspondences Selection in Structure from Motion: A Game-Theoretic Perspective

**Andrea Albarelli · Emanuele Rodolà · Andrea Torsello**

Received: 8 September 2010 / Accepted: 7 March 2011


**Abstract** Most Structure from Motion pipelines are based on the iterative refinement of an initial batch of feature correspondences. Typically this is performed by selecting a set of match candidates based on their photometric similarity; an initial estimate of camera intrinsic and extrinsic parameters is then computed by minimizing the reprojection error. Finally, outliers in the initial correspondences are filtered by enforcing some global geometric property such as the epipolar constraint. In the literature many different approaches have been proposed to deal with each of these three steps, but almost invariably they separate the first inlier selection step, which is based only on local image properties, from the enforcement of global geometric consistency. Unfortunately, these two steps are not independent since outliers can lead to inaccurate parameter estimation or even prevent convergence, leading to the well known sensitivity of all filtering approaches to the number of outliers, especially in the presence of structured noise, which can arise, for example, when the images present several repeated patterns. In this paper we introduce a novel stereo correspondence selection scheme that casts the problem into a Game-Theoretic framework in order to guide the inlier selection towards a consistent subset of correspondences. This is done by enforcing geometric constraints that do not depend on full knowledge of the motion parameters but rather on some semi-local property that can be estimated from the local appearance of the image features. The practical effectiveness of the proposed approach is confirmed by an extensive set of experiments and comparisons with state-of-the-art techniques.

**Keywords** Inlier selection · Game-Theory · Structure from Motion

A. Albarelli (✉) · E. Rodolà · A. Torsello
Dipartimento di Scienze Ambientali, Informatica, Statistica, Università Ca' Foscari Venezia, Venice, Italy
e-mail: albarelli@unive.it

E. Rodolà
e-mail: rodola@dsi.unive.it

A. Torsello
e-mail: torsello@unive.it

## 1 Introduction

The common goal of all Structure from Motion (SfM) techniques is to infer as many 3D clues as possible by analyzing a set of 2D images. In general the 3D knowledge that can be obtained by such methods can be classified into two different (but related) classes: *scene* and *camera* information. Scene information is referred to the actual shape of the objects depicted in the images. This often boils down to assigning a plausible location in space to some significant subset of the acquired 2D points. These newly reconstructed 3D points are the "structure" part of SfM. By contrast, camera information includes all the parameters that characterize the abstract model of the image acquisition process. These can in turn be classified into intrinsic and extrinsic parameters. Intrinsic parameters are related to the physical characteristics of the camera itself, such as its focal length and principal point, while the extrinsic parameters define the camera pose, that is its position and rotation with respect to a conventional origin in the 3D space. Unlike the structure part, which is physically bound to a particular 3D configuration, the intrinsic and extrinsic parameters can vary in each shot; for this reason they are usually referred to as "motion".

Given the wide range of practical applications that could take advantage of a 3D reconstruction, it is not surprising that SfM has been a very active research topic during the last decades. In fact, many different approaches have been

Published online: 24 March 2011

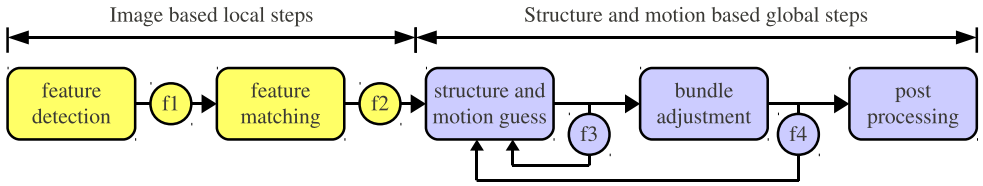



**Fig. 1** A simplified schema that captures the general steps found in many SfM approaches. The main loop is usually based on an iterative refinement of the candidate scene points based of their geometric consistency with respect to the estimated motion. *Circles* between steps represent the applied outlier filtering strategies

proposed in literature: some are aimed at solving the most general scenarios, others specialize to sub-domains, both in terms of the number of free parameters allowed and in terms of the assumptions made on some characteristics of the scene to be inferred. While the most relevant SfM approaches will be discussed with more detail in Sect. 2.3, in this section we will resort to the simplified general workflow presented in Fig. 1 in order to introduce the key ideas and contributions of the proposed approach. To this end, the typical pipeline can be roughly split in two subsequent macro steps (respectively dubbed as *Image based* and *Structure and Motion based* in Fig. 1). The first step deals with the localization in the source 2D images of salient feature points that are both distinctive and repeatable. Such points are meant to be tracked between different images, thus creating multiple sets of correspondences that will be used in the scene reconstruction step. The use of a reduced set of relevant points is crucial as their repeatable characterization allows us to minimize the chance of including wrong correspondences. Typically, filters are applied to the selection and matching phase in an attempt to make this phase more robust. In Fig. 1 the extracted features are further culled by using filter *f1*, which eliminates points that exhibit very common descriptors or that are not distinctive or stable enough. A second refinement can be achieved after the matching: most implementations of filter *f2* remove correspondences that are not reliable enough, that is pairs where the second best match has a very similar score to the first one or that involve too different descriptors. Once a suitable set of point pairs has been found among all the images, the second macro step of the pipeline uses them to perform the actual structure and motion estimation. This happens by building a reasonable guess for both the camera parameters and the spatial locations of the correspondences found, and then, almost invariably, by applying a bundle adjustment optimization to refine them. Also, at this stage, filtering techniques can be adopted in order to remove outliers from the initial set of matches. Specifically, a filter that removes pairs that do not agree with the estimated epipolar constraints can be applied after combining some or all the images into the initial guesses (*f3*), or after bundle adjustment optimized the structure and motion estimates (*f4*). Depending on the result of the filtering a new initial estimation can be triggered, taking advantage of the (hopefully) more accurate selection of corresponding features. This kind of process leads to an iterative refinement that usually stops when the inlier set does not change or becomes stable enough. While this approach works well in many scenarios, it inherently contains a limitation that might drive it to poor results or even prevent it from converging at all: The main criterion for the elimination of erroneous matches is to exclude points that exhibit a large reprojection error or adhere poorly to the epipolar constraint after a first round of scene and pose estimation. Unfortunately this afterthought is based upon an error evaluation that depends on the initial matches; this leads to a quandary that can only be solved by avoiding wrong matches from the start. This is indeed a difficult goal, mainly because the macro step from which the initial matches are generated is only able to exploit strictly local information, such as the appearance of a point or of its immediate surroundings. By contrast the following step would be able to take advantage of global knowledge, but this cannot be trusted enough to perform an extremely selective trimming and thus most methods settle with rather loose thresholds. In order to alleviate this limitation, in this paper we introduce a robust matching technique that allows to operate a very accurate inlier selection at an early stage of the process and without any need to rely on preliminary structure and motion estimations. This is obtained by enforcing properties that are inferable from image regions at a local or semi-local scale and then by extending their validation to a global scale. Similar approaches have already been used to obtain better camera pose estimations when dealing with complex multi-component scenes, where local observations can be handled in a decoupled way, thus leading to a better resilience to outliers (Fermuller et al. 1999). In this paper the inlier validation happens by casting the selection process into a Game-Theoretic setting, where feature-correspondences are allowed to compete with one another, receiving support from correspondences that satisfy the same semi-local constraints, and competitive pressure from the rest. The surviving correspondences form a small cohesive set of mutually compatible correspondences, satisfying the semi-local constraint globally. Of course many alternative selection techniques exist and can be adopted to perform the inlier set optimization, nevertheless the proposed Game-Theoretic approach offers the unique advan-

tage of a strong tendency to limit false negatives rather than concentrating on low false positives as most matching techniques in the literature. This property allows for a strong resilience to the large number of outliers normally encountered in general SfM scenarios. Further, the approach is quit3e general; in fact, in Sect. 3 we will show how the definition of different payoff functions between strategies leads to optimizers with task-specific goals. Finally, in order to assess the advantage provided by our approach, in the experimental section we compare our technique with a reference implementation of the structure-from-motion system presented in Snavely et al. (2006, 2008).

## 2 Background

Before discussing our robust matching approach we will briefly review the most significant related contributions available in literature and introduce some basic notions about the geometry of the SfM process.

### 2.1 Features Extraction and Matching

The selection of 2D point correspondences is arguably the most critical step in image based multi-view reconstruction and, differently from techniques augmented by structured light or known markers, there is no guarantee that pixel patches exhibiting strong photo consistency are actually located on the projection of the same physical point. Further, even when correspondences are correctly assigned, the interest point detectors themselves introduce displacement errors that can be as large as several pixels. Such errors can easily lead to sub-optimal parameter estimation or, in the worst cases, to the inability of the optimization algorithm to obtain a feasible solution. For this reasons, reconstruction approaches adopt several specially crafted expedients to avoid the inclusion of outliers as much as possible. In the first place correspondences are not searched throughout the whole image plane, but only points that are both repeatable and well characterized are considered. This selection is carried out by means of interest point detectors and feature descriptors. Salient points are localized with sub-pixel accuracy by general detectors, such as Harris Operator (Harris and Stephens 1988) and Difference of Gaussians (Marr and Hildreth 1980), or by using techniques that are able to locate affine invariant regions, such as Maximally Stable Extremal Regions (MSER) (Matas et al. 2004) and Hessian-Affine (Mikolajczyk and Schmid 2002). The affine invariance property is desirable since the change in appearance of a scene region after a small camera motion can be locally approximated with an affine transformation. Once interesting points are found, they must be matched to form the candidate pairs to be fed to the subsequent parameter optimization steps. Most of the currently used techniques for point matching are based on the computation of some affine invariant feature descriptor. Specifically, to each point is assigned a feature vector with tens to hundreds of dimensions, plus a scale and a rotation value. Among the most used feature descriptor algorithms are the Scale-Invariant Feature Transform (SIFT) (Lowe 1999, 2003), Speeded Up Robust Features (SURF) (Herbert and Gool 2006), Gradient Location and Orientation Histogram (GLOH) (Mikolajczyk and Schmid 2005) and more recently the Local Energy based Shape Histogram (LESH) (Sarfraz and Hellwich 2008), the SIFT algorithm being the first of the lot and arguably the most widely adopted. The complete SIFT technique, introduced and patented by Lowe, describes in detail both the detection step and the computation of repeatable descriptors to be associated with the found keypoints. Specifically, the localization of potentially relevant features happens by first applying to the image a Gaussian filter at different scales and then by selecting points that are maxima or minima of the Difference of Gaussians (DoG) that occur at multiple scales. This is done by comparing each pixel in the DoG images to its eight neighbors at the same scale and nine corresponding neighboring pixels in each of the neighboring scales. Subsequently the found candidates are interpolated to nearby data in order to ensure an accurate and repeatable position and thus they are filtered by discarding points that exhibit a low contrast or that are located along an edge (which could hinder the precision of the localization). Finally, an orientation based on the local image gradient is assigned to each one of the surviving points. The computation of the descriptor vector is then performed on the image closest in scale to the keypoint's scale and rotates accordingly to the keypoint's orientation. To this end, a set of histograms are computed based on the magnitude and orientation values picked from the neighborhood of the feature. The magnitudes are further weighted by a Gaussian function with $\sigma$ equal to half the width of the descriptor window. The histograms are then packed in a vector which is typically long 128 or 256 elements and that is normalized to unit length in order to enhance invariance to changes in illumination. Given the great success of the SIFT detector/descriptor, several enhancements and specializations were introduced since the original paper by Lowe; for instance, PCA-SIFT (Ke and Sukthankar 2004) applies PCA to the normalized gradient patch to gain more distinctiveness, PHOW (Bosch et al. 2007) makes the descriptor denser and allows to use color information, ASIFT (Morel and Yu 2009) extends the method to cover the tilt of the camera in addition to scale, skew and rotation. In all these techniques, the descriptor vector is robust with respect to affine transformations: i.e., similar image regions exhibit descriptor vectors with small mutual Euclidean distance. This property is used to match each point with the candidate with

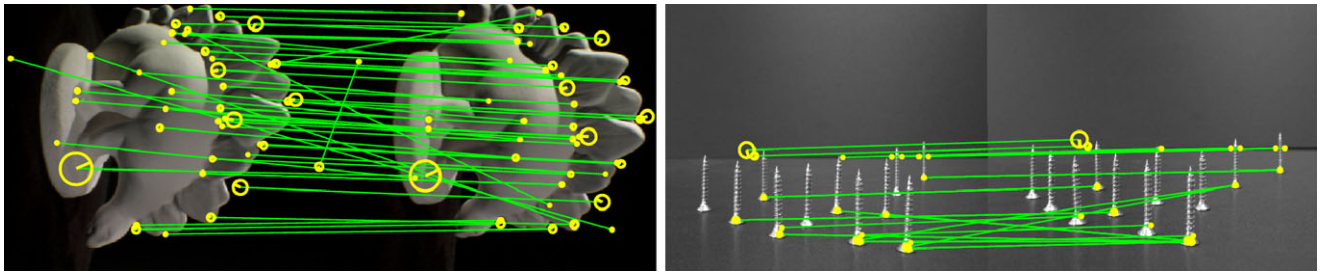


**Fig. 2** Example of SIFT features extracted and matched using the VLFeat package. Each feature in the *first image* has been matched with the feature in the *second image* that exhibits the most similar descriptor. Note that, while most of the correspondences are correct, many mismatches are still present

the nearest descriptor vector. However, if the descriptor is not distinctive enough this approach is prone to select many outliers. A common optimization involves the definition of a maximum threshold over the distance ratio between the first and the second nearest neighbors. In addition, points that are matched multiple times are deemed as ambiguous and discarded (i.e., one-to-one matching is enforced). Despite any effort made in this direction, any filter that operates at a local level is fated to fail when the matched regions are very similar or identical, a situation that is not uncommon as it happens every time an object is repeated multiple times in the scene or there is a repeated texture. In Fig. 2 we show two examples of SIFT features extracted and matched by using the VLFeat (Vedaldi and Fulkerson 2008) Matlab toolkit. In the first example almost all the correspondences are correct, still some clear mismatches are visible both between the plates of the saurus (which are similar in shape) and on the black background (which indeed contains some amount of noise). In the second example several identical screws are matched and, as expected, features coming from different objects are confused and almost all the correspondences are wrong. It should be noted that such mismatches are not a fault of the descriptor itself as it performs exactly its duty by assigning similar description vectors to features that are almost identical from a photometric standpoint. In fact, this particular example is specially crafted to break traditional matchers that rely on local properties. In the experimental section, we will show how introducing some level of global awareness in the process allows to deal well also with these cases that are indeed very common in the highly repetitive world of human-made objects and urban environments.

## 2.2 Camera Model and Epipolar Geometry

The pinhole projection (Fig. 3) is the most common camera model used in reconstruction frameworks. Its wide adoption is due to its ability to approximate well the behaviour of many real cameras. In practical scenarios radial and tangential lens distortions are the main sources of divergence from the pinole model, however it is easy to fit polynomial models to them and compensate for their effect (Tsai 1987; Weng et al. 1992). The most important parameters of this model are the pose of the camera with respect to the world (represented by a rotation matrix $R$ and a translation vector $T$), the distance of the projection center from the image plane (the focal length $f$ in Fig. 3), and the coordinates on the image plane of the intersection between the optical axis and the plane itself (the principal point $c = (c_x, c_y)^T$ in Fig. 3). The projection of a world point $m$ on the image plane happens in two steps. The first step is a rigid body transformation from the world coordinate system to the camera system. This can be easily expressed (using homogeneous coordinates) as:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \sim \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}.$$

The second step is the projection of the point in camera coordinates on the image planes, which happens by applying a camera calibration matrix $\mathbf{K}$ containing the intrinsic parameters of the model. The most general version of the calibration matrix allows for a different vertical ($f_y$) and horizontal ($f_x$) focal length to accommodate for non-square pixels, and for a skewness factor ($s$) to account for non-rectangular pixels:

$$\mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}.$$

In practice, for most real cameras, pixels can be approximated by perfect squares, thus we can resort to the basic model of Fig. 3 and assume $s = 0$ and $f_x = f_y = f$. Usually the camera pose and calibration matrices are combined into a single $3 \times 4$ projection matrix $\mathbf{P} = \mathbf{K}[\mathbf{R}\,\mathbf{T}]$. This projection matrix can be directly applied to a point in (homogeneous) world coordinates to obtain its corresponding 2D point on

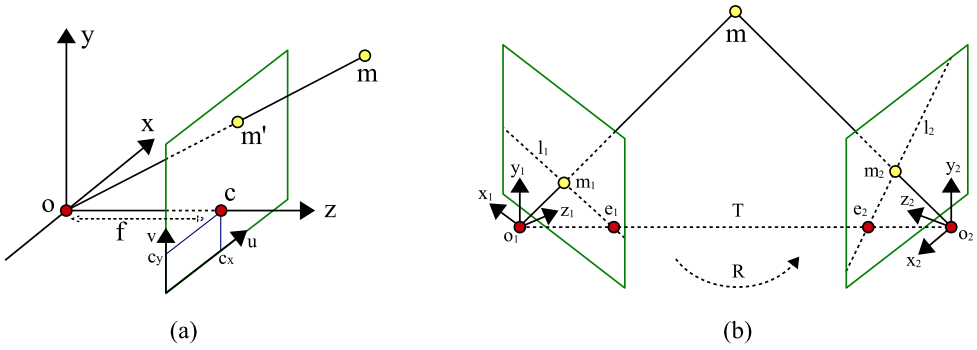



**Fig. 3** Illustration scheme of the pinhole camera model (**a**) and of the epipolar geometry (**b**). See text for details

the image plane:

$$\mathbf{m}' = \mathbf{Pm} = \mathbf{K}[\mathbf{R}\,\mathbf{T}]\mathbf{m}.$$

When a point is observed by two cameras its projections on the respective image planes are not independent. In fact, given the projection $m_1$ of point $m$ in the first camera, its projection $m_2$ on the second image plane must lie on the projection $l_2$ of the line that connects $m_1$ to $m$ (see Fig. 3). This line is called the *epipolar line* and can be found for each point $m_1$ in the first image plane by intersecting the plane defined by $o_1$, $o_2$ and $m_1$ (the *epipolar plane*) with the second image plane. The epipolar constraint can be enforced exactly only if the position of $m_1$ and $m_2$ and the camera parameters are known without error. In practice, however, there will always be some distance between a projected point and the epipolar line it should belong to. This discrepancy is a useful measure for verification tasks such as the detection of outliers among alleged matching image points, or the evaluation of the quality of estimated camera parameters. The epipolar constraint can be expressed algebraically in a straightforward manner. If we know the rotation matrix and translation vector that move one camera reference system to the other we have that:

$$\mathbf{x}_1^T\mathbf{E}\mathbf{x}_2 = \mathbf{x}_1^T \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \mathbf{R}\mathbf{x}_2 = 0,$$

where the *essential matrix* $\mathbf{E}$ is the product between the cross product matrix of the translation vector $\mathbf{T}$ and the rotation matrix $\mathbf{R}$, and $x_1$ and $x_2$ are points expressed in the reference systems of the first and second camera respectively, belonging to the same epipolar plane. If the calibration matrices of both cameras are known, this constraint can also be expressed in terms of image points by applying the inverse of the two calibration matrices to the image points:

$$(\mathbf{K}_1^{-1}\mathbf{m}_1)^T\mathbf{E}(\mathbf{K}_2^{-1}\mathbf{m}_2^T) = \mathbf{m}_1^T(\mathbf{K}_1^{-1T}\mathbf{E}\mathbf{K}_2^{-1})\mathbf{m}_2 = 0,$$

where $\mathbf{F} = \mathbf{K}_1^{-1T}\mathbf{E}\mathbf{K}_2^{-1}$ is called the *fundamental matrix*. It is clear that if intrinsic camera parameters are known the epipolar constraint can be verified on image points by using just the essential matrix, which has only five degrees of freedom; otherwise it is necessary to resort to the use of the fundamental matrix, which has seven degrees of freedom. Many algorithms are known to estimate both $\mathbf{E}$ or $\mathbf{F}$ from image point correspondences (Hartley 1995; Zhang et al. 1995; Torr and Zisserman 1998).

### 2.3 Structure from Motion

Structure from Motion (SfM) has been a core Computer Vision topic for a long time and a large number of different problem formulations and algorithms have been introduced over the last few decades (Aggarwal and Duda 1975; Weng et al. 1993; Zhang 1995). The distinctive traits of many SfM techniques recently proposed in literature are usually to be found in the approach used for the initial estimate and in the refinement technique adopted. In general this refinement happens by iteratively applying a bundle adjustment algorithm (Triggs et al. 2000) to an initial set of correspondences, 3D points and motion hypotheses. This optimization is almost invariably carried out by means of the Levenberg-Marquardt algorithm (Levenberg 1944), which is very sensitive to the presence of outliers in the input data. For this reason any possible care should be taken in order to supply the optimizer with good hypotheses or at least a minimal number of outliers. When a reasonable subset of all the points is visible in all the images global methods can be used to obtain such initial hypothesis. This approach, commonly called *factorization*, was initially proposed only for simplified camera models that are not able to fully capture the pinhole projection (Tomasi and Kanade 1992; Weinshall and Tomasi 1995). More recently, similar approaches have been presented also for perspective cameras (Sturm and Triggs 1996; Heyden et al. 1999), however their need for having each point visible in each camera severely reduces their usability in practical scenarios where occlusion is usually abundant. For this reason incremental methods, which allow to add one or a few images at a time, are by far more popular in SfM applications. Usually such methods start from a reliable image pair (for in-

stance the pair with the higher number of good correspondences), then an initial reconstruction is obtained by triangulation and finally extended sequentially. The extension can happen by virtue of common 2D points between a new camera and one or more images already in the batch. If internal camera parameters are known (at least roughly) rotation and translation direction can be extracted from the essential matrix and translation magnitude can be found using the projection in the new image of an already reconstructed 3D point. In the more general case intrinsic parameters are not known and the new camera can be added by exploiting the correspondences between its 2D features and previously triangulated 3D points to estimate the projection matrix (Beardsley et al. 1997; Pollefeys et al. 1999). Finally, it is possible to merge partial reconstructions by using corresponding 3D points (Fitzgibbon and Zisserman 1998). Many modern approaches iterate this process by including and excluding point correspondences or entire images by validating them with respect to the currently estimated structure and camera poses (Brown and Lowe 2005; Vergauwen and Van Gool 2006; Snavely et al. 2008).

## 3 Non-cooperative Games for Inlier Selection

The selection of matching points based on the feature descriptors is only able to exploit local information. This limitation conflicts with the richness of information that is embedded in the scene structure. For instance, under the assumption of rigidity and small camera motion, intuition suggests that features that are close in one view cannot be too far apart in the other one. Further, if a pair of features exhibit a certain difference of angles or ratio of scales, this relation should be maintained among their respective matches. Our basic idea is to formalize this intuitive notion of consistency between pairs of feature matches into a real-valued compatibility function and to find a large set of matches that express a high level of mutual compatibility. Of course, the ability to define a meaningful pairwise compatibility function and a reliable technique for finding a consistent set is at the basis of the effectiveness of the approach. Following (Torsello et al. 2006; Albarelli et al. 2009), we model the matching process in a Game-Theoretic framework, where two players select a pair of matching points from two images. Each player then receives a payoff proportional to how compatible his match is with respect to the other player's choice. Clearly, it is in each player's interest to pick matches that are compatible with those the other players are likely to choose. In general, as the game is repeated, players will adapt their behavior to prefer matchings that yield larger payoffs, driving all inconsistent hypotheses to extinction, and settling for an equilibrium where the pool of matches from which the players are still actively selecting their associations forms a cohesive set with high mutual support. Within this formulation, the solutions of the matching problem correspond to evolutionary stable states (ESS's), a robust population-based generalization of the notion of a Nash equilibrium. In a sense, this matching process can be seen as a contextual voting system, where each time the game is repeated the previous selections of the other players affect the future vote of each player in an attempt to reach consensus. This way the evolving context brings global information into the selection process. Since the evolutionary process is driven entirely by the payoff between strategies, it is clear that by adopting an appropriate compatibility function it is possible to suit the framework to achieve different goals. In this paper we will introduce two payoff functions to address our multi-view point matching problem. In Sect. 3.2 we will define a compatibility among pairs of correspondences that is proportional to the similarity of the affine transformation inferred from each match; this is done to exploit the expected local spatial and scale coherence among image patches. In Sect. 3.3 we will propose a refinement step that filters out groups of matches by letting them play an evolutionary game where the payoff is bound to their mutual ability to comply with the epipolar constraint.

### 3.1 Game-Theoretic Selection

Originated in the early 40's, Game Theory was an attempt to formalize a system characterized by the actions of entities with competing objectives, which is thus hard to characterize with a single objective function (Weibull 1995). According to this view, the emphasis shifts from the search of a local optimum to the definition of equilibria between opposing forces, providing an abstract theoretically-founded framework to model complex interactions. In this setting multiple players have at their disposal a set of strategies and their goal is to maximize a payoff that depends also on the strategies adopted by other players.

Here we will concentrate on symmetric two player games, i.e., games between two players that have the same set of available strategies and that receive the same payoff when playing against the same strategy. More formally, let $O = \{1, \ldots, n\}$ be the set of available strategies (*pure strategies* in the language of Game-Theory), and $C = (c_{ij})$ be a matrix specifying the payoffs, then an individual playing strategy $i$ against someone playing strategy $j$ will receive payoff $c_{ij}$. A *mixed strategy* is a randomization of the available strategies, i.e., a probability distribution $\mathbf{x} = (x_1, \ldots, x_n)^T$ over the set $O$. Clearly, mixed strategies are constrained to lie in the n-dimensional standard simplex

$$\Delta^n = \left\{ \mathbf{x} \in \mathbb{R}^n \ : \ x_i \geq 0 \text{ for all } i \in 1, \ldots, n, \sum_{i=1}^{n} x_i = 1 \right\}.$$

The *support* of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, is defined as the set of elements chosen with non-zero probability: $\sigma(\mathbf{x}) = \{i \in O | x_i > 0\}$. The expected payoff received by a player choosing element $i$ when playing against a player adopting a mixed strategy $\mathbf{x}$ is $(C\mathbf{x})_i = \sum_j c_{ij} x_j$, hence the expected payoff received by adopting the mixed strategy $\mathbf{y}$ against $\mathbf{x}$ is $\mathbf{y}^T C\mathbf{x}$. The *best replies* against mixed strategy $\mathbf{x}$ is the set of mixed strategies

$$\beta(\mathbf{x}) = \left\{ \mathbf{y} \in \Delta | \mathbf{y}^T C\mathbf{x} = \max_{\mathbf{z}} (\mathbf{z}^T C\mathbf{x}) \right\}.$$

The best reply is not necessarily unique. Indeed, except in the extreme case in which there is a unique best reply that is a pure strategy, the number of best replies is always infinite. A central notion of Game-Theory is that of a Nash equilibrium. A strategy $\mathbf{x}$ is said to be a *Nash equilibrium* if it is the best reply to itself, i.e., $\forall \mathbf{y} \in \Delta, \mathbf{x}^T C\mathbf{x} \geq \mathbf{y}^T C\mathbf{x}$. This implies that $\forall i \in \sigma(\mathbf{x})$ we have $(C\mathbf{x})_i = \mathbf{x}^T C\mathbf{x}$; that is, the payoff of every strategy in the support of $\mathbf{x}$ is constant. The idea underpinning the concept of Nash equilibrium is that a rational player will consider a strategy viable only if no player has an incentive to deviate from it.

We undertake an evolutionary approach to the computation of Nash equilibria. Evolutionary Game-Theory originated in the early 70's as an attempt to apply the principles and tools of Game-Theory to biological contexts. It considers an idealized scenario where pairs of individuals are repeatedly drawn at random from a large population to perform a two-player game. In contrast to traditional Game-Theoretic models, players are not supposed to behave rationally, but rather act according to a pre-programmed behavior, or mixed strategy. Further, it is supposed that some selection process operates over time on the distribution of behaviors favoring players that receive higher payoffs.

In this dynamic setting, the concept of stability, or resistance to invasion by new strategies, becomes central. A strategy $\mathbf{x}$ is said to be an *evolutionary stable strategy* (ESS) if it is a Nash equilibrium and

$$\forall \mathbf{y} \in \Delta \quad \mathbf{x}^T C\mathbf{x} = \mathbf{y}^T C\mathbf{x} \quad \Longrightarrow \quad \mathbf{x}^T C\mathbf{y} > \mathbf{y}^T C\mathbf{y}. \tag{1}$$

This condition guarantees that any deviation from the stable strategies does not pay.

The search for a stable state is performed by simulating the evolution of a natural selection process. Under very loose conditions, any dynamics that respect the payoffs is guaranteed to converge to Nash equilibria (Weibull 1995) and (hopefully) to ESS's; for this reason, the choice of an actual selection process is not crucial and can be driven mostly by considerations of efficiency and simplicity. We chose to use the replicator dynamics (Taylor and Jonker 1978), a well-known formalization of the selection process governed by the following equation

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) \frac{(C\mathbf{x}(t))_i}{\mathbf{x}(t)^T C\mathbf{x}(t)} \tag{2}$$

where $\mathbf{x}_i$ is the $i$-th element of the population and $C$ the payoff matrix.

A point $x$ is said to be a *stationary* (or equilibrium) point of our dynamical system, if $\dot{x}_i = 0$, for all $i = 1, \ldots, n$. A stationary point $x$ is said to be *asymptotically stable* if any trajectory starting sufficiently close to $x$ converges to $x$.

It can be shown (Weibull 1995) that a point $x \in \Delta$ is the limit of a trajectory of the replicator dynamics starting from the interior of $\Delta$ if and only if it is a Nash equilibrium. Further, if point $x \in \Delta$ is an ESS, then it is asymptotically stable for the replicator dynamics.

In our approach, we let matches compete with one another, each obtaining support from compatible associations and competitive pressure from all the others. The selection process is simulated by running the recurrence (2) and, at equilibrium, only pairings that are mutually compatible should survive and are then taken to be inliers.

### 3.2 Affine Preserving Matching Game

Central to this framework is the definition of a *matching game*, or, specifically, the definition of the strategies available to the players and of the payoffs related to these strategies. Given a set $M$ (model) of feature points in a source image and a set $D$ (data) of features in a target image, we call a *matching strategy* any pair $(a_1, a_2)$ with $a_1 \in M$ and $a_2 \in D$. We call the set of all the matching strategies $S \subseteq M \times D$. The total number of matching strategies in $S$ can, in theory, be as large as the Cartesian product of the sets of features detected in the images. Since most interest point detectors extract thousands of features from an image, a suitable selection should be made in order to keep its size limited. To this end we can exploit unary information such as the distance between descriptors or the photo-consistency of local image patches to select only feasible pairs. Specifically, for each source feature we can generate $k$ matching strategies that connect it to the $k$ nearest destination features in terms of descriptor distance. Since our Game-Theoretic approach operates inlier selection regardless of the descriptor, we do not need to set any threshold with respect to the absolute descriptor distance or the distinctiveness between the first and the second nearest point. In this sense, the only constraint that we need to impose over $k$ is that it should be large enough that we can expect the correct correspondence to be among the candidates for a significant proportion of the source features. In our preliminary work (Albarelli et al. 2010) we already analyzed the influence of $k$ over the quality of the matches obtained and we found that a very small

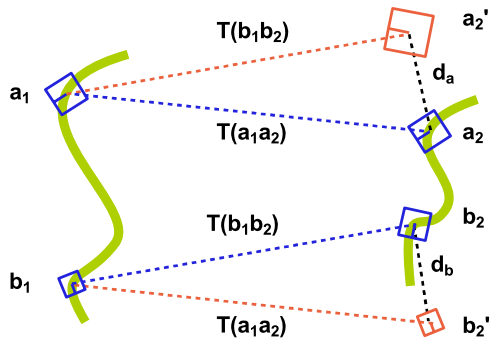



**Fig. 4** The payoff between two matching strategies is inversely proportional to the maximum reprojection error obtained by applying the affine transformation estimated by a match to the other

amount of candidates (typically 3 or 4) are enough to guarantee a satisfactory performance, however, in the presence of highly repeating patterns, a larger value might be needed. By reducing the number of correspondences per source feature to a constant value, we limit the growth of the number of strategies to be linear with the number of (source) features to be matched.

Once $S$ has been selected, our goal becomes to extract from it a large subset of correspondences that includes only correctly matched features: that is, strategies that associate a physical point in the source image with the same physical point (if visible) in the destination image. To this end, it is necessary to define a payoff function $\Pi : S \times S \to \mathbb{R}^+$ that exploits some pairwise information available at this early stage (i.e. before estimating camera and scene parameters) and that can be used to impose consistency globally. Since location, scale, and rotation are associated to each feature, we can associate to each correspondence $(a, b)$ between feature $a$ in the source image and feature $b$ in the target image a similarity transform $T(a, b)$ that maps the neighborhood of $a$ into the neighborhood of $b$, transforming the location, orientation, and scale measured in the source image into the location, orientation, and scale observed in the target image. Under small motion assumptions, we can expect these similarity transformations to be very similar locally. Thus, imposing the conservation of the similarity transform, we aim to extract clusters of feature matches that belong to the same region of the object and that tend to lie at the same level of depth. While this could seem to be an unsound assumption for general camera motion, in the experimental section we will show that it holds well with the typical disparity found in standard multiple view and stereo data sets. Further, it should be noted that with large camera motion, most, if not all, commonly used feature detectors fail, thus any inlier selection attempt becomes meaningless.

In order to define the payoff function $\Pi$ we need a way to measure the distance between similarity transforms. In order to avoid the problem of mixing incommensurable quantities, we compute the distance in terms of the reprojection error expressed in pixels. Specifically, given two matching strategies $(a_1, a_2)$ and $(b_1, b_2)$ and their respective associated similarities $T(a_1, a_2)$ and $T(b_1, b_2)$, we calculate virtual points $a_2'$ and $b_2'$ by applying the other strategy transformation to the source features $a_1$ and $b_1$ (see Fig. 4). More formally,

$$a_2' = T(b_1, b_2)a_1,$$
$$b_2' = T(a_1, a_2)b_1.$$

Given virtual points $a_2'$ and $b_2'$, we can measure the similarity between $(a_1, a_2)$ and $(b_1, b_2)$ as:

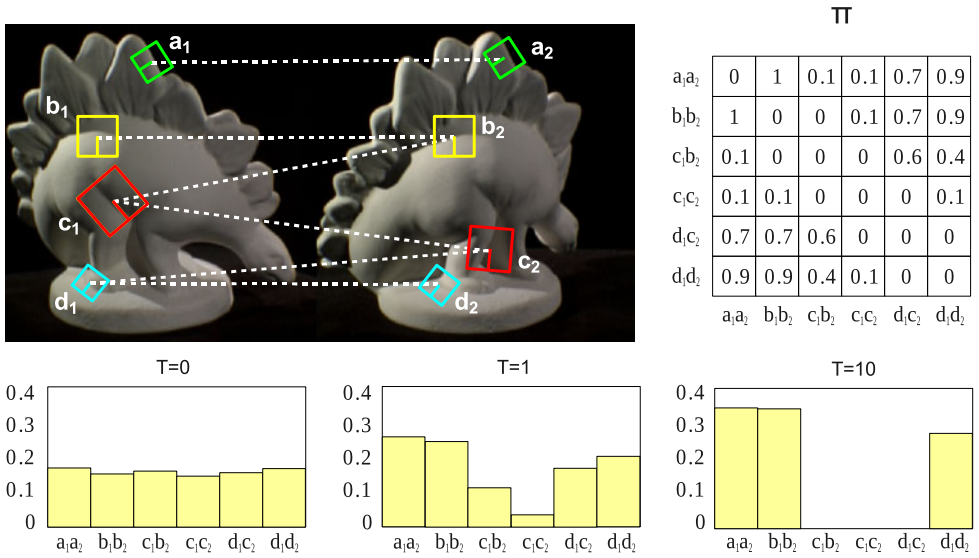
$$\text{sim}((a_1, a_2), (b_1, b_2)) = e^{-\lambda \max(|a_2 - a_2'|, |b_2 - b_2'|)} \tag{3}$$

where $\lambda$ is a selectivity parameter: If $\lambda$ is small, then the similarity function (and thus the matching) is more tolerant with respect to deviation in the similarity transformations, becoming more selective as $\lambda$ grows. Since each source feature can correspond with at most one destination point, it is desirable to avoid any kind of multiple match. It is easy to show that a pair of strategies with zero mutual payoff cannot belong to the support of an ESS (see Albarelli et al. 2009), thus any payoff function $\Pi$ can be easily adapted to enforce one-to-one matching by defining:

$$\Pi((a_1, a_2), (b_1, b_2)) = \begin{cases} \text{sim}((a_1, a_2), (b_1, b_2)), & a_1 \neq b_1, \\ & a_2 \neq b_2, \\ 0 & \text{else.} \end{cases} \tag{4}$$

We define payoff (4) a *similarity enforcing payoff function* and we call an *affine matching game* any symmetric two player game that involves a matching strategies set $S$ and a similarity enforcing payoff function $\Pi$.

The main idea of the proposed approach is that by playing a matching game driven by a similarity enforcing payoff function such as (4), the strategies (i.e. correspondence candidates) that share a similar locally affine transformation are advantaged from an evolutionary point of view and shall emerge in the surviving population. In Fig. 5 we illustrate a simplified example of this process. Once the population has reached a local maximum, all the non-extinct mating strategies can be considered valid, however, technically strategies become truly extinct only after an infinite number of iterations. Since we halt the evolution when the population ceases to change significantly, it is necessary to introduce some criteria to distinguish correct from non-correct matches. To avoid a hard threshold we chose to keep as valid all the played strategies whose population size exceeds a percentage of the most popular strategy. We call this percentage *quality threshold* ($q$). This criterion further limits




| Π | $a_1a_2$ | $b_1b_2$ | $c_1b_2$ | $c_1c_2$ | $d_1c_2$ | $d_1d_2$ |
|---|---|---|---|---|---|---|
| $a_1a_2$ | 0 | 1 | 0.1 | 0.1 | 0.7 | 0.9 |
| $b_1b_2$ | 1 | 0 | 0 | 0.1 | 0.7 | 0.9 |
| $c_1b_2$ | 0.1 | 0 | 0 | 0 | 0.6 | 0.4 |
| $c_1c_2$ | 0.1 | 0.1 | 0 | 0 | 0 | 0.1 |
| $d_1c_2$ | 0.7 | 0.7 | 0.6 | 0 | 0 | 0 |
| $d_1d_2$ | 0.9 | 0.9 | 0.4 | 0.1 | 0 | 0 |

**Fig. 5** An example of the affine-based evolutionary process. Four feature points are extracted from two images and a total of six matching strategies are selected as initial hypotheses. The matrix $\Pi$ shows the compatibilities between pairs of matching strategies according to a one-to-one similarity-enforcing payoff function. Each matching strategy got zero payoff with itself and with strategies that share the same source or destination point (i.e., $\Pi((b_1, b_2), (c_1, b_2)) = 0$). Strategies that are coherent with respect to a similarity transformation exhibit high payoff values (i.e., $\Pi((a_1, a_2), (b_1, b_2)) = 1$ and $\pi((a_1, a_2), (d_1, d_2)) = 0.9$)), while less compatible pairs get lower scores (i.e., $\pi((a_1, a_2), (c_1, c_2)) = 0.1$). Initially (at $T = 0$) the population is set to the barycenter of the simplex and slightly perturbed. After just one iteration, $(c_1, b_2)$ and $(c_1, c_2)$ have lost a significant amount of support, while $(d_1, c_2)$ and $(d_1, d_2)$ are still played by a sizable amount of population. After ten iterations ($T = 10$) $(d_1, d_2)$ has finally prevailed over $(d_1, c_2)$ (note that the two are mutually exclusive). Note that in the final population $((a_1, a_2), (b_1, b_2))$ have a larger support than $(d_1, d_2)$ since they are a little more coherent with respect to similarity

the number of selected strategies, but increases their consistency, since the population proportion is linked to the coherence of the strategy with the other surviving strategies. Each evolution process selects only a single group of matching strategies that are mutually coherent with respect to a local similarity transformation. This means that if we want to cover a large portion of the image we need to iterate the process many times, pruning the previously selected matches at each new iteration. Note that by imposing a minimal size for a group to be deemed as valid, the odds of recognizing structured outliers as false positives get lower. In fact, the probability of a large group to be coherent with respect to local affinity by chance is reduced as the minimal group size increases. Of course the usual trade-off between the desired precision and recall parameters must be taken into account when setting this kind of threshold.

### 3.3 Refinement by Epipolar Constraint Enforcement

The game formulation we just introduced shifts the matching problem to a more global scope by producing a set of correspondences between groups of features. While the affine camera model extracts very coherent groups, making such *macro features* more robust and descriptive than single points, in principle there is nothing that prevents the system to still produce wrong or weak matches. To reduce this chance we propose a different game setup that allows for a further refinement. In this game the strategies set $S$ corresponds to the set of paired feature groups extracted from the affine matching game and the payoff between them is related to the features' agreement to a common epipolar geometry. More specifically, given two pairs of matching groups $a \subseteq M \times D$ and $b \subseteq M \times D$, each one made up of model and data features, we estimate the epipolar geometry from $a \cup b$ and define the payoff among them as:

$$\Pi(a, b) = e^{-\lambda \sum_{(s,t)\in a\cup b} d(t,l(s))} \tag{5}$$

where $l(p)$ is a function that gives the epipolar line in the data image from the feature point $p$ in the model image, according to the estimated epipolar geometry, and $d(p, l)$ calculates the distance between point $p$ and the epipolar line $l$. It is clear that this distance is small (and thus the payoff is big) if the two groups share a common projective interpretation and large otherwise. Of course, different pairs of groups can agree on different epipolar geometry, but the transitive closure induced by the selection process ensures that the strategies in the surviving population will agree on the same (or very similar) projective transformation (see Fig. 6 for a complete example of this process). Regarding the estimation of the epipolar geometry, it can be done in two different ways: if we have at least the intrinsic calibration of the camera we can estimate the essential matrix, by contrast, if we

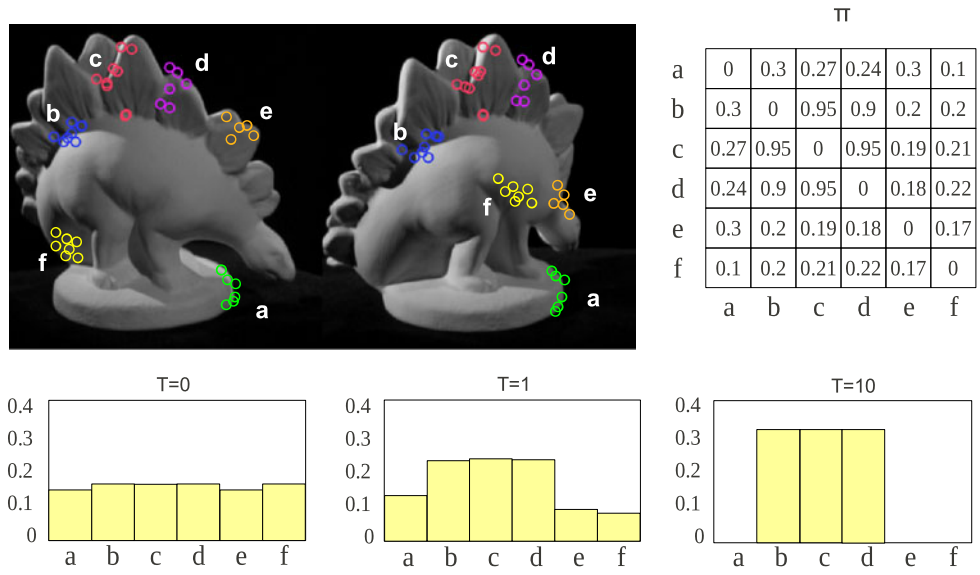



**Fig. 6** An example of the selection of groups of features that agree with respect to a common epipolar geometry. Six matching groups are selected by the affine matching step (labelled from $a$ to $f$ in the figure). Each pair of feature sets is modeled as a matching strategy and the payoff among them is reported in matrix $\Pi$. Note that groups $b$, $c$ and $d$ are correctly matched and thus exhibit a high mutual payoff. By contrast, group $a$ (which is consistent both in terms of photometric and affine properties), $e$ and $f$ are clearly mismatched with respect to the overall scene geometry, which in turn leads to a large error on the epipolar check and thus to a low score in the payoff matrix. At the beginning of the evolutionary process each strategy obtains a fair amount of players ($T = 0$). As expected, after just one iteration of the replicator dynamics the most consistent strategies ($b$, $c$ and $d$) obtain a clear advantage. Finally, after ten iterations ($T = 10$) the other groups have no more support in the population and only the correct matches survived

do not have any hint about the camera geometry, we must resort to a more relaxed set of constraints and use the fundamental matrix instead. In the experimental section we will test both scenarios.

## 4 Experimental Results

We performed an extensive set of tests in order to validate the proposed techniques and to explore their limits. Both quantitative and qualitative results are shown and performances are compared with those achieved by a standard baseline method, i.e. the default feature matcher in the Bundler suite (Snavely et al. 2008).

### 4.1 General Setup and Data Sets

All the following experiments have been made by applying a common basic pattern: first a set of features is extracted from the images by using the SIFT keypoint detector made freely available in Lowe (2003), then these interest points are paired using the matcher we want to test, finally scene and camera parameters are estimated by using the final portion of Bundler pipeline (i.e. the part of the suite that applies Levenberg-Marquardt optimization to a set of proposed matches). We evaluate three different approaches: The first, referred to as Affine Game-Theoretic approach (AGT), uses the affine matching game without the further refinement provided by the enforcement of the epipolar geometry. In this case the iterative extraction and elimination of the groups is image-based, i.e., after a group of matches is selected, all the matches that have sources or targets close to the source and target points of the extracted correspondences are eliminated, and then the evolutionary process is reiterated on the reduced set of strategies. The process is stopped when an extracted group is smaller than a given threshold or has average payoff smaller than a given threshold. This approach is the same described in Albarelli et al. (2010). The second and third approaches, referred to as Calibrated Projective Game-Theoretic approach (CPGT) and Uncalibrated Projective Game-Theoretic approach (UPGT) respectively, make use of the epipolar refinement. CPGT assumes that the camera intrinsic parameters are (approximately) known and estimate the epipolar geometry through the essential matrix, while UPGT uses the fundamental matrix. In both these approaches the iterative extraction and elimination of the groups is strategy-based, i.e., after a group of matches is selected only those matches are eliminated from the strategy set, thus allowing for the same features to appear in several groups, while the stopping criterion here is the same as that of AGT. In our experiments the intrinsic parameters for CPGT have been estimated from the images EXIF information. The three approaches are compared against the default feature matcher in the Bundler suite (BKM). This is a

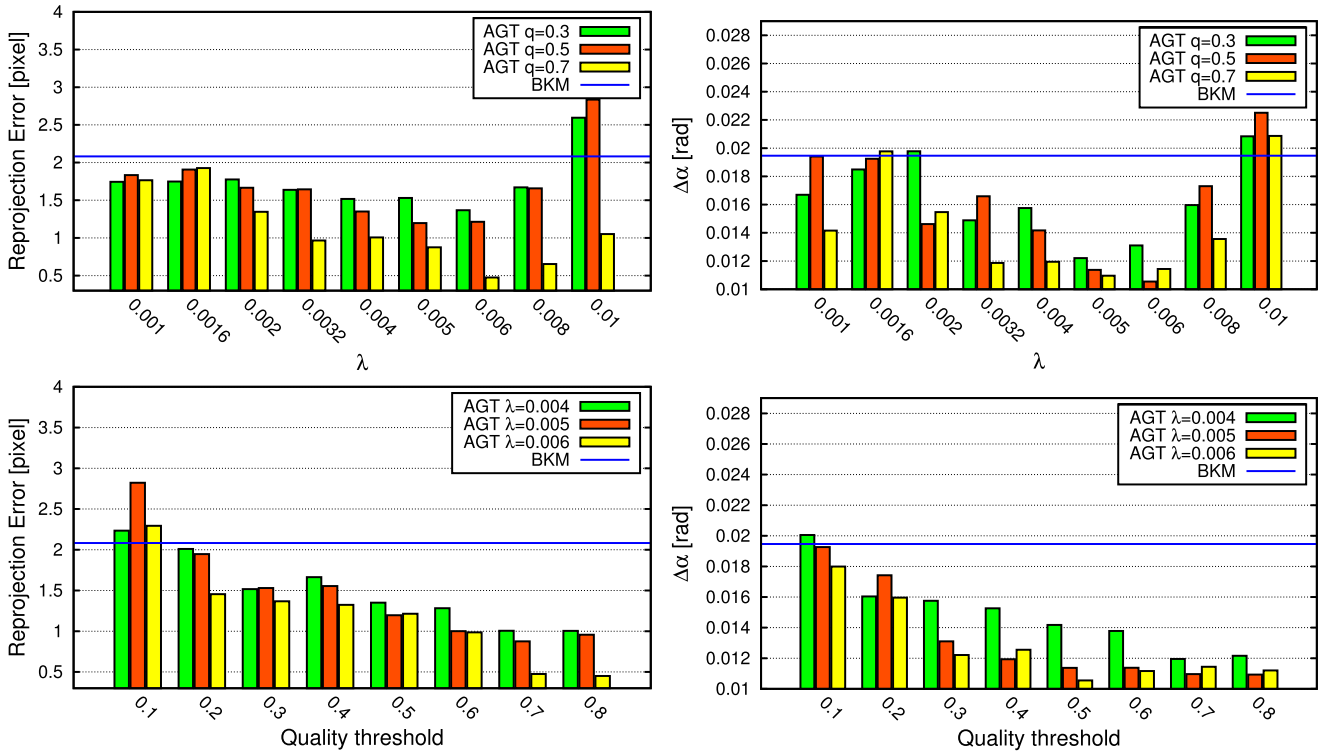



**Fig. 7** Analysis of the performance of the Affine Game-Theoretic approach with respect to variation of the parameters of the algorithm

reasonable choice for several reasons: BKM is optimized to work with SIFT descriptors and, obviously, with the Bundler suite; in addition it is very popular in literature since Bundler itself has been used as the default matcher in many of the recent papers about SfM and dense stereo reconstruction. For each test we evaluated two quality measures: the average reprojection error (expressed in pixels) and the differences in radians between the ground-truth and the estimated rotation angle ($\Delta\alpha$). The first measure aims to capture the cumulative error made in the reconstruction of the structure and the estimation of the motion, while the second measure aims to decouple the error on the camera orientation from the one related to the scene reconstruction. This is possible since we used images pairs coming from a calibrated camera head or image sets with an available ground-truth. Specifically we used a pair of cameras previously calibrated through a standard procedure and took stereo pictures of 20 different, isolated objects; in addition we also included in the data set the shots coming from the "DinoRing" and "TempleRing" sequences from the Middlebury Multi-View Stereo dataset (Seitz et al. 2006). We conducted two main sets of experiments. The goal of the first set is to analyze the impact of the parameters, namely $\lambda$ and *quality threshold* ($q$), over the accuracy of the results. Since AGT and CPGT/UPGT have different payoff functions and the selectivity $\lambda$ is not directly comparable we investigate its influence separately. In addition, all the experiments regarding the refinement methods are made using very relaxed parameters for the AGT step. This is due to the fact that we are willing to accept a slightly higher number of outliers in the first step in exchange for a higher number of candidate groups, in the hope that the refinement process is able to eliminate the spurious groups, but still resulting in a larger number of good correspondences from which to perform parameter estimation. In the second batch of experiments we compare our techniques with the default Bundler matcher. In these experiments the parameters are set to the optimal values estimated previously. We provide both quantitative and qualitative results: the quantitative analysis is based on the errors in reprojection and motion estimation, while the qualitative results are based on a dense reconstruction obtained using the recovered parameters as an input to the PMVS suite (Furukawa and Ponce 2010).

### 4.2 Influence of Parameters

The AGT method depends on two explicit parameters: the sensitivity parameter $\lambda$, which modulates the steepness of the payoff function (4), and $q$, i.e. the percentage of population density with respect to the most represented strategy that one match must obtain to be considered not-extinct. As stated in Sect. 3.2, $\lambda$ controls the selectivity of the selection process, while $q$ allows to further filter the extracted group based on its cohesiveness. Higher values will lead

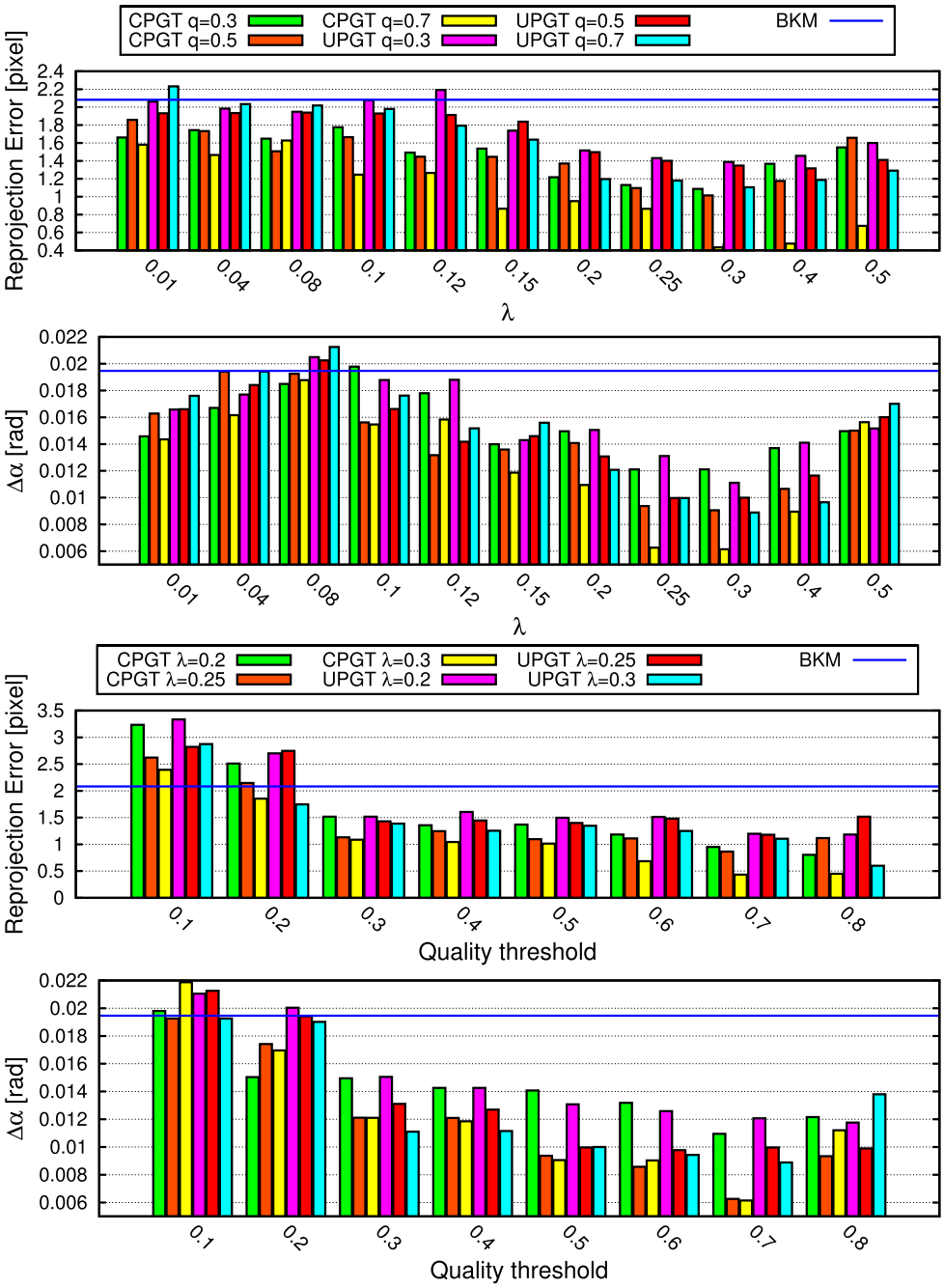



**Fig. 8** Analysis of the performance of the Calibrated and Uncalibrated Projective Game-Theoretic approaches with respect to variation of the parameters of the algorithm

to a more selective culling, while lower values will allow more strategies to pass the screening. Figure 7 reports the results of these experiments averaged over the full set of 20 stereo pairs taken with a previously calibrated camera pair. The first row shows the effect of the selectivity parameter λ. This is evaluated for three different $q$ levels, from 0.3 to 0.7. As expected, both low and high values lead to larger errors, mainly with respect to the estimation of the angle between the two cameras. This is probably due to a too tight and a too relaxed enforcement of local coherence respectively. It could be argued that the estimation of the optimal λ can be tricky in practical situations; however, we must note that, with a reasonable high $q$, it takes a very large sensitivity parameter to obtain a worse performance than that obtained with the default Bundler matcher. Regarding the quality threshold, we can see in the second row of Fig. 7 that the best results are achieved by setting a high level of quality: this is clearly due to the fact that, in practice, the replicator dynamics have converged to a stable ESS and thus most of the non-zero strategies are indeed inliers and are mostly subject only to the (small) feature localization error, thus exhibiting an equally high density. In Fig. 8 we show the results obtained

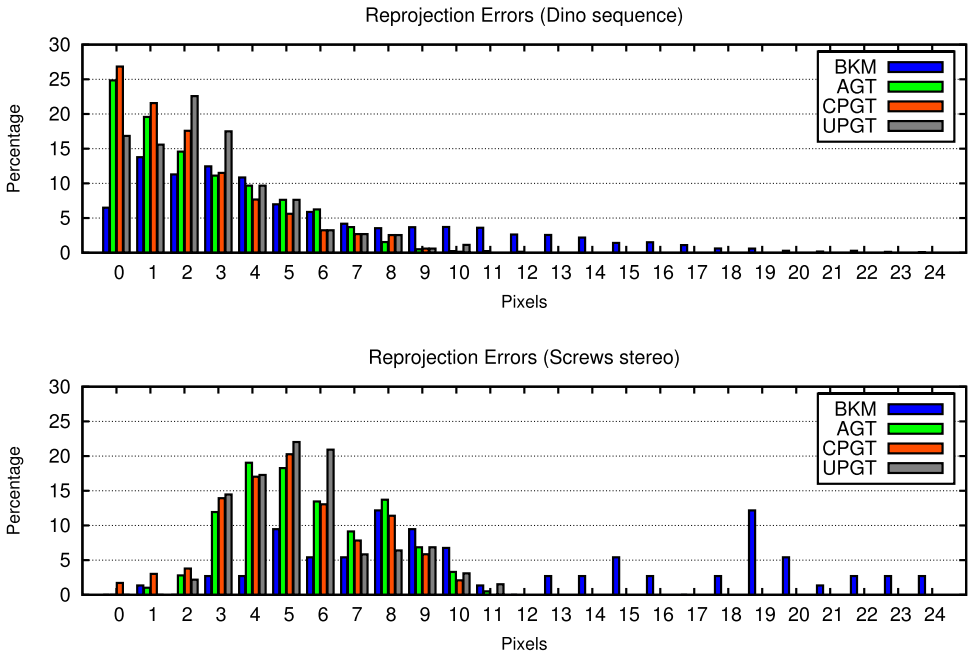
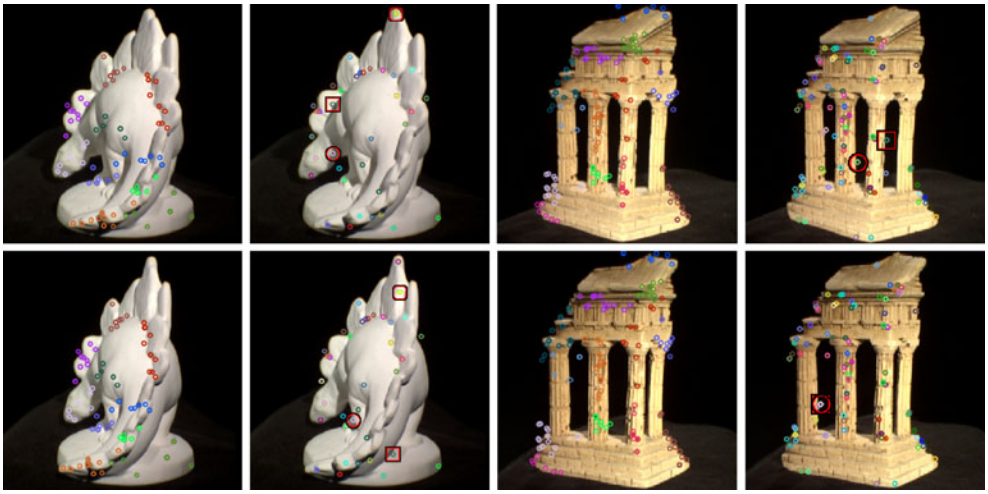



**Fig. 9** Distribution of the reprojection error in one multiple view (*top*) and one stereo pair (*bottom*) example

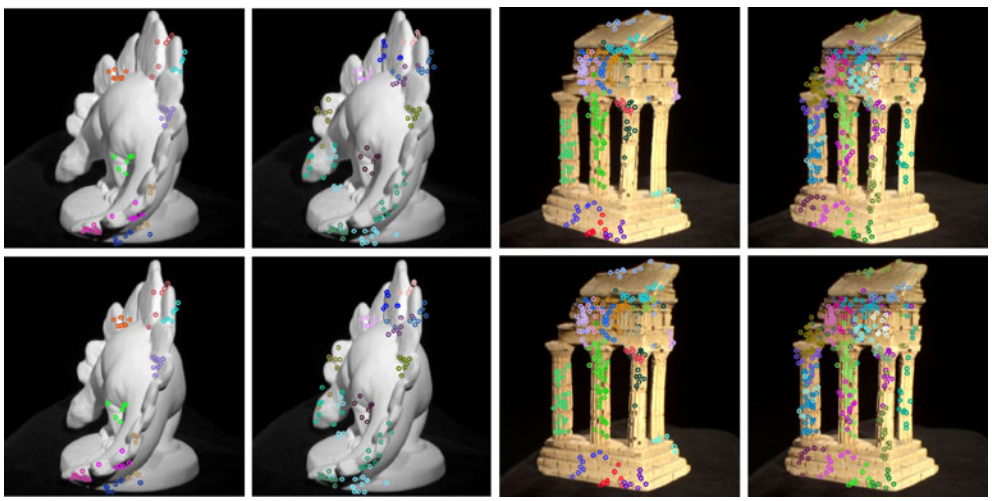
by trying different parameters with CPGT and UPGT. As previously stated, these experiments were made by performing an affine matching step with relaxed parameters: namely a $\lambda$ value of 0.09 and a $q$ of 0.6. The overall behavior with respect to these parameters is similar to what observed with AGT: very low and very high values for $\lambda$ lead to less satisfactory results (whereas in general better than those obtained with the Bundler key matcher), and high $q$ seems to guarantee good estimates. Overall it seems that CPGT always gives better results than UPGT. We will analyze this behavior with more detail in the next section.

### 4.3 Comparisons Between Approaches

To further explore the differences among the proposed techniques and the Bundler matcher, we executed two sets of experiments. The first set applies the approaches to unordered images coming from the DinoRing and TempleRing sequences from the Middlebury Multi-View Stereo dataset for these models, the camera extrinsic parameters are provided and used as a ground-truth. The rationale for using these sets (in opposite to simple stereo pairs) is to allow Bundler to optimize the parameters and correspondences over the complete sequence. The second set is composed of two calibrated stereo scenes selected from the previously acquired collection of 20 items, specifically a statue of Ganesha and a handful of screws placed on a table. For all the sets of experiments we evaluated both the rotation error of all the cameras and the reprojection error of the detected feature points. In the Middlebury sets the results are presented as averages. The Dino model is a difficult case in general, as it provides very few distinctive features; the upper part of Fig. 10 shows the correspondences produced by AGT (left column) in comparison with BKM (right column). The parameters were set to the optimal values estimated in the previous experiments ($\lambda = 0.06$ and $q = 0.8$). This resulted in the detection of many correct matches organized in groups, each corresponding to a different depth level, and visualized with a unique color in the figure. As can be seen, the different depth levels are properly estimated; this is particularly evident throughout the arched back going from the tail (in foreground) to the head of the model (in background), where clustered sets of feature points follow one after the other. Furthermore, these sets of interest points maintain the right correspondences within the pair of images. The Bundler matcher on the other hand, while still achieving good results in the whole process, also outputs erroneous correspondences (marked in the figure). In the lower part of Fig. 10 we can see the results obtained with CPGT and UPGT with $\lambda = 0.3$ and $q = 0.7$ after an affine matching step performed with $\lambda = 0.09$ and $q = 0.9$. We can observe that CPGT gives a significant boost to all the statistics. By contrast UPGT performed worse than AGT (albeit still better than BKM). This is probably due to the higher number of degrees of freedom in the estimation of the fundamental matrix and, thus, to the reduced ability to discriminate incompatible groups. In fact, we can see that the size of the groups obtained with AGT is generally rather small (from 4 to about 10 points), and it is easy to justify such a small number of correspondences under a common fundamental matrix. The quality of reconstruction following the application of all methods can be compared visually by looking at the distribution of the reprojection error in the top row of Fig. 9. While most reprojections fall within 1–3 pixels for the Game-Theoretic ap-

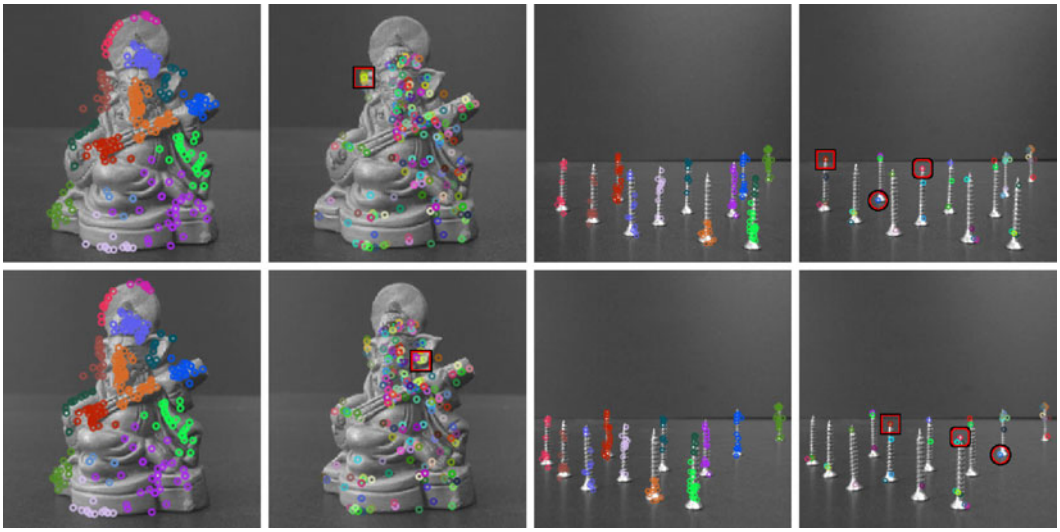


| | | Dino sequence | | Temple sequence | |
|---|---|---|---|---|---|
| | | AGT | BKM | AGT | BKM |
| Matches | | 14573 | 9245 | 25785 | 22317 |
| $\epsilon$ | $\leq$ 1 pix | 24.83 | 6.49406 | 22.6049 | 24.6729 |
| | $\leq$ 5 pix | 54.94 | 48.3659 | 62.7737 | 61.8957 |
| | $\geq$ 5 pix | 20.21 | 45.1401 | 14.6214 | 13.4314 |
| | Avg. | 2.3086 | 4.5255 | 2.3577 | 2.3732 |
| $\Delta\alpha$ | Avg. | 0.005751 | 0.005561 | 0.010514 | 0.009376 |
| | S. dev. | 0.003242 | 0.003184 | 0.005282 | 0.004646 |
| | Max | 0.012057 | 0.011475 | 0.021527 | 0.017016 |
| Avg. levels | | 8.42 | - | 9.27 | - |



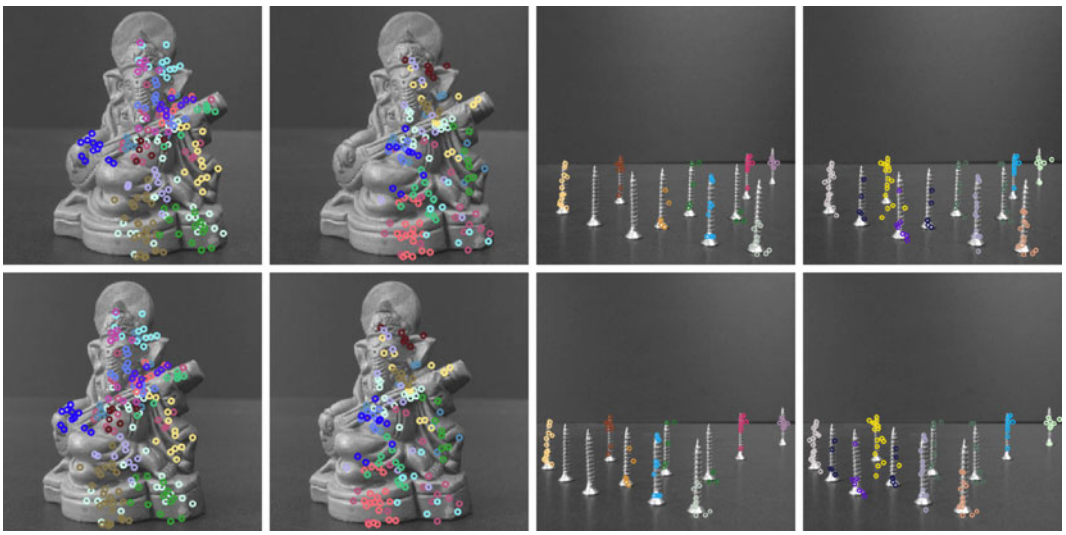
| | | Dino sequence | | Temple sequence | |
|---|---|---|---|---|---|
| | | CPGT | UPGT | CPGT | UPGT |
| Matches | | 15018 | 15231 | 28106 | 28407 |
| $\epsilon$ | $\leq$ 1 pix | 32.1731 | 20.0126 | 25.7232 | 18.3715 |
| | $\leq$ 5 pix | 61.4826 | 75.4671 | 64.5294 | 78.5347 |
| | $\geq$ 5 pix | 6.3518 | 4.5203 | 9.7474 | 3.0938 |
| | Avg. | 1.7051 | 2.9841 | 2.1642 | 3.6713 |
| $\Delta\alpha$ | Avg. | 0.004823 | 0.006437 | 0.009411 | 0.01328 |
| | S. dev. | 0.003671 | 0.004514 | 0.005143 | 0.006545 |
| | Max | 0.013147 | 0.017421 | 0.019725 | 0.027832 |
| Avg. levels | | 17.21 | 18.34 | 20.13 | 22.05 |

**Fig. 10** (Color online) Results obtained with two multiple view data sets



**Ganesha stereo** **Screws stereo**

| | AGT | BKM | AGT | BKM |
|---|---|---|---|---|
| Matches | 280 | 200 | 211 | 46 |
| $\epsilon$ $\leq$ 1 pix | 98.2824 | 20 | 0 | 0 |
| $\leq$ 5 pix | 1.7175 | 80 | 34.7716 | 6.75676 |
| $\geq$ 5 pix | 0 | 0 | 65.2284 | 93.2432 |
| Avg. | 0.321248 | 1.67583 | 5.86237 | 10.2208 |
| $\Delta\alpha$ | 0.001014 | 0.007424 | 0.020822 | 0.030995 |
| Levels | 14 | - | 12 | - |



**Ganesha stereo** **Screws stereo**

| | CPGT | UPGT | CPGT | UPGT |
|---|---|---|---|---|
| Matches | 315 | 282 | 72 | 108 |
| $\epsilon$ $\leq$ 1 pix | 99.0017 | 83.4812 | 2.1637 | 0 |
| $\leq$ 5 pix | 0.9983 | 16.5188 | 37.5721 | 26.3417 |
| $\geq$ 5 pix | 0 | 0 | 60.2642 | 73.6583 |
| Avg. | 0.300272 | 1.2311 | 3.92133 | 4.6379 |
| $\Delta\alpha$ | 0.001623 | 0.00466 | 0.025341 | 0.03945 |
| Levels | 15 | 13 | 8 | 9 |

**Fig. 11** (Color online) Results obtained with two stereo view data sets

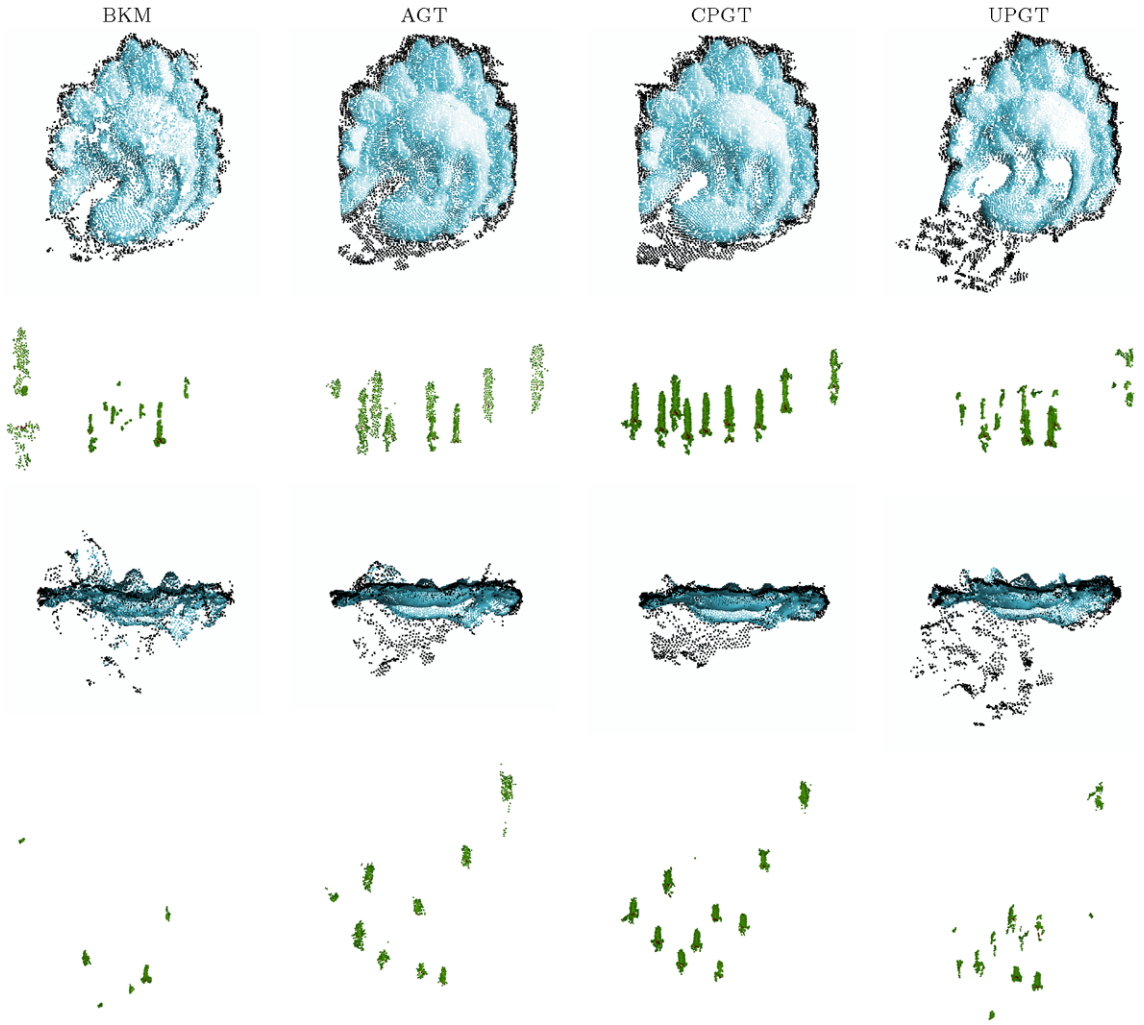



**Fig. 12** Comparisons of the point clouds produced by PMVS using the motion estimated with different matching methods. Respectively the Bundler default keymatcher (BKM), the Affine Game-Theoretic technique (AGT) and the calibrated and uncalibrated projective techniques (CPGT and UPGT)

proaches, the Bundler matcher exhibits a long-tailed trend, with reprojection errors reaching 20 pixels. Unlike the Dino model, the Temple model is quite rich of features: for visualization purposes we only show a subset of the detected matches for all the techniques. While the effectiveness of our approaches is not negatively impacted by the model characteristics, several mismatches are extracted by BKM. In particular, the symmetric parts of the object (mainly the pillars) result in very similar features and this causes the matcher to establish one-to-many correspondences over them. In the calibrated stereo scenario, the Ganesha images are rich of distinctive features and pose no particular difficulty to any of the methods. The Bundler matcher provides very good results, with only one evident false match out of a total of 200 matches (see Fig. 11). The resulting bundle adjustment is quite accurate, giving very small rotation errors and reprojection distances. Nevertheless, our methods perform considerably better: reprojection errors dramatically decrease, with around 98 percent of the feature points falling below one pixel of reprojection error for AGT and 99 percent for CPGT. Unfortunately UPGT is unable to refine the results obtained with AGT, but still achieves smaller errors than BKM. The second calibrated stereo scene, "Screws stereo", is an emblematic case and provides some meaningful insight. The images depict a dozen screws standing on a table, placed by hand at different depth levels. This configuration, together with the abundance of features, should provide enough information for the algorithms to extract significant matches. However, the scene is a difficult one due to the very nature of the objects depicted, which are all identical and highly symmetric, resulting in several features with very similar descriptors and a difficulty in extracting good matches based only on photometric information. Indeed, several false matches are established by the Bundler

matcher (see the last column of Fig. 11). Still, BKM results in a reasonable estimation of the rigid transformation linking the two cameras, as erroneous pairings are removed *a posteriori* during the subsequent phases of bundle adjustment. By contrast, the AGT approach outputs large and accurate sets of matches, roughly one per object, and even difficult cases, such as the left-right parallactic swaps taking place at the borders are correctly dealt with. It is interesting to note that in this case the boost given by CPGT is even more significant than in the previous experiments, with a lower average reprojection error and an overall better error distribution. Unlike with the previous cases, this happens by reducing the number of total matches rather than increasing it, as the refinement process eliminates correspondences that are not globally consistent. In addition this time even UPGT gives better results than AGT: a histogram of the reprojection errors for this object is shown in Fig. 9. Finally, a qualitative analysis of the different approaches is shown in Fig. 12, where the estimated parameters and correspondences are fed to the PMVS dense multiview stereo reconstruction tool. The first and the second rows show the Dino and Screws scenes from a frontal view, while the other two show a top view of the same scenes. AGT and CPGT give the best results for Dino with CPGT providing a more correct representation of the hollow area between the neck and the first leg of the figurine and a smaller number of spurious points. With the screws scene CPGT allows by far the more consistent reconstruction, while BKM is substantially unable to offer to PMVS a satisfactory pose estimation.

### 4.4 Complexity and Running Time

With respect to complexity all the Game-Theoretic approaches are dominated by the steps of the replicator dynamics. Each step is quadratic in the number of strategies, but there is no guarantee about the total number of steps that are needed to reach an ESS. We chose to stop the iterations when the variation of the population was below a minimum threshold. Execution times for the matching steps of our technique are plotted in Fig. 13; the scatter plot shows a weak quadratic growth of convergence time as the number of matching strategies increases with a very small constant in the quadratic term, resulting in computation times below half a second even with a large number of strategies.

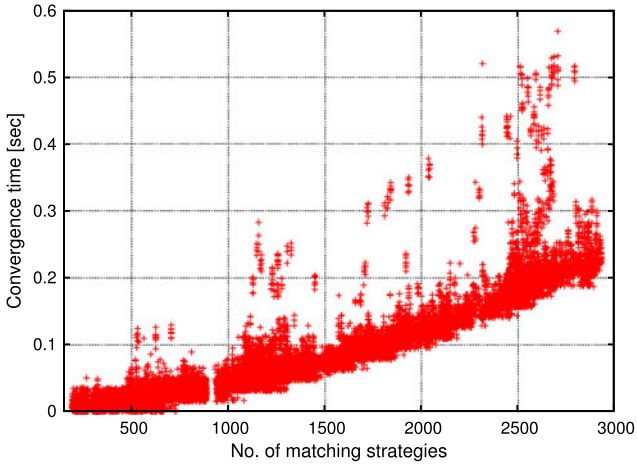



**Fig. 13** Plot of the convergence time of the replicator dynamics with respect to the number of matching strategies

## 5 Conclusions

In this paper we introduced a novel Game-Theoretic technique that performs an accurate feature matching as a preliminary step for multi-view 3D reconstruction using Structure from Motion techniques. Unlike other approaches, we do not rely on a first estimation of scene and camera parameters in order to obtain a robust inlier selection, but rather, we enforce geometric constraints based only on semi-local properties that can be estimated from the images. In particular, we define two selection games, one that selects local groups of compatible correspondences, enforcing a weak affine camera model, and a second consolidation game that filters out groups of matches by considering their compliance with the epipolar constraint. Experimental comparisons with a widely used technique show the ability of our approach to obtain a tighter inlier selection and thus a more accurate estimation of the scene parameters.

**Acknowledgements** We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.

## References

Albarelli, A., Rota Bulò, S., Torsello, A., & Pelillo, M. (2009). Matching as a non-cooperative game. In *Proc. IEEE international conference on computer vision—ICCV'09*.

Albarelli, A., Rodolà, E., & Torsello, A. (2010). Robust game-theoretic inlier selection for bundle adjustment. In *Proc. 3D data processing, visualization and transmission—3DPVT'10*.

Aggarwal, J. K., & Duda, R. O. (1975). Computer analysis of moving polygonal images. *IEEE Transactions on Computers*, *24*, 966–976.

Beardsley, P. A., Zisserman, A., & Murray, D. W. (1997). Sequential updating of projective and affine structure from motion. *International Journal of Computer Vision*, *23*(3), 235–259.

Bosch, A., Zisserman, A., & Munoz, X. (2007). Image classification using random forests and ferns. In *Proc. 11th IEEE international conference on computer vision—ICCV'07* (pp. 1–8).

Brown, M., & Lowe, D. G. (2005). Unsupervised 3d object recognition and reconstruction in unordered datasets. In *3DIM'05: Proceedings of the fifth international conference on 3-D digital imaging and modeling* (pp. 56–63). Los Alamitos: IEEE Computer Society.

Fermuller, C., Brodsky, T., & Aloimonos, Y. (1999). Motion segmentation: a synergistic approach. In *IEEE computer society conference on computer vision and pattern recognition* (Vol. 2, pp. 637–643).

Fitzgibbon, A. W., & Zisserman, A. (1998). Automatic camera recovery for closed or open image sequences. In *ECCV'98: Proceedings of the 5th European conference on computer vision* (Vol. I, pp. 311–326). Berlin: Springer.

Furukawa, Y., & Ponce, J. (2010). Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*, 1362–1376. doi:10.1109/TPAMI.2009.161.

Harris, C., & Stephens, M. (1988). A combined corner and edge detector. In *Proc. fourth Alvey vision conference* (pp. 147–151).

Hartley, R. I. (1995). In defence of the 8-point algorithm. In *Proceedings of IEEE international conference on computer vision* (pp. 1064–1070). Los Alamitos: IEEE Comput. Soc.

Herbert Bay, T. T., & Gool, L. V. (2006). SURF: Speeded up robust features. In *9th European conference on computer vision* (Vol. 3951, pp. 404–417).

Heyden, A., Berthilsson, R., & Sparr, G. (1999). An iterative factorization method for projective structure and motion from image sequences. *Image and Vision Computing*, *17*(13), 981–991.

Ke, Y., & Sukthankar, R. (2004). PCA-SIFT: a more distinctive representation for local image descriptors. In *Proc. IEEE comp. soc. conf. on computer vision and pattern recognition—CVPR'04* (Vol. 2, pp. 506–513).

Levenberg, K. (1944). A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Mechanics and Applied Mathematics*, *II*(2), 164–168.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proc. of the international conference on computer vision ICCV* (pp. 1150–1157).

Lowe, D. (2003). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *20*, 91–110.

Marr, D., & Hildreth, E. (1980). Theory of edge detection. *Proceedings of the Royal Society of London, Series B*, *207*, 187–217.

Matas, J., Chum, O., Urban, M., & Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, *22*(10), 761–767.

Mikolajczyk, K., & Schmid, C. (2002). An affine invariant interest point detector. In *Proc. 7th European conference on computer vision—ECCV 2002* (pp. 128–142). Berlin: Springer.

Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(10), 1615–1630.

Morel, J. M., & Yu, G. (2009). ASIFT: A new framework for fully affine invariant image comparison. *Journal of Imaging Science*, *2*(2), 438–469.

Pollefeys, M., Koch, R., Vergauwen, M., & Gool, L. V. (1999). Hand-held acquisition of 3d models with a video camera. In *3D digital imaging and modeling, international conference on 0:0014*.

Sarfraz, M. S., & Hellwich, O. (2008). Head pose estimation in face recognition across pose scenarios. In *VISAPP (1)* (pp. 235–242).

Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., & Szeliski, R. (2006). A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. of the IEEE conference on computer vision and pattern recognition* (pp. 519–528).

Snavely, N., Seitz, S. M., & Szeliski, R. (2006). Photo tourism: exploring photo collections in 3d. In *ACM SIGGRAPH'06* (pp. 835–846).

Snavely, N., Seitz, S. M., & Szeliski, R. (2008). Modeling the world from Internet photo collections. *International Journal of Computer Vision*, *80*(2), 189–210.

Sturm, P. F., & Triggs, B. (1996). A factorization based algorithm for multi-image projective structure and motion. In *ECCV'96: Proceedings of the 4th European conference on computer vision* (Vol. II, pp. 709–720). Berlin: Springer.

Taylor, P., & Jonker, L. (1978). Evolutionarily stable strategies and game dynamics. *Mathematical Biosciences*, *40*, 145–156.

Tomasi, C., & Kanade, T. (1992). Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, *9*, 137–154. doi:10.1007/BF00129684.

Torr, P., & Zisserman, A. (1998). Robust computation and parametrization of multiple view relations. In *ICCV'98: Proceedings of the sixth international conference on computer vision*. Los Alamitos: IEEE Computer Society.

Torsello, A., Rota Bulò, S., & Pelillo, M. (2006). Grouping with asymmetric affinities: A game-theoretic perspective. In *Proc. of the IEEE conference on computer vision and pattern recognition—CVPR'06* (pp. 292–299).

Triggs, B., McLauchlan, P., Hartley, R., & Fitzgibbon, A. (2000). Bundle adjustment—a modern synthesis. In B. Triggs, A. Zisserman, & R. Szeliski (Eds.), *Lecture notes in computer science: Vol. 1883. Vision algorithms: theory and practice* (pp. 298–372).

Tsai, R. (1987). A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, *3*(4), 323–344.

Vedaldi, A., & Fulkerson, B. (2008) VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/.

Vergauwen, M., & Van Gool, L. (2006). Web-based 3d reconstruction service. *Machine Vision and Applications*, *17*(6), 411–426.

Weibull, J. (1995). *Evolutionary game theory*. Cambridge: MIT Press.

Weinshall, D., & Tomasi, C. (1995). Linear and incremental acquisition of invariant shape models from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *17*, 512–517.

Weng, J., Cohen, P., & Herniou, M. (1992). Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *14*(10), 965–980.

Weng, J., Ahuja, N., & Huang, T. S. (1993). Optimal motion and structure estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *15*(9), 864–884.

Zhang, Z. (1995). Estimating motion and structure from correspondences of line segments between two perspective images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *17*(12), 1129–1139.

Zhang, Z., Deriche, R., Faugeras, O., & Luong, Q. T. (1995). A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, *78*(1–2), 87–119.

# Graph Transduction as a Non-cooperative Game

Aykut Erdem[1] and Marcello Pelillo[2]

[1] Hacettepe University, Beytepe, 06800, Ankara, Turkey
aykut.erdem@hacettepe.edu.tr
[2] "Ca' Foscari" University of Venice, Mestre, Venezia, 30172, Italy
pelillo@dsi.unive.it

**Abstract.** Graph transduction is a popular class of semi-supervised learning techniques, which aims to estimate a classification function defined over a graph of labeled and unlabeled data points. The general idea is to propagate the provided label information to unlabeled nodes in a consistent way. In contrast to the traditional view, in which the process of label propagation is defined as a graph Laplacian regularization, here we propose a radically different perspective that is based on game-theoretic notions. Within our framework, the transduction problem is formulated in terms of a non-cooperative multi-player game where any equilibrium of the proposed game corresponds to a consistent labeling of the data. An attractive feature of our formulation is that it is inherently a multi-class approach and imposes no constraint whatsoever on the structure of the pairwise similarity matrix, being able to naturally deal with asymmetric and negative similarities alike. We evaluated our approach on some real-world problems involving symmetric or asymmetric similarities and obtained competitive results against state-of-the-art algorithms.

## 1 Introduction

In the machine learning community, *semi-supervised learning* (SSL) has gained considerable popularity over the last decade [3,19] and within the existing paradigms, graph-based approaches to SSL, namely the *graph transduction* methods, constitute an important class. These approaches model the geometry of the data as a graph with nodes corresponding to the labeled and unlabeled points and edges being weighted by the similarity between points, and try to estimate the labels of unlabeled points by propagating the coarse information available at the labeled nodes to the unlabeled ones. Performing this propagation in a consistent way relies on a common a priori assumption, known as the "*cluster assumption*" [17,3], which states that (1) points which are close to each other are expected to have the same label, and (2) points in the same cluster (or on the same manifold) are expected to have the same label. Building on this assumption, traditional graph-based approaches formalize graph transduction as a regularized function estimation problem on an undirected graph [9,20,17].

In this paper, we present a novel framework for graph transduction, which is derived from a game-theoretic formulation of the competition between the multi-population of hypotheses of class membership. Specifically, we cast the problem of graph transduction as a *multi-player non-cooperative game* where the players are the data points that

X. Jiang, M. Ferrer, and A. Torsello (Eds.): GbRPR 2011, LNCS 6658, pp. 195–204, 2011.

play a classification game over and over until an equilibrium is reached in their respective strategies. In this game, the strategies played by the labeled points are already decided at the outset, as each of them knows which class it belongs to. On the other hand, the strategies available to unlabeled points are the whole set of hypotheses of being a member of one of the provided classes. The players compete with each other by selecting their own strategies, each choice obtains support from the compatible ones and competitive pressure from all the others. In the long run, the competition will reduce the population of strategies which assume the hypotheses that do not receive strong support from the rest, while it will allow populations with strong support to flourish. In this study, this evolutionary dynamics is modeled by a classic formalization of natural selection process used in the *evolutionary game theory* [16], commonly referred to as the *replicator dynamics*. It is worth-mentioning that our formulation is intrinsically a *multi-class* approach and does not impose any constraint on the value of the payoffs (similarities); in particular, *payoffs do not have to be nonnegative or symmetric*.

The remainder of this paper is structured as follows. In Section 2, we review basic notions from non-cooperative game theory. In Section 3, we formulate graph transduction in terms of a non-cooperative multi-player game. In Section 4, we present our experimental results on a number of real-world classification problems. Finally, in Section 5, we conclude the paper with a summary and directions for future work.

## 2 Non-cooperative Games and Nash Equilibria

Following the notations used in [16], a game with many players can be expressed in normal form as a triple $G = (\mathcal{I}, S, \pi)$, where $\mathcal{I} = \{1, \ldots, n\}$, with $n \geq 2$, is the set of *players*, $S = \times_{i \in \mathcal{I}} S_i$ is the *joint strategy space* defined as the Cartesian product of the individual pure strategy sets $S_i = \{1, \ldots, m_i\}$, and $\pi : S \rightarrow \mathbb{R}^n$ is the *combined payoff function* which assigns a real valued payoff $\pi_i(s) \in \mathbb{R}$ to each *pure strategy profile* $s \in S$ and player $i \in \mathcal{I}$.

A *mixed strategy* of player $i \in \mathcal{I}$ is a probability distribution over its pure strategy set $S_i$, which can be described as the vector $x_i = (x_{i1}, \ldots, x_{im_i})^T$ such that each component $x_{ih}$ denotes the probability that the player chooses to play its $h^{th}$ pure strategy among all the available strategies. Mixed strategies for each player $i \in \mathcal{I}$ are constrained to lie in the *standard simplex* of the $m_i$-dimensional Euclidean space $\mathbb{R}^{m_i}$, $\Delta_i = \{x_i \in \mathbb{R}^{m_i} : \sum_{h=1}^{m_i} x_{ih} = 1, \text{and } x_{ih} \geq 0 \text{ for all } h\}$. Accordingly, a *mixed strategy profile* $x = (x_1, \ldots, x_n)$ is defined as a vector of mixed strategies, each $x_i \in \Delta_i$ representing the mixed strategy assigned to player $i \in \mathcal{I}$, and each mixed strategy profile lives in the *mixed strategy space* of the game, given by the Cartesian product $\Theta = \times_{i \in \mathcal{I}} \Delta_i$.

For the sake of simplicity, let $z = (x_i, y_{-i}) \in \Theta$ denote the strategy profile where player $i$ plays strategy $x_i \in \Delta_i$ whereas other players $j \in \mathcal{I} \setminus \{i\}$ play based on the strategy profile $y \in \Theta$, that is to say, $z_i = x_i$ and $z_j = y_j$ for all $j \neq i$. The expected value of the payoff that player $i$ obtains can be determined by a weighted sum for any $i, j \in \mathcal{I}$ as

$$u_i(x) = \sum_{s \in S} x(s)\pi_i(s) = \sum_{k=1}^{m_j} u_i\left(e_j^k, x_{-j}\right) x_{jk} \tag{1}$$

where $u_i\left(e_j^k, x_{-j}\right)$ denotes the payoff that player $i$ receives when player $j$ adopts its $k^{th}$ pure strategy, and $e_j^k \in \Delta_j$ stands for the *extreme mixed strategy* corresponding the vector of length $m_j$ whose components are all zero except the $k^{th}$ one which is equal to one.

The *mixed best replies* for player $i$ against a mixed strategy $y \in \Theta$, denoted by $\beta_i(y)$, is the set of mixed strategies which is constructed in such a way that no other mixed strategy other than the ones included in this set gives a higher payoff to player $i$ against strategy $y$, defined as the set $\beta_i(y) = \{x_i \in \Delta_i : u_i\left(x_i, y_{-i}\right) \geq u_i\left(z_i, y_{-i}\right)\ \forall z_i \in \Delta_i\}$. Subsequently, the combined mixed best replies is defined as the Cartesian product of best replies of all the players $\beta(y) = \times_{i \in \mathcal{I}} \beta_i(y) \subset \Theta$.

**Definition 1.** *A mixed strategy $x^* = (x_1^*, \ldots, x_n^*)$ is said to be a Nash equilibrium if it is the best reply to itself, $x^* \in \beta(x^*)$, that is*

$$u_i(x_i^*, x_{-i}^*) \geq u_i(x_i, x_{-i}^*) \tag{2}$$

*for all $i \in \mathcal{I}, x_i \in \Delta_i$, and $x_i \neq x_i^*$. Furthermore, a Nash equilibrium $x^*$ is called* strict *if each $x_i^*$ is the unique best reply to $x^*$, $\beta(x^*) = \{x^*\}$*

Nash equilibrium constitutes the key concept of game theory. It is proven by Nash that any non-cooperative game with finite set of strategies has at least one mixed Nash equilibrium [11]. The algorithmic issue of computing a Nash equilibria for the proposed transduction game will be discussed later in Section 3.2.

## 3 Graph Transduction Game (GTG)

Consider the following *graph transduction game*. Assume each player $i \in \mathcal{I}$ participating in the game corresponds to a particular point in a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and can choose a strategy among the set of strategies $S_i = \{1, \ldots, c\}$, each expressing a certain hypothesis about its membership to a class and $|S_i|$ being the total number of classes. Hence, the mixed strategy profile of each player $i \in \mathcal{I}$ lies in the $c$-dimensional simplex $\Delta_i$. By problem definition, we can categorize the players of the game into two disjoint groups: those which already have the knowledge of their membership, which we call *determined players* and denote them with the symbol $\mathcal{I}_{\mathcal{D}}$, and those which don't have any idea about this in the beginning of the game, which are hence called *undetermined players* and correspondingly denoted with $\mathcal{I}_{\mathcal{U}}$.

The so-called determined players of the game can further be distinguished based on the strategies they follow without hesitation, coming from their membership information. In formal terms, $\mathcal{I}_{\mathcal{D}} = \{\mathcal{I}_{\mathcal{D}|1}, \ldots, \mathcal{I}_{\mathcal{D}|c}\}$, where each disjoint subset $\mathcal{I}_{\mathcal{D}|k}$ stands for the set of players always playing their $k^{th}$ pure strategies. It thus follows from this statement that each player $i \in \mathcal{I}_{\mathcal{D}|k}$ plays its extreme mixed strategy $e_i^k \in \Delta_i$. In other words, $x_i$ is constrained to belong to the minimal face of the simplex $\Delta_i$ spanned by $\{e_i^k\}$. In this regard, it can be argued that the determined players do not play the game to maximize their payoffs since they have already chosen their strategies. In fact, the transduction game can be easily reduced to a game with only undetermined players $\mathcal{I}_{\mathcal{U}}$ where the definite strategies of determined players $\mathcal{I}_{\mathcal{D}}$ act as bias over the choices of undetermined players.

It should be noted that any instance of the proposed transduction game will always have a Nash equilibrium in mixed strategies [11]. Recall that, for the players, such an equilibrium corresponds to a steady state such that each player plays a strategy that could yield the highest payoff when the strategies of the remaining players are kept fixed, and it provides us a globally consistent labeling of the data set. Once an equilibrium is reached, the label of a data point (player) $i$ is simply given by the strategy with the highest probability in the equilibrium mixed strategy of player $i$ as $y_i = \arg\max_{h \leq c} x_{ih}$.

### 3.1 Defining Payoff Functions

A crucial step in formulating transduction as a non-cooperative game is how the payoff function of the game is specified. Here, we make a simplification and assume that the payoffs associated to each player are additively separable, and this makes the proposed game a member of a special subclass of multi-player games, known as *polymatrix games* [8,7]. Formally speaking, for a pure strategy profile $s = (s_1, \ldots, s_n) \in S$, the payoff function of every player $i \in \mathcal{I}$ is in the form:

$$\pi_i(s) = \sum_{j=1}^{n} A_{ij}(s_i, s_j) \tag{3}$$

where $A_{ij} \in \mathbb{R}^{c \times c}$ is the *partial payoff* matrix between players $i$ and $j$. It follows that, in terms of a mixed strategy profile $x = (x_1, \ldots, x_n)$, the payoffs are computed as $u_i(e_i^h) = \sum_{j=1}^{n} (A_{ij} x_j)_h$ and $u_i(x) = \sum_{j=1}^{n} x_i^T A_{ij} x_j$.

In an instance of the transduction game, since each determined player is restricted to play a definite strategy of its own, all of these fixed choices can be reflected directly in the payoff function of a undetermined player $i \in \mathcal{I}_{\mathcal{U}}$ as follows:

$$u_i(e_i^h) = \sum_{j \in \mathcal{I}_{\mathcal{U}}} (A_{ij} x_j)_h + \sum_{k=1}^{c} \sum_{j \in \mathcal{I}_{\mathcal{D}|k}} A_{ij}(h, k) \tag{4}$$

$$u_i(x) = \sum_{j \in \mathcal{I}_{\mathcal{U}}} x_i^T A_{ij} x_j + \sum_{k=1}^{c} \sum_{j \in \mathcal{I}_{\mathcal{D}|k}} x_i^T (A_{ij})_k \tag{5}$$

Now, we are left with specifying the partial payoff matrices between each pair of players. Let the geometry of the data be modeled with a weighted graph $\mathcal{G} = (\mathcal{X}, \mathcal{E}, w)$ in which $\mathcal{X}$ is the set of nodes representing both labeled and unlabeled points, and $w : \mathcal{E} \rightarrow \mathbb{R}$ is a weight function assigning a similarity value to each edge $e \in \mathcal{E}$. Representing the graph with its weighted adjacency matrix $W = (w_{ij})$, we set the partial payoff matrix between two players $i$ and $j$ as $A_{ij} = I_c \times w_{ij}$ where $I_c$ is the identity matrix of size $c$[1]. Note that when partial payoff matrices are represented in block form as $A = (A_{ij})$, the matrix $A$ is given by the Kronecker product $A = I_c \otimes W$. Our experiments demonstrate that in specifying the payoffs, it is preferable to use the normalized

[1] The rationale for specifying partial payoffs in this way depends on the analysis of graph transduction on a unweighted undirected graph. Due to the page limit, the details will be reported in a longer version.

similarity data matrix $\widehat{W} = D^{-1/2} W D^{-1/2}$ where $D = (d_{ii})$ is the diagonal degree matrix of $W$ with its elements given by $d_{ii} = \sum_j w_{ij}$.

### 3.2 Computing Nash Equilibria

In the recent years, there has been a growing interest in the computational aspects of Nash equilibria. The general problem of computing a Nash equilibrium is shown to belong to the complexity class PPAD-complete, a newly defined subclass of NP [4]. Nevertheless, there are many refinements and extensions of Nash equilibria which can be computed efficiently and moreover, the former result does not apply to certain classes of games. Here, we restrict ourselves to the well-established evolutionary approach [16], initiated by J. Maynard Smith [10]. This dynamic interpretation of the concept imagines that the game is played repeatedly, generation after generation, during which a selection process acts on the multi-population of strategies, thereby resulting in the evolution of the fittest strategies. The selection dynamics is commonly modeled by the following set of ordinary differential equations:

$$\dot{x}_{ih} = g_{ih}(x) x_{ih} \tag{6}$$

where a dot signifies derivative with respect to time, and $g(x) = (g_1(x), \ldots, g_n(x))$ is the growth rate function with open domain containing $\Theta = \times_{i \in \mathcal{I}} \Delta_i$, each component $g_i(x)$ being a vector-valued growth rate function for player $i$. Hence, $g_{ih}$ specifies the growth rate at which player $i$'s pure strategy $h$ replicates. It is generally required that the function $g$ be *regular* [16], *i.e.* (1) $g$ is Lipschitz continuous and (2) $g_i(x) \cdot x_i = 0$ for all $x \in \Theta$ and players $i \in \mathcal{I}$. While the first condition guarantees that the system (6) has a unique solution through every initial state, the condition $g_i(x) \cdot x_i = 0$ ensures that the simplex $\Delta_i$ is invariant under (6).

The class of regular selection dynamics includes a wide subclass known as *payoff monotonic dynamics*, in which the ratio of strategies with a higher payoff increase at a higher rate. Formally, a regular selection dynamics (6) is said to be payoff monotonic if

$$u_i\left(e_i^h, x_{-i}\right) > u_i\left(e_i^k, x_{-i}\right) \Leftrightarrow g_{ih}(x) > g_{ik}(x) \tag{7}$$

for all $x \in \Theta$, $i \in \mathcal{I}$ and pure strategies $h, k \in S_i$.

A particular subclass of payoff monotonic dynamics, which is used to model the evolution of behavior by imitation processes, is given by

$$\dot{x}_{ih} = x_{ih} \left[ \sum_{l \in S_i} x_{il} \Big( \phi_i \left[ u_i \left( e_i^h - e_i^l, x_{-i} \right) \right] - \phi_i \left[ u_i \left( e_i^l - e_i^h, x_{-i} \right) \right] \Big) \right] \tag{8}$$

where $\phi_i(u_i)$ is a strictly increasing function of $u_i$. When $\phi_i$ is taken as the identity function, *i.e.* $\phi_i(u_i) = u_i$, we obtain the multi-population version of the replicator dynamics:

$$\dot{x}_{ih} = x_{ih} \left( u_i(e_i^h, x_{-i}) - u_i(x) \right) \tag{9}$$

The following theorem states that the fixed points of (9) are Nash equilibria.

**Theorem 1.** *A point $x \in \Theta$ is the limit of a trajectory of (9) starting from the interior of $\Theta$ if and only if $x$ is a Nash equilibrium. Further, if point $x \in \Theta$ is a strict Nash equilibrium then it is asymptotically stable, additionally implying that the trajectories starting from all nearby states converge to $x$.*

*Proof.* See [16].

In the experiments, we utilized the following discrete-time counterpart of (9), where we initialize the mixed strategies of each undetermined player to uniform probabilities, *i.e.* the barycenter of the simplex $\Delta_i$.

$$x_{ih}(t+1) = x_{ih}(t)\frac{u_i(e_i^h)}{u_i(x(t))} \tag{10}$$

The discrete-time replicator dynamics (10) has the same properties as the continuous version (See [12] for a detailed analysis). The computational complexity of finding a Nash equilibrium of a transduction game using (10) can be given by $\mathcal{O}(kcn^2)$, where $n$ is the number of players (data points), $c$ is the number of pure strategies (classes) and $k$ is the number of iterations needed to converge. In theory, it is difficult to predict the number of required iterations, but experimentally, we noticed that it typically grows linearly on the number of data points[2]. We note that the complexity of popular graph transduction methods such as [20,17] is also close to $\mathcal{O}(n^3)$.

## 4 Experimental Results

Our experimental evaluation is divided into two groups based on the structure of similarities that arise in the problems. Basically, we test our approach on some real-world problems involving *symmetric* or *asymmetric* similarities. It is noteworthy to mention that the standard methods are restricted to work with *symmetric* and *non-negative* similarities but our game-theoretic interpretation imposes no constraint whatsoever, being able to naturally deal with *asymmetric* and *negative* similarities alike.

### 4.1 Experiments with Symmetric Similarities

We conducted experiments on three well-known data sets: *USPS*[3], *YaleB* [5] and *20-news*[4]. *USPS* contains images of hand-written digits 0-9 down-sampled to $16 \times 16$ pixels and it has 7291 training and 2007 test examples. As used in [17], we only selected the digits 1 to 4 from the training and test sets, which gave us a total of 3874 data points. *YaleB* is composed of face images of 10 subjects captured under varying poses and illumination conditions. As in [2], we down-sampled each image to $30 \times 40$

[2] We observed that the dynamics always converged to a fixed point in our experiments with symmetric and asymmetric similarities. It should be added that in the asymmetric case, the convergence is in fact not guaranteed since there is no Lyapunov function for the dynamics. Still, by Theorem 1, if the dynamics converges to a fixed point, it will definitely be a Nash equilibrium.

[3] `http://www-stat.stanford.edu/~tibs/ElemStatLearn/`

[4] `http://people.csail.mit.edu/jrennie/20newsgroups/`

pixels and considered a subset of 1755 images which corresponds to the individuals 2, 5 and 8. *20-news* is the text classification data set used in [17], which contains 3970 newsgroup articles selected from the 20-newsgroups data set, all belonging to the topic `rec` which is composed of the subjects `autos, motorcycles, sport.baseball` and `sport.hockey`. As described in [17], each article is represented in 8014-dimensional space based on the TFIDF representation scheme.

For *USPS* and *YaleB*, we treated each image pixel as a single feature, thus each example was represented in 256-, and 1200-dimensional space, respectively. We computed the similarity between two examples $\mathbf{x}_i$ and $\mathbf{x}_j$ using the Gaussian kernel as $w_{ij} = exp(-\frac{d(\mathbf{x}_i,\mathbf{x}_j)^2}{2\sigma^2})$ where $d(\mathbf{x}_i, \mathbf{x}_j)$ is the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ and $\sigma$ is the kernel width parameter. Among several choices for the distance measure $d(\cdot)$, we evaluated the Euclidean distance $\|\mathbf{x}_i - \mathbf{x}_j\|$ for *USPS* and *YaleB*, and the cosine distance $d(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{\langle \mathbf{x}_i, \mathbf{x}_j \rangle}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|}$ for *20-news*.

In the experiments, we compared our approach, which we denote as GTG, against four well-known graph-based SSL algorithms, namely the Spectral Graph Transducer (SGT) [9][5], the *Gaussian fields and harmonic functions* based method (GFHF) [20][6], the *local and global consistency* method (LGC) [17][7] and Laplacian Regularized Least Squares (LapRLS) [1][8]. A crucial factor in the success of graph-based algorithms is the construction of the input graph as it represents the data manifold. To be fair in our evaluation, for all the methods, we used a fixed set of kernel widths and generated 9 different candidate 20-NN graphs by setting $w_{ij} = 0$ if $x_j$ is not amongst the 20-nearest neighbors of $x_i$. In particular, the kernel width $\sigma$ ranges over the set $linspace(0.1r, r, 5) \cup linspace(r, 10r, 5))$ with $r$ being the average distance from each example to its $20^{th}$ nearest neighbor and $linspace(a, b, n)$ denoting the set of $n$ linearly spaced numbers between and including $a$ and $b$.

In Fig. 1, we show the test errors of all methods averaged over 100 trials with different sizes of labeled data where we randomly select labeled samples so that each set contains at least one sample from each class. As it can be seen, LapRLS method gives the best results for the relatively small data set *YaleB*. However, for the other two, its performance is poor. In general, the proposed GTG algorithm is either the best or the second best algorithm; while its success is almost identical to that of the LGC method in *USPS* and *Yale-B*, it gives superior results for *20-news*.

### 4.2 Experiments with Asymmetric Similarities

We carried out experiments on three document data sets – *Cora*, *Citeseer* [14][9], and *WebKB*[10]. *Cora* contains 2708 machine learning publications classified into seven classes, and there are 5429 citations between the publications. *Citeseer* consists of 3312

[5] We select the optimal value of the parameter $c$ with the best mean performance from the set $\{400, 800, 1600, 3200, 6400, 12800\}$.

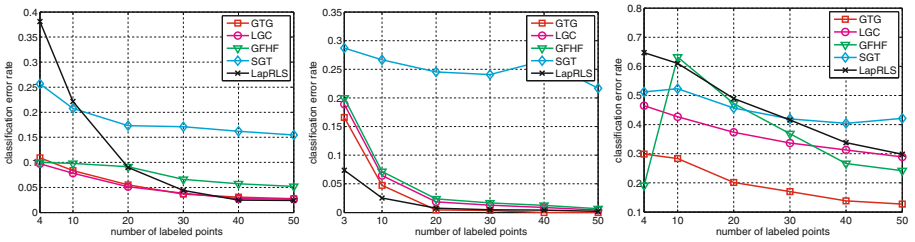
[6] In obtaining the hard labels, we employ the *class mass normalization* step suggested in [20].

[7] As in [17], we set the parameter $\alpha$ as 0.99.

[8] We select the optimal values of the extrinsic and intrinsic regularization parameters $\gamma_A$ and $\gamma_I$ from the set $\{10^{-6}, 10^{-4}, 10^{-2}, 1\}$ for the best mean performance.

[9] Both data sets are available at `http://www.cs.umd.edu/projects/linqs/projects/lbc/`

[10] Available at `http://www.nec-labs.com/~zsh/files/link-fact-data.zip`



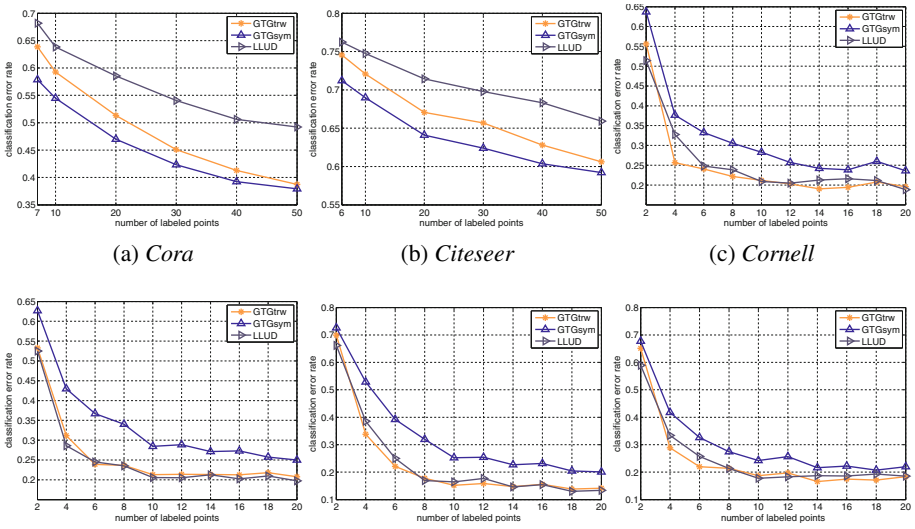

(a) *USPS* (b) *YaleB* (c) *20-news*

**Fig. 1.** Performance comparisons on classification problems with *symmetric* similarities

scientific publications, each of which belongs to one of six classes, and there are a total of 4732 links. *WebKB* contains webpages collected from computer science departments of four universities (*Cornell*, *Texas*, *Washington* and *Wisconsin*), and each classified into seven categories. Following the setup in [18], here we concentrate on classifying student pages from the others. Each subset respectively contains 827, 814, 1166 and 1210 webpages and 1626, 1480, 2218 and 3200 links. In our experiments, as in [18], we only considered the citation structure, even though one can also assign some weights by utilizing the textual content of the documents. Specifically, we worked on the link matrix $W = (w_{ij})$, where $w_{ij} = 1$ if document $i$ cites document $j$ and $w_{ij} = 0$ otherwise.

Unlike our approach, the standard methods mentioned before, namely SGT, GFHF, LGC and LapRLS, are subject to symmetric similarities. Hence, in this context, they can be applied only after rendering the similarities symmetric but this could result in loss of relevant information in some cases. In our evaluation, we restrict ourselves to the graph-based methods which can directly deal with asymmetric similarities. Specifically, we compared our game-theoretic approach against our implementation of the method in [18], denoted here with LLUD. We note that LLUD is based on the notion of random walks on directed graphs and it reduces to LGC in the case of symmetric similarities. However, it assumes the input similarity graph to be strongly connected, so in [18] the authors consider the *teleporting random walk (trw)* transition matrix as input, which is given by $P^{\eta} = \eta P + (1-\eta)P^{u}$ where $P = D^{-1}W$ and $P^{u}$ is the uniform transition matrix. For asymmetric similarity data, we also define the payoffs in terms of this transition matrix and denote this version with GTGtrw. In the experiments, we fixed $\eta = 0.99$ for both LLUD and GTGtrw. To provide a baseline, we also report the results of our approach that works on the symmetrized similarity matrices, denoted with GTGsym. For that case, we used the transformation $\widetilde{W} = 0.5 \times (W + W^{T})$ for *SCOP*, and the symmetrized link matrix $\widetilde{W} = (w_{ij})$ for the others, where $w_{ij} = 1$ if either document $i$ cites document $j$ or vice versa, and $w_{ij} = 0$ otherwise.

The test errors averaged over 100 trials are shown in Fig. 2. Notice that the performances of GTGtrw and LLUD are quite similar on the classification problems in *WebKB* data sets. On the other hand, GTGtrw is superior in the multi-class problems of *Cora* and *Citeseer*. We should add that symmetrization sometimes can provide good results. In *Cora* and *Citeseer*, GTGsym performs better than the other two methods.




(a) *Cora* (b) *Citeseer* (c) *Cornell*

(d) *Texas* (e) *Washington* (f) *Wisconsin*

**Fig. 2.** Performance comparisons on classification problems with *asymmetric* similarities

## 5 Summary and Discussion

In this paper, we provided a game-theoretic interpretation to graph transduction. In the suggested approach, the problem of transduction is formulated in terms of a multi-player non-cooperative game where any equilibrium of the game coincides with the notion of a consistent labeling of the data. As compared to existing approaches, the main advantage of the proposed framework is that there is no restriction on the pair-wise relationships among data points; similarities and thus the payoffs can be negative or asymmetric. The experimental results show that the our approach is not only more general but also competitive with standard approaches. In the future, we plan to continue exploring the generality of our approach when both similarity and dissimilarity relations exist in data [15,6]. Another possible future direction is to focus on improving the efficiency. In our current implementation, we use the standard replicator dynamics to reach an equilibrium but we can study other selection dynamics that are much faster [13].

### Acknowledgments

We acknowledge financial support from the FET programme within EU FP7, under the SIMBAD project (contract 213250).

## References

1. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. J. Mach. Learn. Res. 7, 2399–2434 (2006)

2. Breitenbach, M., Grudic, G.Z.: Clustering through ranking on manifolds. In: ICML, pp. 73–80 (2005)
3. Chapelle, O., Schölkopf, B., Zien, A. (eds.): Semi-Supervised Learning. MIT Press, Cambridge (2006)
4. Daskalakis, C., Goldberg, P.W., Papadimitriou, C.H.: The complexity of computing a Nash equilibrium. Commun. ACM 52(2), 89–97 (2009)
5. Georghiades, A., Belhumeur, P., Kriegman, D.: From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans. on PAMI 23(6), 643–660 (2001)
6. Goldberg, A., Zhu, X., Wright, S.: Dissimilarity in graph-based semi-supervised classification. In: AISTATS (2007)
7. Howson, J.T.: Equilibria of polymatrix games. Management Science 18(5), 312–318 (1972)
8. Janovskaya, E.B.: Equilibrium points in polymatrix games (in Russian) Litovskii Matematicheskii Sbornik 8, 381–384 (1968); (Math. Reviews 39 #3831)
9. Joachims, T.: Transductive learning via spectral graph partitioning. In: ICML, pp. 290–297 (2003)
10. Maynard Smith, J.: Evolution and the theory of games. Cambridge University Press, Cambridge (1982)
11. Nash, J.: Non-cooperative games. The Annals of Mathematics 54(2), 286–295 (1951)
12. Pelillo, M.: The dynamics of nonlinear relaxation labeling processes. J. Math. Imaging Vis. 7(4), 309–323 (1997)
13. Rota Bulò, S., Bomze, I.M.: Infection and immunization: a new class of evolutionary game dynamics. Games and Economic Behaviour (Special issue in honor of John F. Nash, jr.) 71, 193–211 (2011)
14. Sen, P., Namata, G.M., Bilgic, M., Getoor, L., Gallagher, B., Eliassi-Rad, T.: Collective classification in network data. AI Magazine 29(3), 93–106 (2008)
15. Tong, W., Jin, R.: Semi-supervised learning by mixed label propagation. In: AAAI, pp. 651–656 (2007)
16. Weibull, J.W.: Evolutionary Game Theory. MIT Press, Cambridge (1995)
17. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: NIPS, pp. 321–328 (2004)
18. Zhou, D., Huang, J., Schölkopf, B.: Learning from labeled and unlabeled data on a directed graph. In: ICML, pp. 1036–1043 (2005)
19. Zhu, X.: Semi-supervised learning literature survey. Tech. Rep. 1530, Computer Sciences, University of Wisconsin-Madison (2005)
20. Zhu, X., Ghahramani, Z., Lafferty, J.: Semi-supervised learning using Gaussian fields and harmonic functions. In: ICML, pp. 912–919 (2003)

# Semantic Image Labelling as a Label Puzzle Game

Peter Kontschieder[1]
kontschieder@icg.tugraz.at
Samuel Rota Bulò[2]
srotabul@dsi.unive.it
Michael Donoser[1]
donoser@icg.tugraz.at
Marcello Pelillo[2]
pelillo@dsi.unive.it
Horst Bischof[1]
bischof@icg.tugraz.at

[1] Institute for Computer Graphics and Vision
Graz University of Technology
Austria

[2] Dipartimento di Scienze Ambientali, Informatica e Statistica
Università Ca' Foscari Venezia
Italy

**Abstract**

In this work we introduce a novel solution to the semantic image labelling problem, *i.e*. the task of assigning semantic object class labels to individual pixels in a test image. Conventional methods are typically relying on random fields for modelling interactions between neighboring pixels and obtaining smooth labelling results using unary and pairwise cost functions. Instead, we consider the labelling problem as a puzzle game, where the final labelling is obtained by assembling discriminatively learned candidate sets of label puzzle pieces, each representing a topological and semantically plausible label configuration. The puzzle game is set up by means of a modified random forest classifier, designed to learn the local, topological label-structure and hence the local context associated to the training data. To solve the puzzle game we propose an iterative optimization technique that maximizes an agreement function by alternatingly seeking for the best label puzzle piece per pixel and the resulting semantic labelling per image. We provide both, theoretical properties of our puzzle solver algorithm as well as experimental results on the challenging MSRC and CamVid databases. In a direct comparison with a conditional random field we obtain superior results, indicating the practicability of our proposed method.

## 1 Introduction

The field of semantic image labelling has received great attention and evolved in a remarkable manner during the past couple of years. Given an input image, it aims for the proper assignment of a-priori learned class labels to each pixel in a test image[1]. For instance, a typical street scene might result in coherently labelled regions of road, car, bicyclist and so on. In order to obtain labellings reflecting the natural statistics of a scene, state-of-the-art approaches [13, 18, 19] combine multiple, complementary cues at different levels within



[1] For brevity reasons we use the terms *image labelling* or solely *labelling* interchangeably instead of *semantic image labelling* throughout the paper.

random field models [20]. These include low-level cues which are mostly computed on a per-pixel basis and incorporate local color or texture statistics. Mid-level cues operate on regions or superpixels to provide shape, continuity or symmetry information. Motivated from perceptual psychology [2, 3], high-level cues introduce global image statistics and information about inter-object or contextual relations, seeking for proper scene configurations at the image level.

Many researchers are following the idea of Shotton *et al*. [31], putting special emphasis on the generation of good unary potentials to be used in conjunction with a conditional random field (CRF) model. Indeed, unary potentials have a significant influence on the success of a labelling algorithm. For instance, [31] uses an adapted version of joint boosting [35] to train unary classifiers integrating textons [23] and shape filters. Similarly, [33] uses a boosting approach to combine an extended set of different feature cues. Recently, several methods [7, 15, 29, 32, 36] used random forests [1, 6, 12] for obtaining properly learned combinations of local features like color, intensity derivatives, covariance features [26], textons, HOG features [10] or motion and 3D structure features [7].

To render the inference process more efficient and include segmentation information, many works [21, 28] consider *mid-level* or super-pixel representations rather than individual pixels. In such a way, the labelling problem is defined over regions, mostly obtained by Mean Shift [9], graph-based approaches [11] or Normalized Cut [30]. To cope with suboptimal segmentations not following the desired object boundaries, [17, 25] combine multiple segmentations or reshape superpixels to recover from errors as presented in [14]. Recently, some researchers [18, 19, 27, 34] also started to incorporate *high-level* information, *e.g*. contextual (semantic) or object detector information, into CRFs. This yields a considerable improvement of the overall labelling results, since contradicting labellings can be resolved by means of global or co-occurrence statistics. In other words, such information helps to improve the labelling of adjacent regions being partially labelled by non-compatible object class configurations.

The label space characterizing an image labelling problem instance does indeed exhibit an inherently topological structure which renders the class labels explicitly interdependent. Many approaches to labelling, however, are not exploiting this information properly, as they rely on classifiers trained on a set of labeled images, which associate pixels only with single, *atomic* class labels acting as arbitrary identifiers without any dependencies among them (*e.g*. [7, 15]). In this way, the structured label space information in the training images remains largely unexploited. As a consequence, labellings obtained at a low-level are quite noisy and exhibit configurations of labels which never appeared in the training images. To alleviate this effect, the atomic labels obtained by the base classifiers are combined in a more or less sophisticated way, *e.g*. by means of a CRF. However, the rules guiding this label relaxation process are typically imposed in a top-down fashion rather than being learnt from training data. At a high-level, some efforts have been put on capturing topological relationships between labels, *e.g*. by collecting co-occurrence statistics of categories in images. However, the integration of this information in the labelling approaches leads typically to simplistic, but expensive, high-order energy terms in CRFs, or to a-posteriori re-elaboration of low- or mid-level labelling solutions.

**Contributions.** In this paper we propose a novel method to include local, contextual information into the low-level classification process. Instead of integrating a series of complementary cues within a random field model, we formulate the image labelling problem as the task of assembling topological label information in a coherent way. Intuitively, our approach

can be explained as a puzzle game where the puzzle pieces are represented by structured object class labels. These structured labels are directly obtained and learned from the ground truth training data and always exhibit a semantically meaningful label configuration. In such a way, the set of possible label puzzle pieces only shows plausible label configurations such as a cow standing on grass but not on water. In concrete terms, the labelling of an image is obtained as a result of a two-stage process. First, the label puzzle game is set up by assigning a set of plausible puzzle pieces to each pixel in the test image, using a modified random forest classifier. Afterwards, we search for a solution of the label puzzle, which consists of both, a per-pixel class label and puzzle piece selection, maximizing a measure of overall agreement. This optimization problem is addressed by means of a heuristic, which alternates between optimizing the image labelling and the per-pixel puzzle piece selection. As a result, we obtain a joint labelling based on the selection of plausible, local label configurations, respecting the local contextual information of neighboring pixels.

**Paper organization.** In Section 2 we describe our image labelling puzzle approach, and provide related definitions and notations. In Section 3 we describe how to set up a label puzzle game by means of a modified random forest classifier, which discriminatively learns structured labels from the training data. In Section 4 we introduce an algorithm to solve the label puzzle game, show its theoretical properties and analyse its complexity. Finally, we provide experimental results and concluding remarks in Sections 5 and 6, respectively.

## 2 The Label Puzzle Game

In this section we propose our novel idea, which considers image labelling as the task of assembling a kind of puzzle, where the pieces are label configurations (*e.g.* in our experiments they are square patches of labels) gathered from the training images during a learning and classification process. Please note that this is in contrast to a common tiling or jigsaw puzzle [8, 37], where the pieces form a partition of the target image in the image domain. Instead, we associate each pixel with a possibly different set of label puzzle pieces in the label domain from which only one must be selected. Additionally, pieces belonging to different pixels may overlap. Given a test image, the goal is to simultaneously assemble the related puzzle and assign labels to pixels in a way such that the agreement of the selected pieces with the underlying labelling is maximized.

**Notations and definitions.** An *image* is a function $f : D \to \mathbb{R}^d$ mapping pixels in $D \subseteq \mathbb{Z}^2$ to $d$-dimensional feature vectors, encoding different local cues of the image (*e.g.* color, gradient features, filter banks). A *labelling* for an image is a function $\ell : D \to Y$ mapping pixels to labels in $Y = \{1, \ldots, k\}$. A (label) puzzle piece is a (local) *label configuration*, *i.e.* a function $p : \mathbb{Z}^2 \to Y \cup \{\perp\}$ mapping two-dimensional points to labels or to void ($\perp$), a special symbol indicating the absence of a label. The set of images, labellings and puzzle pieces (*i.e.* label configurations) are denoted by $\mathscr{I}$, $\mathscr{L}$ and $\mathscr{P}$, respectively. A *puzzle configuration* is a function $z : D \to \mathscr{P}$ associating each pixel in $D$ with a puzzle piece in $\mathscr{P}$. The set of puzzle configurations is denoted by $\mathscr{Z}$. Note that, for notational convenience, we will write in the sequel $z_{i,j} \in \mathscr{P}$ instead of $z(i, j)$ and we will denote with $z_{i,j}(u, v)$ the label in position $(u, v)$ in puzzle piece $z_{i,j}$.

The *agreement* of a puzzle piece $p \in \mathscr{P}$ located in $(i, j) \in D$ with a labelling $\ell \in \mathscr{L}$ is defined as the number of corresponding pixels sharing the same label, *i.e.*

$$\phi^{(i,j)}(p, \ell) = \sum_{(u,v) \in D} [p(u - i, v - j) = \ell(u, v)] \,, \tag{1}$$

where $[P]$ are the Iverson brackets yielding 1 if proposition $P$ is true, 0 otherwise. Given a puzzle configuration $z \in \mathscr{Z}$ and a labelling $\ell \in \mathscr{L}$, the *total agreement* $\Phi(z,\ell)$ of the image labelling puzzle is the sum of the agreements of each puzzle piece in $z$ with the labelling $\ell$, *i.e.*

$$\Phi(z,\ell) = \sum_{(i,j)\in D} \phi^{(i,j)}(z_{i,j},\ell) \,. \tag{2}$$

**The label puzzle game.** A *label puzzle game* for an image $f \in \mathscr{I}$ is a function $\pi_f$ mapping each pixel $(i,j) \in D$ to a non-empty set of puzzle pieces $\pi_f(i,j) \subseteq \mathscr{P}$. This function restricts the possible choices of puzzle pieces per pixel and, hence, also the set of admissible puzzle configurations to

$$\mathscr{Z}|_{\pi_f} = \left\{ z \in \mathscr{Z} \,\middle|\, z_{i,j} \in \pi_f(i,j) \right\} .$$

A solution of a label puzzle game $\pi_f$ is a pair $(z^*,\ell^*) \in \mathscr{Z}|_{\pi_f} \times \mathscr{L}$ consisting of an admissible puzzle configuration and a labelling for $f$ yielding the maximum total agreement:

$$(z^*,\ell^*) \in \arg\max_{(z,\ell)} \left\{ \Phi(z,\ell) \,\middle|\, (z,\ell) \in \mathscr{Z}|_{\pi_f} \times \mathscr{L} \right\} . \tag{3}$$

A heuristic for finding a solution of (3) will be discussed in Section 4.

An important component of our framework is the *label puzzle game generator* providing the label puzzle game $\pi_f$ for any image $f \in \mathscr{I}$ we want to label. The generator is obtained as the result of a supervised learning process, involving a set of labelled training images, and will be discussed in the next section.

# 3 Label Puzzle Game Generator

In this section, we describe a puzzle game generator built upon an adapted random forest classifier [16]. The basic idea is to collect a set of admissible label puzzle pieces for each pixel in a test image, which will be used to create the label puzzle game. To this end, we augment random forests with the ability of performing structured label predictions rather than single and atomic classifications. In such a way we directly obtain structured labels as output of our classifier, subsequently denoted as *puzzle pieces*.

Before moving into the details of our approach, we briefly review the traditional random forest framework [1, 17]. A random forest is an ensemble of binary decision trees, each of which is a classifier mapping samples in $\mathscr{X}$ to class labels in $Y$. In the context of image labelling, the sample space $\mathscr{X}$ consists of a set of labelled patches (*e.g.* square regions of pixels) extracted from the training images, where each patch is associated with the label of a specific pixel it contains (typically the one in the center). The prediction for a sample in a decision tree takes place by routing it from the root node to a leaf holding a class label. The path followed by a sample moving along the tree is determined by split functions $\psi : \mathscr{X} \to \{\text{left}, \text{right}\}$ located in each node, according to which a sample is forwarded to the left or right child. The prediction for the whole forest is computed using a majority vote criterion from the predictions cast by its single decision trees. As for the learning part, decision trees in a random forest are recursively trained by selecting in each node a split function from a set of randomly generated ones, which induces a partition of the training set showing the best information gain about the class label distributions due to the split. According to the chosen split function in a node, the training set is then partially forwarded to its left and right child, respectively. If the training samples reaching a node are less than a given threshold, if they exhibit a low entropy in their class label distribution or if a maximum

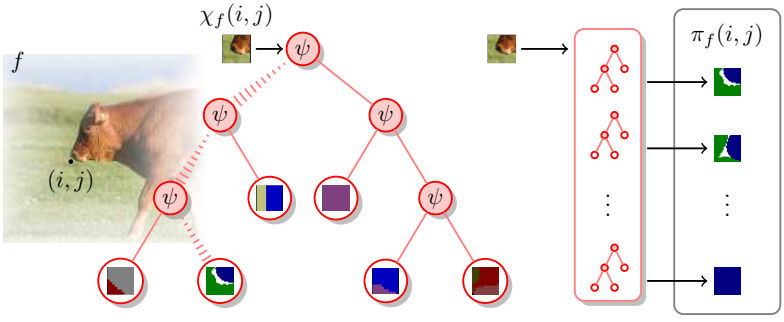



Figure 1: Pipeline of the construction of a label puzzle game. For each pixel $(i,j)$ of the test image $f$, we extract an image patch $\chi_f(i,j)$ and compute a set of plausible puzzle pieces $\pi_f(i,j)$ for it by means of a modified random forest.

depth in the tree is reached, the recursion stops and a leaf is grown. Finally, the class label best represented in the training samples is assigned to the leaf.

To adapt the random forest to collect puzzle pieces, we change the label space from $Y$ to the set of puzzle pieces $\mathscr{P}$. Hence, each decision tree can be considered as a function $h : \mathscr{X} \to \mathscr{P}$ mapping image patches in $\mathscr{X}$ to puzzle pieces in $\mathscr{P}$. Accordingly, the training set consists of image patches with a corresponding structured label, *i.e.* a puzzle piece $p \in \mathscr{P}$ collected from the ground truth. The shape of a puzzle piece may be arbitrary but, for simplicity, we assume all puzzle pieces to have the same shape. Note that in our experiments we considered simple square regions of labels as puzzle pieces, as illustrated in Figure 1.

Besides dealing with a structured label space, our decision trees present other significant differences with respect to standard ones. First, we changed the way the best split function is selected at each tree node in order to take the new label space into account. Specifically, we randomly select for each node a point $(i,j) \in \mathbb{Z}^2$ and any training sample $(x,p) \in \mathscr{X} \times \mathscr{P}$ reaching that node is given label $p(i,j)$. By so doing, the same training sample may be considered with different labels in different nodes, thereby exploiting the whole structure of the puzzle piece during the tree construction. Moreover, the split function selection can still be efficiently carried out using, *e.g.* the technique based on information gain. A second difference is in the way a puzzle piece is selected as representative in a leaf: Given a leaf of the tree, let $T \subseteq \mathscr{X} \times \mathscr{P}$ be the subset of the training set that reached the leaf during the training procedure. Since we would like to select a representative close to the mode of the distribution of puzzle pieces in the leaf, we estimate a conditional probability $\Pr(p|T)$ of a puzzle piece given $T$. For simplicity, we make a pixel independence assumption, thereby obtaining:

$$\Pr(p|T) = \prod_{(i,j)\in\mathbb{Z}^2} \Pr^{(i,j)}(p(i,j)|T) \,,$$

as product of the marginal class label distributions $\Pr^{(i,j)}(y|T)$ of pixels in position $(i,j)$ given $T$, where

$$\Pr^{(i,j)}(y|T) = \frac{1}{|T|} \sum_{(x,p)\in T} [p(i,j) = y] \,.$$

The puzzle piece representative $p^*$ for the leaf is then selected as the one maximizing the joint probability over the set of available puzzle pieces:

$$p^* \in \arg\max_p \{\Pr(p|T) \,|\, (x,p) \in T \text{ for some } x \in \mathscr{X}\} \,.$$

Finally, a random forest $\{h_1,\ldots,h_n\}$ consisting of $n$ trees can be considered as a function $H$ mapping image patches $x \in \mathscr{X}$ to non-empty sets of puzzle pieces $H(x) \subseteq \mathscr{P}$ in the following way:

$$H(x) = \bigcup_{k=1}^{n} \{h_k(x)\} .$$

Note that, as opposed to standard random forests, the predictions gathered from the single decision trees are not merged into a single puzzle piece, but we keep them all as a set of puzzle pieces. The modified random forest can then be used to generate a label puzzle game for an image $f \in \mathscr{I}$ as follows:

$$\pi_f(i,j) = H(\chi_f(i,j)) , \tag{4}$$

where $\chi_f(i,j) \in \mathscr{X}$ denotes the patch extracted from image $f \in \mathscr{I}$ in position $(i,j) \in D$. In Figure 1, we summarize the label puzzle game generation process for a particular image.

# 4 Label Puzzle Game Solver

The optimization problem in (3) underlying our image labelling approach is in general non-trivial to solve. The algorithm we propose in this section is a heuristic, which is simple and effective as shown in the experiments conducted (see Section 5). It is based on an alternating optimization technique, where we iteratively switch between optimizing the labelling variable $\ell \in \mathscr{L}$ and the puzzle configuration variable $z \in \mathscr{Z}|_{\pi_f}$.

Let $\ell^{(t)}$ be the labelling of the image at a given time $t \geq 0$. The puzzle configuration $z^{(t+1)}$ at time $t+1$ can be obtained according to the following updating scheme:

$$z_{i,j}^{(t+1)} \in \arg\max_{p} \left\{ \phi^{(i,j)}\left(p, \ell^{(t)}\right) \middle| \, p \in \pi_f(i,j) \right\} , \tag{5}$$

which selects for each pixel $(i,j) \in D$ a puzzle piece in the set $\pi_f(i,j)$ maximizing the agreement with the labelling $\ell^{(t)}$. On the other hand, given the puzzle configuration $z^{(t+1)} \in \mathscr{Z}|_{\pi_f}$ at time $t+1$, we compute the new labelling $\ell^{(t+1)} \in \mathscr{L}$ by taking a majority vote over all puzzle pieces as follows:

$$\ell^{(t+1)}(u,v) \in \arg\max_{y} \left\{ \sum_{(i,j)\in D} \left[ z_{i,j}^{(t+1)}(u-i, v-j) = y \right] \middle| \, y \in Y \right\} . \tag{6}$$

The iterative process is started from an initial labelling $\ell^{(0)} \in \mathscr{L}$ and, by repeatedly applying rules (5) and (6), it will eventually converge towards a local solution of (3). Theorem 1, indeed, provides a theoretical guarantee that the iterative scheme never decreases the value of the objective function $\Phi$.

**Theorem 1.** *Let $\pi_f$ be a label puzzle game for image $f \in \mathscr{I}$, let $\ell^{(0)} \in \mathscr{L}$ be an initial labelling for $f$, and let $z^{(0)} \in \mathscr{Z}|_{\pi_f}$ be an initial puzzle configuration. Then for any $t \geq 0$ we have*

$$\Phi\left(z^{(t+1)}, \ell^{(t)}\right) \geq \Phi\left(z^{(t)}, \ell^{(t)}\right) \tag{7}$$

*and*

$$\Phi\left(z^{(t+1)}, \ell^{(t+1)}\right) \geq \Phi\left(z^{(t+1)}, \ell^{(t)}\right) \tag{8}$$

*where $z^{(t+1)}$ and $\ell^{(t+1)}$ are computed according to (5) and (6), respectively.*

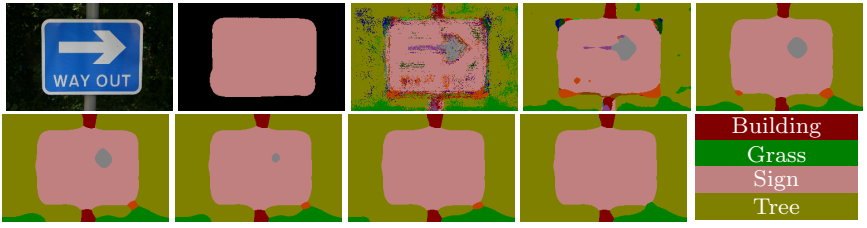



Figure 2: A labelling example of the proposed approach. Top to bottom, left to right: Image to be labelled, groundtruth labelling, initial random forest classification, labellings obtained by our approach after $t = 0, 5, 10, 20, 35, 50$ iterations, final label captions.

*Proof.* By (5) we have for all $t \geq 0$ and $(i, j) \in D$:

$$\phi\left(z_{i,j}^{(t+1)}, \ell^{(t)}\right) \geq \phi\left(z_{i,j}^{(t)}, \ell^{(t)}\right).$$

By summing up each side of this inequality for all pixels $(i, j) \in D$ we obtain (7).

As for the second inequality, note that by (6) and (1) we have

$$\sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t+1)}(u,v)\right] \geq \sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t)}(u,v)\right]$$

for all $(u, v) \in D$. This together with a trivial re-ordering of the summations yields

$$\begin{aligned}\Phi\left(z^{(t+1)}, \ell^{(t+1)}\right) &= \sum_{(u,v)\in D}\sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t+1)}(u,v)\right] \\ &\geq \sum_{(u,v)\in D}\sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t)}(u,v)\right] = \Phi\left(z^{(t+1)}, \ell^{(t)}\right)\end{aligned}$$

from which the result derives. □

As for the computational complexity of the solver, let $N$ be the number of pixels, $K$ the average number of puzzle pieces per pixel, $M$ the number of non-void elements of a puzzle piece, $k$ the number of labels, and $\gamma$ the number of iterations. An update step for the pixel configuration $z$ has complexity $O(K \cdot M \cdot N)$, while an update step for the labelling $\ell$ has complexity $O((k+M) \cdot N)$. The overall complexity is thus given by $O(\gamma \cdot (k+M+KM) \cdot N)$. Note that in our experiments we stopped the iterative process if either a fixed point or a maximum number $\gamma = 75$ of iterations was reached.

# 5 Experiments

In order to demonstrate the quality of our method, we evaluate on the challenging and widely known CamVid [7] and MSRCv2 [31] databases. We use almost the same setup for both databases, *i.e.* we collect the training samples on a regular grid with a stride of 10 (CamVid) or 5 (MSRCv2) and apply an inverse weighting scheme to correct the imbalance of the training sample distribution. We train forests consisting of 15 decision trees with 500 iterations per node test, stopping when less than 5 samples were available per leaf node. The feature patch sizes are fixed to $20 \times 20$ while we evaluate different puzzle piece sizes on the MSRCv2 database.

We use the following feature cues: CIELab raw channel intensities, first and second order derivatives of the luminance channel and correlation coefficients between covariances

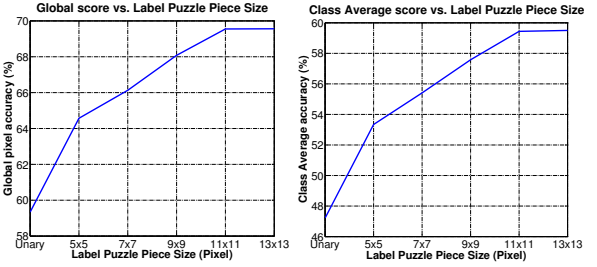



Figure 3: Evaluation of label puzzle piece sizes and their impact on the global pixel (left) and class average (right) accuracies for the MSRCv2 database.

of the RGB raw channel intensities and the first order derivatives of the grayscale intensity image, similar to [15, 26]. The pixel-wise classifications (*unary*) are computed according to the class label distribution of the central pixels of the puzzle piece returned by the trees. In order to obtain the initial pixelwise labelling $\ell^{(0)}$ for the puzzle solver, we take a majority vote decision based on the label statistics collected over all overlapping puzzle pieces. For performance evaluation, we use standard criteria as *e.g.* used in [31]. These are the *Global Pixel Average*, *i.e.* the fraction of correctly classified pixels computed over all classes and test images, and the more strict *Class Average*, defined as the fraction of correctly classified pixels belonging to a specific category over all test images.

On both databases, we compare to the labelling results obtained by minimizing the energy term of a conditional random field (CRF) model with graph cuts, when supplied with our random forest classification results. We use the publicly available GCO implementation[2] [5] and the alpha expansion solver. As unary or data terms, we provide the central label statistics over the puzzle pieces of the entire forest. For the pairwise or smoothness term we use the standard, contrast-sensitive Potts model as suggested in [4].

## 5.1 MSRCv2 Database

This database consists of 532 images containing 21 object classes and predefined splits into 276 training and 256 test images as suggested in [31]. Our random forests obtain pixel classification scores of $(59.3/47.2\%)$ (global/class average) which are higher than the scores obtained by related random forest approaches of Kluckner *et al.* [15] $(55.8/42.2\%)$, the naive, supervised approach of Shotton *et al.* [32] $(49.7/34.5\%)$ and Lazebnik *et al.* [22] $(53.3/40.7\%)$ using combinations of color, textons and SIFT [24] features. There are however methods starting with a significantly better baseline as in Schroff *et al.* [29] $(69.7/-\%)$ which we were not able to reproduce.

**Influence of puzzle piece sizes** In Figure 3 we show the influence of the puzzle piece size on the obtained classification scores. The correlation between label puzzle size and classification score is clearly indicated for both performance measures. This strengthens our initial assumption that the introduction of contextual information at the local level is viable for image labelling. Further increase of the label puzzle pieces will likely introduce smoothing effects along the object boundaries unless a sufficient amount of label transitions are captured during the training phase.

**Comparison to CRF** As illustrated in Table 5.1, we obtain superior results for both, the global pixel labelling accuracy and the more strict per-class average score when compared to a CRF. With our method we always improve over the baseline classification and are superior

[2] http://vision.csd.uwo.ca/code/

| Method | Global | Class Avg | Building | Grass | Tree | Cow | Sheep | Sky | Aeroplane | Water | Face | Car | Bicycle | Flower | Sign | Bird | Book | Chair | Road | Cat | Dog | Body | Boat |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Unary | 59 | 47 | 28 | 93 | 75 | 53 | 62 | 94 | 44 | 53 | 63 | 36 | 57 | 60 | 32 | 19 | 44 | 19 | 61 | 36 | 29 | 25 | 5 |
| Unary + CRF | 67 | 57 | 28 | 96 | 83 | 66 | 74 | 93 | 58 | 56 | 70 | 45 | **81** | **80** | 39 | 23 | **64** | **32** | 75 | **56** | **42** | 31 | 4 |
| Unary + Puzzle | **70** | **60** | **43** | 96 | 83 | **78** | **81** | **96** | **70** | **59** | **71** | **55** | 79 | 73 | **42** | **25** | 59 | 29 | 75 | 53 | 40 | **37** | **6** |

Table 1: Comparison of scores on MSRCv2 database in [%] for puzzle piece size of $11 \times 11$. *Unary* are the scores obtained by the structured random forest classifications alone, *Unary + CRF* are the results using the conditional random field and *Unary + Puzzle* refers to the final labelling result, obtained by our proposed method. Bold style indicates best score while underlined scores are same among CRF and our approach.

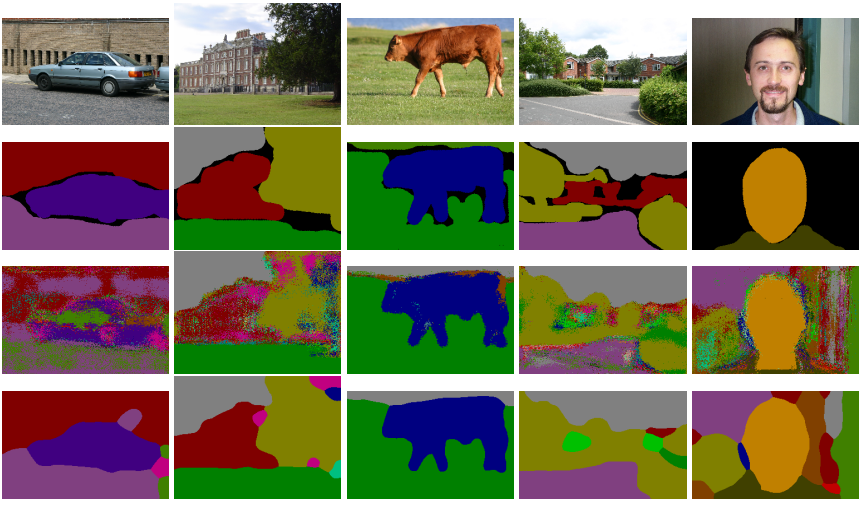


Figure 4: Qualitative labelling results obtained by our method on the MSRCv2 database. First row: Original image, Second row: Ground truth labelling, Third row: Unary classifications, Last row: Our proposed labelling approach.

or equal to the CRF in 15/21 classes. Our final labelling result is in a comparable range of those reported in [15, 22, 29, 31]. We are aware that state-of-the-art methods [18] achieve higher scores on the MSRCv2 database, however, they are using higher-order terms in the CRF and globally learned contextual information while we deliberatly restrict our method to local classification results.

## 5.2 CamVid Database

The Cambridge-driving Labeled Video Database (CamVid) [7] is a collection of videos captured on road driving scenes, consisting of more than 10 minutes of high quality ($970 \times 720$), 30 Hz footage. A subset of 711 images is almost entirely annotated into 32 categories, however, in our experiments on this database we used only the 11 commonly used categories with the same splits for training and testing as presented in [7, 33]. Our unary classifaction results are (70.7/44.8%) (global/class average) which are improved to (75.0/47.8%) when using the CRF model. However, with our proposed label puzzle approach we can boost the scores to (81.7/49.6%), showing competitive results in comparison to Brostow *et al.* [7] (69.1/53.0%), and Sturgess *et al.* [33] (76.4/59.8%) and (79.8/59.9%) in a CRF setting with only unary terms and unary+pairwise terms, respectively.

# 6 Conclusion

In this paper we have proposed a novel approach for the task of image labelling, which allows to exploit local contextual information and the label topological structure observed in the training data. This is achieved by defining a label puzzle game, where a final labelling is obtained by maximizing the mutual agreement of structured class labels (our label puzzle pieces), which are associated with every pixel. We introduced a modification of the random forest classifiers in order to discriminatively learn and provide the structured class labels needed for the construction of a label puzzle game. We showed how the optimization problem underlying our approach can be optimized in order to obtain the final labelling, and we provided theoretical properties and a complexity analysis of our algorithm. Our approach achieved superior results in experiments on the MSRCv2 and CamVid databases when directly compared to a standard CRF formulation, supporting our claim that high-quality labelling results can be obtained by properly learning and integrating local contextual information at a low-level. As a future work, we plan to extend our approach by incorporating additional mid-level cues and global co-occurrence statistics.

**Acknowledgements** We acknowledge the financial support of the Austrian Science Fund (FWF) from project Fibermorph (P22261-N22), the Research Studios Austria Project $\mu$STRUCSCOP (818651) and the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project 'SIMBAD' (213250).

# References

[1] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9(7):1545–1588, 1997.

[2] I. Biederman. Perceiving real-world scenes. *Science*, 177(43):77–80, 1972.

[3] I. Biederman, R. J. Mezzanotte, and J. C. Rabinowitz. Scene perception: detecting and judging objects undergoing relational violations. *Cognitive Psychology*, 14(2), 1982.

[4] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. *(ICCV)*, 2001.

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *(PAMI)*, 2001.

[6] L. Breiman. Random forests. In *Machine Learning*, pages 5–32, 2001.

[7] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *(ECCV)*, 2008.

[8] T. S. Cho, S. Avidan, and W. T. Freeman. A probabilistic image jigsaw puzzle solver. In *(CVPR)*, 2010.

[9] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *(PAMI)*, 2002.

[10] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. In *(CVPR)*, 2005.

[11] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *(IJCV)*, 2004.

[12] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 36(1):3–42, 2006.

[13] J. M. Gonfaus, X. Boix, J. van de Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzalez. Harmony potentials for joint classification and segmentation. In *(CVPR)*, 2010.

[14] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *(ICCV)*, 2009.

[15] S. Kluckner, T. Mauthner, P. M. Roth, and H. Bischof. Semantic image classification using consistent regions and individual context. In *(BMVC)*, 2009.

[16] P. Kontschieder, S. Rota Bulò, H. Bischof, and M. Pelillo. Structured class-labels in random forests for semantic image labelling. In *(ICCV)*, 2011.

[17] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Associative hierarchical CRFs for object class image segmentation. In *(ICCV)*, 2009.

[18] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Graph cut based inference with co-occurrence statistics. In *(ECCV)*, 2010.

[19] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr. What, where & how many? Combining object detectors and CRFs. In *(ECCV)*, 2010.

[20] J. Lafferty, A. Mccallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *(ICML)*, 2001.

[21] D. Larlus and F. Jurie. Combining appearance models and markov random fields for category level object segmentation. In *(CVPR)*, 2008.

[22] S. Lazebnik and M. Raginsky. An empirical bayes approach to contextual region classification. In *(CVPR)*, 2009.

[23] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *(IJCV)*, 2001.

[24] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *(IJCV)*, 2004.

[25] T. Malisiewicz and A. A. Efros. Improving spatial support for objects via multiple segmentations. In *(BMVC)*, 2007.

[26] F. Porikli, O. Tuzel, and P. Meer. Covariance tracking using model update based on lie algebra. In *(CVPR)*, 2006.

[27] A. Rabinovich, A. Vedfaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *(ICCV)*, 2007.

[28] B. Russell, W. Freeman, A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *(CVPR)*, 2006.

[29] F. Schroff, A. Criminisi, and A Zisserman. Object class segmentation using random forests. In *(BMVC)*, 2008.

[30] J. Shi and J. Malik. Normalized cuts and image segmentation. *(PAMI)*, 2000.

[31] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *(ECCV)*, 2006.

[32] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *(CVPR)*, 2008.

[33] P. Sturgess, K. Alahari, L. Ladicky, and P.H.S. Torr. Combining appearance and structure from motion features for road scene understanding. In *(BMVC)*, 2009.

[34] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *(ICCV)*, 2003.

[35] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Sharing features: Efficient boosting procedures for multiclass object detection. In *(CVPR)*, 2004.

[36] J. M. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *(CVPR)*, 2006.

[37] X. Yang, N. Adluru, and L. J. Latecki. Particle filter with state permutations for solving image jigsaw puzzles. In *(CVPR)*, 2011.

# Structured Class-Labels in Random Forests for Semantic Image Labelling

Peter Kontschieder*, Samuel Rota Bulò△, Horst Bischof* and Marcello Pelillo△

*Institute for Computer Graphics and Vision
Graz University of Technology - Austria
{kontschieder,bischof}@icg.tugraz.at

△Dipartimento di Scienze Ambientali,
Informatica e Statistica
Università Ca' Foscari Venezia - Italy
{srotabul,pelillo}@dsi.unive.it

## Abstract

*In this paper we propose a simple and effective way to integrate structural information in random forests for semantic image labelling. By structural information we refer to the inherently available, topological distribution of object classes in a given image. Different object class labels will not be randomly distributed over an image but usually form coherently labelled regions. In this work we provide a way to incorporate this topological information in the popular random forest framework for performing low-level, unary classification. Our paper has several contributions: First, we show how random forests can be augmented with structured label information. In the second part, we introduce a novel data splitting function that exploits the joint distributions observed in the structured label space for learning typical label transitions between object classes. Finally, we provide two possibilities for integrating the structured output predictions into concise, semantic labellings. In our experiments on the challenging MSRC and CamVid databases, we compare our method to standard random forest and conditional random field classification results.*

## 1. Introduction

The field of visual object classification has received great attention and evolved in a remarkable manner during the past couple of years. Besides major progresses in the development of new image representations, a large variety of novel machine learning algorithms have been developed and applied to problems in the computer vision domain like object detection, classification, tracking, or action recognition. In this work we present a novel classification algorithm based on random forests, customized to the application of semantic image labelling [27, 9], *i.e.* a per-pixel classification of an image.

Using supervised learning algorithms for semantic image labelling typically requires a large amount of densely labelled training data. A label image corresponding to a natural scene exhibits structured information, respecting the topology shown in the scene. For instance, a typical street scene might result in coherently labelled regions of road, car, bicyclist and so on. Structured learning [30] provides ideas to take this form of additional, structural information into account and therefore intuitively fits to the needs of semantic image labelling or segmentation. Considering our example of the street scene, structured learning allows to integrate the actual label topology in the training process, *e.g.* a car should be driving on a road, but not on top of a building. However, exploiting this form of topological structure of the label images directly in machine learning algorithms for computer vision problems is still widely ignored.

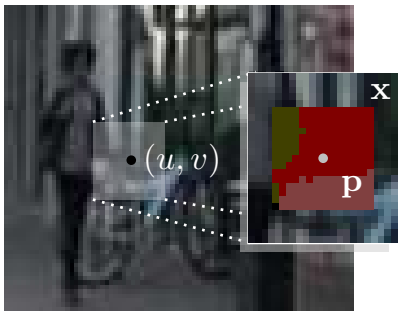



Figure 1. Training data example, as used in our proposed structured learning random forest. While standard random forests associate only the center label at position $(u, v)$ to an image patch $\mathbf{x}$, we incorporate the topology of the local label neighborhood $\mathbf{p}$ and therefore learn valid labelling transitions among adjacent object categories. Here: person, building and bicycle.

For the task of semantic image labelling, state-of-the-art approaches [16, 17, 13, 24] are typically using complementary features at different levels within random field models [18]. Low-level features are mostly calculated on a per-pixel basis and incorporate local color or texture statistics, while mid-level features operate on regions or superpixels to provide shape, continuity or symmetry information. Motivated by perceptual psychology [3], high-level features introduce global image statistics and information about inter-

object or contextual relations, seeking for proper scene configurations on the image level. In such a way, structural information is mostly incorporated on the highest, semantic level. Recently, [31] presented a way to effectively learn a contextual model named Auto-context. A boosting classifier is trained by iteratively learning from appearance and contextual information, collecting typical structures from the images. However, the learning phase is computationally very demanding. Other approaches like boosted random fields [29] or SpatialBoost [2] share both the disadvantage of significant computational complexity when considering contextual beliefs as weak learners. The work in [4] introduces a generalization of a support vector machine (SVM) for structured output regression, used to predict bounding boxes for the task of object localization. Finally, we refer to [23], which gives a comprehensive tutorial on structured learning and prediction in computer vision.

In this paper, we provide a simple but effective way to incorporate structural information in the popular random forest [8, 1, 12] learning algorithm which is considered to be competitive to other state-of-the-art learning techniques like boosting or SVMs. Inspired by ideas of structured learning we provide a novel way to incorporate joint statistics about the local label neighborhood in the random forest framework for learning typical labelling transitions among object class categories, as illustrated in Figure 1. In contrast to standard classification, which can only deal with a single (atomic) label per training sample during the training process, we take structured labelling information of the label neighborhood into account. Including this information at the classification level drastically improves the results and simultaneously counteracts the assignment of meaningless label configurations, as experienced when using standard random forests. Our proposed method is easy to implement and we show superior results on all our conducted experiments for the task of semantic image labelling on the challenging CamVid [9] and MSRCv2 [27] databases in comparison to standard random forests. To sum up, using our proposed structured learning method possesses several advantages when used for semantic image labelling:

- Including the label topology in the training stage yields a classification stage, that respects the label configurations observed during training
- Using structured label information in the classification avoids assigning implausible label transitions

The major drawback of our method is the need for densely labelled training data. However, this problem is shared with state-of-the-art image labelling algorithms and the results of our experiments on the MSRCv2 database indicate that also non-completely labelled training data are well handled by our method.

## 2. Randomized Decision Forests

We start by providing a brief review of the randomized decision forests [1, 12] and introducing some notations which will be used in the subsequent sections. Randomized decision forests exhibit several appealing properties: They are extremely fast for training and classification, can be easily parallelized [25], are inherently multi-class capable, tend not to overfit and are robust to label noise [8].

A (binary) *decision tree* is a tree-structured classifier which makes a prediction by routing a feature sample $\mathbf{x} \in \mathcal{X}$ through the tree to a leaf, where the actual classification is taking place. A *leaf* $\textsc{Lf}(\pi) \in \mathbb{T}$ is the simplest form of a decision tree and is able to cast a class prediction $\pi \in \mathcal{Y}$ for any sample it is reached by. In all other cases, a decision tree is a *node* $\textsc{Nd}(\psi, t_l, t_r) \in \mathbb{T}$, which is characterized by a binary test (or split) function $\psi(\mathbf{x}) : \mathcal{X} \to \{0, 1\}$, a left decision sub-tree $t_l \in \mathbb{T}$ and a right decision sub-tree $t_r \in \mathbb{T}$. The role of the test function is to decide whether a sample feature $\mathbf{x}$ reaching the node should be forwarded to its left decision sub-tree $t_l$ if $\psi(\mathbf{x}) = 0$, or to its right decision sub-tree $t_r$ if $\psi(\mathbf{x}) = 1$.

A (binary) *decision forest* is an ensemble $F \subseteq \mathbb{T}$ of (binary) decision trees which makes a prediction about a sample feature by averaging over the single predictions collected from the trees in the ensemble.

**Class prediction.** A class prediction for a sample $\mathbf{x} \in \mathcal{X}$ can be obtained from a tree $t \in \mathbb{T}$ by recursively branching the sample down the tree until a leaf is reached. Formally, we write the tree prediction function $h(\mathbf{x} \,|\, t) : \mathcal{X} \to \mathcal{Y}$ for a decision tree $t \in \mathbb{T}$ recursively as

$$h(\mathbf{x} \,|\, \textsc{Nd}(\psi, t_l, t_r)) = \begin{cases} h(\mathbf{x} \,|\, t_l) & \text{if } \psi(\mathbf{x}) = 0\,, \\ h(\mathbf{x} \,|\, t_r) & \text{if } \psi(\mathbf{x}) = 1\,, \end{cases}$$
$$h(\mathbf{x} \,|\, \textsc{Lf}(\pi)) = \pi\,.$$

The class prediction of a sample $\mathbf{x} \in \mathcal{X}$ given a forest $F$ can then be obtained from the individual decision tree predictions as the one receiving the majority of the votes, *i.e.*,

$$y^* = \arg\max_{y \in \mathcal{Y}} \sum_{t \in F} [h(\mathbf{x}|t) = y]\,. \tag{1}$$

where $[Q]$ is the Iverson bracket which gives 1 if proposition $Q$ is true and 0 otherwise. Combining the outputs of multiple decision trees into a single classifier supports the ability to generalize and mitigates the risk of overfitting, which may affect single decision trees.

**Randomized training.** We train the binary decision forest according to the extremely randomized trees algorithm [12]. Each tree in a forest is trained independently on a random subset of the training set $\mathcal{D} \subseteq \mathcal{X} \times \mathcal{Y}$ according to a

recursive learning procedure. If $\mathcal{D}$ is smaller than a minimum size or if the entropy of its class distribution $E(\mathcal{D})$ is below a given threshold, a leaf $\textsc{Lf}(\pi)$ is grown where the class prediction $\pi$ is set to the most represented class in the training data $\mathcal{D}$, *i.e.*,

$$\pi \in \arg\max_{z \in \mathcal{Y}} \sum_{(\mathbf{x},y)\in\mathcal{D}} [y = z] \,. \tag{2}$$

Otherwise a node $\textsc{Nd}(\psi, t_l, t_r)$ is grown, where $\psi$ is a test function selected from a randomly generated set $\Psi$, maximizing the expected information gain about the label distribution due to the split $\{\mathcal{D}_l^\psi, \mathcal{D}_r^\psi\}$ of the training data, which has been induced by $\psi$ [21]:

$$\begin{aligned}\psi &= \arg\max_{\psi'\in\Psi} \{E(\mathcal{D}) - E(\mathcal{D}\,;\,\psi')\} = \arg\min_{\psi'\in\Psi} E(\mathcal{D}\,;\,\psi') \\ &= \arg\min_{\psi'\in\Psi} \left\{ \frac{|\mathcal{D}_l^{\psi'}|}{|\mathcal{D}|} E(\mathcal{D}_l^{\psi'}) + \frac{|\mathcal{D}_r^{\psi'}|}{|\mathcal{D}|} E(\mathcal{D}_r^{\psi'}) \right\} .\end{aligned}$$

Finally, the trees $t_l$ and $t_r$ are recursively grown with their respective training data $\mathcal{D}_l^\psi$ and $\mathcal{D}_r^\psi$.

In case of unbalanced training data among the different classes to be learned, the tree classifiers can be trained by weighting each label $z \in \mathcal{Y}$ according to the inverse class frequencies observed in the training data $\mathcal{D}$, *i.e.*, $\omega_z = \left(\sum_{(\mathbf{x},y)\in\mathcal{D}} [y = z]\right)^{-1}$. The weights are also considered in the computation of the expected (weighted) information gain, which determines the selection of the best test function during the training procedure. This allows to reduce the class average prediction error.

## 3. Random Forests in Computer Vision

Recently, random forests were customized for a large variety of tasks in computer vision [5, 11, 20, 19, 21, 22]. Typically, computer vision applications have used random forests for classification tasks in the image domain, where the feature space is anchored to a pixel grid topology. They are trained on a specific feature space $\mathcal{X}$, which consists of a set of $d \times d$ patches extracted from a set of multi-channel images $\mathcal{I}$, where channels may include color features such as gradients, filter banks, *etc*.

More formally, a multi-channel training image is a 3-dimensional matrix $I$ and $I_{(u,v,c)}$ denotes the value at pixel $(u, v)$ and channel $c$ in the image. A patch is simply a triplet $(u, v, I) \in \mathcal{X}$, representing the coordinates $(u, v)$ of the patch center in image $I \in \mathcal{I}$. The label space $\mathcal{Y} = \{1, \ldots, k\}$ is given by the set of $k$ object classes we are going to find in the images.

Different types of test functions for a patch $\mathbf{x} = (u, v, I) \in \mathcal{X}$ have been investigated for the classification task. The following are the most commonly used ones:

$$\begin{aligned}\psi^{(1)}(\mathbf{x}\,|\,\theta_1, \tau) &= [I_{(u,v,0)+\theta_1} > \tau] \,, \\ \psi^{(2)}(\mathbf{x}\,|\,\theta_1, \theta_2, \tau) &= [I_{(u,v,0)+\theta_1} - I_{(u,v,0)+\theta_2} > \tau] \,, \\ \psi^{(3)}(\mathbf{x}\,|\,\theta_1, \theta_2, \tau) &= [I_{(u,v,0)+\theta_1} + I_{(u,v,0)+\theta_2} > \tau] \,, \\ \psi^{(4)}(\mathbf{x}\,|\,\theta_1, \theta_2, \tau) &= \left[\left|I_{(u,v,0)+\theta_1} - I_{(u,v,0)+\theta_2}\right| > \tau\right] \,,\end{aligned}$$

where $\theta_i = (\delta u_i, \delta v_i, c_i)$, $i = 1, 2$, are displacement parameters relative to the patch center used to index a point in the patch, and $\tau \in \mathbb{R}$ is a threshold. Note that test functions of random type and with randomly generated parameters are drawn during the training procedure to form the sets $\Psi$ of split functions in each node of the decision trees.

Once a random forest $F$ has been trained, the classification of a test image can be naively obtained by labelling each pixel with the most probable class predicted by the forest, centered on the $d \times d$ patch.

## 4. Structured Learning in Random Forests

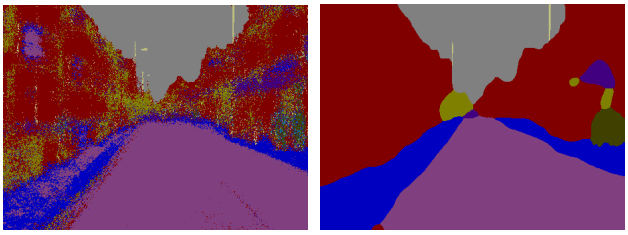
In traditional classification approaches like the one presented in the previous section, input data samples are assigned to single, *atomic* class labels, acting as arbitrary identifiers without any dependencies among them. For many computer vision problems however, this model is limited because the label space of a classification task does exhibit an inherently topological structure, which renders the class labels explicitly interdependent. Although this structured label space is already present in the training data, it remains largely unexploited by standard classification approaches, like the random forests introduced in the previous sections. Consequently, when applying standard random forest classifiers for semantic image labelling, the obtained results are quite noisy (*e.g.*, see Figure 2(c)). Indeed, a random patch extracted from the labelled image will likely show a configuration which never appeared in the ground-truth classification used to train the classifiers.

To overcome this limitation, we propose a novel way of enriching the standard random forest classifiers by rendering them aware of the local topological structure of the output label space. Towards this end, we depart from the traditional classification paradigm and address the problem from a structured learning perspective [30] within the random forest framework.

### 4.1. Structured Label Space

Our structured label space $\mathcal{P}$ consists of $d' \times d'$ patches of object class labels, *i.e.*, $\mathcal{P} = \mathcal{Y}^{d'\times d'}$. With $p_{ij} \in \mathcal{Y}$ we denote the $ij$-entry of the label patch $\mathbf{p}$. Additionally, we index the entries in a way that index $(0, 0)$ takes the central position. To distinguish between a patch $\mathbf{x}$ from the feature space $\mathcal{X}$ (see, Section 3) and a patch $\mathbf{p}$ from the

(a) Original (b) Ground truth



(c) Random Forest (d) Our method

Figure 2. Examples of object class segmentations using unary classifiers. Best viewed in color.

structured label space $\mathcal{P}$, we refer to them as *feature patch* and *label patch*, respectively. Each training feature patch $\mathbf{x} = (u, v, I)$ has an associated label patch $\mathbf{p}$ which holds the labels of all pixels of image $I$ within a $d' \times d'$ neighborhood of $(u, v)$ (see, Figure 1). In other words, $p_{ij}$ is the label of pixel $(u + i, v + j)$ in image $I$. Please note that the size $d'$ of the label patch may differ from the size $d$ of the feature patch.

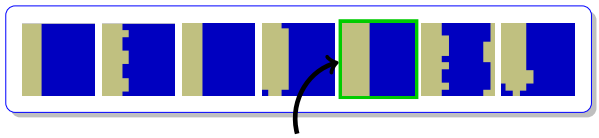
In the next subsection we show how the split function selection strategy in the nodes of the random forest will be adapted to account for the new label space. However, for the moment we will simply assume that the training patches from $\mathcal{D} \subseteq \mathcal{X} \times \mathcal{P}$ have been routed through the tree to the leafs. Consider now a leaf $t$ and let $\mathcal{P}_t \subseteq \mathcal{P}$ be the set of label patches present in the training data used to grow the leaf (see Figure 8). Then, the class label $\boldsymbol{\pi}$ parametrizing the leaf is now a structured label of size $d' \times d'$ from $\mathcal{P}$ and not just an atomic label from $\mathcal{Y}$ as in the standard random forest. A good selection for the structured class label should represent a mode of the joint distribution of the label patches in $\mathcal{P}_t$.

In order to keep the complexity of this step low, we compute the joint probability by making a pixel independence assumption as

$$\Pr(\mathbf{p}|\mathcal{P}_t) = \prod_{i,j} \Pr^{(i,j)}(p_{ij}|\mathcal{P}_t)\,, \tag{3}$$

where $\Pr^{(i,j)}(c|\mathcal{P}_t)$ represents the marginal class distribution over all the label patches of pixel position $(i, j)$. The label patch $\boldsymbol{\pi}$ is finally selected for leaf $t$ as the one in $\mathcal{P}_t$ maximizing the joint probability:

$$\boldsymbol{\pi} = \arg\max_{\mathbf{p} \in \mathcal{P}_t} \Pr(\mathbf{p}|\mathcal{P}_t)\,. \tag{4}$$



Selection of $\boldsymbol{\pi}$ based on joint probability

Figure 3. Example of label patches reaching a leaf during training. Based on the joint probability distribution of labels in the leaf a label patch $\boldsymbol{\pi}$ is selected.

### 4.2. Test Function Selection for Structured Labels

The change introduced in the label space should be coupled with an adaptation of the way a test function is selected in each node of the random forest during the training procedure in order to account for the additional information carried by the structured labels. One naive solution is to port the test selection criterion actually used in the standard random forest to our context, *e.g.*, by simply associating each patch with the label we find in the center of the associated label patch $\mathbf{p}$. This, however, results in a split of the training set which is identical to what the standard random forest implementation does, without thus properly exploiting the label topology.

In order to take advantage of the new label space, we propose to select the best split function at each node based on the information gain with respect to a two-label joint distribution. Specifically, we associate each training pair $(\mathbf{x}, \mathbf{p})$ with two labels: One label is provided by the patch central pixel label $p_{00}$, whereas the second one is given by $p_{ij}$, where $(i, j)$ is a patch label position which has been uniformly drawn (once per node). By adopting this new test function selection criterion, all entries of a label patch have the chance to actively influence the way a feature patch is branched through the tree during the training procedure.

One drawback of this new test selection method is the increased complexity deriving from the evaluation of the 2-label joint distribution ($|\mathcal{Y}|^2$ elements) instead of the simple, single label distribution ($|\mathcal{Y}|$ elements). To overcome this we consider also a different test function selection method, which consists in associating each training pair $(\mathbf{x}, \mathbf{p})$ with just one label, but instead of considering label $p_{00}$ of the central pixel, we consider a label $p_{ij}$ from a random position $(i, j)$, which is generated once per node. By so doing, we still have the effect that all entries of the label patch may influence the learning procedure, but at no higher computational cost.

### 4.3. Structured Label Predictions

The structured predictions gathered from the trees of a forest have to be combined into a single label patch prediction. To this end, we follow a procedure which is similar to the one adopted in order to select the label patch $\boldsymbol{\pi}$ in a leaf (see Section 4.1).

Consider a trained forest $F$, a test patch $\mathbf{x} = (u, v, I)$. and let $\mathcal{P}_F$ be the set of predictions for $\mathbf{x}$ gathered from each tree $t \in F$:

$$\mathcal{P}_F = \{h(\mathbf{x}|t) \in \mathcal{P} \,:\, t \in F\}\,. \tag{5}$$

Similarly to (4), the label patch prediction of the forest $F$ for feature patch $\mathbf{x}$ is given by the one maximizing the patch label joint probability estimated from $\mathcal{P}_F$, *i.e.*,

$$\mathbf{p}^* = \arg\max_{\mathbf{p}\in\mathcal{P}_F} \Pr(\mathbf{p}\,|\,\mathcal{P}_F)\,, \tag{6}$$

where $\Pr(\mathbf{p}|\mathcal{P}_F)$ is defined as in (3).

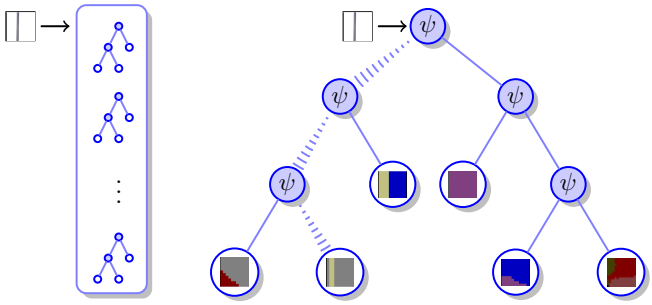



Figure 4. Prediction of the structured label of a feature patch in a random forest. The feature patch is routed through each tree in the forest according to the test functions $\psi$ in the tree nodes until a leaf is reached, holding the learned label transitions between color-coded classes. The structured label in the leaf is then assigned to the feature patch. Best viewed in color.

### 4.4. Simple Fusion of Structured Predictions

As opposed to standard classification algorithms which, given a test image $I$, directly assign an object class label to a each pixel, our classifiers cast a prediction for each pixel, involving also the neighboring ones. Indeed, if $\mathbf{p} \in \mathcal{P}$ is the patch label predicted for pixel $(u, v)$ in a test image then a neighbor $(u + i, v + j)$ in a $d' \times d'$ neighborhood of $(u, v)$ could be classified as $p_{ij} \in \mathcal{Y}$. Hence, for each test pixel we collect $d' \times d'$ class predictions, which have to be integrated into a single class prediction. A simple way of performing this operation consists in selecting the most voted class per pixel. This process is illustrated in Figure 5.

The outcome of this fusion step is a labelling $\boldsymbol{\ell}$ from the set $\mathcal{L}$ of all possible labellings for the image, $\ell_{uv} \in \mathcal{Y}$ being the class label associated with pixel $(u, v)$.

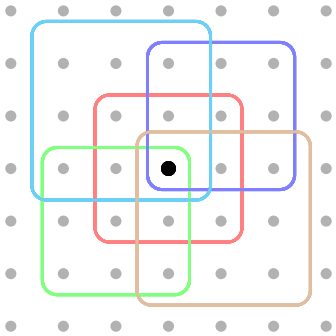


Figure 5. Fusion of structured predictions. Each pixel collects class hypotheses from the structured labels predicted for itself and neighboring pixels, which have to be fused into a single class prediction. For clarity reasons, only 5/9 label patches are drawn. Best viewed in color.

### 4.5. Optimizing the Label Patch Selection

A different and more principled approach to the computation of the final labelling can be obtained by optimizing the label patch selection with respect to a given labelling rather than solely taking (6) for each pixel. This allows to better exploit the label patch diversity in the set of predictions $\mathcal{P}_F$ obtained from (5).

We define the *agreement* of an individual label patch $\mathbf{p}$ located at $(i, j) \in I$ with a given labelling $\ell \in \mathcal{L}$ as the number of corresponding pixels sharing the same label, *i.e.*

$$\phi^{(i,j)}(\mathbf{p}, \boldsymbol{\ell}) = \sum_{(u,v)\in I} \left[p_{(u-i)(v-j)} = \ell_{uv}\right]. \tag{7}$$

Furthermore, let $\mathbf{z} \in \mathcal{Z}_I$ be an assignment of label patches to pixels in $I$, $z_{uv} \in \mathcal{P}_F$ being a label patch for pixel $(u, v)$ taken from (5), where $\mathcal{Z}_I$ denotes the set of all such assignments for image $I$. Then, for a particular configuration $\mathbf{z} \in \mathcal{Z}_I$ and a labelling $\boldsymbol{\ell} \in \mathcal{L}$, the total agreement $\Phi(\mathbf{z}, \boldsymbol{\ell})$ is defined as the sum of agreements of each label patch in $\mathbf{z}$ with the labelling $\boldsymbol{\ell}$ according to

$$\Phi(\mathbf{z}, \boldsymbol{\ell}) = \sum_{(u,v)\in I} \phi^{(u,v)}(z_{uv}, \boldsymbol{\ell})\,. \tag{8}$$

As we want to find the label patch configuration that leads to the maximum total agreement with the labelling of a test image $I$, we can write the optimal solution as a pair $(\mathbf{z}^*, \boldsymbol{\ell}^*) \in \mathcal{Z}_I \times \mathcal{L}$, where

$$(\mathbf{z}^*, \boldsymbol{\ell}^*) \in \arg\max_{(\mathbf{z},\boldsymbol{\ell})} \{\Phi(\mathbf{z}, \boldsymbol{\ell}) \,|\, (\mathbf{z}, \boldsymbol{\ell}) \in \mathcal{Z}_I \times \mathcal{L}\}\,. \tag{9}$$

To solve (9), we use a simple, iterative optimization method that alternates between selecting the best agreeing label patch per pixel and producing a new labelling as described in Section 4.4. For a more detailed description, we refer the interested reader to our recent work [15].

## 5. Experiments

In this section we evaluate our proposed structured learning random forest algorithm on the challenging CamVid [9] and MSRCv2 [27] databases for the task of semantic image labelling. For performance reasons, we implemented our method in C++ and ran all experiments on a standard desktop computer with 2.9 GHz and 2 GB RAM.

In all our experiments we show a comparison to a standard random forest implementation (denoted as 'Our Baseline RF'), which is actually a special instance of our method with a label patch size of $1 \times 1$ and a fixed label center position. Where available, we list results of state-of-the-art methods [27, 9, 14] that are also using random forests (but not the same features), in order to show that our baseline random forest implementation achieves state-of-the-art performance. Additionally, we compare to the results obtained when minimizing the energy term of a pairwise, conditional random field (CRF) model with graph cuts, using the publicly available GCO [7] implementation[1]. To this end, we provide the class label statistics of the baseline random forest as unary or data terms and use the standard, contrast-sensitive Potts model as suggested in [6] for the pairwise or smoothness term.

To show the impact of the respective stages of our method, we evaluate different training ['Structure' / 'Full'] and classification ['Simple Fusion' / 'Optimized Selection'] procedures as follows: 'Structure' considers the structured label patches but only takes one random label position, *i.e.* a single label distribution into account for training. 'Full' considers structured label patches and the two-label joint distribution in the split functions (see Section 4.2 for more details). 'Simple Fusion' and 'Optimized Selection' refer to the fusion methods of the structured output predictions as described in Sections 4.4 and 4.5, respectively.

We used the same, primitive low-level features for training both, our baseline and our novel structured learning random forests, since our primary intention is to show the improvement when the label topology is taken into account: CIELab raw channel intensities, first and second order derivatives as well as HOG-like features, computed on the L-Channel. In all experiments we fixed the feature patch size to $24 \times 24$ and trained 10 trees, using 500 iterations for the node tests and stopping when less then 5 samples per leaf were available.

We list the scores of our experiments according to the same evaluation criteria as used in [27, 9, 14] and additionally include the more strict average intersection vs. union score as *e.g.* used in the PASCAL VOC challenges [10]. In particular, '*Global*' refers to the percentage of all pixels that were correctly classified, '*Avg(Class)*'[2] expresses the average recall over all classes and '*Avg(Pascal)*'[3] denotes the average intersection vs. union score.

| Method | *Global* | *Avg(Class)* | *Avg(Pascal)* |
|---|---|---|---|
| RF using Motion and Structure cues [9] | 61.8 | 43.6 | - |
| Our Baseline RF | 69.9 | 42.2 | 30.6 |
| Our Baseline RF + CRF | 74.5 | 45.4 | 33.8 |
| Our method (Structure + Simple Fusion) | 74.8 | 45.0 | 34.1 |
| Our method (Full + Simple Fusion) | 76.8 | 46.1 | 35.4 |
| Our method (Full + Optimized Selection) | 79.2 | 46.0 | 36.2 |

Table 1. Classification results on CamVid database for label patch size $13 \times 13$.

### 5.1. CamVid Database Experiments

The Cambridge-driving Labeled Video Database (CamVid) [9] is a collection of videos captured on road driving scenes. It consists of more than 10 minutes of high quality ($970 \times 720$), 30 Hz footage and is divided into four sequences. Three sequences were taken during daylight and one at dusk. A subset of 711 images is almost entirely annotated into 32 categories, but we used only the 11 commonly used categories with the same splits for training and testing as presented in [9, 28].

We resized the training images by a factor of 0.5 and randomly collected training samples on a regular lattice with a stride of 10, resulting in approximately 850k training samples. The training time per tree is 23 minutes when using the single label test and 30 minutes with the joint label test. For the experiment where we only consider the labelling transitions, we reduced the stride to 8. In order to correct the imbalance among samples of different classes, we applied an inverse frequency weighting as mentioned in Section 2.

**CamVid - 11 Classes.** The standard protocol for evaluating on the CamVid database considers the following 11 object categories, forming a majority of the overall labelled pixels (89.16%): ROAD, BUILDING, SKY, TREE, SIDEWALK, CAR, COLUMN_POLE, SIGN-SYMBOL, FENCE, PEDESTRIAN and BICYCLIST. In Table 1 we list our results using a label patch size of $13 \times 13$, clearly indicating the performance boost when using our proposed structured learning method over the standard random forest. We can achieve comparable results to the CRF implementation with the Simple Fusion approach and significantly increase the scores using the Optimized Selection. We explain this by the fact that our method is restricted to pick from a candidate set of semantically plausible label patches provided by the trees, rather than propagating arbitray label configurations in the associated graph.

In Figure 6 we show the influence of the label patch size,

[1] http://vision.csd.uwo.ca/code/

[2] $\frac{\text{True Positives}}{\text{True Positives + False Negatives}}$

[3] $\frac{\text{True Positives}}{\text{True Positives + False Negatives + False Positives}}$

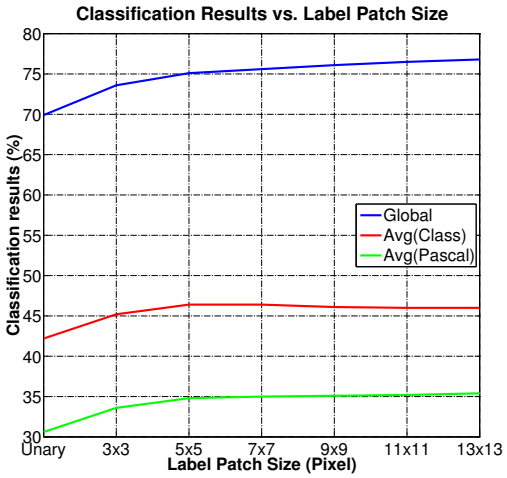



Figure 6. Classification results on CamVid database as a function of the label patch size using Simple Fusion.

| Method | *Global* | *Avg(Class)* | *Avg(Pascal)* |
|---|---|---|---|
| Our Baseline RF | 63.8 | 44.2 | 29.8 |
| Our Baseline RF + CRF | 68.2 | 48.2 | 33.3 |
| Our method (Structure + Simple Fusion) | 69.9 | 50.4 | 35.0 |
| Our method (Full + Simple Fusion) | 71.6 | 50.1 | 35.8 |
| Our method (Full + Optimized Selection) | 72.5 | 51.4 | 36.4 |

Table 2. Classification results for labelling transitions on the CamVid database for label patch size $11 \times 11$.

*i.e*. the size of the considered label topology during training and classification using the configuration 'Full + Simple Fusion'. It is clearly shown that even a small neighborhood ($\geq 5 \times 5$) leads to a significant boost in the classification stage.

**Labelling Transition Evaluation.** In this experiment we evaluate only the transitions between object classes to demonstrate the impact of structured predictions on the label border classification results. To perform this experiment, we discarded all labels in the ground truth information when they were outside a radius of 24 pixels to a transition between two or more classes. This results in a drop to $41.9\%$ of the original amount of labelled pixels. In Table 2, the corresponding results are listed when using a label neighborhood of $11 \times 11$. Although the global score has slightly dropped compared to the previous experiment, we obtain improvements on the (stricter) *Avg(Class)* and *Avg(Pascal)* criteria. This strengthens our assumptions that the proposed framework yields to superior results, especially when classifying transitions between object classes.

## 5.2. MSRCv2 Database Experiments

To show that our method also yields to an improvement when the images are not entirely labelled, we performed another experiment on the MSRCv2 Database [27].

| Method | *Global* | *Avg(Class)* | *Avg(Pascal)* |
|---|---|---|---|
| Texton forests naïve (supervised) [26] | 49.7 | 34.5 | - |
| RF using covariance features [14] | 55.8 | 42.2 | - |
| Our Baseline RF | 54.8 | 43.4 | 28.3 |
| Our Baseline RF + CRF | 61.0 | 52.8 | 35.1 |
| Our method (Structure + Simple Fusion) | 60.8 | 51.0 | 33.8 |
| Our method (Full + Simple Fusion) | 60.8 | 51.1 | 33.9 |
| Our method (Full + Optimized Selection) | 63.9 | 55.6 | 37.6 |

Table 3. Classification results on MSRCv2 database for label patch size $11 \times 11$.

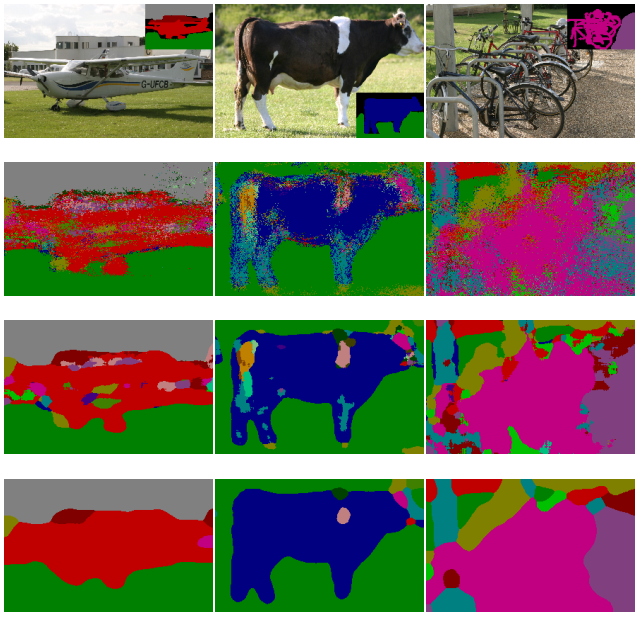


Figure 7. Qualitative labelling results on images of the MSRCv2 database. Top row: Original images with ground truth annotations. Second row: Labelling using our baseline random forest classifier. Third row: Full + Simple Fusion. Last row: Full + Optimized Selection. Best viewed in color.

This database consists of 532 images containing 21 object classes and predefined splits into 276 training and 256 test images. We collected the training samples on a regular lattice with a stride of 5, leading to approximately 500k training samples and training times of 13 and 17 minutes per tree using single or joint label distributions, respectively. In contrast to the almost completely labelled CamVid database, the labellings for MSRCv2 are only available for $71.9\%$ of the pixels, hence more roughly sketching the object classes of interest. In Figure 7 we show some qualitative results and in Table 3, we provide the scores for a label neighborhood size of $11 \times 11$ and again find an improvement with our structured learning algorithm. The gain of using the joint statistics over the single label distribution seems to vanish in the Simple Fusion approach, however, we explain this by the fact that our algorithm does not see enough properly labelled transitions between different classes.

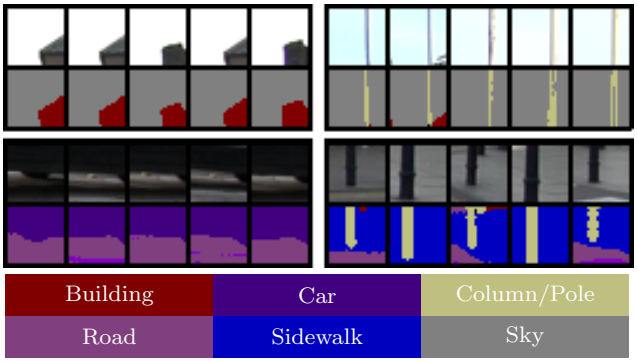



Figure 8. Illustration of feature patches with corresponding label patches, collected from different leaf nodes when trained on CamVid database. Bottom rows: Label sets and associated colors. Best viewed in color.

## 6. Conclusions

In this paper we presented a simple and effective way to integrate ideas from structured learning into the popular random forest framework for the task of semantic image labelling. In particular, we incorporated the topology of the local label neighborhood in the training process and therefore intuitively learned valid labelling transitions among adjacent object categories. During the tree construction, we used topological joint label statistics of the training data in the node split functions for exploring the structured label space. For classification, we provided two possibilities for fusing the structured label predictions: A simple method using overlapping predictions and a more principled approach, selecting most compatible label patches in the neighborhood. We provided several experiments on the callenging CamVid and MSRCv2 databases and found superior results when compared to standard random forest or conditional random field (using pairwise potentials) classification results. In our future work we will investigate how the output of our classifiers can be used as higher-order potential generator in a CRF.

**Acknowledgements.** We acknowledge the financial support of the Austrian Science Fund (FWF) from project 'Fibermorph' with number P22261-N22 and the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project 'SIMBAD', grant no. 213250.

## References

[1] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 1997. 2

[2] S. Avidan. Spatialboost: Adding spatial reasoning to adaboost. In *(ECCV)*, 2006. 2

[3] I. Biederman. Perceiving real-world scenes. *Science*, 1972. 1

[4] M. Blaschko and C. Lampert. Learning to localize objects with structured output regression. In *(ECCV)*, 2008. 2

[5] A. Bosch, A. Zisserman, and X. Muñoz. Image classification using random forests and ferns. In *(ICCV)*, 2007. 3

[6] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *(ICCV)*, 2001. 6

[7] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *(PAMI)*, 2001. 6

[8] L. Breiman. Random forests. In *Machine Learning*, 2001. 2

[9] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *(ECCV)*, 2008. 1, 2, 6

[10] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *(IJCV)*, 2010. 6

[11] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky. Hough forests for object detection, tracking, and action recognition. *(PAMI)*, 2011. 3

[12] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 2006. 2

[13] J. M. Gonfaus, X. Boix, J. van de Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzalez. Harmony potentials for joint classification and segmentation. In *(CVPR)*, 2010. 1

[14] S. Kluckner, T. Mauthner, P. M. Roth, and H. Bischof. Semantic image classification using consistent regions and individual context. In *(BMVC)*, 2009. 6, 7

[15] P. Kontschieder, S. Rota Bulò, M. Donoser, M. Pelillo, and H. Bischof. Semantic image labelling as a label puzzle game. In *(BMVC)*, 2011. 5

[16] L. Ladicky, C. Russell, P. Kohli, and P. Torr. Graph cut based inference with co-occurrence statistics. In *(ECCV)*, 2010. 1

[17] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, and P. Torr. What, where & how many? combining object detectors and CRFs. In *(ECCV)*, 2010. 1

[18] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *(ICML)*, 2001. 1

[19] C. Leistner, A. Saffari, and H. Bischof. MIForests: Multiple-instance learning with randomized trees. In *(ECCV)*, 2010. 3

[20] C. Leistner, A. Saffari, J. Santner, and H. Bischof. Semi-supervised random forests. In *(ICCV)*, 2009. 3

[21] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *(CVPR)*, 2005. 3

[22] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. In *(NIPS)*, 2006. 3

[23] S. Nowozin and C. Lampert. Structured learning and prediction in computer vision. In *Foundations and Trends in Computer Graphics and Vision*, 2011. 2

[24] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *(ICCV)*, 2007. 1

[25] T. Sharp. Implementing decision trees and forests on a GPU. In *(ECCV)*, 2008. 2

[26] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *(CVPR)*, 2008. 7

[27] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *(ECCV)*, 2006. 1, 2, 6, 7

[28] P. Sturgess, K. Alahari, L. Ladicky, and P. Torr. Combining appearance and structure from motion features for road scene understanding. In *(BMVC)*, 2009. 6

[29] A. Torralba, K. P. Murphy, and W. T. Freeman. Contextual models for object detection using boosted random fields. In *(NIPS)*. 2005. 2

[30] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector learning for interdependent and structured output spaces. In *(ICML)*, 2004. 1, 3

[31] Z. Tu. Auto-context and its application to high-level vision tasks. In *(CVPR)*, 2008. 2

# High Order Structural Matching Using Dominant Cluster Analysis*

Peng Ren, Richard C. Wilson, and Edwin R. Hancock

Department of Computer Science, The University of York, York, YO10 5GH, UK
{pengren,wilson,erh}@cs.york.ac.uk

**Abstract.** We formulate the problem of high order structural matching by applying *dominant cluster analysis* (DCA) to a direct product hypergraph (DPH). For brevity we refer to the resulting algorithm as DPH-DCA. The DPH-DCA can be considered as an extension of the game theoretic algorithms presented in [8] from clustering to matching, and also as a reduced version of reduced version of the method of ensembles of affinity relations presented in [6]. The starting point for our method is to construct a $K$-uniform direct product hypergraph for the two sets of higher-order features to be matched. Each vertex in the direct product hypergraph represents a potential correspondence and the weight on each hyperedge represents the agreement between two $K$-tuples drawn from the two feature sets. Vertices representing correct assignment tend to form a strongly intra-connected cluster, i.e. a dominant cluster. We evaluate the association of each vertex belonging to the dominant cluster by maximizing an objective function which maintains the $K$-tuple agreements. The potential correspondences with nonzero association weights are more likely to belong to the dominant cluster than the remaining zero-weighted ones. They are thus selected as correct matchings subject to the one-to-one correspondence constraint. Furthermore, we present a route to improving the matching accuracy by invoking prior knowledge. An experimental evaluation shows that our method outperforms the state-of-the-art high order structural matching methods[10][3].

## 1 Introduction

Many problems in computer vision and machine learning can be posed as that of establishing the consistent correspondences between two sets of features. Traditional matching approaches are usually confined to structures with pairwise relations. Recently, a number of researchers have attempted to extend the matching process to incorporate higher order relations. Zass *et al.* [10] are among the first to investigate this problem by introducing a probabilistic hypergraph matching framework, in which higher order relationships are marginalized to unary order. It has already been pointed out in [1] that this graph approximation is just a low pass representation of the original hypergraph and causes information loss and inaccuracy. On other hand, Duchenne *et al.* [3] have developed the spectral technique for graph matching [4] into a higher order matching

* We acknowledge the financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250). Edwin R. Hancock is supported by a Royal Society Wolfson Research Merit Award.

G. Maino and G.L. Foresti (Eds.): ICIAP 2011, Part I, LNCS 6978, pp. 1–8, 2011.

framework using the so called *tensor power iteration*. Although they adopt an $L_1$ norm constraint in computation, the original objective function is subject to an $L_2$ norm and does not satisfy the basic probabilistic properties.

We present a framework based on applying *dominant cluster analysis* (DCA) to a direct product hypergraph (DPH). The idea is to extend the main cluster method of Leordeanu and Hebert [4] for graphs and its generalization for higher order matching [3], using dominant cluster analysis. Furthermore, we present a method for initializing our algorithm that can be used to suppress outliers. This improves the matching performance of our method, and comparable results can not be achieved by using alternative high order matching algorithms [3][10]. Similar ideas have recently been presented in [6]. Our method however, generalises the methods descrbibed in [3][10] from graphs to hypergraphs, and is more pricipled in its formulation.

## 2 Problem Formulation

We represent the set of $K$th order feature relationships by a $K$-uniform hypergraph $HG(V,E)$, whose hyperedges have identical cardinality $K$. Each vertex $v_i \in V$ in the $K$-uniform hypergraph $HG(V,E)$ represents one element in the feature set. Each hyperedge $e_i \in E$ represents one $K$-tuple $\{v_{i_1}, \cdots, v_{i_K}\} \in V$ and the weight attached to each hyperedge represents the similarity measure on the $K$-tuple encompassed by the hyperedge. For simplicity, we denote a vertex $v_i$ by its index $i$ in the remainder of our work. The $K$-uniform hypergraph $HG(V,E)$ can be represented as a $K$th order tensor $\mathcal{H}$, whose element $H_{i_1,\cdots,i_K}$ is the hyperedge weight if there is a hyperedge encompassing the vertex subset $\{i_1, \cdots, i_K\} \in V$, and zero otherwise. The problem of matching two feature sets both constituted by $K$th order relationships can then be transformed to that of matching the two associated $K$-uniform hypergraphs $HG(V,E)$ and $HG'(V',E')$. To this end, we establish the high order compatibility matrix $\mathcal{C}$, i.e. compatibility tensor, for $HG(V,E)$ and $HG'(V',E')$. The elements of the $K$th order compatibility tensor $\mathcal{C}$ are defined as follows

$$C_{i_1 i'_1,\cdots,i_K i'_K} = \begin{cases} 0 \quad \text{if } H_{i_1,\cdots,i_K} = 0 \quad \text{or } H'_{i'_1,\cdots,i'_K} = 0; \\ s(H_{i_1,\cdots,i_K}, H'_{i'_1,\cdots,i'_K}) \text{ otherwise;} \end{cases} \tag{1}$$

where $s(\cdot,\cdot)$ is a function that measures hyperedge similarity. We define the hyperedge similarity using a Gaussian kernel $s(H_{i_1,\cdots,i_K}, H'_{i'_1,\cdots,i'_K}) = \exp(-\|H_{i_1,\cdots,i_K} - H'_{i'_1,\cdots,i'_K}\|_2^2/\sigma_1)$ where $\sigma_1$ is a scaling parameter. Many alternative similarity measures can be used instead. Each element of the compatibility tensor $\mathcal{C}$ represents a similarity measure between the two corresponding hyperedges. The hyperedge pair $\{i_1, \cdots, i_K\}$ and $\{i'_1, \cdots, i'_K\}$ with a large similarity measure has a large probability $\Pr(\{i_1, \cdots, i_K\} \leftrightarrow \{i'_1, \cdots, i'_K\}|H, H')$ for matching. Here the notation $\leftrightarrow$ denotes a possible matching between a pair of hyperedges or a pair of vertices. Under the conditional independence assumption of the matching process [10], the hyperedge matching probability can be factorized over the associated vertices of the hypergraphs as $\Pr(\{i_1, \cdots, i_K\} \leftrightarrow \{i'_1, \cdots, i'_K\}|HG, HG') = \prod_{n=1}^{K} \Pr(i_n \leftrightarrow i'_n|HG, HG')$ where $\Pr(i_n \leftrightarrow i'_n|HG, HG')$ denotes the probability for the possible matching $i_n \leftrightarrow i'_n$

to be correct. For two hypergraphs $HG(V,E)$ and $HG(V',E')$ with $|V| = N$ and $|V'| = N'$ respectively, we denote their $N \times N'$ matching matrix by $\mathbf{P}$ with entries $P_{ii'} = \Pr(i \leftrightarrow i'|HG, HG')$. High order matching problems can be formulated as locating the matching probability that most closely accords with the elements of the compatibility tensor, i.e. seeking the optimal $\mathbf{P}$ by maximizing the objective function

$$f(\mathbf{P}) = \sum_{i_1=1}^{N}\sum_{i'_1=1}^{N'}\cdots\sum_{i_K=1}^{N}\sum_{i'_K=1}^{N'} C_{i_1 i'_1,\cdots,i_K i'_K}\Pr(\{i_1,\cdots,i_K\}\leftrightarrow\{i'_1,\cdots,i'_K\}|HG,HG')$$
$$= \sum_{i_1=1}^{N}\sum_{i'_1=1}^{N'}\cdots\sum_{i_K=1}^{N}\sum_{i'_K=1}^{N'} C_{i_1 i'_1,\cdots,i_K i'_K}\prod_{n=1}^{K} P_{i_n i'_n} \quad (2)$$

subject to $\forall i,j,\ P_{ii} \geq 0$ and $\sum_{i=1}^{N}\sum_{i'=1}^{N'} P_{ii'} = 1$. Let $\widehat{\Pr}(i \leftrightarrow i'|HG,HG') = \widehat{P}_{ii'}$ where $\widehat{P}_{ii'}$ is the $(i,i')$th entry of $\widehat{\mathbf{P}}$. We refer to $\widehat{\Pr}(i \leftrightarrow i'|HG,HG')$ as the matching probability for vertex $i$ and $i'$, and the set of matching probabilities $\{\widehat{\Pr}(i \leftrightarrow i'|HG,HG')|i \in V; i' \in V'\}$ obtained by maximizing (2) reveal how likely it is that each correspondence is correct according to structural similarity between the two hypergraphs $HG$ and $HG'$. This formulation has also been adopted in tensor power iteration for higher order matching [3]. However, the difference between our method and the existing algorithms is that we restrict the solution of (2) to obey the the fundamental axioms of probability, i.e. positiveness and unit total probability mass. This constraint not only provides an alternative probabilistic perspective for hypergraph matching, but also proves convenient for optimization.

Once the set of matching probabilities satisfying (2) are computed, correspondences between vertices drawn from $HG$ and $HG'$ can be established. Matchings with a zero probability are the least likely correspondences, and matchings with nonzero probabilities tend to be those with significant similarity between their structural contexts. Our aim is to seek the subset of possible matchings with nonzero probabilities which satisfy (2) and that are subject to the one-to-one matching constraint.

## 3 High Order Matching as Dominant Cluster Analysis on a Direct Product Hypergraph

In this section we pose the high order relational matching problem formulated in (2) as one of *dominant cluster analysis* on a *direct product hypergraph*. We commence by establishing a direct product hypergraph for the two hypergraphs to be matched. Optimal matching can be achieved by extracting the dominant cluster of vertices from the direct product hypergraph.

### 3.1 Direct Product Hypergraph

The construction of a direct product hypergraph for two $K$-uniform hypergraphs is a generalization of that of the direct product graph [9], which can be used to construct kernels for graph classification. We extend the concept of a direct product graph to

encapsulate high order relations residing in a hypergraph and apply this generalization to hypergraph matching problems. For two $K$-uniform hypergraphs $HG(V,E)$ and $HG'(V',E')$, the direct product $HG_\times$ is a hypergraph with vertex set

$$V_\times = \{(i,i')|i \in V, i' \in V'\}; \tag{3}$$

and edge set

$$E_\times = \{\{(i_1,i'_1)\cdots(i_K,i'_K)\}|\{i_1,\cdots,i_K\} \in E, \{i'_1,\cdots,i'_K\} \in E'\}. \tag{4}$$

The vertex set of the direct product hypergraph $HG_\times$ consists of Cartesian pairs of vertices drawn from $HG$ and $HG'$ separately. Thus the cardinality of the vertex set of $HG_\times$ is $|V_\times| = |V||V'| = NN'$. The direct product hypergraph $HG_\times$ is $K$-uniform, and each $K$-tuple of vertices in $HG_\times$ is encompassed in a hyperedge if and only if the corresponding vertices in $HG$ and $HG'$ are both encompassed by a hyperedge in the relevant hypergraph. Each hyperedge in a direct product hypergraph is weighted by the similarity between the two associated hyperedges from $HG$ and $HG'$.

Furthermore, from our definition of direct product hypergraph, it is clear that the compatibility tensor $\mathcal{C}$ defined in (1) is in fact the tensor $\mathcal{C}_\times$ associated with the direct product hypergraph $HG_\times$ for $HG$ and $HG'$. Every possible matching $i \leftrightarrow i'$ is associated with the vertex $(i,i')$ in $HG_\times$. For simplicity we let $\alpha$ denote a vertex in $HG_\times$ instead of $(i,i')$, and let $\mathbb{D}$ denote the subset of vertices in $HG_\times$ which represent the correct vertex matching for $HG$ and $HG'$. We denote the probability for the vertex $\alpha$ belonging to $\mathbb{D}$ by $\Pr(\alpha \in \mathbb{D}|HG_\times)$. For a direct product hypergraph with $N_\times$ vertices, we establish a $N_\times \times 1$ vector $\mathbf{p}$ with its $\alpha$th element $p_\alpha = \Pr(\alpha \in \mathbb{D}|HG_\times)$. With these ingredients the optimal model satisfying the condition (2) reduces to

$$\widehat{\mathbf{p}} = \underset{\mathbf{p}}{\operatorname{argmax}} \sum_{\alpha_1=1}^{N_\times} \cdots \sum_{\alpha_K=1}^{N_\times} C_{\alpha_1,\cdots,\alpha_K} \prod_{n=1}^{K} p_{\alpha_n} \tag{5}$$

subject to the constraints $\forall\alpha,\ p_\alpha \geq 0$ and $\sum_{\alpha=1}^{N_\times} p_\alpha = 1$. Following the construction of a direct product hypergraph, the objective function (5) is a natural extension of that in [8] from clustering to matching. It is also a reduced version of the objective function of ensembles of affinity relations [6], with no manual threshold on the optimization.

According to (5), zero probability will be assigned to the vertices that do not belong to $\mathbb{D}$. We refer to the probability $\widehat{\Pr}(\alpha \in \mathbb{D}|HG_\times) = \widehat{p}_\alpha$ where $\widehat{p}_\alpha$ is the $\alpha$th element of the vector $\widehat{\mathbf{p}}$ satisfying the optimality condition in (5) as the association probability for the vertex $\alpha$. Therefore, the matching problem can be solved by extracting the cluster of vertices with nonzero association probabilities in the direct product hypergraph.

### 3.2 Dominant Cluster Analysis

In this subsection, we formulate the problem of high order structural matching by applying *dominant cluster analysis* (DCA) to a direct product hypergraph (DPH). A dominant cluster of a hypergraph is the subset of vertices with the greatest average similarity, i.e. average similarity will decrease subject to any vertex deletion from or vertex addition to

the subset. Drawing on the concept of the dominant set in a graph [7] and its game theoretic generalization [8], we can easily perform DPH-DCA by applying the following update until convergence is reached [2]

$$p_{\alpha}^{\mathtt{new}} = \frac{p_{\alpha} \sum_{\alpha_2=1}^{N_{\times}} \cdots \sum_{\alpha_K=1}^{N_{\times}} C_{\alpha,\alpha_2,\cdots,\alpha_K} \prod_{n=2}^{K} p_{\alpha_n}}{\sum_{\beta=1}^{N_{\times}} p_{\beta} \sum_{\beta_2=1}^{N_{\times}} \cdots \sum_{\beta_K=1}^{N_{\times}} C_{\beta,\beta_2,\cdots,\beta_K} \prod_{n=2}^{K} p_{\beta_n}} \tag{6}$$

At convergence the weight $\widehat{p}_{\alpha}$ is equal to the association probability $\widehat{\Pr}(\alpha \in \mathbb{D}|HG_{\times})$, i.e. the probability for the corresponding potential matching $i \leftrightarrow i'$ to be correct.

## 4 Matching with Prior Rejections

The high order structural matching algorithm described in Section 3 is a unsupervised process. The weight of each vertex in the direct product hypergraph can be initialized by using a uniform distribution of probability. However, if two vertices in a hypergraph have the same structural context, i.e. their interchange does not change the hypergraph structure, they can cause ambiguity when matching is attempted. Two alternative state-of-the-art methods, namely probabilistic hypergraph matching [10] and tensor power iteration [3], also suffer from this shortcoming.

However, if prior knowledge about outliers (i.e. hypergraph vertices for which no match exists) is available, we can to a certain extent avoid the ambiguity and improve matching accuracy by using a different weight initialization strategy. We refer to the vertex subset $V_{\times}^{o} \subseteq V_{\times}$ (i.e. possible correspondences) associated with available outliers as prior rejections, and the adopted initialization in the light of prior rejections is as follows

$$w(\alpha) = \begin{cases} 0 & \text{if } \alpha \in V_{\times}^{o}; \\ 1/(N_{\times} - N_{\times}^{o}) & \text{otherwise;} \end{cases} \tag{7}$$

where $N_{\times}^{o}$ is the cardinality of $V_{\times}^{o}$.

The initialization scheme (7) improves the matching accuracy within the DPH-DCA framework because the vertex weight $w(\alpha)$ in the numerator of the update formula (6) plays an important role in maintaining the initial rejection. It enables the prior rejections to maintain a zero weight and does not affect the matching scores for other possible correspondences at each update until converged. The extent to which the matching accuracy can be improved depends on the amount of prior rejections available. The more prior knowledge concerning the outliers that is available, the more accurate the matching that can be obtained. This will be verified in our experimental section.

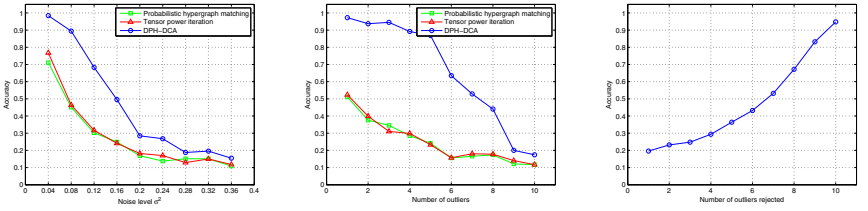
In [6], the authors have described the same initialization step as a disadvantage. On the other hand, we argue that the initialization scheme (7) does not apply to the alternative methods[10][3] even when identified outliers are available. The probabilistic hypergraph matching method [10] initializes a matching score by a fixed value obtained from the marginalization of the compatibility tensor, and thus can not accommodate the prior rejections by using (7). The tensor power iteration method [3], though manually initialized, converges to a fixed matching score for different initializations.

# 5 Experiments

We test our algorithm for high order structural matching on two types of data. Firstly, we test our method on synthetic data to evaluate its robustness to noise and outliers. Secondly, we conduct experiments to match features extracted from images. Prior rejections are considered for both types of data to improve the matching accuracy. We compare our method with two state-of-the-art methods, i.e. probabilistic hypergraph matching (PHM) [10] and tensor power iteration (TPI) [3].

## 5.1 Matching Synthetic Data

We commence with the random generation of a structural prototype with 15 vertices. The distance $d_{ij}$ between each pair of vertices $i$ and $j$ of the prototype is randomly distributed subject to the Gaussian distribution $N(1, 0.5)$. We test our method by establishing correspondences between the prototype structure and a modified structure. The alternative modifications include a) noise addition, b) vertex deletion, c) rescaling and d) rotation. Since neither the probabilistic hypergraph matching method nor the tensor power iteration method relies upon a specific initialization, we test our DPH-DCA matching method without prior rejections to make a fair comparison with these two alternative methods. To test the performance of different methods for hypergraph matching we re-scaled the distance between each of vertex pairs by a random factor and rotate the structure by a random angle. In this case, the pairwise relationships no longer holds for the matching task. We use the sum of polar sines presented in [5] as a high order similarity measure for point tuples. We measure the similarity of every 3-tuple within the vertex set and thus establish a weighted 3-uniform hypergraph for the structure. The compatibility tensor $\mathcal{C}$ for two structures is computed according to (1) with $\sigma_1 = 0.1$. Figure 1(a) illustrates the results of the matching accuracy as a function of noise level. It is clear that our DPH-DCA framework outperforms the two alternative methods at each noise level. To take the investigation one step further, we study the performance of our method for matching structures of different vertex cardinality. To this end, we extract a substructure from a prototype and slightly perturb the distance between each vertex pair by adding random noise normally distributed according to $N(0, 0.04)$ . The cardinality of the vertex set of the substructure varies from $14$ down to $5$. Vertices not in the substructure are outliers for the matching process. For each vertex cardinality of a substructure, 100 trials are performed. Figure 1(b) illustrates the matching accuracy as a function of outlier number for the three methods. It is clear that our DPH-DCA framework outperforms the two alternative methods at each number of outliers. We have also evaluated the matching accuracy of our DPH-DCA framework at different levels of available prior rejection. To this end, we have extracted a $5$-vertex substructure from a prototype and slightly perturb the distance between each vertex pair by adding random noise normally distributed according to $N(0, 0.04)$. We involve prior rejections by rejecting the matchings associated with a varying number of outliers. Figure 1(c) illustrates the matching accuracy as a function of the number of rejected outliers. It is clear that the matching accuracy grows monotonically as the number of rejected outliers increases.

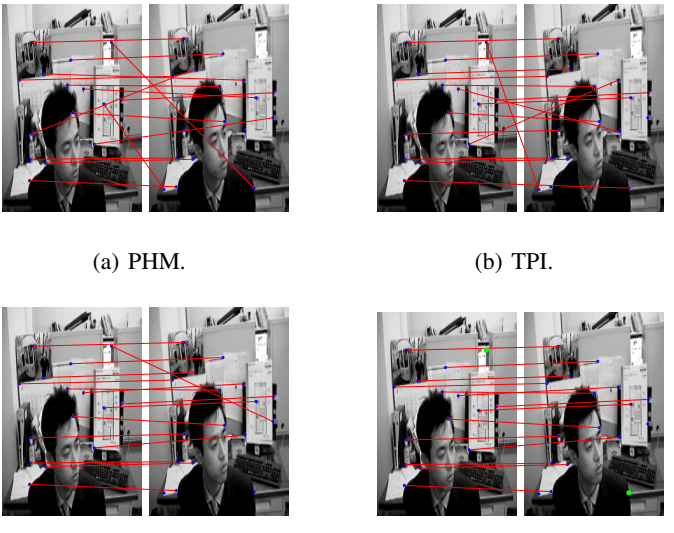



(a) Noise. (b) Outliers. (c) Prior rejections.

**Fig. 1.** Matching performance

### 5.2 Image Correspondences

To visualize the matching for real world images we test the alternative methods on frames of video[1]. We use the Harris detector to extract corner points from the first and 30th frames. We use the sum of polar sines presented in [5] to measure the similarity of every 3-tuple within the corner points and thus establish a weighted 3-uniform hypergraph for each image. Figure 2 illustrates the matching performances for alternative methods. The matching results for the two comparison methods are visualized in Figures 2(a) and 2(b), where 11 correct correspondences and 4 incorrect ones are



(a) PHM. (b) TPI.

(c) DPH-DCA. (d) DPH-DCA with two prior rejections.

**Fig. 2.** Image correspondences

[1] http://www.suri.it.okayama-u.ac.jp/e-program-separate.html

obtained by using the tensor power iteration, and 12 correct correspondences and 3 incorrect ones by the probabilistic hypergraph matching. For DCA without prior rejections (visualized in Figure 2(c)), we obtain 14 correct correspondences and 1 incorrect ones. Figure 2(d) visualizes the matching result by rejecting two outliers (green marked). It is clear that the false matching is eliminated by incorporating the proper prior rejections.

## 6 Conclusion and Future Work

We have presented a novel approach to high order structural matching. We have transformed the matching problem to that of extracting the dominant cluster from the direct product hypergraph for two feature sets with high order relationships. Prior knowledge about outliers can be easily involved in our framework by initializing the matchings associated with the outliers by a zero weight. Experiments have shown that our method outperforms the state-of-the-art methods.

## References

1. Agarwal, S., Lim, J., Zelnik-Manor, L., Perona, P., Kriegman, D., Belongie, S.: Beyond pairwise clustering. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (2005)
2. Baum, L.E., Eagon, J.A.: An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology. Bulletin of the American Mathematical Society 73, 360–363 (1967)
3. Duchenne, O., Bach, F.R., Kweon, I.S., Ponce, J.: A tensor-based algorithm for high-order graph matching. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (2009)
4. Leordeanu, M., Hebert, M.: A spectral technique for correspondence problems using pairwise constraints. In: Proceedings of IEEE International Conference on Computer Vision (2005)
5. Lerman, G., Whitehouse, J.T.: On $d$-dimensional $d$-semimetrics and simplex-type inequalities for high-dimensional sine functions. Journal of Approximation Theory 156(1), 52–81 (2009)
6. Liu, H., Latecki, L.J., Yan, S.: Robust clustering as ensembles of affinity relations. In: Proceedings of Advances in Neural Information Processing Systems (2010)
7. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(1), 167–172 (2007)
8. Rota-Bulo, S., Pelillo, M.: A game-theoretic approach to hypergraph clustering. In: Proceedings of Advances in Neural Information Processing Systems (2009)
9. Vishwanathan, S.V.N., Borgwardt, K.M., Kondor, I.R., Schraudolph, N.N.: Graph kernels. Journal of Machine Learning Research 11, 1201–1242 (2010)
10. Zass, R., Shashua, A.: Probabilistic graph and hypergraph matching. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 234–778 (2008)

# A Game-Theoretic Approach to Robust Selection of Multi-View Point Correspondence

Emanuele Rodolà, Andrea Albarelli, and Andrea Torsello
*Dipartimento di Informatica - Università Ca' Foscari*
*Via Torino, 155 - 30172 Venice Italy*

## Abstract

*In this paper we introduce a robust matching technique that allows very accurate selection of corresponding feature points from multiple views. Robustness is achieved by enforcing global geometric consistency at an early stage of the matching process, without the need of subsequent verification through reprojection. The global consistency is reduced to a pairwise compatibility making use of the size and orientation information provided by common feature descriptors, thus projecting what is a high-order compatibility problem into a pairwise setting. Then a game-theoretic approach is used to select a maximally consistent set of candidate matches, where highly compatible matches are enforced while incompatible correspondences are driven to extinction.*

## 1 Introduction

The selection of 3D point correspondences from their 2D projections is arguably one of the most important steps in image based multi-view reconstruction, as errors in the initial correspondences can lead to suboptimal parameter estimation. The selection of corresponding points is usually carried out by means of interest point detectors and feature descriptors. Once salient and well-identifiable points are found on each image, correspondences between the features in the various views must be extracted and fed to the bundle adjustment algorithm. To this end, each point is associated a descriptor vector with tens to hundreds of dimensions, which usually include a scale and a rotation value. Arguably the most famous of such descriptors is the Scale-invariant feature transform (SIFT) [3].Features are designed so that similar image regions subject to similarity transformation exhibit descriptor vectors with small Euclidean distance. This property is used to match each point with a candidate with similar descriptor. However, if the descriptor is not distinctive enough this approach is prone to select many outliers since the approach only exploits local information. This limitation conflicts with the richness of information that is embedded in the scene structure. For instance, under the assumption of rigidity and small camera motion, features that are close in one view are expected to be close in the other one as well. In addition, if a pair of feature exhibit a certain difference of angles or ratio of scales, this relation should be maintained among their respective matches. This prior information about scene structure can be accounted for by using a feature tracker [4, 6] to extract correspondences, but this requires that the view positions be not far apart. Further, in the presence of strong parallax, a locally uniform 3D motion does not result in a locally uniform 2D motion, and for these reason the geometric constraints can be enforced only locally. A common heuristic for the enforcement of global structure is to eliminate points that exhibit a large reprojection error after a first round of Bundle Adjustment [7]. Unfortunately this post-filtering technique requires good initial estimates to begin with.

In this paper we introduce a robust matching technique that allows to operate a very accurate inlier selection at an early stage of the process and without any need to rely on 3D reprojections. The approach selects feasible matches by enforcing global geometric consistency. Specifically, it enforces that all pairs of correspondences between 2D views are consistent with a common 3D rigid transformation. This constraint is in general underspecified, as a whole manifold of pairs of correspondences are consistent with a rigid 3D transformation, as it is well known that at least seven matching points are needed to solve the epipolar equation [2]. However, by accumulating mutual support through a large set of mutually compatible correspondences, one can expect to reduce the ambiguity to a single 3D rigid transformation. In the proposed approach high order consistency constraint are reduced to a second order compatibility where sets of 2D point correspondences that can be interpreted as projections of rigidly-transformed 3D points all have high mutual support. Then, following [8, 1], a game-theoretic approach is used to select a set of candidate matches, enforcing highly compatible matches while driving to extinction incompatible correspondences.

## 2 Pairwise Geometric Consistency

There are two fundamental hypotheses underlying the reduction to second order of the high-order 3D geometric consistency. First, We assume that the views have the same set of camera parameters, second, we assume that the feature descriptor provides scale and orientation information and that this is related to actual local information in the 3D objects present in the scene. The effect of the first assumption is that the geometric consistency is reduced to a rigidity constraint that can be cast as a conservation along views of the distances between the unknown 3D position of the feature points, while the effect of the second assumption is that we can recover the missing depth information as a variation in scale between two views of the same point is inversely proportional to variation in projected size of the local patch around the 3D point and, thus, to the projected size of the feature descriptor.

More formally, assume that we have two points $p_1$ and $p_2$, which in one view have coordinates $(u_1^1, v_1^1)$ and $(u_2^1, v_2^1)$ respectively, while in a second image they have coordinates $(u_1^2, v_1^2)$ and $(u_2^2, v_2^2)$. These points, in the coordinate system of the first camera, have 3D coordinates $z_1^1(u_1^1, v_1^1, f)$ and $z_2^1(u_2^1, v_2^1, f)$ respectively, while in the reference frame of the second camera they have coordinates $z_1^2(u_1^2, v_1^2, f)$ and $z_2^1(u_2^2, v_2^2, f)$. Up to a change in units, these coordinates can be rewritten as $p_1^1 = \frac{1}{s_1^1}\begin{pmatrix} u_1^1 \\ v_1^1 \\ 1 \end{pmatrix}$, $p_2^1 = \frac{a}{s_2^1}\begin{pmatrix} u_2^1 \\ v_2^1 \\ 1 \end{pmatrix}$, $p_1^2 = \frac{1}{s_1^2}\begin{pmatrix} u_1^2 \\ v_1^2 \\ 1 \end{pmatrix}$, and $p_2^2 = \frac{a}{s_2^2}\begin{pmatrix} u_2^2 \\ v_2^2 \\ 1 \end{pmatrix}$, where $a$ is the ratio between the actual scales of the local 3D patches around points $p_1$ and $p_2$, whose projections on the two views give the perceived scales $s_1^1$ and $s_1^2$ for point $p_1$ and $s_2^1$ and $s_2^2$ for point $p_2$.

The assumption that both scale and orientation are linked with actual properties of the local patch around each 3D point is equivalent to having 2 points for each feature correspondence: the actual location of the feature, plus a virtual point located along the axis of orientation of the feature at a distance proportional to the actual scale scale of the patch. These pair of 3D points must move rigidly going from the coordinate system of one camera to the other, so that given any two sets of correspondences with 3D points $p_1$ and $p_2$ and their corresponding virtual points $q_1$ and $q_2$, the distances between these four points must be preserved in the reference frames of every view (see Fig. 1).

Under a frontal-planar assumption for each local patch, or, less stringently, under small variation in viewpoints, we can assign 3D coordinates to the virtual

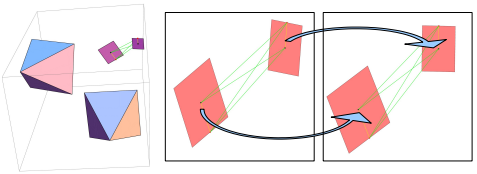


**Figure 1. Scale and orientation offer depth information and a second virtual point. the conservation of the distances in green enforce consistency with a 3D rigid transformation.**

points in the reference frames of the two images:

$$q_1^1 = p_1^1 + \begin{pmatrix} \cos\theta_1^1 \\ \sin\theta_1^1 \\ 0 \end{pmatrix} \quad q_2^1 = p_2^1 + a\begin{pmatrix} \cos\theta_2^1 \\ \sin\theta_2^1 \\ 0 \end{pmatrix}$$
$$q_1^2 = p_1^2 + \begin{pmatrix} \cos\theta_1^2 \\ \sin\theta_1^2 \\ 0 \end{pmatrix} \quad q_2^2 = p_2^2 + a\begin{pmatrix} \cos\theta_2^2 \\ \sin\theta_2^2 \\ 0 \end{pmatrix},$$

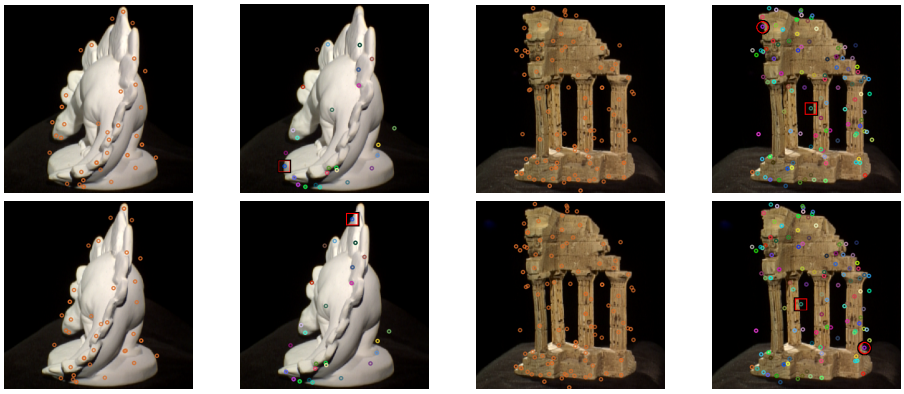
where $\theta_i^j$ is the perceived orientation of feature $i$ in image $j$. At this point, given two sets of correspondences between points in two images, namely the correspondence $m_1$ between a feature point in the first image with coordinates, scale and orientation $(u_1^1, v_1^1, s_1^1, \theta_1^1)$ with the feature point in the second image $(u_1^2, v_1^2, s_1^2, \theta_1^2)$, and the correspondence $m_2$ between the points $(u_2^1, v_2^1, s_2^1, \theta_2^1)$ and $(u_2^2, v_2^2, s_2^2, \theta_2^2)$ in the first and second image respectively, we can compute a distance from the manifold of feature descriptors compatible with a single 3D rigid transformation as

$$d(m_1, m_2, a) = (||p_1^1 - p_2^1||^2 - ||p_1^2 - p_2^2||^2)^2 + (||p_1^1 - q_2^1||^2 - ||p_1^2 - q_2^2||^2)^2 + (||q_1^1 - p_2^1||^2 - ||q_1^2 - p_2^2||^2)^2 + (||q_1^1 - q_2^1||^2 - ||q_1^2 - q_2^2||^2)^2 .$$

From this we define the compatibility between correspondences as $C(m_1, m_2) = \max_a e^{-\gamma d(m_1, m_2, a)}$, where $a$ is maximized over a reasonable range of ratio of scales of local 3D patches. In our experiments $a$ was optimized in the interval $[0.5; 2]$.

## 3 Game-Theoretic Feature Matching

We model the matching process in a game-theoretic framework [1], where two players extracted from a large population select a pair of matching points from two images. The player then receives a payoff from the other players proportional to how compatible his match is with respect to the other player's choice. Clearly, it is in each player's interest to pick matches that are compatible with those the other players are likely to choose. It is supposed that some selection process operates over time on the distribution of behaviors favoring players



Game-Theoretic Bundler Keymatcher Game-Theoretic Bundler Keymatcher

Dino sequence Temple sequence

| | Dino sequence | | Temple sequence | |
|---|---|---|---|---|
| | Game-Theoretic | Bundler Keymatcher | Game-Theoretic | Bundler Keymatcher |
| Matches | $262.5 \pm 61.4$ | $172.4 \pm 79.5$ | $535.7 \pm 38.7$ | $349.3 \pm 36.2$ |
| $\Delta\alpha$ | $0.0668 \pm 0.0777$ | $0.0767 \pm 0.1172$ | $0.1326 \pm 0.0399$ | $0.1414 \pm 0.0215$ |
| $\Delta\gamma$ | $0.4393 \pm 0.4963$ | $0.6912 \pm 0.8793$ | $0.0809 \pm 0.0144$ | $0.0850 \pm 0.0065$ |

**Figure 2. Results obtained with the Dino and Temple data sets (images best viewed in color).**

that receive larger payoffs and driving all inconsistent hypotheses to extinction, finally settling for an equilibrium where the pool of matches from which the players are still actively selecting their associations forms a cohesive set with high mutual support. More formally, let $O = \{1, \cdots, n\}$ be the set of available strategies (*pure strategies* in the language of game theory) and $C = (c_{ij})$ be a matrix specifying the payoff that an individual playing strategy $i$ receives against someone playing strategy $j$. A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, \ldots, x_n)^T$ over the available strategies $O$, thus lying in the n-dimensional standard simplex $\Delta^n = \{\mathbf{x} \in \mathbb{R}^n : \forall i \in 1 \ldots n \; x_i \geq 0, \; \sum_{i=1}^{n} x_i = 1\}$. The expected payoff received by a player choosing element $i$ when playing against a player adopting a mixed strategy $\mathbf{x}$ is $(C\mathbf{x})_i = \sum_j c_{ij} x_j$, hence the expected payoff received by adopting the mixed strategy $\mathbf{y}$ against $\mathbf{x}$ is $\mathbf{y}^T C \mathbf{x}$. A strategy $\mathbf{x}$ is said to be a *Nash equilibrium* if it is the best reply to itself, i.e., $\forall \mathbf{y} \in \Delta$, $\mathbf{x}^T C \mathbf{x} \geq \mathbf{y}^T C \mathbf{x}$. A strategy $\mathbf{x}$ is said to be an *evolutionary stable strategy* (ESS) if it is a Nash equilibrium and $\forall \mathbf{y} \in \Delta \quad \mathbf{x}^T C \mathbf{x} = \mathbf{y}^T C \mathbf{x} \Rightarrow \mathbf{x}^T C \mathbf{y} > \mathbf{y}^T C \mathbf{y}$. This condition guarantees that any deviation from the stable strategies does not pay. The search for a stable state is performed by simulating the evolution of a natural selection process. Under very loose conditions, any dynamics that respect the payoffs is guaranteed to converge to Nash equilibria and (hopefully) to ESS's; for this reason, the choice of an actual selection process is not crucial and can be driven mostly by considerations of efficiency and simplicity. We chose to use the replicator dynamics, a well-known formalization of the selection process governed by the recurrence $\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^t \frac{(C\mathbf{x}^t)_i}{\mathbf{x}^{t\,T} C \mathbf{x}^t}$, where $\mathbf{x}_i^t$ is the proportion of the population that plays the $i$-th strategy at time $t$. Once the population has reached a local maximum, all the non-extincted pure strategies can be considered selected by the game.

## 4 Experimental Results

To evaluate the performance of our proposal, we compared the results with those obtained with the keymatcher included in the structure-from-motion suite Bundler [7]. For the first set of experiments we selected pair of adjacent views from the "DinoRing" and "TempleRing" sequences from the Middlebury Multi-View Stereo dataset [5]; for these models, camera parameters are provided and used as a ground-truth. For all the sets of experiments we evaluated the differences in radians between the (calibrated) ground-truth and respectively the estimated rotation angle ($\Delta\alpha$) and rotation axis ($\Delta\gamma$). The "Dino" model is a difficult case in general, as it provides very few features; the upper part of Fig. 2 shows the correspondences produced by our method (left column) in comparison with the other matcher (right column). The "Temple" model richer in features and for visualization purposes we only show a subset of the detected matches for both the techniques. The Bundler keymatcher, while still achieving good results, provides some mismatches in both cases. This

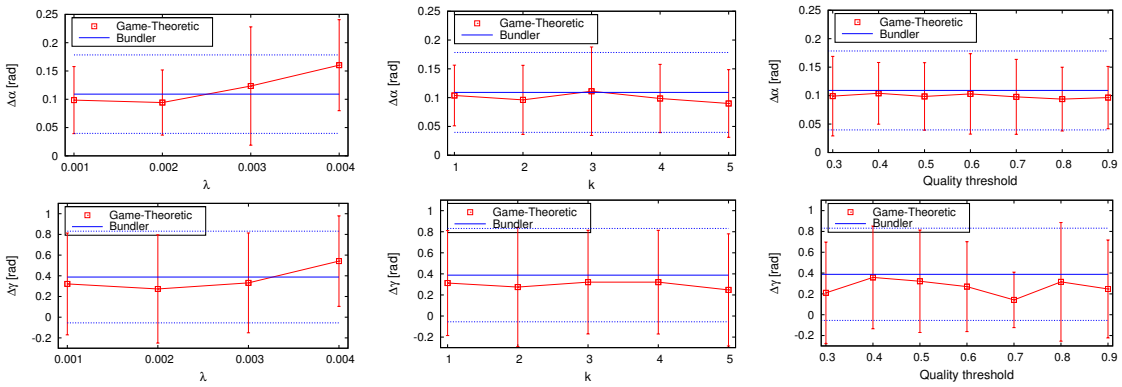



**Figure 3. Analysis of the performance of the approach with respect to variation of the parameters of the algorithm.**

can be explained by the fact that the symmetric parts of the object, e.g. the pillars in the temple model, result in very similar features that are hard to disambiguate by a purely local matcher. Our method, on the other hand, by enforcing global 3D consistency, can effectively disambiguate the matches. Looking at the results we can see that our approach extracts around 50% more correspondence, providing a slight increase in precision and reduction in variance of the estimates. Note that selected measures evaluate the quality of the underlying least square estimates of the motion parameters after a reprojection step, thus small variations are expected.

Next, we analyzed the impact of the algorithm parameters over the quality of the results obtained. To this end we investigated three parameters: the similarity decay $\lambda$, the number $k$ of candidate mates per features, and the *quality threshold*, that is the minimum support for a correspondence to be considered non-extinct, divided by the maximum support in the population. Figure 3 reports the results of these experiments. Overall, these experiments suggest that those parameters have little influence over the quality of the result. However the Game-Theoretic approach achieves better average results and smaller standard deviations for almost all reasonable values of the parameters.

## 5 Conclusions

In this paper we introduced a robust matching technique for feature points from multiple views. Robustness is achieved by enforcing global geometric consistency in a pairwise setting. This is achieved by using the scale and orientation information offered by SIFT features and projecting what is left of a high-order compatibility problem into a pairwise compatibility measure, by enforcing the conservation of distances between the unknown 3D positions of the points. Finally, a game-theoretic approach is used to select a maximally consistent set of candidate matches, where highly compatible matches are enforced while incompatible correspondences are driven to extinction. Experimental comparisons with a widely used technique show the ability of our approach to obtain a more accurate estimation of the scene parameters.

### Acknowledgments

We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.

## References

[1] A. Albarelli, S. Rota Bulò, A. Torsello, and M. Pelillo. Matching as a non-cooperative game. In *ICCV 2009*, 2009.

[2] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.

[3] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003.

[4] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[5] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR '06*, pages 519–528, 2006.

[6] J. Shi and C. Tomasi. Good features to track. In *CVPR*, pages 593–600, 1994.

[7] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *Int. J. Comput. Vision*, 80(2):189–210, 2008.

[8] A. Torsello, S. Rota Bulò, and M. Pelillo. Grouping with asymmetric affinities: A game-theoretic perspective. In *CVPR '06*, pages 292–299, 2006.

# A Game-Theoretic Approach to the Enforcement of Global Consistency in Multi-View Feature Matching

Emanuele Rodolà, Andrea Albarelli, and Andrea Torsello
Dipartimento di Informatica – Università Ca' Foscari – Venice, Italy

**Abstract.** In this paper we introduce a robust matching technique that allows to operate a very accurate selection of corresponding feature points from multiple views. Robustness is achieved by enforcing global geometric consistency at an early stage of the matching process, without the need of ex-post verification through reprojection. Two forms of global consistency are proposed, but in both cases they are reduced to pairwise compatibilities making use of the size and orientation information provided by common feature descriptors. Then a game-theoretic approach is used to select a maximally consistent set of candidate matches, where highly compatible matches are enforced while incompatible correspondences are driven to extinction. The effectiveness of the approach in estimating camera parameters for bundle adjustment is assessed and compared with state-of-the-art techniques.

## 1 Introduction

The selection of 3D point correspondences from their 2D projections is arguably one of the most important steps in image based multi-view reconstruction, as errors in the initial correspondences can lead to sub-optimal parameter estimation. The selection of corresponding points is usually carried out by means of interest point detectors and feature descriptors. Salient points are localized with subpixel accuracy by general detectors, such as Harris Operator [2] and Difference of Gaussians [6], or by using techniques that are able to locate affine invariant regions, such as Maximally stable extremal regions (MSER) [7] and Hessian-Affine [8]. This latter affine invariance property is desirable since the change in appearance of a scene region after a small camera motion can be locally approximated with an affine transformation. Once salient and well-identifiable points are found on each image, correspondences between the features in the various views must be extracted and fed to the bundle adjustment algorithm. To this end, each point is associated a descriptor vector with tens to hundreds of dimensions, which usually include a scale and a rotation value. Arguably the most famous of such descriptors are the Scale-invariant feature transform (SIFT) [4], the Speeded Up Robust Features (SURF) [3], and the Gradient Location and Orientation Histogram (GLOH) [9], and more recently the Local Energy based Shape Histogram (LESH) [10]. Features are designed so that similar image regions subject to similarity transformation exhibit descriptor vectors with small Euclidean distance. This property is used to match each point with a candidate with similar descriptor. However, if the descriptor is not distinctive enough this

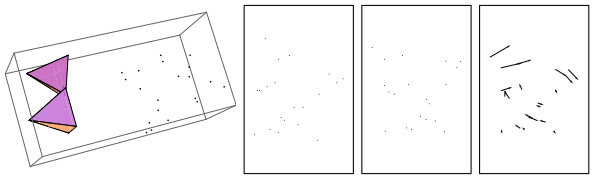


**Fig. 1.** Locally uniform 3D motion does not result in a locally uniform 2D motion. From left to right: 3D scene, left and right views, and motion estimation.

approach is prone to select many outliers since it only exploits local information. This limitation conflicts with the richness of information that is embedded in the scene structure. For instance, under the assumption of rigidity and small camera motion, features that are close in one view are expected to be close in the other one as well. In addition, if a pair of features exhibit a certain difference of angles or ratio of scales, this relation should be maintained among their respective matches. This prior information about scene structure can be accounted for by using a feature tracker [5, 12] to extract correspondences, but this requires that the view positions be not far apart. Further, in the presence of strong parallax, a locally uniform 3D motion does not result in a locally uniform 2D motion, and for these reasons the geometric constraints can be enforced only locally (see Fig. 1 for an example). A common heuristic for the enforcement of global structure is to eliminate points that exhibit a large reprojection error after a first round of Bundle Adjustment [13]. Unfortunately this post-filtering technique requires good initial estimates to begin with.

In this paper we introduce a robust matching technique that allows to operate a very accurate inlier selection at an early stage of the process and without any need to rely on 3D reprojections. The approach selects feasible matches by enforcing global geometric consistency. Two geometric consistency models are presented. The first enforces that all pairs of correspondences between 2D views are consistent with a common 3D rigid transformation. Here, as is common in similar point-matching approaches, we assume that we have reasonable guesses for the intrinsic camera parameters and reduce the problem space to the search of a 3D rigid transformation from one image space to the other. This condition is in general underspecified, as a whole manifold of pairs of correspondences are consistent with a rigid 3D transformation. However, by accumulating mutual support through a large set of mutually compatible correspondences one can expect to reduce the ambiguity to a single 3D rigid transformation. In the proposed approach, high order consistency constraints are reduced to a second order compatibility where sets of 2D point correspondences that can be interpreted as projections of rigidly-transformed 3D points all have high mutual support. The reduction is obtained by making use of the scale and orientation information linked with each feature point in the SIFT descriptor [4] and a further reprojection that can be considered a continuous form of hypergraph clique expansion [15].

The second geometric consistency constraint assumes a weak perspective camera and matches together points whose maps are compatible with a common

affine transformation. This allows us to extract small coherent clusters of points all laying at similar depths. The locally affine hypothesis could seem to be an unsound assumption for general camera motion, and in effect cannot account for point inversion due to parallax, but in the experimental section we will show that it holds well with the typical disparity found in standard data sets. Further, it should be noted that with large camera motion most, if not all, commonly used feature detectors fail, thus any inlier selection attempt becomes meaningless.

Once the geometric consistency contraints are specified, we can use them to drive the matching process. Following [14, 1], we model the matching process in a game-theoretic framework, where two players extracted from a large population select a pair of matching points from two images. The player then receives a payoff from the other players proportional to how compatible his match is with respect to the other player's choice, where the compatibility derives from some utility function that rewards pair of matches that are consistent. Clearly, it is in each player's interest to pick matches that are compatible with those the other players are likely to choose. In general, as the game is repeated, players will adapt their behavior to prefer matchings that yield larger payoffs, driving all inconsistent hypotheses to extinction, and settling for an equilibrium where the pool of matches from which the players are still actively selecting their associations forms a cohesive set with high mutual support. Within this formulation, the solutions of the matching problem correspond to evolutionary stable states (ESS's), a robust population-based generalization of the notion of a Nash equilibrium. In a sense, this matching process can be seen as a contextual voting system, where each time the game is repeated the previous selections of the other players affect the future vote of each player in an attempt to reach consensus. This way, the evolving context brings global information into the selection process.

## 2 Pairwise Geometric Consistency

In what follows we will describe the two geometric constraints that will be used to drive the matching process. The first approach tries to impose that the points be consistent with a common 3D rigid transformation.

There are two fundamental hypotheses underlying the reduction to second order of this high-order 3D geometric consistency. First, we assume that the views have the same set of camera parameters, that we have reasonable guesses for the intrinsic parameters, and we can ignore lens distortion. Thus, the geometric consistency is reduced to the compatibility of the projected points with a single 3D rigid transformation related to the relative positions of the cameras. Second, we assume that the feature descriptor provides scale and orientation information and that this is related to actual local information in the 3D objects present in the scene. The effect of the first assumption is that the geometric consistency is reduced to a rigidity constraint that can be cast as a conservation along views of the distances between the unknown 3D position of the feature points, while the effect of the second assumption is that we can recover the missing depth information as a variation in scale between two views of the same point and that this variation is inversely proportional to variation in projected

size of the local patch around the 3D point and, thus, to the projected size of the feature descriptor. More formally, assume that we have two points $p_1$ and $p_2$, which in one view have coordinates $(u_1^1, v_1^1)$ and $(u_2^1, v_2^1)$ respectively, while in a second image they have coordinates $(u_1^2, v_1^2)$ and $(u_2^2, v_2^2)$. These points, in the coordinate system of the first camera, have 3D coordinates $z_1^1(u_1^1, v_1^1, f)$ and $z_2^1(u_2^1, v_2^1, f)$ respectively, while in the reference frame of the second camera they have coordinates $z_1^2(u_1^2, v_1^2, f)$ and $z_2^2(u_2^2, v_2^2, f)$. Up to a change in units, these coordinates can be re-written as

$$p_1^1 = \frac{1}{s_1^1}\begin{pmatrix} u_1^1 \\ v_1^1 \\ f \end{pmatrix},\ p_2^1 = \frac{a}{s_2^1}\begin{pmatrix} u_2^1 \\ v_2^1 \\ f \end{pmatrix},\ p_1^2 = \frac{1}{s_1^2}\begin{pmatrix} u_1^2 \\ v_1^2 \\ f \end{pmatrix},\ p_2^2 = \frac{a}{s_2^2}\begin{pmatrix} u_2^2 \\ v_2^2 \\ f \end{pmatrix},$$

where $f$ is the focal lenght and $a$ is the ratio between the actual scales of the local 3D patches around points $p_1$ and $p_2$, whose projections on the two views give the perceived scales $s_1^1$ and $s_1^2$ for point $p_1$ and $s_2^1$ and $s_2^2$ for point $p_2$.

The assumption that both scale and orientation are linked with actual properties of the local patch around each 3D point is equivalent to having 2 points for each feature correspondence: the actual location of the feature, plus a virtual point located along the axis of orientation of the feature at a distance proportional to the actual scale of the patch. These pairs of 3D points must move rigidly going from the coordinate system of one camera to the other, so that given any two sets of correspondences with 3D points $p_1$ and $p_2$ and their corresponding virtual points $q_1$ and $q_2$, the distances between these four points must be preserved in the reference frames of every view (see Fig. 2).

Under a frontal-planar assumption for each local patch, or, less stringently, under small variation in viewpoints, we can assign 3D coordinates to the virtual points in the reference frames of the two images:

$$\begin{aligned} q_1^1 &= p_1^1 + \begin{pmatrix} \cos\theta_1^1 \\ \sin\theta_1^1 \\ 0 \end{pmatrix} & q_2^1 &= p_2^1 + a\begin{pmatrix} \cos\theta_2^1 \\ \sin\theta_2^1 \\ 0 \end{pmatrix} \\ q_1^2 &= p_1^2 + \begin{pmatrix} \cos\theta_1^2 \\ \sin\theta_1^2 \\ 0 \end{pmatrix} & q_2^2 &= p_2^2 + a\begin{pmatrix} \cos\theta_2^2 \\ \sin\theta_2^2 \\ 0 \end{pmatrix}, \end{aligned}$$

where $\theta_i^j$ is the perceived orientation of feature $i$ in image $j$. At this point, given two sets of correspondences between points in two images, namely the correspondence $m_1$ between a feature point in the first image with coordinates, scale and orientation $(u_1^1, v_1^1, s_1^1, \theta_1^1)$ with the feature point in the second image $(u_1^2, v_1^2, s_1^2, \theta_1^2)$, and the correspondence $m_2$ between the points $(u_2^1, v_2^1, s_2^1, \theta_2^1)$ and $(u_2^2, v_2^2, s_2^2, \theta_2^2)$ in the first and second image respectively, we can compute a distance from the manifold of feature descriptors compatible with a single 3D rigid transformation as

$$\begin{aligned} d(m_1, m_2, a) = (||p_1^1 - p_2^1||^2 - ||p_1^2 - p_2^2||^2)^2 + (||p_1^1 - q_2^1||^2 - ||p_1^2 - q_2^2||^2)^2 + \\ (||q_1^1 - p_2^1||^2 - ||q_1^2 - p_2^2||^2)^2 + (||q_1^1 - q_2^1||^2 - ||q_1^2 - q_2^2||^2)^2 . \end{aligned}$$

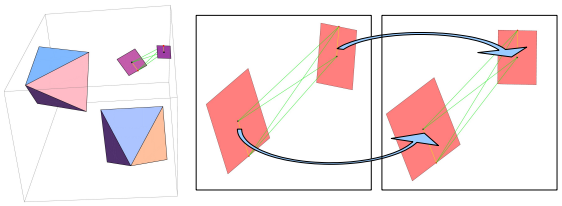


**Fig. 2.** Scale and orientation offer depth information and a second virtual point. the conservation of the distances in green enforces consistency with a 3D rigid transformation.

From this we define the compatibility between correspondences as $C(m_1, m_2) = \max_a e^{-\gamma d(m_1, m_2, a)}$, where $a$ is maximized over a reasonable range of ratio of scales of local 3D patches. In our experiments $a$ was optimized in the interval $[0.5; 2]$.

The second geometric consistency constraint assumes a weak perspective camera and matches together points whose maps are compatible with a common affine transformation. Specifically, we are able to associate to each matching strategy $(a_1, a_2)$ one and only one similarity transformation, that we call $T(a_1, a_2)$. When this transformation is applied to $a_1$ it produces the point $a_2$, but when applied to the source point $b_1$ of the matching strategy $(b_1, b_2)$ it does not need to produce $b_2$. In fact it will produce $b_2$ if and only if $T(a_1, a_2) = T(b_1, b_2)$, otherwise it will give a point $b'_2$ that is as near to $b_2$ as the transformation $T(a_1, a_2)$ is similar $T(b_1, b_2)$. Given two matching strategies $(a_1, a_2)$ and $(b_1, b_2)$ and their respective associated similarities $T(a_1, a_2)$ and $T(b_1, b_2)$, we calculate their reciprocal reprojected points as:

$$a'_2 = T(b_1, b_2)a_1$$
$$b'_2 = T(a_1, a_2)b_1$$

That is the virtual points obtained by applying to each source point the similarity transformation associated to the other match (see Fig 3). Given virtual points $a'_2$ and $b'_2$ we are finally able to calculate the payoff between $(a_1, a_2)$ and $(b_1, b_2)$ as:

$$\Pi((a_1, a_2), (b_1, b_2)) = e^{-\lambda \max(||a_2 - a'_2||, ||b_2 - b'_2||)} \quad (1)$$

Where $\lambda$ is a selectivity parameter that allows to operate a more or less strict inlier selection. If $\lambda$ is small, then the payoff function (and thus the matching) is more tolerant, otherwise the evolutionary process becomes more selective as $\lambda$ grows.

The rationale of the payoff function proposed in equation 1 is that, while by changing point of view the similarity relationship between features is not mantained (as the object is not planar and the transformation is projective), we can expect the transformation to be a similarity at least "locally". This means that we aim to extract clusters of feature matches that belong to the same region of the object and that tend to lie in the same level of depth.

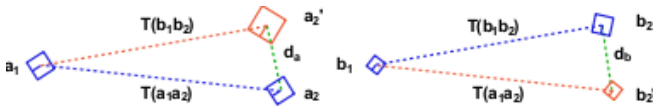



**Fig. 3.** The payoff between two matching strategies is inversely proportional to the maximum reprojection error obtained by applying the affine transformation estimated by a match to the other.

Each matching process selects a group of matching strategies that are coherent with respect to a local similarity transformation. This means that if we want to cover a large portion of the subject we need to iterate many times and prune the previously selected matches at each new start. Obviously, after all the depth levels have been swept, small and not significative residual groups start to emerge from the evolution. To avoid the selection of this spurious matches we fixed a minimum cardinality for each valid group.

## 3 Game-Theoretic Feature Matching

We model the matching process in a game-theoretic framework [1], where two players extracted from a large population select a pair of matching points from two images. The player then receives a payoff from the other players proportional to how compatible his match is with respect to the other player's choice. Clearly, it is in each player's interest to pick matches that are compatible with those the other players are likely to choose. It is supposed that some selection process operates over time on the distribution of behaviors favoring players that receive larger payoffs and driving all inconsistent hypotheses to extinction, finally settling for an equilibrium where the pool of matches from which the players are still actively selecting their associations forms a cohesive set with high mutual support. More formally, let $O = \{1, \cdots, n\}$ be the set of available strategies (*pure strategies* in the language of game theory) and $C = (c_{ij})$ be a matrix specifying the payoff that an individual playing strategy $i$ receives against someone playing strategy $j$. A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, \ldots, x_n)^T$ over the available strategies $O$, thus lying in the n-dimensional standard simplex $\Delta^n = \{\mathbf{x} \in \mathbb{R}^n : \forall i \in 1 \ldots n \; x_i \geq 0, \; \sum_{i=1}^{n} x_i = 1\}$. The expected payoff received by a player choosing element $i$ when playing against a player adopting a mixed strategy $\mathbf{x}$ is $(C\mathbf{x})_i = \sum_j c_{ij} x_j$, hence the expected payoff received by adopting the mixed strategy $\mathbf{y}$ against $\mathbf{x}$ is $\mathbf{y}^T C\mathbf{x}$. A strategy $\mathbf{x}$ is said to be a *Nash equilibrium* if it is the best reply to itself, i.e., $\forall \mathbf{y} \in \Delta, \; \mathbf{x}^T C\mathbf{x} \geq \mathbf{y}^T C\mathbf{x}$. A strategy $\mathbf{x}$ is said to be an *evolutionary stable strategy* (ESS) if it is a Nash equilibrium and $\forall \mathbf{y} \in \Delta \quad \mathbf{x}^T C\mathbf{x} = \mathbf{y}^T C\mathbf{x} \Rightarrow \mathbf{x}^T C\mathbf{y} > \mathbf{y}^T C\mathbf{y}$. This condition guarantees that any deviation from the stable strategies does not pay. The search for a stable state is performed by simulating the evolution of a natural selection process. Under very loose conditions, any dynamics that respect the payoffs is guaranteed to converge to Nash equilibria and (hopefully) to ESS's; for this reason, the choice of an actual selection process is not crucial and can be driven mostly by considerations of efficiency and simplicity. We chose to use the repli-

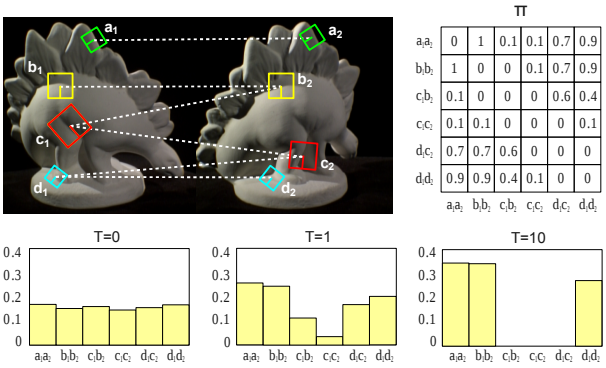
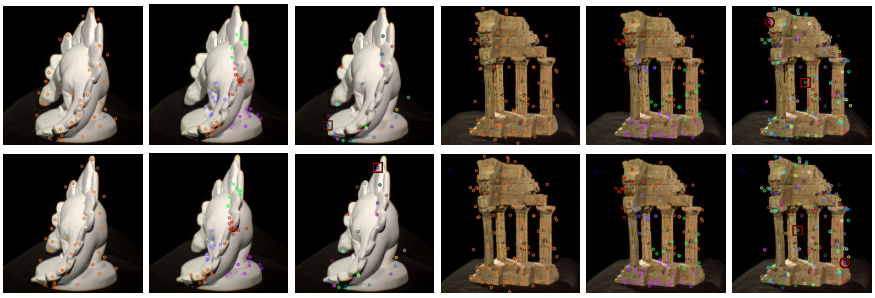



**Fig. 4.** An example of the evolutionary process. Four feature points are extracted from two images and a total of six matching strategies are selected as initial hypotheses. The matrix $\Pi$ shows the compatibilities between pairs of matching strategies according to a one-to-one similarity-enforcing payoff function. Each matching strategy got zero payoff with itself and with strategies that share the same source or destination point (i.e., $\Pi((b_1, b_2), (c_1, b_2)) = 0$). Strategies that are coherent with respect to a similarity transformation exhibit high payoff values (i.e., $\Pi((a_1, a_2), (b_1, b_2)) = 1$ and $\pi((a_1, a_2), (d_1, d_2)) = 0.9)$), while less compatible pairs get lower scores (i.e., $\pi((a_1, a_2), (c_1, c_2)) = 0.1$). Initially (at T=0) the population is set to the barycenter of the simplex and slightly perturbed. After just one iteration, $(c_1, b_2)$ and $(c_1, c_2)$ have lost a significant amount of support, while $(d_1, c_2)$ and $(d_1, d_2)$ are still played by a sizable amount of population. After ten iterations (T=10) $(d_1, d_2)$ has finally prevailed over $(d_1, c_2)$ (note that the two are mutually exclusive). Note that in the final population $((a_1, a_2), (b_1, b_2))$ have a larger support than $(d_1, d_2)$ since they are a little more coherent with respect to similarity.

cator dynamics, a well-known formalization of the selection process governed by the recurrence $\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^t \frac{(C\mathbf{x}^t)_i}{\mathbf{x}^{tT} C \mathbf{x}^t}$, where $\mathbf{x}_i^t$ is the proportion of the population that plays the $i$-th strategy at time $t$. Once the population has reached a local maximum, all the non-extincted pure strategies can be considered selected by the game. One final note should be made about one-to-one matching. Since each source feature can correspond with at most one destination point, it is desirable to avoid any kind of multiple match. It is easy to show that a pair of strategies with mutual zero payoff cannot belong to the support of an ESS (see [1]), thus any payoff function can easily be adapted to enforce one-to-one matching by setting to 0 the payoff of mates that share either the source or the destination point.

## 4 Experimental Results

To evaluate the performance of our proposals, we compared the results with those obtained with the keymatcher included in the structure-from-motion suite Bundler [13]. For the first set of experiments we selected pairs of adjacent views from the "DinoRing" and "TempleRing" sequences from the Middlebury Multi-View Stereo dataset [11]; for these models, camera parameters are provided and used as a ground-truth. For all the sets of experiments we evaluated the



GT-3Drigid GT-2Daffine Bundler GT-3Drigid GT-2Daffine Bundler

**Dino sequence**

| | GT-3Drigid | GT-2Daffine | Bundler |
|---|---|---|---|
| Matches | $262.5 \pm 61.4$ | $271.1 \pm 64.2$ | $172.4 \pm 79.5$ |
| $\Delta\alpha$ | $0.0668 \pm 0.0777$ | $0.0497 \pm 0.0810$ | $0.0767 \pm 0.1172$ |
| $\Delta\gamma$ | $0.4393 \pm 0.4963$ | $0.3184 \pm 0.3247$ | $0.6912 \pm 0.8793$ |

**Temple sequence**

| | GT-3Drigid | GT-2Daffine | Bundler |
|---|---|---|---|
| Mathces | $535.7 \pm 38.7$ | $564.3 \pm 37.2$ | $349.3 \pm 36.2$ |
| $\Delta\alpha$ | $0.1326 \pm 0.0399$ | $0.0989 \pm 0.0224$ | $0.1414 \pm 0.0215$ |
| $\Delta\gamma$ | $0.0809 \pm 0.0144$ | $0.0792 \pm 0.0091$ | $0.0850 \pm 0.0065$ |

**Fig. 5.** Results obtained with the Dino and Temple data sets.

differences in radians between the (calibrated) ground-truth and respectively the estimated rotation angle ($\Delta\alpha$) and rotation axis ($\Delta\gamma$). The "Dino" model is a difficult case in general, as it provides very few features; the upper part of Fig. 5 shows the correspondences produced by our game -theoretic matching approach with geometric constraints enforcing a 3D rigid transformation (GT-3Drigid), the approach with the weak perspective camera assumptions (GT-2Daffine), and the Bundler matcher (Bundler). The color of the points matched using GT-2Daffine relate to the extraction group, i.e., points with the same color have been matched at th same re-iteration of the game-theoretic matching process. The "Temple" model is richer in features and for visualization purposes we only show a subset of the detected matches for all three techniques. The Bundler matcher, while still achieving good results, provides some mismatches in both cases. This can be explained by the fact that the symmetric parts of the object, e.g. the pillars in the temple model, result in very similar features that are hard to disambiguate by a purely local matcher. Both our methods, on the other hand, by enforcing global consistency, can effectively disambiguate the matches. Looking at the results we can see that both our approaches extract around 50% more correspondences than Bundler. The first approach provides a slight increase in precision and reduction in variance of the estimates. Note, however, that the

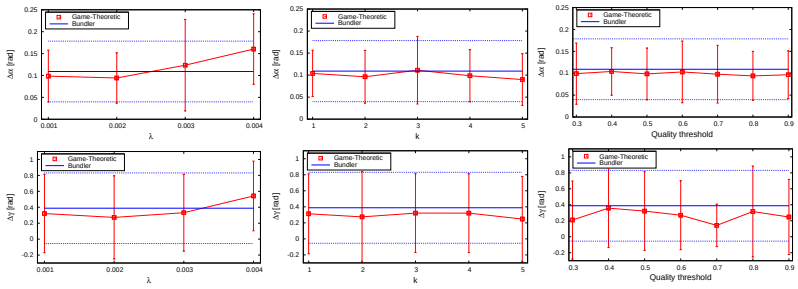



**Fig. 6.** Analysis of the performance of the approach with respect to variation of the parameters of the algorithm.

selected measures evaluate the quality of the underlying least square estimates of the motion parameters after a reprojection step, thus small variations are expected. The approach enforcing a global 2D affine transformation exhibits a larger increase in precision and reduction in variance. This can be explained by the fact that the adjacent views of the two sequences have very little parallax effects, thus the weak persective camera assumption holds quite well. In this context the stricter model is better specified and thus more discriminative.

Next, we analyzed the impact of the algorithm parameters over the quality of the results obtained. To this end, we investigated three parameters: the similarity decay $\lambda$, the number $k$ of candidate mates per features, and the *quality threshold*, that is the minimum support for a correspondence to be considered non-extinct, divided by the maximum support in the population. Figure 4 reports the results of these experiments. The goal of these experiments was to show the sensitivity to the matcher's parameters, not to choose between constraints, so only the 3D geometric constraint was used. Overall, these experiments show that almost all reasonable values of the parameters give similar values for the match, thus those parameters have little influence over the quality of the result, with the Game-Theoretic approach achieving better average results and smaller standard deviation than the Bundler matcher.

## 5 Conclusions

In this paper we introduced a robust matching technique for feature points from multiple views. Robustness is achieved by enforcing global geometric consistency in a pairwise setting. Two different geometric consistency models are proposed. The first enforces the compatibility with a single 3D rigid transformation of the points. This is achieved by using the scale and orientation information offered by SIFT features and projecting what is left of a high-order compatibility problem into a pairwise compatibility measure, by enforcing the conservation of distances between the unknown 3D positions of the points. The second model assumes a weak perspective camera model and enforces that points are subject to an affine transformation. This extracts only local groups at similar depths, but the matching process is repeated to cover the whole scene. In both cases, a game-theoretic approach is used to select a maximally consistent set of candidate matches, where

highly compatible matches are enforced while incompatible correspondences are driven to extinction. Experimental comparisons with a widely used technique show the ability of our approach to obtain more accurate estimates of the scene parameters.

**Acknowledgment**

We acknowledge the financial support of the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project SIMBAD grant no. 213250.

## References

1. Andrea Albarelli, Samuel Rota Bulò, Andrea Torsello, and Marcello Pelillo. Matching as a non-cooperative game. In *ICCV 2009*, 2009.
2. C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Fourth Alvey Vision Conference*, pages 147–151, 1988.
3. Tinne Tuytelaars Herbert Bay and Luc Van Gool. Surf: Speeded up robust features. In *9th European Conference on Computer Vision*, volume 3951, pages 404–417, 2006.
4. D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003.
5. Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.
6. D. Marr and E. Hildreth. Theory of Edge Detection. *Royal Soc. of London Proc. Series B*, 207:187–217, February 1980.
7. J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004. British Machine Vision Computing 2002.
8. K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I*, pages 128–142, London, UK, 2002. Springer-Verlag.
9. K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630, 2005.
10. M. Saquib Sarfraz and Olaf Hellwich. Head pose estimation in face recognition across pose scenarios. In *VISAPP (1)*, pages 235–242, 2008.
11. Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR '06*, pages 519–528, 2006.
12. Jianbo Shi and Carlo Tomasi. Good features to track. In *CVPR*, pages 593–600, 1994.
13. Noah Snavely, Steven M. Seitz, and Richard Szeliski. Modeling the world from internet photo collections. *Int. J. Comput. Vision*, 80(2):189–210, 2008.
14. Andrea Torsello, Samuel Rota Bulò, and Marcello Pelillo. Grouping with asymmetric affinities: A game-theoretic perspective. In *CVPR '06*, pages 292–299, 2006.
15. J. Y. Zien, M. D. F. Schlag, and P. K. Chan. Multi-level spectral hypergraph partitioning with arbitrary vertex sizes. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 18:1389–1399, 1999.

# Fast Population Game Dynamics for Dominant Sets and Other Quadratic Optimization Problems

Samuel Rota Bulò[1], Immanuel M. Bomze[2], and Marcello Pelillo[1]

[1] Dipartimento di Informatica - Univ. of Venice - Italy
{srotabul,pelillo}@dsi.unive.it
[2] Department of Statistics and Decision Support Systems - Univ. of Vienna - Austria
immanuel.bomze@univie.ac.at

**Abstract.** We propose a fast population game dynamics, motivated by the analogy with infection and immunization processes within a population of "players," for finding dominant sets, a powerful graph-theoretical notion of a cluster. Each step of the proposed dynamics is shown to have a linear time/space complexity and we show that, under the assumption of symmetric affinities, the average population payoff is strictly increasing along any non-constant trajectory, thereby allowing us to prove that dominant sets are asymptotically stable (i.e., attractive) points for the proposed dynamics. The approach is general and can be applied to a large class of quadratic optimization problems arising in computer vision. Experimentally, the proposed dynamics is found to be orders of magnitude faster than and as accurate as standard algorithms.

## 1 Introduction

Dominant sets are a graph-theoretical notion of a cluster [1], which have found application in problems as diverse as the analysis of fMRI data [2], content-based image retrieval [3], detection of anomalous activities in video streams [4], bioinformatics [5], human action recognition [6] and matching problems [7,8].

Computationally, the standard approach to finding dominant sets in an edge-weighted graph is to use *replicator dynamics*, a class of evolutionary game-theoretic algorithms inspired by Darwinian selection processes. However, a typical problem associated with these algorithms is the scaling behavior with the number of data. On a dataset containing $N$ examples, the computationally complexity of each replicator dynamics step is $\mathcal{O}(N^2)$, thereby hindering their applicability to problems involving very large data sets, such as high-resolution imagery and spatio-temporal data.

In order to avoid this drawback, in this paper we propose a new population game dynamics for finding dominant sets which turns out to be dramatically faster and even more accurate than standard approaches from evolutionary game theory. Our approach is motivated by the analogy with infection and immunization processes within a population of "players." The selection mechanism

E.R. Hancock et al. (Eds.): SSPR & SPR 2010, LNCS 6218, pp. 275–285, 2010.

governing our dynamics iteratively performs an infection step, which consists of spreading (or suppressing) the most successful (unsuccessful) strategies in the population. The infection phase is then protracted as long as the selected "infective" strategy performs better (or worse, if not extinct) than the average population's payoff. As opposed to standard techniques, such as the replicator dynamics or best-response dynamics, which can be considered interior-point methods, our algorithm resembles a vertex-pivoting method. Each step of the proposed dynamics is shown to have a linear time/space complexity and we show that, under the assumption of symmetric affinities, the average population payoff is strictly increasing along any non-constant trajectory, thereby allowing us to prove that dominant sets (i.e., ESS equilibria of the underlying "grouping game" [9]) are asymptotically stable points for the proposed dynamics.

We provide experimental evidence that the proposed algorithm is orders of magnitude faster than standard dynamics on two computer vision applications, namely image segmentation and region-based hierarchical image matching, while preserving the quality of the solutions found.

Although the main focus in this paper is dominant sets, we note that the proposed approach is general and can be applied to a large class of optimization problem, instances of which abound in computer vision and pattern recognition (e.g., graph matching, stereo matching, image labeling, etc. ).

## 2 Basics of Evolutionary Game Theory

Evolutionary game theory considers an idealized scenario whereby pairs of individuals are repeatedly drawn at random from a large, ideally infinite, population to play a symmetric two-player game. Let $O = \{1, \ldots, n\}$ be the set of *pure strategies* available to the players and let $A$ be the $n \times n$ payoff or utility matrix [10], where $a_{ij}$ is the payoff that a player gains when playing the strategy $i$ against an opponent playing strategy $j$. A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, x_2, \ldots, x_n)^\top$ over the available strategies in $O$. Mixed strategies lie in the standard simplex $\Delta$ of the $n$-dimensional Euclidean space, which is defined as

$$\Delta = \left\{ \mathbf{x} \in \mathbb{R}^n : \sum_{i=1}^{n} x_i = 1 \text{ and } x_i \geq 0,\ i = 1, \ldots, n \right\}.$$

We denote by $\mathbf{e}^i$ the $i$th column of the identity matrix. The *support* of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, defines the set of elements with non-zero probability: $\sigma(\mathbf{x}) = \{i \in O : x_i > 0\}$. The expected payoff that a player obtains by playing the pure strategy $i$ against an opponent playing a mixed strategy $\mathbf{x}$ is $\pi(\mathbf{e}^i|\mathbf{x}) = (A\mathbf{x})_i = \sum_j a_{ij}x_j$ .Hence, the expected payoff received by adopting a mixed strategy $\mathbf{y}$ is given by $\pi(\mathbf{y}|\mathbf{x}) = \mathbf{y}^\top A\mathbf{x}$ while the population expected payoff is $\pi(\mathbf{x}) = \pi(\mathbf{x}|\mathbf{x}) = \mathbf{x}^\top A\mathbf{x}$ . For notational compactness, in the sequel we will write $\pi(\mathbf{y} - \mathbf{x}|\mathbf{z})$ for the payoff difference $\pi(\mathbf{y}|\mathbf{z}) - \pi(\mathbf{x}|\mathbf{z})$, and $\pi(\mathbf{y} - \mathbf{x})$ for $\pi(\mathbf{y} - \mathbf{x}|\mathbf{y}) - \pi(\mathbf{y} - \mathbf{x}|\mathbf{x})$.

A mixed strategy $\mathbf{x}$ is a *(symmetric) Nash (equilibrium) strategy* if for all $\mathbf{y} \in \Delta$, we have $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) \leq 0$. This implies that $\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) \leq 0$ for all $i \in O$, which in turn implies that $\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) = 0$ for all $i \in \sigma(\mathbf{x})$. Hence, the payoff is constant across all (pure) strategies in the support of $\mathbf{x}$, while all strategies outside the support of $\mathbf{x}$ earn a payoff that is less than or equal $\pi(\mathbf{x})$.

A strategy $\mathbf{x}$ is said to be an *Evolutionary Stable Strategy* (ESS) if it is a Nash strategy (*equilibrium condition*) and for all $\mathbf{y} \in \Delta \backslash \{\mathbf{x}\}$ satisfying $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$ we have $\pi(\mathbf{y} - \mathbf{x}|\mathbf{y}) < 0$ (*stability condition*). Intuitively, ESS's are strategies such that any small deviation from them will lead to an inferior payoff. ESS's can be found by *replicator dynamics* (RD), a classic formalization of a natural selection process [10].

## 3 Dominant Sets and Their Characterizations

The dominant set framework is a pairwise clustering approach [1] that is based on the notion of a dominant set, which can be seen as an edge-weighted generalization of a clique. The framework is based on a recursive characterization of the weight $W_S(i)$ of element $i$ with respect to a set $S$ of elements, and characterizes a group as a *dominant set*, i.e., a set that satisfies:

1. $W_S(i) > 0$, for all $i \in S$,
2. $W_{S\cup\{i\}}(i) < 0$, for all $i \notin S$.

These conditions correspond to the two main properties of a cluster: the first regards internal homogeneity, whereas the second regards external heterogeneity.

The characteristic vector $\mathbf{x}^S$ of a set $S \subseteq V$ is defined as

$$x_i^S = \begin{cases} \frac{W_S(i)}{W(S)} & \text{if } i \in S\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

The following result establishes a one-to-one correspondence between ESS's and dominant sets [9].

**Theorem 1.** *If $S \subseteq V$ is a dominant set with respect to affinity matrix $A$, then $\mathbf{x}^S$ is an ESS for a two-player game with payoff matrix $A$.*

*Conversely, if $\mathbf{x}$ is an ESS for a two-person game with payoff matrix $A$, then $S = \sigma(\mathbf{x})$ is a dominant set with respect to $A$, provided that $W_{S\cup\{i\}}(i) \neq 0$ for all $i \notin S$.*

Under the assumption of a symmetric affinity matrix $A$ there exists a one-to-one correspondence between dominant sets and the (strict) local solutions of the following so-called standard quadratic program (StQP) [1]:

$$\max \left\{ \mathbf{x}^\top A \mathbf{x} \,:\, \mathbf{x} \in \Delta \right\}\,. \tag{1}$$

## 4 A New Class of Evolutionary Dynamics

Let $\mathbf{x} \in \Delta$ be the *incumbent* population state, $\mathbf{y}$ be the *mutant* population invading $\mathbf{x}$ and let $\mathbf{z} = (1-\varepsilon)\mathbf{x} + \varepsilon\mathbf{y}$ be the population state obtained by injecting into $\mathbf{x}$ a small share of $\mathbf{y}$-strategists. The *score function* of $\mathbf{y}$ versus $\mathbf{x}$ [11] is given by:

$$h_{\mathbf{x}}(\mathbf{y}, \varepsilon) = \pi(\mathbf{y} - \mathbf{x}|\mathbf{z}) = \varepsilon\pi(\mathbf{y} - \mathbf{x}) + \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) \,.$$

Following [12], we define the *(neutral) invasion barrier* $b_{\mathbf{x}}(\mathbf{y})$ of $\mathbf{x} \in \Delta$ against any mutant strategy $\mathbf{y}$ as the largest population share $\varepsilon_{\mathbf{y}}$ of $\mathbf{y}$-strategists such that for all smaller positive population shares $\varepsilon$, $\mathbf{x}$ earns a higher or equal payoff than $\mathbf{y}$ in the post-entry population $\mathbf{z}$. Formally:

$$b_{\mathbf{x}}(\mathbf{y}) = \inf(\{\varepsilon \in (0,1) : h_{\mathbf{x}}(\mathbf{y}, \varepsilon) > 0\} \cup \{1\}) \,.$$

Given populations $\mathbf{x}, \mathbf{y} \in \Delta$, we say that $\mathbf{x}$ is *immune* against $\mathbf{y}$ if $b_{\mathbf{x}}(\mathbf{y}) > 0$. Trivially, a population is always immune against itself. Note that, $\mathbf{x}$ is immune against $\mathbf{y}$ if and only if either $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) < 0$ or $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$ and $\pi(\mathbf{y} - \mathbf{x}) \leq 0$. If $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) > 0$ we say that $\mathbf{y}$ is *infective* for $\mathbf{x}$. We denote the set of infective strategies for $\mathbf{x}$ as

$$\Upsilon(\mathbf{x}) = \{\mathbf{y} \in \Delta : \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) > 0\} \,.$$

Consider $\mathbf{y} \in \Upsilon(\mathbf{x})$; clearly, this implies $b_{\mathbf{x}}(\mathbf{y}) = 0$. If we allow for invasion of a share $\varepsilon$ of $\mathbf{y}$-strategists as long as the score function of $\mathbf{y}$ versus $\mathbf{x}$ is positive, at the end we will have a share of $\delta_{\mathbf{y}}(\mathbf{x})$ mutants in the post-entry population, where

$$\delta_{\mathbf{y}}(\mathbf{x}) = \inf\left(\{\varepsilon \in (0,1) : h_{\mathbf{x}}(\mathbf{y}, \varepsilon) \leq 0\} \cup \{1\}\right) .$$

Note that if $\mathbf{y}$ is infective for $\mathbf{x}$, then $\delta_{\mathbf{y}}(\mathbf{x}) > 0$, whereas if $\mathbf{x}$ is immune against $\mathbf{y}$, then $\delta_{\mathbf{y}}(\mathbf{x}) = 0$. Since score functions are (affine-)linear, there is a simpler expression $\delta_{\mathbf{y}}(\mathbf{x}) = \min\left[\frac{\pi(\mathbf{x}-\mathbf{y}|\mathbf{x})}{\pi(\mathbf{y}-\mathbf{x})}, 1\right]$, if $\pi(\mathbf{y} - \mathbf{x}) < 0$, and $\delta_{\mathbf{y}}(\mathbf{x}) = 1$, otherwise.

**Proposition 1.** *Let* $\mathbf{y} \in \Upsilon(\mathbf{x})$ *and* $\mathbf{z} = (1-\delta)\mathbf{x} + \delta\mathbf{y}$*, where* $\delta = \delta_{\mathbf{y}}(\mathbf{x})$*. Then* $\mathbf{y} \notin \Upsilon(\mathbf{z})$*.*

The proof of this result is straightforward by linearity and can be found, e.g., in [13].

The core idea of our method is based on the fact that $\mathbf{x} \in \Delta$ is a Nash equilibrium if and only if $\Upsilon(\mathbf{x}) = \emptyset$ (we prove this in Theorem 2). Therefore, as long as we find a strategy $\mathbf{y} \in \Upsilon(\mathbf{x})$, we update the population state according to Proposition 1 in order obtain a new population $\mathbf{z}$ such that $\mathbf{y} \notin \Upsilon(\mathbf{z})$ and we reiterate this process until no infective strategy can be found, or in other words, a Nash equilibrium is reached.

The formalization of this process provides us with a class of new dynamics which, for evident reasons, is called *Infection and Immunization Dynamics* (InImDyn ):

$$\mathbf{x}^{(t+1)} = \delta_{\mathcal{S}(\mathbf{x}^{(t)})}(\mathbf{x}^{(t)})[\mathcal{S}(\mathbf{x}^{(t)}) - \mathbf{x}^{(t)}] + \mathbf{x}^{(t)} \,. \tag{2}$$

Here, $\mathcal{S} : \Delta \to \Delta$ is a generic *strategy selection* function which returns an infective strategy for $\mathbf{x}$ if it exists, or $\mathbf{x}$ otherwise:

$$\mathcal{S}(\mathbf{x}) = \begin{cases} \mathbf{y} & \text{for some } \mathbf{y} \in \Upsilon(\mathbf{x}) \text{ if } \Upsilon(\mathbf{x}) \neq \emptyset\,, \\ \mathbf{x} & \text{otherwise}\,. \end{cases} \tag{3}$$

By running these dynamics we aim at reaching a population state that can not be infected by any other strategy. In fact, if this is the case, then $\mathbf{x}$ is a Nash strategy, which happens if and only if it is fixed (i.e., stationary) under dynamics (2):

**Theorem 2.** *Let $\mathbf{x} \in \Delta$ be a strategy. Then the following statements are equivalent:*
*(a) $\Upsilon(\mathbf{x}) = \emptyset$: there is no infective strategy for $\mathbf{x}$;*
*(b) $\mathbf{x}$ is a Nash strategy;*
*(c) $\mathbf{x}$ is a fixed point under dynamics* (2).

*Proof.* A strategy $\mathbf{x}$ is a Nash strategy if and only if $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) \leq 0$ for all $\mathbf{y} \in \Delta$. This is true if and only if $\Upsilon(\mathbf{x}) = \emptyset$. Further, $\delta = 0$ implies $\mathcal{S}(\mathbf{x}) = \mathbf{x}$. Conversely, if $\mathcal{S}(\mathbf{x})$ returns $\mathbf{x}$, then we are in a fixed point. By construction of $\mathcal{S}(\mathbf{x})$ this happens only if there is no infective strategy for $\mathbf{x}$.

The following result shows that average payoff is strictly increasing along any non-constant trajectory of the dynamics (2), provided that the payoff matrix is symmetric.

**Theorem 3.** *Let $\{\mathbf{x}^{(t)}\}_{t\geq 0}$ be a trajectory of* (2). *Then for all $t \geq 0$,*

$$\pi(\mathbf{x}^{(t+1)}) \geq \pi(\mathbf{x}^{(t)})\,,$$

*with equality if and only if $\mathbf{x}^{(t)} = \mathbf{x}^{(t+1)}$, provided that the payoff matrix is symmetric.*

*Proof.* Again, let us write $\mathbf{x}$ for $\mathbf{x}^{(t)}$ and $\delta$ for $\delta_{\mathcal{S}(\mathbf{x})}(\mathbf{x})$. As shown in [13], we have

$$\pi(\mathbf{x}^{(t+1)}) - \pi(\mathbf{x}^{(t)}) = \delta\left[h_{\mathbf{y}}(\mathbf{x}, \delta) + \pi(\mathbf{y} - \mathbf{x}|\mathbf{x})\right]\,.$$

If $\mathbf{x}^{(t+1)} \neq \mathbf{x}^{(t)}$, then $\mathbf{x}$ is no Nash strategy, and $\mathbf{y} = \mathcal{S}(\mathbf{x})$ returns an infective strategy. Hence $\delta > 0$ and

$$h_{\mathbf{y}}(\mathbf{x}, \delta) + \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) \geq \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) > 0$$

(in fact, if $\delta < 1$, then even $h_{\mathbf{y}}(\mathbf{x}, \delta) = 0$), so that we obtain a strict increase of the population payoff. On the other hand, if $\pi(\mathbf{x}^{(t+1)}) = \pi(\mathbf{x}^{(t)})$, then the above equation implies $\delta = 0$ or $h_{\mathbf{x}}(\mathbf{x}, \delta) = \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$, due to nonnegativity of both quantities above. In particular, we have $\delta = 0$ or $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$. In both cases, $\mathbf{y} = \mathcal{S}(\mathbf{x})$ cannot be infective for $\mathbf{x}$. Thus $\Upsilon(\mathbf{x}) = \emptyset$ and $\mathbf{x}$ must be a fixed point, according to Theorem 2. This establishes the last assertion of the theorem.

Theorem 3 shows that by running InImDyn , under symmetric payoff function, we strictly increase the population payoff until we reach a Nash equilibrium at a fixed point. This of course holds for any selection function $\mathcal{S}(\mathbf{x})$ satisfying (3). However, the way we choose $\mathcal{S}(\mathbf{x})$ may affect the efficiency of the dynamics. The next section introduces a particular selection function that leads to a well-performing dynamics for our purposes.

## 5 A Pure Strategy Selection Function

Depending on how we choose the function $\mathcal{S}(\mathbf{x})$ in (2), we may obtain different dynamics. One in particular, which is simple and leads to nice properties, consists in allowing only infective pure strategies.

Given a population $\mathbf{x}$, we define the co-strategy of $\mathbf{e}^i$ with respect to $\mathbf{x}$ as

$$\overline{\mathbf{e}^i}_{\mathbf{x}} = \frac{x_i}{x_i - 1}(\mathbf{e}^i - \mathbf{x}) + \mathbf{x}\,.$$

Note that if $\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) \neq 0$ then either $\mathbf{e}^i \in \Upsilon(\mathbf{x})$ or $\overline{\mathbf{e}^i}_{\mathbf{x}} \in \Upsilon(\mathbf{x})$.

Consider the strategy selection function $\mathcal{S}_{Pure}(\mathbf{x})$, which finds a pure strategy $i$ maximizing $|\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x})|$, and returns $\mathbf{e}^i$, $\overline{\mathbf{e}^i}_{\mathbf{x}}$ or $\mathbf{x}$ according to whether $\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x})$ is positive, negative or zero. Let $\mathcal{M}(\mathbf{x})$ be a pure strategy such that

$$\mathcal{M}(\mathbf{x}) \in \arg\max_{i=1,\ldots,n} |\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x})|\,.$$

Then $\mathcal{S}_{Pure}(\mathbf{x})$ can be written as

$$\mathcal{S}_{Pure}(\mathbf{x}) = \begin{cases} \mathbf{e}^i & \text{if } \pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) > 0 \text{ and } i = \mathcal{M}(\mathbf{x}) \\ \overline{\mathbf{e}^i}_{\mathbf{x}} & \text{if } \pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) < 0 \text{ and } i = \mathcal{M}(\mathbf{x}) \\ \mathbf{x} & \text{otherwise}\,. \end{cases}$$

Note that the search space for an infective strategy is reduced from $\Delta$ to a finite set. Therefore, it is not obvious that $\mathcal{S}_{Pure}(\mathbf{x})$ is a well-defined selection function, i.e., it satisfies (3). The next theorem shows that indeed it is.

**Proposition 2.** *Let $\mathbf{x} \in \Delta$ be a population. There exists an infective strategy for $\mathbf{x}$, i.e., $\Upsilon(\mathbf{x}) \neq \emptyset$, if and only if $\mathcal{S}_{Pure}(\mathbf{x}) \in \Upsilon(\mathbf{x})$.*

*Proof.* Let $\mathbf{y} \in \Upsilon(\mathbf{x})$. Then $0 < \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = \sum_{i=1}^{n} y_i \pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x})$. But this implies that there exists at least one infective pure strategy for $\mathbf{x}$, i.e., $\mathbf{e}^i \in \Upsilon(\mathbf{x})$ for some $i = 1, \ldots, n$. The converse trivially holds.

A fixed point of InImDyn is *asymptotically stable* if any trajectory starting sufficiently close to $\mathbf{x}$ converges to $\mathbf{x}$.

**Theorem 4.** *A state $\mathbf{x}$ is asymptotically stable for* InImDyn *with $\mathcal{S}_{Pure}$ as strategy selection function if and only if $\mathbf{x}$ is an ESS, provided that the payoff matrix is symmetric.*

*Proof.* If the payoff matrix is symmetric, every accumulation point of InImDyn with $\mathcal{S}_{Pure}$ is a Nash equilibrium [13]. Moreover $ESS$s are strict local maximizers of $\pi(\mathbf{x})$ over $\Delta$ and vice versa [10].

If $\mathbf{x}$ is asymptotically stable, then there exists a neighborhood $U$ of $\mathbf{x}$ in $\Delta$ such that any trajectory starting in $U$ converges to $\mathbf{x}$. By Theorem 3 this implies that $\pi(\mathbf{x}) > \pi(\mathbf{y})$ for all $\mathbf{y} \in U$, $\mathbf{y} \neq \mathbf{x}$. Hence, $\mathbf{x}$ is a strict local maximizer of $\pi(\mathbf{x})$ and therefore $\mathbf{x}$ is an ESS.

Conversely, if $\mathbf{x}$ is an ESS then $\mathbf{x}$ is a strict local maximizer of $\pi(\mathbf{x})$ and an isolated Nash equilibrium. Hence, there exists a neighborhood $U$ of $\mathbf{x}$ in $\Delta$ where $\pi(\mathbf{x})$ is strictly concave and $\mathbf{x}$ is the only accumulation point. This together with Theorem 3 implies that any trajectory starting in $U$ will converge to $\mathbf{x}$. Hence, $\mathbf{x}$ is asymptotically stable.

This selection function exhibits the nice property of rendering the complexity per iteration of our new dynamics linear in both space and time, as opposed to the replicator dynamics, which have quadratic space/time complexity per iteration.

**Theorem 5.** *Given the iterate $\mathbf{x}^{(t)}$ and its linear transformation $A\mathbf{x}^{(t)}$, both space and time requirement of one iteration step is linear in $n$, the number of objects.*

*Proof.* Again abbreviate $\mathbf{x} = \mathbf{x}^{(t)}$. Now, given $A\mathbf{x}$ we can straightforwardly compute in linear time and space $\pi(\mathbf{x})$ and $\mathcal{S}_{Pure}(\mathbf{x})$. Assume that $\mathcal{S}_{Pure}(\mathbf{x}) = \mathbf{e}^i$, then the computation of $\delta_{\mathbf{e}^i}(\mathbf{x})$ has a linear complexity, since $\pi(\mathbf{x} - \mathbf{e}^i|\mathbf{x}) = (A\mathbf{x})_i - \pi(\mathbf{x})$ and $\pi(\mathbf{e}^i - \mathbf{x}) = a_{ii} - 2A\mathbf{x} + \pi(\mathbf{x})$. Moreover, $A\mathbf{x}^{(t+1)}$ can be also computed in linear time and space since

$$A\mathbf{x}^{(t+1)} = \delta_{\mathbf{e}^i}(\mathbf{x})\,[A_i - A\mathbf{x}] + A\mathbf{x}\,,$$

where $A_i$ is the $i$th column of $A$. Similar arguments hold if $\mathcal{S}_{Pure}(\mathbf{x}) = \overline{\mathbf{e}^i}_{\mathbf{x}}$. Indeed,

$$\pi(\overline{\mathbf{e}^i}_{\mathbf{x}} - \mathbf{x}|\mathbf{x}) = \frac{x_i}{x_i - 1}\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x})\,,$$
$$\pi(\overline{\mathbf{e}^i}_{\mathbf{x}} - \mathbf{x}) = \left(\frac{x_i}{x_i - 1}\right)^2 \pi(\mathbf{e}^i - \mathbf{x})\,,$$

and finally,

$$A\mathbf{x}^{(t+1)} = \left(\frac{x_i}{x_i - 1}\right)\delta_{\overline{\mathbf{e}^i}_{\mathbf{x}}}(\mathbf{x})\,[A_i - A\mathbf{x}] + A\mathbf{x}\,.$$

Hence the result.

The only step of quadratic complexity is the first one, where we need to compute $A\mathbf{x}^{(0)}$. Even this can be reduced to linear complexity, if we start from a pure strategy $\mathbf{e}^i$, in which case we have $A\mathbf{x}^{(0)} = A_i$. Note that the latter is impossible, e.g., for the replicator dynamics.

## 6 Experimental Results

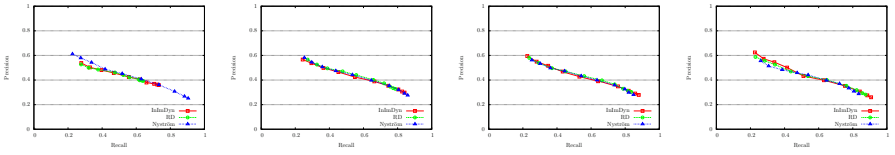
In order to test the effectiveness of our algorithm, we present experiments on some computer vision problems, which have been attacked using the dominant-set framework or related quadratic optimization problems. Our aim is to show the computational gain over the standard algorithm used in the literature, namely the replicator dynamics (RD). Specifically, we present comparisons on image segmentation [1] and region-based hierarchical image matching [8].

The stopping criterion adopted with our dynamics is a measure of the accuracy of the Nash equilibrium, which is given by $\epsilon(\mathbf{x}) = \sum_i \min\left\{x_i, \pi(\mathbf{x} - \mathbf{e}^i|\mathbf{x})\right\}^2$. Indeed, $\epsilon(\mathbf{x})$ is 0 if and only if $\mathbf{x}$ is a Nash equilibrium. In the experiments, we stopped the dynamics at accurate solutions, namely when $\epsilon(\mathbf{x}) < 10^{-10}$. As for RD, we stopped the dynamics either when $\epsilon(\mathbf{x}) < 10^{-10}$ or when a maximum number of iterations was exceeded.

### 6.1 Image Segmentation

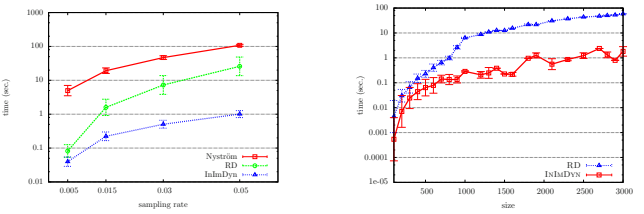
We performed image segmentation experiments over the whole Berkeley dataset [14] using the dominant-set framework as published in [1]. The affinity between two pixels $i$ and $j$ was computed based on color and using the standard Gaussian kernel. Our InImDyn algorithm was compared against standard replicator dynamics (RD) [1] (using the out-of-sample extension described in [15]) as well as the Nyström method [16]. The algorithms were coded in C and run on a AMD Sempron 3 GHz computer with 1GB RAM. To test the behavior of the algorithms under different input sizes we performed experiments at different pixel sampling rates, namely 0.005, 0.015, 0.03 and 0.05, which roughly correspond to affinity matrices of size 200, 600, 1200 and 2000, respectively. Since the Nyström method, as opposed to the dominant set approach, needs as input the desired number of clusters, we selected an optimal one after a careful tuning phase.

In Figure 2(a) we report (in logarithmic scale) the average computational times (in seconds) per image obtained with the three approaches. The computational gain of InImDyn over the replicator dynamics is remarkable and it clearly increases at larger sampling rates. It is worth mentioning that InImDyn other than being faster, achieved also better approximations of Nash equilibriums as




(a) s.r. 0.005 (b) s.r. 0.015 (c) s.r. 0.03 (d) s.r. 0.05

**Fig. 1.** Precision/Recall plots obtained on the Berkeley Image Database (s.r.=sampling rate)




(a) Image segmentation (b) Image matching

**Fig. 2.** Average execution times (in logarithmic scale) for the image segmentation and region-based hierarchical image matching applications

opposed to RD. As for the quality of the segmentation results, we report in Figure 1 the average precision/recall obtained in the experiment with the different sampling rates. As can be seen, all the approaches perform equivalently, in particular RD and InImDyn achieved precisely the same results as expected.

### 6.2 Region-Based Hierarchical Image Matching

In [8] the authors present an approach to region-based hierarchical image matching, aimed at identifying the most similar regions in two images, according to a similarity measure defined in terms of geometric and photometric properties. To this end, each image is mapped into a tree of recursively embedded regions, obtained by a multiscale segmentation algorithm. In this way the image matching problem is cast into a tree matching problem, that is solved recursively through a set of sub-matching problems, each of which is then attacked using replicator dynamics (see [8] for details). Given that typically hundreds of sub-matching problems are generated by a single image matching instance, it is of primary importance to have at one's disposal a fast matching algorithm. This makes our solution particularly appealing for this application.

We compared the running time of InImDyn and RD over a set of images taken from the original paper [8]. We run the experiments on a machine equipped with 8 Intel Xeon 2.33 GHz CPUs and 8 GB RAM. Figure 2(b) shows the average computation times (in seconds) needed by RD and InImDyn to solve the set of sub-matching problems generated from 10 image matching instances. Since each image matching problem generated sub-matching problems of different sizes, we grouped the instances having approximately the same size together. We plotted the average running time within each group (in logarithmic scale) as a function of the instance sizes and reported the standard deviations as error bars. Again, as can be seen, InImDyn turned out to be orders of magnitude faster than RD.

## 7 From QPs to StQPs

Although in this paper we focused mainly on dominant sets, which lead to quadratic optimization problems over the standard simplex (StQPs), the

proposed approach is indeed more general and can be applied to a large class of quadratic programming problems (QPs), instances of which frequently arise in computer vision and pattern recognition.

In fact, consider a general QP over a bounded polyhedron

$$\max\left\{\frac{1}{2}\mathbf{x}^\top Q\mathbf{x}+\mathbf{c}^\top\mathbf{x}\,:\,\mathbf{x}\in M\right\}, \tag{4}$$

where $M=\operatorname{conv}\{\mathbf{v}_1,\ldots,\mathbf{v}_k\}\subseteq\mathbb{R}^n$ is the convex hull of the points $\mathbf{v}_i$, which form the columns of a $n\times k$-matrix $V$. Then we can write the QP in (4) as the following StQP:

$$\max\left\{\mathbf{y}^\top\hat{Q}\mathbf{y}\,:\,\mathbf{y}\in\Delta\right\},$$

where $\hat{Q}=\frac{1}{2}\left(V^\top QV+\mathbf{e}^\top V^\top\mathbf{c}+\mathbf{c}^\top V\mathbf{e}\right)$.

Thus every QP over a polytope can be expressed as an StQP. This approach is of course only practical if the vertices $V$ are known and $k$ is not too large. This is the case of QPs over the $\ell^1$ ball, where $V=[I|-I]$, $I$ the $n\times n$ identity matrix and $\Delta\subset\mathbb{R}^{2n}$ and, more generally, for box-constrained QPs [17]. However, even for general QPs, where the constraints are expressed as $M=\{\mathbf{x}\in\mathbb{R}^n_+\,:\,A\mathbf{x}=b\}$, we can use StQP as a relaxation without using all vertices (see [18] for details).

## Acknowledgements

We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250).

## References

1. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. IEEE Trans. Pattern Anal. Machine Intell. 29(1), 167–172 (2007)
2. Neumann, J., von Cramon, D.Y., Forstmann, B.U., Zysset, S., Lohmann, G.: The parcellation of cortical areas using replicator dynamics in fMRI. NeuroImage 32(1), 208–219 (2006)
3. Wang, M., Ye, Z.L., Wang, Y., Wang, S.X.: Dominant sets clustering for image retrieval. Signal Process. 88(11), 2843–2849 (2008)
4. Hamid, R., Johnson, A., Batta, S., Bobick, A., Isbell, C., Coleman, G.: Detection and explanation of anomalous activities: representing activities as bags of event n-grams. In: CVPR, vol. 1, pp. 20–25 (2005)
5. Frommlet, F.: Tag SNP selection based on clustering according to dominant sets found using replicator dynamics. Adv. in Data Analysis (in press, 2010)
6. Wei, Q.D., Hu, W.M., Zhang, X.Q., Luo, G.: Dominant sets-based action recognition using image sequence matching. In: ICIP, vol. 6, pp. 133–136 (2007)
7. Albarelli, A., Torsello, A., Rota Bulò, S., Pelillo, M.: Matching as a non-cooperative game. In: ICCV (2009)
8. Todorovic, S., Ahuja, N.: Region-based hierarchical image matching. Int. J. Comput. Vision 78(1), 47–66 (2008)

9. Torsello, A., Rota Bulò, S., Pelillo, M.: Grouping with asymmetric affinities: a game-theoretic perspective. In: CVPR, pp. 292–299 (2006)
10. Weibull, J.W.: Evolutionary game theory. Cambridge University Press, Cambridge (1995)
11. Bomze, I.M., Pötscher, B.M.: Game Theoretical Foundations of Evolutionary Stability. Springer, Heidelberg (1989)
12. Bomze, I.M., Weibull, J.W.: Does neutral stability imply Lyapunov stability? Games and Econ. Behaviour 11, 173–192 (1995)
13. Rota Bulò, S.: A game-theoretic framework for similarity-based data clustering. PhD thesis, University of Venice (2009)
14. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: ICCV, July 2001, vol. 2, pp. 416–423 (2001)
15. Pavan, M., Pelillo, M.: Efficient out-of-sample extension of dominant-set clusters. NIPS 17, 1057–1064 (2005)
16. Fowlkes, C., Belongie, S., Malik, F.C.J.: Spectral grouping using the Nyström method. IEEE Trans. Pattern Anal. Machine Intell. 26(2), 214–225 (2004)
17. Pardalos, P.M.: Quadratic problems defined on a convex hull of points. BIT Num. Math. 28(2), 323–329 (1988)
18. Bomze, I.M., Locatelli, M., Tardella, F.: Efficient and cheap bounds for (standard) quadratic optimization. Technical report, University "La Sapienza" of Rome (2005)

# Pairwise Probabilistic Clustering Using Evidence Accumulation

Samuel Rota Bulò[1], André Lourenço[3], Ana Fred[2,3], and Marcello Pelillo[1]

[1] Dipartimento di Informatica - University of Venice - Italy
{srotabul,pelillo}@dsi.unive.it
[2] Instituto Superior Técnico - Lisbon - Portugal
[3] Instituto de Telecomunicações - Lisbon - Portugal
{arlourenco,afred}@lx.it.pt

**Abstract.** In this paper we propose a new approach for consensus clustering which is built upon the evidence accumulation framework. Our method takes the co-association matrix as the only input and produces a soft partition of the dataset, where each object is probabilistically assigned to a cluster, as output. Our method reduces the clustering problem to a polynomial optimization in probability domain, which is attacked by means of the Baum-Eagon inequality. Experiments on both synthetic and real benchmarks data, assess the effectiveness of our approach.

## 1 Introduction

There is a close connection between the concepts of pairwise similarity and probability in the context of unsupervised learning. It is a common assumption that, if two objects are similar, it is very likely that they are grouped together by some clustering algorithm, the higher the similarity, the higher the probability of co-occurrence in a cluster. Conversely, if two objects co-occur very often in the same cluster (high co-occurrence probability), then it is very likely that they are very similar. This duality and correspondence between pairwise similarity and pairwise probability within clusters forms the core idea of the clustering ensemble approach known as evidence accumulation clustering (EAC) [1].

Evidence accumulation clustering combines the results of multiple clusterings into a single data partition by viewing each clustering result as an independent evidence of pairwise data organization. Using a pairwise frequency count mechanism amongst a clustering committee, the method yields, as an intermediate result, a co-association matrix that summarizes the evidence taken from the several members in the clustering ensemble. This matrix corresponds to the maximum likelihood estimate of the probability of pairs of objects being in the same group, as assessed by the clustering committee. One of the main advantages of EAC is that it allows for a big diversification within the clustering committee. Indeed, no assumption is made about the algorithms used to produce the data partitions, it is robust to incomplete information, i.e., we may include partitions over sub-sampled versions of the original data set, and no restriction is made on the number of clusters of the partitions.

E.R. Hancock et al. (Eds.): SSPR & SPR 2010, LNCS 6218, pp. 395–404, 2010.

Once a co-association matrix is produced according to the EAC framework, a consensus clustering is obtained by applying a clustering algorithm, which typically induces a hard partition, to the co-association matrix. Although having crisp partitions as baseline for the accumulation of evidence of data organization is reasonable, this assumption is too restrictive in the phase of producing a consensus clustering. This is for instance the case for many important applications such as clustering micro-array gene expression data, text categorization, perceptual grouping, labeling of visual scenes and medical diagnosis. In fact, the importance of dealing with overlapping clusters has been recognized long ago [2] and recently, in the machine learning community, there has been a renewed interest around this problem [3,4]. Moreover, by inducing hard partitions we loose important information like the level of uncertainty of each label assignment. It is also worth considering that the underlying clustering criteria of ad hoc algorithms do not take advantage of the probabilistic interpretation of the computed similarities, which is an intrinsic part of the EAC framework.

In this paper we propose a new approach for consensus clustering which is built upon the evidence accumulation framework. Our idea was inspired by a recent work due to Zass and Sashua [5]. Our method takes the co-association matrix as the only input and produces a soft partition of the data set, where each object is probabilistically assigned to a cluster, as output. In order to find the unknown cluster assignments, we fully exploit the fact that each entry of the co-association matrix is an estimation of the probability of two objects to be in a same cluster, which is derived from the ensemble of clusterings. Indeed, it is easy to see that under reasonable assumptions, the probability that two objects $i$ and $j$ will occur in the same cluster is a function of the unknown cluster assignments of $i$ and $j$. By minimizing the divergence between the estimation derived from the co-association matrix and this function of the unknowns, we obtain the result of the clustering procedure. More specifically, our method reduces the clustering problem to a polynomial optimization in the probability domain, which is attacked by means of the Baum-Eagon inequality [6]. This inequality, indeed, provides us with a class of nonlinear transformations that serve our purpose. In order to assess the effectiveness of our findings we conducted experiments on both synthetic and real benchmark data sets.

## 2 A Probabilistic Model for Clustering

Let $O = \{1, \ldots, n\}$ be a set of data objects (or simply objects) to cluster into $K$ classes and let $\mathcal{E} = \{cl_i\}_{i=1}^{N}$ be an ensemble of $N$ clusterings of $O$ obtained by running different algorithms with different parameterizations on (possibly) sub-sampled versions of the original data set $O$. Data sub-sampling is herein put forward as a most general framework for the following reasons: it favors the diversification of the clustering ensemble; it models situations of distributed clustering where local clusterers have only partial access to the data; by using this type of data perturbation, the co-association matrix has an additional interpretation of pairwise stability that can further be used for the purpose of cluster validation [7].

Each clustering in the ensemble $\mathcal{E}$ is a function $cl_i : O_i \to \{1, \ldots, K_i\}$ from the set of objects $O_i \subseteq O$ to a class label. For the afore-mentioned reasons, $O_i$ is a subset of the original data set $O$ and, moreover, each clustering may assume a different number of classes $K_i$. We denote by $\Omega_{ij}$ the indices of the clusterings where $i$ and $j$ have been classified, which is given by

$$\Omega_{ij} = \{p = 1 \ldots N : i, j \in O_p\} .$$

Consider also $N_{ij} = |\Omega_{ij}|$, where $|\cdot|$ provides the cardinality of the argument, which is the number of clusterings where $i$ and $j$ have been both classified.

The aim of our work is to learn, from the ensemble of clusterings $\mathcal{E}$, how to cluster the objects into $K$ classes, without having, in principle, any other information about the objects we are going to cluster. To this end, we start from the assumption that objects can be softly assigned to clusters. Hence, the clustering problem consists in estimating, for each object $i \in O$, an unknown assignment $\mathbf{y}_i$, which is a probability distribution over the set of cluster labels $\{1, \ldots, K\}$, or, in other words, an element of the *standard simplex* $\Delta_K$ given by

$$\Delta_K = \{\boldsymbol{x} \in \mathbb{R}_+^K : \|\mathbf{x}\|_1 = 1\} ,$$

where $\mathbb{R}_+$ is the set of nonnegative reals, and $\|\cdot\|_1$ is the $\ell^1$-norm. The $k$th entry of $\mathbf{y}_i$ thus provides the probability of object $i$ to be assigned to cluster $k$. Given the unknown cluster assignments $\mathbf{y}_i$ and $\mathbf{y}_j$ of objects $i$ and $j$, respectively, and assuming independent cluster assignments, the probability of them to occur in a same cluster can be easily derived as $\mathbf{y}_i^\top \mathbf{y}_j$. Suppose now $Y = (\mathbf{y}_1, \ldots, \mathbf{y}_n) \in \Delta_K^n$ to be the matrix formed by stacking the $\mathbf{y}_i$'s, which in turn form the columns of $Y$. Then, the $n \times n$ matrix $Y^\top Y$ provides the co-occurrence probability of any pair of objects in $O$.

For each pair of objects $i$ and $j$, let $X_{ij}$ be a Bernoulli distributed random variable (r.v.) indicating whether objects $i$ and $j$ occur in a same cluster. Note that, according to our model, the mean (and therefore the parameter) of $X_{ij}$ is $\mathbf{y}_i^\top \mathbf{y}_j$, i.e., the probability of co-occurrence of $i$ and $j$. For each pair of objects $i$ and $j$, we collect from the clusterings ensemble $N_{ij}$ independent realizations $x_{ij}^{(p)}$ of $X_{ij}$, which are given by:

$$x_{ij}^{(p)} = \begin{cases} 1 & \text{if } cl_p(i) = cl_p(j) , \\ 0 & \text{otherwise} . \end{cases}$$

for $p \in \Omega_{ij}$. By taking their mean, we obtain the empirical probability of co-occurrence $c_{ij}$, which is the fraction of times objects $i$ and $j$ have been assigned to a same cluster:

$$c_{ij} = \frac{1}{N_{ij}} \sum_{p \in \Omega_{ij}} x_{ij}^{(p)} .$$

The matrix $C = (c_{ij})$, derived from the empirical probabilities of co-occurrence of any pair of objects, is known as the *co-association matrix* within the evidence

accumulation-based framework for clustering [8,1]. Since $C$ is the maximum likelihood estimate of $Y^\top Y$ given the observations from the clustering ensemble $\mathcal{E}$, we will refer to the former as the *empirical co-association matrix*, and to the latter as the *true co-association matrix*.

At this point, by minimizing the divergence, in a least-square sense, of the true co-association matrix from the empirical one, with respect to $Y$, we find a solution $Y^*$ of the clustering problem. This leads to the following optimization problem:

$$\begin{aligned} Y^* = \arg\min \quad & \|C - Y^\top Y\|_F^2 \\ \text{s.t.} \quad & Y \in \Delta_K^n \,. \end{aligned} \tag{1}$$

where $\|\cdot\|_F$ is the Frobenius norm. Note that $Y^*$ provides us with soft assignments of the objects to the $K$ classes. Indeed, $y^*_{ki}$ gives the probability of object $i$ to be assigned to class $k$. If a hard partition is needed, this can be forced by assigning each object $i$ to the highest probability class, which is given by: $\arg\max_{k=1\ldots K}\{y^*_{ki}\}$. Moreover, by computing the entropy of each $\mathbf{y}_i$, we can obtain an indication of the uncertainty of the cluster assignment for object $i$.

## 3 Related Work

In [5] a similar approach is proposed for pairwise clustering. First of all, a preprocessing on the similarity matrix $W$ looks for its closest doubly-stochastic matrix $F$ under $\ell_1$ norm, or Frobenius norm, or relative entropy [9]. The $k$-clustering problem is then tackled by finding a completely-positive factorization of $F = (f_{ij})$ in the least-square sense, i.e., by solving the following optimization problem:

$$\begin{aligned} G^* = \arg\min \quad & \|F - G^\top G\|_F^2 \\ \text{s.t.} \quad & G \in \mathbb{R}_+^{k\times n} \,. \end{aligned} \tag{2}$$

Note that this leads to an optimization program, which resembles (1), but is inherently different. The elements $g_{ri}$ of the resulting matrix $G$ provide an indication of object $i$ to be assigned to class $r$. However, unlike our formulation, these quantities are not explicit probabilities and it may happen for instance that $g_{ri} = 0$ for all $r = 1\ldots k$, i.e., some objects may remain in principle unclassified.

The approach proposed to find a local solution of (2) consists in iterating the following updating rule:

$$g_{ri} \leftarrow \frac{g_{ri}\sum_{j\neq i}^n g_{rj} f_{ij}}{\sum_{s=1}^k g_{si}\sum_{j\neq i}^n g_{sj} g_{rj}} \,.$$

The computational complexity for updating all entries in $G$ once (complete iteration) is $O(kn^2)$, while we expect to find a solution in $O(\gamma kn^2)$, where $\gamma$ is the average number of complete iterations required to converge. A disadvantage of this iterative scheme is that updates must be sequential, i.e., we cannot update all entries of $G$ in parallel.

## 4 The Baum-Eagon Inequality

In the late 1960s, Baum and Eagon [6] introduced a class of nonlinear transformations in probability domain and proved a fundamental result which turns out to be very useful for the optimization task at hand. The next theorem introduces what is known as the Baum-Eagon inequality.

**Theorem 1 (Baum-Eagon).** *Let $X = (x_{ri}) \in \Delta_k^n$ and $Q(X)$ be a homogeneous polynomial in the variables $x_{ri}$ with nonnegative coefficients. Define the mapping $Z = (z_{ri}) = \mathcal{M}(X)$ as follows:*

$$z_{ri} = x_{ri} \frac{\partial Q(X)}{\partial x_{ri}} \Big/ \sum_{s=1}^{k} x_{si} \frac{\partial Q(X)}{\partial x_{si}}, \tag{3}$$

*for all $i = 1 \ldots n$ and $r = 1 \ldots k$. Then $Q(\mathcal{M}(X)) > Q(X)$, unless $\mathcal{M}(X) = X$. In other words $\mathcal{M}$ is a growth transformation for the polynomial $Q$.*

This result applies to homogeneous polynomials, however in a subsequent paper, Baum and Sell [10] proved that Theorem 1 still holds in the case of arbitrary polynomials with nonnegative coefficients, and further extended the result by proving that $\mathcal{M}$ increases $Q$ homotopically, which means that for all $0 \leq \eta \leq 1$, $Q(\eta\mathcal{M}(X) + (1-\eta)X) \geq Q(X)$ with equality if and only if $\mathcal{M}(X) = X$.

The Baum-Eagon inequality provides an effective iterative means for maximizing polynomial functions in probability domains, and in fact it has served as the basis for various statistical estimation techniques developed within the theory of probabilistic functions of Markov chains [11]. It is indeed not difficult to show that, by starting from the interior of the simplex, the fixed points of the Baum-Eagon dynamics satisfy the first-order Karush-Kuhn-Tucker necessary conditions for local maxima and that we have a strict local solution in correspondence to asymptotically stable point.

## 5 The Algorithm

In order to use the Baum-Eagon theorem for optimizing (1) we need to meet the requirement of having a polynomial to maximize with nonnegative coefficients in the simplex-constrained variables. To this end, we consider the following optimization program, which is proved to be equivalent to (1):

$$\begin{aligned} \max \quad & 2Tr(CY^\top Y) + \|Y^\top E_K Y\|^2 - \|Y^\top Y\|^2 \\ \text{s.t.} \quad & Y \in \Delta_K^n, \end{aligned} \tag{4}$$

where $E_K$ is the $K \times K$ matrix of all 1's, and $Tr(\cdot)$ is the matrix trace function.

**Proposition 1.** *The maximizers of* (4) *are minimizers of* (1) *and vice versa. Moreover, the objective function of* (4) *is a polynomial with nonnegative coefficients in the variables $y_{ki}$, which are elements of $Y$.*

*Proof.* Let $P(Y)$ and $Q(Y)$ be the objective functions of (1) and (4), respectively.

To prove the second part of the proposition note that trivially every term of the polynomial $\|Y^\top Y\|^2$ is also a term of $\|Y^\top E_K Y\|^2$. Hence, $Q(Y)$ is a polynomial with nonnegative coefficients in the variables $y_{ki}$.

As for the second part, by simple algebra, we can write $Q(Y)$ in terms of $P(Y)$ as follows:

$$\begin{aligned} Q(Y) &= \|C\|^2 - P(Y) + \|Y^\top E_K Y\|^2 \\ &= \|C\|^2 - P(Y) + 1\,, \end{aligned}$$

where we used the fact that $\|Y^\top E_K Y\| = 1$. Note that the removal of the constant terms from $Q(Y)$ leaves its maximizers over $\Delta_K^n$ unaffected. Therefore, maximizers of (4) are also maximizers of $-P(Y)$ over $\Delta_K^n$ and thus minimizers of (1). This concludes the proof.

By Proposition 1 we can use Theorem 1 to locally optimize (4). This allows us to find a solution of (1). Note that, in our case, the objective function is not a homogeneous polynomial but, as mentioned previously, this condition is not necessary [10]. By applying (3), we obtain the following updating rule for $Y = (y_{ki})$:

$$y_{ki}^{(t+1)} = y_{ki}^{(t)} \frac{n + [Y(C - Y^\top Y)]_{ki}}{n + \sum_k y_{ki}^{(t)} [Y(C - Y^\top Y)]_{ki}}\,, \tag{5}$$

where we abbreviated $Y^{(t)}$ with $Y$ and any non-constant iteration of (5) strictly decreases the objective function of (1).

The computational complexity of the proposed dynamics is $O(\gamma k n^2)$, where $\gamma$ is the average number of iterations required to converge (note that in our experiments we kept $\gamma$ fixed). One remarkable advantage of this dynamics is that it can be easily parallelized in order to benefit from modern multi-core processors. Additionally, it can be easily implemented with few lines of Matlab code.

## 6 Experiments

We conducted experiments on different real data-sets from the UCI Machine Learning Repository: iris, house-votes, std-yeast-cell and breast-cancer. Additionally, we considered also the image-complex synthetic data-set, shown in figure 1. For each data-set, we produced the clustering ensemble $\mathcal{E}$ by running different clustering algorithms, with different parameters, on subsampled versions of the original data-set (the sampling rate was fixed to 0.9). The clustering algorithms used to produce the ensemble were the following [12]: Single Link (SL), Complete Link (CL), Average Link (AL) and K-means (KM).

Table 1 summarizes the experimental setting that has been considered. For each data-set, we report the optimal number of clusters $K$ and the size $n$ of the data-set, respectively. As for the ensemble, each algorithm was run several times in order to produce clusterings with different number of classes, $K_i$. For each clustering approach and each parametrization of the same we generated $N = 100$ different subsampled versions of the data-set.

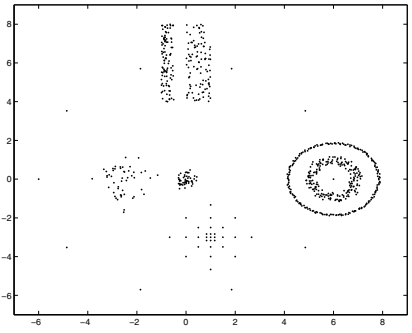


**Fig. 1.** Image Complex Synthetic data-set

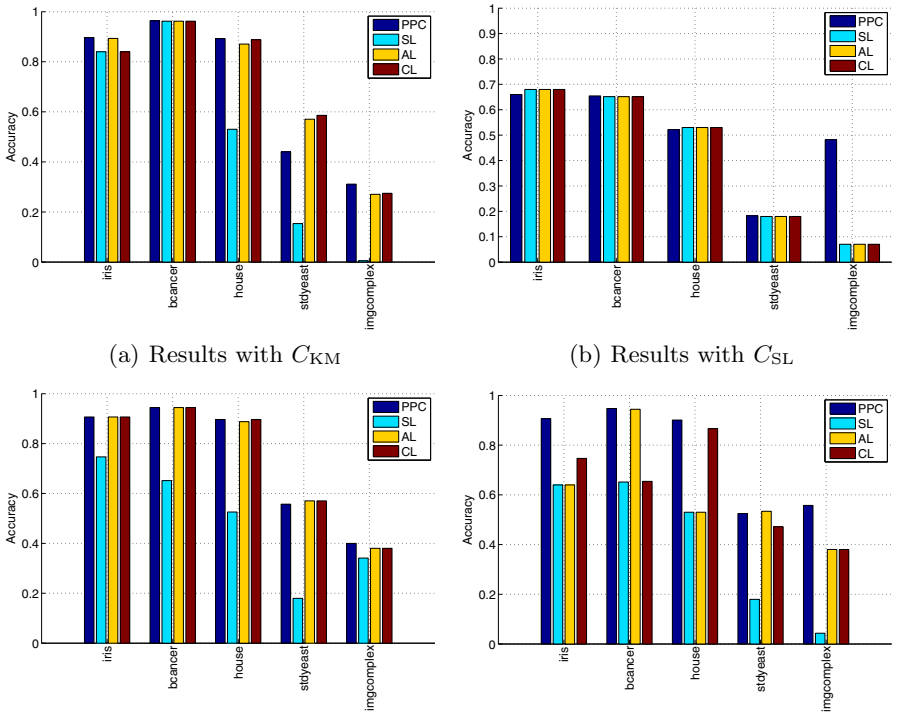
**Table 1.** Benchmark data-sets and parameter values used with different clustering algorithms (see text for description)

| Data-Sets | $K$ | $n$ | Ensemble |
|---|---|---|---|
| | | | $K_i$ |
| iris | 3 | 150 | 3-10,15,20 |
| house-votes | 2 | 232 | 2-10,15,20 |
| std-yeast-cell | 5 | 384 | 5-10,15,20 |
| breast-cancer | 2 | 683 | 2-10,15,20 |
| image-complex | 8 | 1000 | 8-15,20,30, 37 |

Once all the clusterings have been generated, we grouped them by algorithm into several *base ensembles*, namely $\mathcal{E}_{\mathrm{SL}}$, $\mathcal{E}_{\mathrm{AL}}$, $\mathcal{E}_{\mathrm{CL}}$ and $\mathcal{E}_{\mathrm{KM}}$. Moreover, we created a large ensemble $\mathcal{E}_{\mathrm{All}}$ from the union of all of them. For each ensemble we created a corresponding co-association matrix, namely $C_{\mathrm{SL}}$, $C_{\mathrm{AL}}$, $C_{\mathrm{CL}}$, $C_{\mathrm{KM}}$ and $C_{\mathrm{All}}$. For each of these co-association matrices, we applied our Pairwise Probabilistic Clustering (PPC) approach, and compared it against the performances obtained with the same matrices by the agglomerative hierarchical algorithms SL, AL and CL. Each method was provided with the optimal number of classes as input parameter.

Figure 2 summarizes the results obtained over the benchmark data-sets. The performances are assessed in terms of accuracy, i.e., the percentage of correct labels. When we consider the base ensembles, i.e., $\mathcal{E}_{\mathrm{SL}}$, $\mathcal{E}_{\mathrm{AL}}$, $\mathcal{E}_{\mathrm{CL}}$ and $\mathcal{E}_{\mathrm{KM}}$, on average our approach achieves the best results, although other approaches, such as the AL, perform comparably well. Our algorithm, however, outperforms the competitors when we take the union $\mathcal{E}_{\mathrm{All}}$ of all the base ensembles into account. Interestingly, the results obtained by PPC on the combined ensemble are as good as the best one obtained in the base ensembles and, in some cases like the image-complex dataset, they are even better.

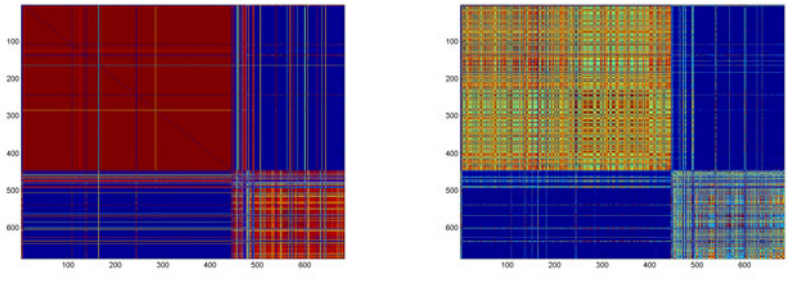
The different levels of performance obtained by the several algorithms over the different clustering ensembles, as shown in Figures 2(a) to 2(d), are illustrative of the distinctiveness between the underlying clustering ensembles, and the diversity of clustering solutions. It is then clear that the ensemble $\mathcal{E}_{\mathrm{All}}$ has




(a) Results with $C_{\mathrm{KM}}$ (b) Results with $C_{\mathrm{SL}}$

(c) Results with $C_{\mathrm{AL}}$ (d) Results with $C_{\mathrm{All}}$

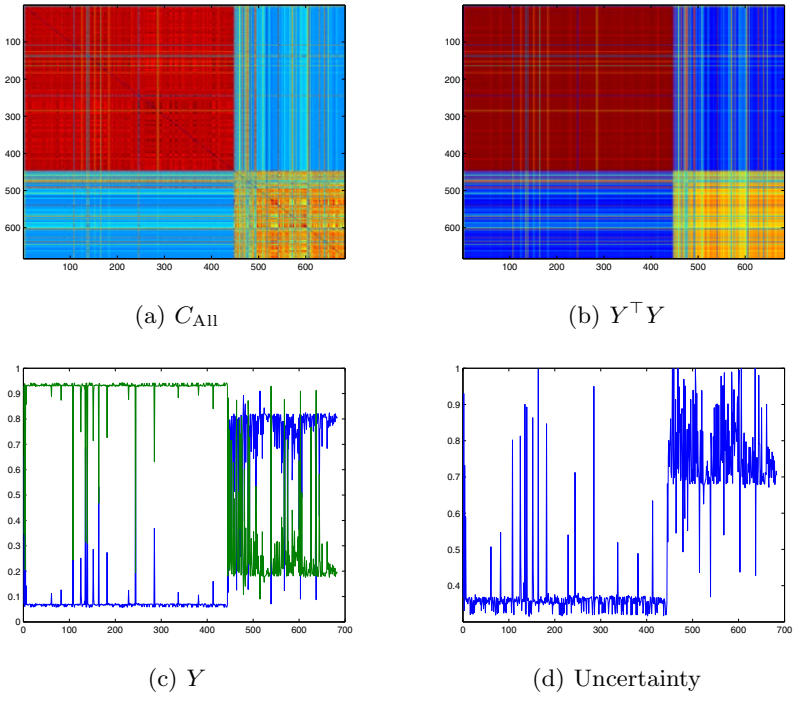
**Fig. 2.** Experiments on benchmark data-sets



(a) $C_{\mathrm{AL}}$ (b) $C_{KM}$

**Fig. 3.** Co-association matrices with ensembles $\mathcal{E}_{AL}$ and $\mathcal{E}_{KM}$

the largest diversity when compared to the individual ensembles; this is quantitatively confirmed when computing average pairwise consistency values between partitions in the individual CEs and the one resulting by the merging of these. This higher diversity causes the appearance of noisy-like structure in the



(a) $C_{\text{All}}$ (b) $Y^{\top}Y$

(c) $Y$ (d) Uncertainty

**Fig. 4.** Results on the breast-cancer data-set

co-association matrices. This is illustrated in Figures 3(a) and 3(b) corresponding to the co-association matrices $C_{\text{AL}}$ and $C_{\text{KM}}$, respectively, when compared to the $C_{\text{All}}$ in Figure 4(a). The better performance of the PPC algorithm on the latter CE, can be attributed to a leveraging effect over these local noisy estimates, thus better unveiling the underlying structure of the data. This is illustrated next.

Figures 4(a) and 4(b) show the empirical co-association matrix $C_{\text{All}}$ and the true one, respectively, for the breast-cancer data-set. While the block structure of two clusters is apparent in both figures, we can see that the true co-association turns out to be less noisy than the empirical one. In Figure 4(c) we plot the soft cluster assignments, $Y$. Here, object indices are on the x-axis, and probabilities are on the y-axis, each curve representing the profile of a cluster. As one can see from the cluster memberships, the two clusters can be clearly evinced, although there is a higher level of uncertainty in the assignments of objects belonging to the smallest cluster. Indeed, this can also be seen in Figure 4(d), where we plot the uncertainty $h_i$ in the cluster assignments, which is computed for each object $i$ as the normalized entropy of $\mathbf{y}_i$, i.e.,

$$h_i = -\frac{\sum_{k=1}^{K} y_{ki} \log(y_{ki})}{\log(K)} \,.$$

## 7 Conclusion

In this paper we introduced a new approach for consensus clustering. Taking advantage of the probabilistic interpretation of the computed similarities of the the co-association matrix, derived from the ensemble of clusterings, using the Evidence Accumulation Clustering, we propose a principled soft clustering method. Our method reduces the clustering problem to a polynomial optimization in probability domain, which is attacked by means of the Baum-Eagon inequality. Experiments on both synthetic and real benchmarks assess the effectiveness of our approach.

## Acknowledgement

We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250). This work was partially supported by the Portuguese Foundation for Science and Technology (FCT), Portuguese Ministry of Science and Technology, under grant PTDC/ EIACCO/ 103230/ 2008.

## References

1. Fred, A., Jain, A.K.: Combining multiple clustering using evidence accumulation. IEEE Trans. Pattern Anal. Machine Intell. 27(6), 835–850 (2005)
2. Jardine, N., Sibson, R.: The construction of hierarchic and non-hierarchic classifications. Computer J. 11, 177–184 (1968)
3. Banerjee, A., Krumpelman, C., Basu, S., Mooney, R.J., Ghosh, J.: Model-based overlapping clustering. In: Int. Conf. on Knowledge Discovery and Data Mining, pp. 532–537 (2005)
4. Heller, K., Ghahramani, Z.: A nonparametric bayesian approach to modeling overlapping clusters. In: Int. Conf. AI and Statistics (2007)
5. Zass, R., Shashua, A.: A unifying approach to hard and probabilistic clustering. In: Int. Conf. Comp. Vision (ICCV), vol. 1, pp. 294–301 (2005)
6. Baum, L.E., Eagon, J.A.: An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. Bull. Amer. Math. Soc. 73, 360–363 (1967)
7. Fred, A., Jain, A.K.: Learning pairwise similarity for data clustering. In: Int. Conf. Patt. Recogn. (ICPR), pp. 925–928 (2006)
8. Fred, A., Jain, A.K.: Data clustering using evidence accumulation. In: Int. Conf. Patt. Recogn. (ICPR), pp. 276–280 (2002)
9. Zass, R., Shashua, A.: Doubly stochastic normalization for spectral clustering. In: Adv. in Neural Inform. Proces. Syst (NIPS), vol. 19, pp. 1569–1576 (2006)
10. Baum, L.E., Sell, G.R.: Growth transformations for functions on manifolds. Pacific J. Math. 27, 221–227 (1968)
11. Baum, L.E., Petrie, T., Soules, G., Weiss, N.: A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Ann. Math. Statistics 41, 164–171 (1970)
12. Jain, A.K., Dubes, R.C.: Algorithms for data clustering. Prentice-Hall, Englewood Cliffs (1988)

# A Game-Theoretic Approach to Hypergraph Clustering

**Samuel Rota Bulò** **Marcello Pelillo**
University of Venice, Italy
{srotabul,pelillo}@dsi.unive.it

## Abstract

Hypergraph clustering refers to the process of extracting maximally coherent groups from a set of objects using high-order (rather than pairwise) similarities. Traditional approaches to this problem are based on the idea of partitioning the input data into a user-defined number of classes, thereby obtaining the clusters as a by-product of the partitioning process. In this paper, we provide a radically different perspective to the problem. In contrast to the classical approach, we attempt to provide a meaningful formalization of the very notion of a cluster and we show that game theory offers an attractive and unexplored perspective that serves well our purpose. Specifically, we show that the hypergraph clustering problem can be naturally cast into a non-cooperative multi-player "clustering game", whereby the notion of a cluster is equivalent to a classical game-theoretic equilibrium concept. From the computational viewpoint, we show that the problem of finding the equilibria of our clustering game is equivalent to locally optimizing a polynomial function over the standard simplex, and we provide a discrete-time dynamics to perform this optimization. Experiments are presented which show the superiority of our approach over state-of-the-art hypergraph clustering techniques.

## 1 Introduction

Clustering is the problem of organizing a set of objects into groups, or *clusters*, in a way as to have similar objects grouped together and dissimilar ones assigned to different groups, according to some similarity measure. Unfortunately, there is no universally accepted formal definition of the notion of a cluster, but it is generally agreed that, informally, a cluster should correspond to a set of objects satisfying two conditions: an *internal coherency* condition, which asks that the objects belonging to the cluster have high mutual similarities, and an *external incoherency* condition, which states that the overall cluster internal coherency decreases by adding to it any external object.

Objects similarities are typically expressed as pairwise relations, but in some applications higher-order relations are more appropriate, and approximating them in terms of pairwise interactions can lead to substantial loss of information. Consider for instance the problem of clustering a given set of $d$-dimensional Euclidean points into lines. As every pair of data points trivially defines a line, there does not exist a meaningful pairwise measure of similarity for this problem. However, it makes perfect sense to define similarity measures over triplets of points that indicate how close they are to being collinear. Clearly, this example can be generalized to any problem of model-based point pattern clustering, where the deviation of a set of points from the model provides a measure of their dissimilarity. The problem of clustering objects using high-order similarities is usually referred to as the *hypergraph clustering problem*.

In the machine learning community, there has been increasing interest around this problem. Zien and co-authors [24] propose two approaches called "clique expansion" and "star expansion", respectively. Both approaches transform the similarity hypergraph into an edge-weighted graph, whose edge-weights are a function of the hypergraph's original weights. This way they are able to tackle

the problem with standard pairwise clustering algorithms. Bolla [6] defines a Laplacian matrix for an unweighted hypergraph and establishes a link between the spectral properties of this matrix and the hypergraph's minimum cut. Rodrìguez [16] achieves similar results by transforming the hypergraph into a graph according to "clique expansion" and shows a relationship between the spectral properties of a Laplacian of the resulting matrix and the cost of minimum partitions of the hypergraph. Zhou and co-authors [23] generalize their earlier work on regularization on graphs and define a hypergraph normalized cut criterion for a $k$-partition of the vertices, which can be achieved by finding the second smallest eigenvector of a normalized Laplacian. This approach generalizes the well-known "Normalized cut" pairwise clustering algorithm [19]. Finally, in [2] we find another work based on the idea of applying a spectral graph partitioning algorithm on an edge-weighted graph, which approximates the original (edge-weighted) hypergraph. It is worth noting that the approaches mentioned above are devised for dealing with higher-order relations, but can all be reduced to standard pairwise clustering approaches [1]. A different formulation is introduced in [18], where the clustering problem with higher-order (super-symmetric) similarities is cast into a nonnegative factorization of the closest hyper-stochastic version of the input affinity tensor.

All the afore-mentioned approaches to hypergraph clustering are partition-based. Indeed, clusters are not modeled and sought directly, but they are obtained as a by-product of the partition of the input data into a fixed number of classes. This renders these approaches vulnerable to applications where the number of classes is not known in advance, or where data is affected by clutter elements which do not belong to any cluster (as in figure/ground separation problems). Additionally, by partitioning, clusters are necessarily disjoint sets, although it is in many cases natural to have overlapping clusters, e.g., two intersecting lines have the point in the intersection belonging to both lines.

In this paper, following [14, 20] we offer a radically different perspective to the hypergraph clustering problem. Instead of insisting on the idea of determining a partition of the input data, and hence obtaining the clusters as a by-product of the partitioning process, we reverse the terms of the problem and attempt instead to derive a rigorous formulation of the very notion of a cluster. This allows one, in principle, to deal with more general problems where clusters may overlap and/or outliers may get unassigned. We found that game theory offers a very elegant and general mathematical framework that serves well our purposes. The basic idea behind our approach is that the hypergraph clustering problem can be considered as a multi-player non-cooperative "clustering game". Within this context, the notion of a cluster turns out to be equivalent to a classical equilibrium concept from (evolutionary) game theory, as the latter reflects both the internal and external cluster conditions alluded to before. We also show that there exists a correspondence between these equilibria and the local solutions of a polynomial, linearly-constrained, optimization problem, and provide an algorithm for finding them. Experiments on two standard hypergraph clustering problems show the superiority of the proposed approach over state-of-the-art hypergraph clustering techniques.

## 2 Basic notions from evolutionary game theory

Evolutionary game theory studies models of strategic interactions (called *games*) among large numbers of anonymous agents. A game can be formalized as a triplet $\Gamma = (P, S, \pi)$, where $P = \{1, \ldots, k\}$ is the set of players involved in the game, $S = \{1, \ldots, n\}$ is the set of *pure strategies* (in the terminology of game-theory) available to each player and $\pi : S^k \rightarrow \mathbb{R}$ is the *payoff function*, which assigns a payoff to each *strategy profile*, i.e., the (ordered) set of pure strategies played by the individuals. The payoff function $\pi$ is assumed to be invariant to permutations of the strategy profile. It is worth noting that in general games, each player may have its own set of strategies and own payoff function. For a comprehensive introduction to evolutionary game theory we refer to [22].

By undertaking an evolutionary setting we assume to have a large population of non-rational agents, which are randomly matched to play a game $\Gamma = (P, S, \pi)$. Agents are considered non-rational, because each of them initially chooses a strategy from $S$, which will be always played when selected for the game. An agent, who selected strategy $i \in S$, is called *$i$-strategist*. Evolution in the population takes place, because we assume that there exists a selection mechanism, which, by analogy with a Darwinian process, spreads the fittest strategies in the population to the detriment of the weakest one, which will in turn be driven to extinction. We will see later in this work a formalization of such a selection mechanism.

The state of the population at a given time $t$ can be represented as a $n$-dimensional vector $\mathbf{x}(t)$, where $x_i(t)$ represents the fraction of $i$-strategists in the population at time $t$. The set of all possible states describing a population is given by

$$\Delta = \left\{ \mathbf{x} \in \mathbb{R}^n \, : \, \sum_{i \in S} x_i = 1 \text{ and } x_i \geq 0 \text{ for all } i \in S \right\},$$

which is called *standard simplex*. In the sequel we will drop the time reference from the population state, where not necessary. Moreover, we denote with $\sigma(\mathbf{x})$ the *support* of $\mathbf{x} \in \Delta$, i.e., the set of strategies still alive in population $\mathbf{x} \in \Delta$: $\sigma(\mathbf{x}) = \{i \in S \, : \, x_i > 0\}$.

If $\mathbf{y}^{(i)} \in \Delta$ is the probability distribution identifying which strategy the $i$th player will adopt if drawn to play the game $\Gamma$, then the average payoff obtained by the agents can be computed as

$$u(\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(k)}) = \sum_{(s_1, \ldots, s_k) \in S^k} \pi(s_1, \ldots, s_k) \prod_{j=1}^{k} y_{s_j}^{(j)} \, . \tag{1}$$

Note that (1) is invariant to any permutation of the input probability vectors.

Assuming that the agents are randomly and independently drawn from a population $\mathbf{x} \in \Delta$ to play the game $\Gamma$, the population average payoff is given by $u(\mathbf{x}^k)$, where $\mathbf{x}^k$ is a shortcut for $\mathbf{x}, \ldots, \mathbf{x}$ repeated $k$ times. Furthermore, the average payoff that an $i$-strategist obtains in a population $\mathbf{x} \in \Delta$ is given by $u(\mathbf{e}^i, \mathbf{x}^{k-1})$, where $\mathbf{e}^i \in \Delta$ is a vector with $x_i = 1$ and zero elsewhere.

An important notion in game theory is that of equilibrium [22]. A population $\mathbf{x} \in \Delta$ is in equilibrium when the distribution of strategies will not change anymore, which intuitively happens when every individual in the population obtains the same average payoff and no strategy can thus prevail on the other ones. Formally, $\mathbf{x} \in \Delta$ is a *Nash equilibrium* if

$$u(\mathbf{e}^i, \mathbf{x}^{k-1}) \leq u(\mathbf{x}^k) \, , \qquad \text{for all } i \in S \, . \tag{2}$$

In other words, every agent in the population performs at most as well as the population average payoff. Due to the multi-linearity of $u$, a consequence of (2) is that

$$u(\mathbf{e}^i, \mathbf{x}^{k-1}) = u(\mathbf{x}^k) \, , \qquad \text{for all } i \in \sigma(\mathbf{x}) \, , \tag{3}$$

i.e., all the agents that survived the evolution obtain the same average payoff, which coincides with the population average payoff.

A key concept pertaining to evolutionary game theory is that of an evolutionary stable strategy [7, 22]. Such a strategy is robust to evolutionary pressure in an exact sense. Assume that in a population $\mathbf{x} \in \Delta$, a small share $\epsilon$ of mutant agents appears, whose distribution of strategies is $\mathbf{y} \in \Delta$. The resulting postentry population is given by $\mathbf{w}_\epsilon = (1 - \epsilon)\mathbf{x} + \epsilon \mathbf{y}$. Biological intuition suggests that evolutionary forces select against mutant individuals if and only if the average payoff of a mutant agent in the postentry population is lower than that of an individual from the original population, i.e.,

$$u(\mathbf{y}, \mathbf{w}_\epsilon^{k-1}) < u(\mathbf{x}, \mathbf{w}_\epsilon^{k-1}) \, . \tag{4}$$

A population $\mathbf{x} \in \Delta$ is *evolutionary stable* (or an ESS) if inequality (4) holds for any distribution of mutant agents $\mathbf{y} \in \Delta \setminus \{\mathbf{x}\}$, granted the population share of mutants $\epsilon$ is sufficiently small (see, [22] for pairwise contests and [7] for $n$-wise contests).

An alternative, but equivalent, characterization of ESSs involves a leveled notion of evolutionary stable strategies [7]. We say that $\mathbf{x} \in \Delta$ is an *ESS of level* $j$ against $\mathbf{y} \in \Delta$, if there exists $j \in \{0, \ldots, k-1\}$ such that both conditions

$$u(\mathbf{y}^{j+1}, \mathbf{x}^{k-j-1}) \;\; < \;\; u(\mathbf{y}^j, \mathbf{x}^{k-j}) \, , \tag{5}$$

$$u(\mathbf{y}^{i+1}, \mathbf{x}^{k-i-1}) \;\; = \;\; u(\mathbf{y}^i, \mathbf{x}^{k-i}) \, , \quad \text{for all } 0 \leq i < j \, , \tag{6}$$

are satisfied. Clearly, $\mathbf{x} \in \Delta$ is an ESS if it satisfies a condition of this form for every $\mathbf{y} \in \Delta \setminus \{\mathbf{x}\}$. It is straightforward to see that any ESS is a Nash equilibrium [22, 7]. An ESS, which satisfies conditions (6) with $j$ never more than $J$, will be called an *ESS of level* $J$. Note that for the generic case most of the preceding conditions will be superfluous, i.e., only ESSs of level 0 or 1 are required [7]. Hence, in the sequel, we will consider only ESSs of level 1. It is not difficult to verify that any ESS (of level 1) $\mathbf{x} \in \Delta$ satisfies

$$u(\mathbf{w}_\epsilon^k) < u(\mathbf{x}^k) \, , \tag{7}$$

for all $\mathbf{y} \in \Delta \setminus \{\mathbf{x}\}$ and small enough values of $\epsilon$.

## 3 The hypergraph clustering game

The hypergraph clustering problem can be described by an edge-weighted hypergraph. Formally, an edge-weighted *hypergraph* is a triplet $H = (V, E, s)$, where $V = \{1, \ldots, n\}$ is a finite set of *vertices*, $E \subseteq \mathcal{P}(V) \setminus \{\emptyset\}$ is the set of (hyper-)edges (here, $\mathcal{P}(V)$ is the power set of $V$) and $s : E \to \mathbb{R}$ is a weight function which associates a real value with each edge. Note that negative weights are allowed too. Although hypergraphs may have edges of varying cardinality, we will focus on a particular class of hypergraphs, called $k$-graphs, whose edges have all fixed cardinality $k \geq 2$.

In this paper, we cast the hypergraph clustering problem into a game, called *(hypergraph) clustering game*, which will be played in an evolutionary setting. Clusters are then derived from the analysis of the ESSs of the clustering game. Specifically, given a $k$-graph $H = (V, E, s)$ modeling a hypergraph clustering problem, where $V = \{1, \ldots, n\}$ is the set of objects to cluster and $s$ is the similarity function over the set of objects in $E$, we can build a game involving $k$ players, each of them having the same set of (pure) strategies, namely the set of objects to cluster $V$. Under this setting, a population $\mathbf{x} \in \Delta$ of agents playing a clustering game represents in fact a cluster, where $x_i$ is the probability for object $i$ to be part of it. Indeed, any cluster can be modeled as a probability distribution over the set of objects to cluster. The payoff function of the clustering game is defined in a way as to favour the evolution of agents supporting highly coherent objects. Intuitively, this is accomplished by rewarding the $k$ players in proportion to the similarity that the $k$ played objects have. Hence, assuming $(v_1, \ldots, v_k) \in V^k$ to be the tuple of objects selected by $k$ players, the payoff function can be simply defined as

$$\pi(v_1, \ldots, v_k) = \begin{cases} \frac{1}{k!} s\left(\{v_1, \ldots, v_k\}\right) & \text{if } \{v_1, \ldots, v_k\} \in E\,, \\ 0 & \text{else}\,, \end{cases} \tag{8}$$

where the term $1/k!$ has been introduced for technical reasons.

Given a population $\mathbf{x} \in \Delta$ playing the clustering game, we have that the average population payoff $u(\mathbf{x}^k)$ measures the cluster's internal coherency as the average similarity of the objects forming the cluster, whereas the average payoff $u(\mathbf{e}^i, \mathbf{x}^{k-1})$ of an agent supporting object $i \in V$ in population $\mathbf{x}$, measures the average similarity of object $i$ with respect to the cluster.

An ESS of a clustering game incorporates the properties of internal coherency and external incoherency of a cluster:

**internal coherency:** since ESSs are Nash equilibria, from (3), it follows that every object contributing to the cluster, i.e., every object in $\sigma(\mathbf{x})$, has the same average similarity with respect to the cluster, which in turn corresponds to the cluster's overall average similarity. Hence, the cluster is internally coherent;

**external incoherency:** from (2), every object external to the cluster, i.e., every object in $V \setminus \sigma(\mathbf{x})$, has an average similarity which does not exceed the cluster's overall average similarity. There may still be cases where the average similarity of an external object is the same as that of an internal object, mining the cluster's external incoherency. However, since $\mathbf{x}$ is an ESS, from (7) we see that whenever we try to extend a cluster with small shares of external objects, the cluster's overall average similarity drops. This guarantees the external incoherency property of a cluster to be also satisfied.

Finally, it is worth noting that this theory generalizes the dominant-sets clustering framework which has recently been introduced in [14]. Indeed, ESSs of pairwise clustering games, i.e. clustering games defined on graphs, correspond to the dominant-set clusters [20, 17]. This is an additional evidence that ESSs are meaningful notions of cluster.

## 4 Evolution towards a cluster

In this section we will show that the ESSs of a clustering game are in one-to-one correspondence with (strict) local solution of a non-linear optimization program. In order to find ESSs, we will also provide a dynamics due to Baum and Eagon, which generalizes the replicator dynamics [22].

Let $H = (V, E, s)$ be a hypergraph clustering problem and $\Gamma = (P, V, \pi)$ be the corresponding clustering game. Consider the following non-linear optimization problem:

$$\text{maximize} \quad f(\mathbf{x}) = \sum_{e \in E} s(e) \prod_{i \in e} x_i\,, \quad \text{subject to} \quad \mathbf{x} \in \Delta\,. \tag{9}$$

It is simple to see that any first-order Karush-Kuhn-Tucker (KKT) point $\mathbf{x} \in \Delta$ of program (9) [13] is a Nash equilibrium of $\Gamma$. Indeed, by the KKT conditions there exist $\mu_i \geq 0$, $i \in S$, and $\lambda \in \mathbb{R}$ such that for all $i \in S$,

$$\nabla f(\mathbf{x})_i + \mu_i - \lambda = \frac{1}{k} u(\mathbf{e}^i, \mathbf{x}^{k-1}) + \mu_i - \lambda = 0 \quad \text{and} \quad \mu_i x_i = 0\,,$$

where $\nabla$ is the gradient operator. From this it follows straightforwardly that $u(\mathbf{e}^i, \mathbf{x}^{k-1}) \leq u(\mathbf{x}^k)$ for all $i \in S$. Moreover, it turns out that any strict local maximizer $\mathbf{x} \in \Delta$ of (9) is an ESS of $\Gamma$. Indeed, by definition, a strict local maximizer of this program satisfies $u(\mathbf{z}^k) = f(\mathbf{z}) < f(\mathbf{x}) = u(\mathbf{x}^k)$, for any $\mathbf{z} \in \Delta \setminus \{\mathbf{x}\}$ close enough to $\mathbf{x}$, which is in turn equivalent to (7) for sufficiently small values of $\epsilon$.

The problem of extracting ESSs of our hypergraph clustering game can thus be cast into the problem of finding strict local solutions of (9). We will address this optimization task using a result due to Baum and Eagon [3], who introduced a class of nonlinear transformations in probability domain.

**Theorem 1** (Baum-Eagon)**.** *Let $P(\mathbf{x})$ be a homogeneous polynomial in the variables $x_i$ with non-negative coefficients, and let $\mathbf{x} \in \Delta$. Define the mapping $\mathbf{z} = \mathcal{M}(\mathbf{x})$ as follows:*

$$z_i = x_i \partial_i P(\mathbf{x}) \Big/ \sum_{j=1}^{n} x_j \partial_j P(\mathbf{x}), \qquad i = 1, \ldots, n. \tag{10}$$

*Then $P(\mathcal{M}(\mathbf{x})) > P(\mathbf{x})$, unless $\mathcal{M}(\mathbf{x}) = \mathbf{x}$. In other words $\mathcal{M}$ is a growth transformation for the polynomial $P$.*

The Baum-Eagon inequality provides an effective iterative means for maximizing polynomial functions in probability domains, and in fact it has served as the basis for various statistical estimation techniques developed within the theory of probabilistic functions of Markov chains [4]. It was also employed for the solution of relaxation labelling processes [15].

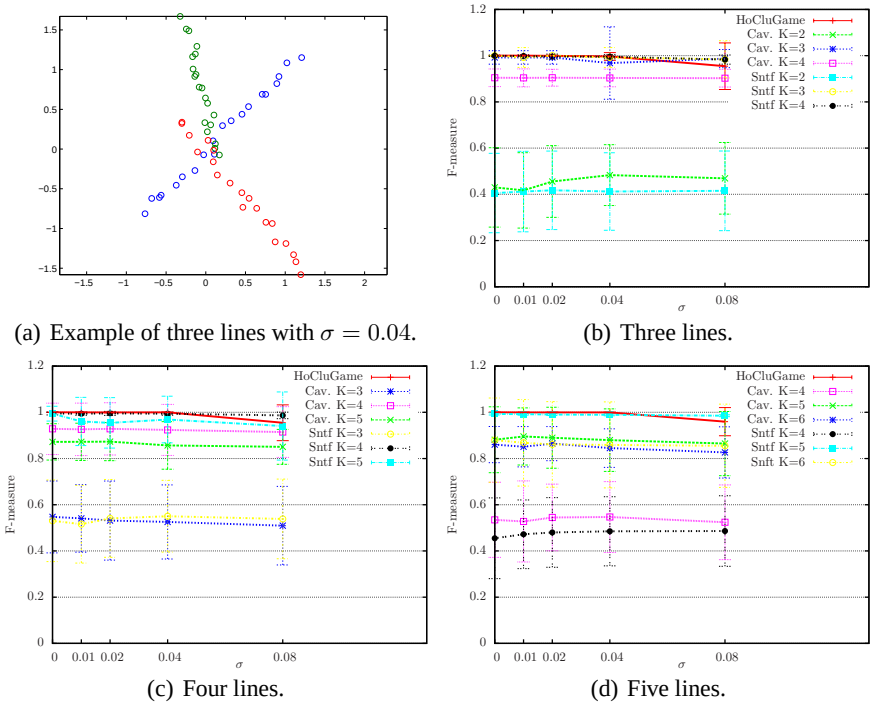
Since $f(\mathbf{x})$ is a homogeneous polynomial in the variables $x_i$, we can use the transformation of Theorem 1 in order to find a local solution $\mathbf{x} \in \Delta$ of (9), which in turn provides us with an ESS of the hypergraph clustering game. By taking the support of $\mathbf{x}$, we have a cluster under our framework. The complexity of finding a cluster is thus $O(\rho|E|)$, where $|E|$ is the number of edges of the hypergraph describing the clustering problem and $\rho$ is the average number of iteration needed to converge. Note that $\rho$ never exceeded $100$ in our experiments.

In order to obtain the clustering, in principle, we have to find the ESSs of the clustering game. This is a non-trivial, although still feasible, task [21], which we leave as a future extension of this work. By now, we adopt a naive *peeling-off strategy* for our cluster extraction procedure. Namely, we iteratively find a cluster and remove it from the set of objects, and we repeat this process on the remaining objects until a desired number of clusters have been extracted. The set of extracted ESSs with this procedure does not technically correspond to the ESSs of the original game, but to ESSs of sub-games of it. The cost of this approximation is that we unfortunately loose (by now) the possibility of having overlapping clusters.

## 5 Experiments

In this section we present two types of experiments. The first one addresses the problem of line clustering, while the second one addresses the problem of illuminant-invariant face clustering. We tested our approach against Clique Averaging algorithm (CAVERAGE), since it was the best performing approach in [2] on the same type of experiments. Specifically, CAVERAGE outperformed Clique Expansion [10] combined with Normalized cuts, Gibson's Algorithm under sum and product model [9], kHMeTiS [11] and Cascading RANSAC [2]. We also compare against Super-symmetric Non-negative Tensor Factorization (SNTF) [18], because it is the only approach, other than ours, which does not approximate the hypergraph to a graph.

Since both CAVERAGE and SNTF, as opposed to our method, require the number of classes $K$ to be specified, we run them with values of $K \in \{K^* - 1, K^*, K^* + 1\}$ among which the optimal one ($K^*$) is present. This allows us to verify the robustness of the approaches under wrong values of $K$, which may occur in general as the optimal number of clusters is not known in advance.




(a) Example of three lines with $\sigma = 0.04$.

(b) Three lines.

(c) Four lines.

(d) Five lines.

Figure 1: Results on clustering $3, 4$ and $5$ lines perturbed with increasing levels of Gaussian noise ($\sigma = 0, 0.01, 0.02, 0.04, 0.08$).

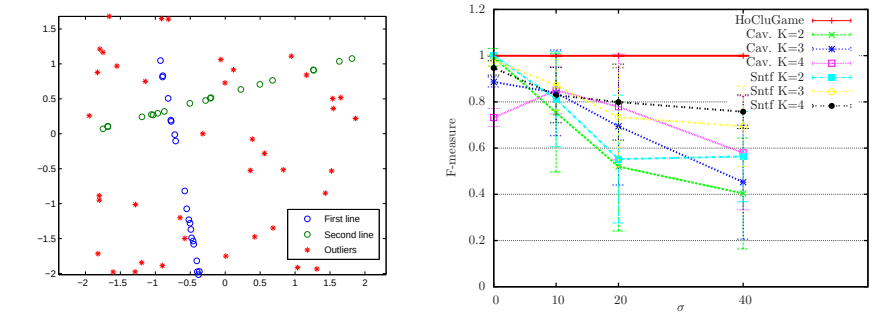
We executed the experiments on a AMD Sempron 3Ghz computer with 1Gb RAM. Moreover, we evaluated the quality of a clustering by computing the average F-measure of each cluster in the ground-truth with the most compatible one in the obtained solution (according to a one-to-one correspondence).

### 5.1 Line clustering

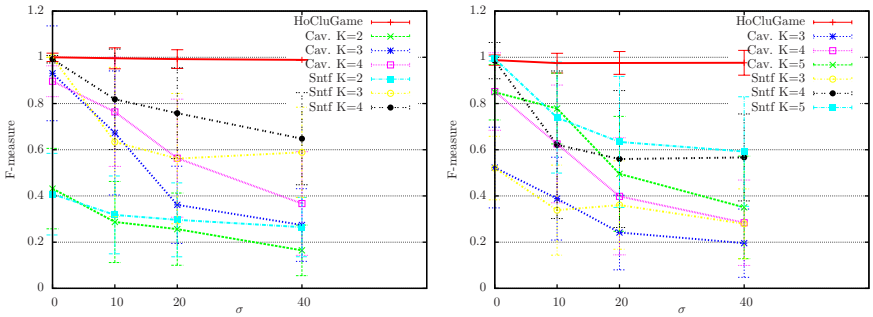
We consider the problem of clustering lines in spaces of dimension greater than two, i.e., given a set of points in $\mathbb{R}^d$, the task is to find sets of collinear points. Pairwise measures of similarity are useless and at least three points are needed. The dissimilarity measure on triplets of points is given by their mean distance to the best fitting line. If $d(i, j, k)$ is the dissimilarity of points $\{i, j, k\}$, the similarity function is given by $s(\{i, j, k\}) = \exp(-d(i, j, k)^2/\sigma^2)$, where $\sigma$ is a scaling parameter, which has been optimally selected for all the approaches according to a small test set.

We conducted two experiments, in order to assess the robustness of the approaches to both local and global noise. Local noise refers to a Gaussian perturbation applied to the points of a line, while global noise consists of random outlier points.

A first experiment consists in clustering $3$, $4$ and $5$ lines generated in the $5$-dimensional space $[-2, 2]^5$. Each line consists of $20$ points, which have been perturbed according to $5$ increasing levels of Gaussian noise, namely $\sigma = 0, 0.01, 0.02, 0.04, 0.08$. With this setting there are no outliers and every point should be assigned to a line (e.g., see Figure 1(a)). Figure 1(b) shows the results obtained with three lines. We reported, for each noise level, the mean and the standard deviation of the average F-measures obtained by the algorithms on 30 randomly generated instances. Note that, if the optimal $K$ is used, CAVERAGE and SNTF perform well and the influence of local noise is minimal. This behavior intuitively makes sense under moderate perturbations, because if the approaches correctly partitioned the data without noise, it is unlikely that the result will change by slightly perturbing them. Our approach however achieves good performances as well, although we can notice that with the highest noise level, the performance slightly drops. This is due to the fact that points deviating too much from the overall cluster average collinearity will be excluded as they undermine the cluster's internal coherency. Hence, some perturbed points will be considered outliers. Nevertheless, it is worth noting that by underestimating the optimal number of classes both CAVERAGE and SNTF exhibit a drastic performance drop, whereas the influence of overestimations




(a) Example of two lines with 40 outliers. (b) Two lines.




(c) Three lines. (d) Four lines.

Figure 2: Results on clustering $2, 3$ and $4$ lines with an increasing number of outliers $(0, 10, 20, 40)$.

has a lower impact on the two partition-based algorithms. By increasing the number of lines involved in the experiment from three to four (Figure 1(c)) and to five (Figure 1(d)) the scenario remains almost the same for our approach and SNTF, while we can notice a slight decrease of CAVERAGE's performance.

The second experiment consists in clustering $2$, $3$ and $4$ slightly perturbed lines (with fixed local noise $\sigma = 0.01$) generated in the 5-dimensional space $[-2, 2]^5$. Again, each line consists of $20$ points. This time however we added also global noise, i.e., $0$, $10$, $20$ and $40$ random points as outliers (e.g., see Figure 2(a)). Figure 2(b) shows the results obtained with two lines. Here, the supremacy of our approach over partition-based ones is clear. Indeed, our method is not influenced by outliers and therefore it performs almost perfectly, whereas CAVERAGE and SNTF perform well only without outliers and with the optimal $K$. It is interesting to notice that, as outliers are introduced, CAVERAGE and SNTF perform better with $K > 2$. Indeed, the only way to get rid of outliers is to group them in additional clusters. However, since outliers are not mutually similar and intuitively they do not form a cluster, we have that the performance of CAVERAGE and SNTF drop drastically as the number of outliers increases. Finally, by increasing the number of lines from two to three (Figure 2(c)) and to four (Figure 2(d)), the performance of CAVERAGE and SNTF get worse, while our approach still achieves good results.

### 5.2 Illuminant-invariant face clustering

In [5] it has been shown that images of a Lambertian object illuminated by a point light source lie in a three dimensional subspace. According to this result, if we assume that four images of a face form the columns of a matrix then $d = s_4^2/(s_1^2 + \cdots + s_4^2)$ provides us with a measure of dissimilarity, where $s_i$ is the $i$th singular value of this matrix [2]. We use this dissimilarity measure for the face clustering problem and we consider as dataset the Yale Face Database B and its extended version [8, 12]. In total we have faces of 38 individuals, each under 64 different illumination conditions. We compared our approach against CAVERAGE and SNTF on subsets of this face dataset. Specifically, we considered cases where we have faces from 4 and 5 random individuals (10 faces per individual), and with and without outliers. The case with outliers consists in 10 additional faces each from a different individual. For each of those combinations, we created 10 random subsets. Similarly to the case of line clustering, we run CAVERAGE and SNTF with values of $K \in \{K^* - 1, K^*, K^* + 1\}$, where $K^*$ is the optimal one.

| n. of classes: | 4 | | 5 | |
|---|---|---|---|---|
| n. of outliers: | 0 | 10 | 0 | 10 |
| CAVERAGE K=3 | 0.63±0.11 | 0.59±0.07 | - | - |
| CAVERAGE K=4 | **0.96±0.06** | 0.84±0.07 | 0.56±0.14 | 0.58±0.07 |
| CAVERAGE K=5 | 0.91±0.06 | 0.79±0.05 | 0.85±0.12 | 0.83±0.06 |
| CAVERAGE K=6 | - | - | 0.84±0.09 | 0.82±0.06 |
| SNTF K=3 | 0.62±0.12 | 0.58±0.10 | - | - |
| SNTF K=4 | 0.87±0.07 | 0.81±0.08 | 0.61±0.13 | 0.59±0.09 |
| SNTF K=5 | 0.82±0.09 | 0.76±0.09 | 0.86±0.12 | 0.80±0.07 |
| SNTF K=6 | - | - | 0.85±0.08 | 0.79±0.11 |
| HoCluGame | 0.95±0.03 | **0.94±0.02** | **0.95±0.05** | **0.94±0.02** |

Table 1: Experiments on illuminant-invariant face clustering.

In Table 1 we report the average F-measures (mean and standard deviation) obtained by the three approaches. The results are consistent with those obtained in the case of line clustering with the exception of SNTF, which performs worse than the other approaches on this real-world application. CAVERAGE and our algorithm perform comparably well when clustering 4 individuals without outliers. However, our approach turns out to be more robust in every other tested case, i.e., when the number of classes increases and when outliers are introduced. Indeed, CAVERAGE's performance decreases, while our approach yields the same good results.

In both the experiments of line and face clustering the execution times of our approach were higher than those of CAVERAGE, but considerably lower than SNTF. The main reason why CAVERAGE run faster is that our approach and SNTF work directly on the hypergraph without resorting to pairwise relations, which is indeed what CAVERAGE does. Further, we mention that our code was not optimized to improve speed and all the approaches were run without any sampling policy.

## 6 Discussion

In this paper, we offered a game-theoretic perspective to the hypergraph clustering problem. Within our framework the clustering problem is viewed as a multi-player non-cooperative game, and classical equilibrium notions from evolutionary game theory turn out to provide a natural formalization of the notion of a cluster. We showed that the problem of finding these equilibria (clusters) is equivalent to solving a polynomial optimization problem with linear constraints, which we solve using an algorithm based on the Baum-Eagon inequality. An advantage of our approach over traditional techniques is the independence from the number of clusters, which is indeed an intrinsic characteristic of the input data, and the robustness against outliers, which is especially useful when solving figure-ground-like grouping problems. We also mention, as a potential positive feature of the proposed approach, the possibility of finding overlapping clusters (e.g., along the lines presented in [21]), although in this paper we have not explicitly dealt with this problem. The experimental results show the superiority of our approach with respect to the state of the art in terms of quality of solution. We are currently studying alternatives to the plain Baum-Eagon dynamics in order to improve efficiency.

**Acknowledgments.** We acknowledge financial support from the FET programme within EU FP7, under the SIMBAD project (contract 213250). We also thank Sameer Agarwal and Ron Zass for providing us with the code of their algorithms.

## References

[1] S. Agarwal, K. Branson, and S. Belongie. Higher order learning with graphs. In *Int. Conf. on Mach. Learning*, volume 148, pages 17–24, 2006.

[2] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman, and S. Belongie. Beyond pairwise clustering. In *IEEE Conf. Computer Vision and Patt. Recogn.*, volume 2, pages 838–845, 2005.

[3] L. E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bull. Amer. Math. Soc.*, 73:360–363, 1967.

[4] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Statistics*, 41:164–171, 1970.

[5] P. Belhumeur and D. Kriegman. What is the set of images of an object under all possible lighting conditions. *Int. J. Comput. Vision*, 28(3):245–260, 1998.

[6] M. Bolla. Spectral, euclidean representations and clusterings of hypergraphs. *Discr. Math.*, 117:19–39, 1993.

[7] M. Broom., C. Cannings, and G. T. Vickers. Multi-player matrix games. *Bull. Math. Biology*, 59(5):931–952, 1997.

[8] A. S. Georghiades., P. N. Belhumeur, and D. J. Kriegman. From few to many: illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Machine Intell.*, 23(6):643–660, 2001.

[9] D. Gibson, J. M. Kleinberg, and P. Raghavan. *VLDB*, chapter Clustering categoral data: An approach based on dynamical systems., pages 311–322. Morgan Kaufmann Publishers Inc., 1998.

[10] T. Hu and K. Moerder. Multiterminal flows in hypergraphs. In T. Hu and E. S. Kuh, editors, *VLSI circuit layout: theory and design*, pages 87–93. 1985.

[11] G. Karypis and V. Kumar. Multilevel k-way hypergraph partitioning. *VLSI Design*, 11(3):285–300, 2000.

[12] K. C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Machine Intell.*, 27(5):684–698, 2005.

[13] D. G. Luenberger. *Linear and nonlinear programming*. Addison Wesley, 1984.

[14] M. Pavan and M. Pelillo. Dominant sets and pairwise clustering. *IEEE Trans. Pattern Anal. Machine Intell.*, 29(1):167–172, 2007.

[15] M. Pelillo. The dynamics of nonlinear relaxation labeling processes. *J. Math. Imag. and Vision*, 7(4):309–323, 1997.

[16] J. Rodrìguez. On the Laplacian spectrum and walk-regular hypergraphs. *Linear and Multilinear Algebra*, 51:285–297, 2003.

[17] S. Rota Bulò. *A game-theoretic framework for similarity-based data clustering*. PhD thesis, University of Venice, 2009.

[18] A. Shashua, R. Zass, and T. Hazan. Multi-way clustering using super-symmetric non-negative tensor factorization. In *Europ. Conf. on Comp. Vision*, volume 3954, pages 595–608, 2006.

[19] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 22:888–905, 2000.

[20] A. Torsello, S. Rota Bulò, and M. Pelillo. Grouping with asymmetric affinities: a game-theoretic perspective. In *IEEE Conf. Computer Vision and Patt. Recogn.*, pages 292–299, 2006.

[21] A. Torsello, S. Rota Bulò, and M. Pelillo. Beyond partitions: allowing overlapping groups in pairwise clustering. In *Int. Conf. Patt. Recogn.*, 2008.

[22] J. W. Weibull. *Evolutionary game theory*. Cambridge University Press, 1995.

[23] D. Zhou, J. Huang, and B. Schölkopf. Learning with hypergraphs: clustering, classification, embedding. In *Adv. in Neur. Inf. Processing Systems*, volume 19, pages 1601–1608, 2006.

[24] J. Y. Zien, M. D. F. Schlag, and P. K. Chan. Multilevel spectral hypergraph partitioning with arbitrary vertex sizes. *IEEE Trans. on Comp.-Aided Design of Integr. Circ. and Systems*, 18:1389–1399, 1999.

2010 International Conference on Pattern Recognition

# Probabilistic Clustering using the Baum-Eagon Inequality

Samuel Rota Bulò and Marcello Pelillo
*DSI - University of Venice - Italy*
{srotabul,pelillo}@dsi.unive.it

## Abstract

*The paper introduces a framework for clustering data objects in a similarity-based context. The aim is to cluster objects into a given number of classes without imposing a hard partition, but allowing for a soft assignment of objects to clusters. Our approach uses the assumption that similarities reflect the likelihood of the objects to be in a same class in order to derive a probabilistic model for estimating the unknown cluster assignments. This leads to a polynomial optimization in probability domain, which is tackled by means of a result due to Baum and Eagon. Experiments on both synthetic and real standard datasets show the effectiveness of our approach.*

## 1. Introduction

Clustering is the unsupervised learning task of organizing a set of data objects (or simply objects) into groups. Commonly, clustering methods work under the assumption that objects are explicitly described in terms of features. However, a more challenging and appealing trend, which has became popular in the last few years, considers a similarity-based scenario, where the information about the objects to be clustered is expressed in terms of their *similarities*.

Unfortunately, the clustering problem is ill-posed, as there is no commonly accepted notion of a cluster. Indeed, there is a large variety of approaches, which tackle this problem by making more or less restrictive assumptions about the result they are aiming to. The most popular one forces objects to be clustered into a fixed number $k$ of classes and, typically, this is also coupled with the requirement that each data object belongs to a single cluster, yielding a hard partition of the data. This last assumption is, however, too restrictive for many important applications such as clustering micro-array gene expression data, text categorization, perceptual grouping, labeling of visual scenes and medical diagnosis.

Inspired by a recent work due to Zass and Sashua [8], we introduce a probabilistic framework for clustering in a similarity-based context. The aim is to cluster objects into a given number of classes without forcing crisp partitions, but allowing for soft assignments of objects to clusters. To this end, we first design a statistical model for the similarities parametrized by the unknown cluster assignments. We derive the model from the assumption that the similarity between two objects follows a Gaussian distribution centered around the likelihood of them to occur in a same cluster, which in turn depends on the unknown cluster assignments of the two objects. We use then the model to estimate the unknown parameters from the similarities by adopting a maximum likelihood approach. This reduces the clustering problem to a polynomial optimization in probability domain, which is solved by means of the Baum-Eagon inequality [1]. This result, indeed, provides us with a class of nonlinear transformations that serve our purpose. Experiments conducted on both synthetic and real standard datasets show the effectiveness of our approach.

## 2. A probabilistic model for $k$-clustering

Let $O = \{1, \dots, n\}$ be a set of data objects (or simply objects) to cluster into $k$ classes and consider a scenario in which objects are not explicitly described in terms of feature vectors, but a $n \times n$ nonnegative real matrix $W = (w_{ij})$ is given, whose entries provide a measure of the likelihood that two objects occur in the same cluster.

In this paper we take a probabilistic perspective by allowing objects to belong to mixtures of clusters, i.e., cluster memberships are discrete distributions over the set $\{1, \dots, k\}$ of clusters or, technically, points of the *standard simplex* $\Delta_k$, which is given by

$$\Delta_k = \left\{ \mathbf{x} \in \mathbb{R}^k_+ : \|\mathbf{x}\|_1 = 1 \right\} .$$

Let $\mathbf{y}_1, \dots, \mathbf{y}_n \in \Delta_k$ be the unknown cluster memberships of the objects in $O$. Then the likelihood of objects $i$ and $j$ to be clustered together (under independence assumption) is given by $\alpha \mathbf{y}_i^\top \mathbf{y}_j$, where $\alpha$ is a positive real value. Suppose also $Y = (\mathbf{y}_1, \dots, \mathbf{y}_n) \in \Delta_k^n$ to be the matrix obtained by stacking the $\mathbf{y}_i$'s in a row. Then, the matrix with the "true" likelihoods is given by $\alpha Y^\top Y$.

We want now to find the values of $Y$ and $\alpha$ such that the true likelihoods $\alpha Y^\top Y$ best represent the empirical ones $W$. To this end, suppose that each entry $w_{ij}$ has a measurement error that is independently random and Gaussian distributed around the "true" likelihood $\alpha \mathbf{y}_i^\top \mathbf{y}_j$. Assume also that the standard deviations $\sigma$ are

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.353

the same for all these distributions. Then we can estimate $\alpha, Y$ and $\sigma^2$ from $W$ by maximizing the likelihood of the data $W$, which is given by

$$\begin{aligned} p(W|Y,\alpha,\sigma^2) &= \prod_{i,j} p(w_{ij}|\mathbf{y}_i,\mathbf{y}_j,\alpha,\sigma^2) \\ &= \prod_{i,j} \mathcal{N}(w_{ij}|\alpha \mathbf{y}_i^\top \mathbf{y}_j, \sigma^2) \end{aligned} \tag{1}$$

where $\mathcal{N}(x|\mu,\sigma^2)$ denotes the Gaussian probability density function with mean $\mu$ and variance $\sigma^2$. Since maximizing (1) is the same as minimizing its negative logarithm, we obtain the following minimization problem

$$\begin{aligned} \min \quad & \frac{1}{\sigma^2}\|W - \alpha Y^\top Y\|^2 + n^2 \log \sigma \\ \text{s.t.} \quad & Y \in \Delta_k^n,\ \alpha, \sigma \in \mathbb{R}\,. \end{aligned} \tag{2}$$

With the view of recovering only the cluster memberships $Y$, we note that the value of the variance $\sigma^2$ in (2) does not affect the value of the optimal $Y$. Hence, $\sigma$ can be discarded and thereby (2) can be simplified as follows:

$$\begin{aligned} \min \quad & \|W - \alpha Y^\top Y\|^2 \\ \text{s.t.} \quad & Y \in \Delta_k^n, \quad \alpha \in \mathbb{R}\,. \end{aligned} \tag{3}$$

Note that the optimal solution in the variables $Y$ and $\alpha$ of (3) and (2) are the same. Moreover, the value of $Y$ provides us with *soft assignments* of the objects to the $k$ classes. Indeed, $y_{ri}$ gives the probability of object $i$ to be assigned to class $r$. If a hard partition is needed, this can be forced by assigning each object $i$ to the most probable class, which is given by: $\arg\max_{r=1\ldots k} y_{ri}$.

## 3. Related works

In [8] a similar approach is proposed. First of all, a preprocessing on the similarity matrix $W$ looks for its closest doubly-stochastic matrix $F$ under $\ell_1$ norm, or Frobenius norm, or relative entropy [9]. The $k$-clustering problem is then tackled by finding a completely-positive factorization of $F = (f_{ij})$ in the least-square sense, i.e., by solving the following optimization problem:

$$\begin{aligned} \min \quad & \|F - G^\top G\|^2 \\ \text{s.t.} \quad & G \in \mathbb{R}_+^{k\times n}\,. \end{aligned} \tag{4}$$

Note that this leads to an optimization program, which resembles (3), but is inherently different. The elements $g_{ri}$ of the resulting matrix $G$ provide an indication of object $i$ to be assigned to class $r$. However, unlike our formulation, these quantities are not explicit probabilities and it may happen for instance that $g_{ri} = 0$ for all $r = 1\ldots k$, i.e., some objects may remain in principle unclassified.

The approach proposed to find a local solution of (4) consists in iterating the following updating rule:

$$g_{ri} \leftarrow \frac{g_{ri}\sum_{j\neq i}^n g_{rj} f_{ij}}{\sum_{s=1}^k g_{si}\sum_{j\neq i}^n g_{sj} g_{rj}}\,.$$

The computational complexity for updating all entries in $G$ once (complete iteration) is $O(kn^2)$, while we expect to find a solution in $O(\gamma kn^2)$, where $\gamma$ is the average number of complete iterations required to converge. A disadvantage of this iterative scheme is that updates must be sequential, i.e., we cannot update all entries of $G$ in parallel.

## 4. The Baum-Eagon inequality

In the late 1960s, Baum and Eagon [1] introduced a class of nonlinear transformations in probability domain and proved a fundamental result which turns out to be very useful for the optimization task at hand. The next theorem introduces what is known as the Baum-Eagon inequality.

**Theorem 1** (Baum-Eagon)**.** *Let $X = (x_{ri}) \in \Delta_k^n$ and $Q(X)$ be a homogeneous polynomial in the variables $x_{ri}$ with nonnegative coefficients. Define the mapping $Z = (z_{ri}) = \mathcal{M}(X)$ as follows:*

$$z_{ri} = x_{ri}\frac{\partial Q(X)}{\partial x_{ri}} \Big/ \sum_{s=1}^k x_{si}\frac{\partial Q(X)}{\partial x_{si}}\,, \tag{5}$$

*for all $i = 1\ldots n$ and $r = 1\ldots k$. Then $Q(\mathcal{M}(X)) > Q(X)$, unless $\mathcal{M}(X) = X$. In other words $\mathcal{M}$ is a growth transformation for the polynomial $Q$.*

This result applies to homogeneous polynomials, however in a subsequent paper, Baum and Sell [3] proved that Theorem 1 still holds in the case of arbitrary polynomials with nonnegative coefficients, and further extended the result by proving that $\mathcal{M}$ increases $Q$ homotopically, which means that for all $0 \leq \eta \leq 1$, $Q(\eta\mathcal{M}(X) + (1-\eta)X) \geq Q(X)$ with equality if and only if $\mathcal{M}(X) = X$.

The Baum-Eagon inequality provides an effective iterative means for maximizing polynomial functions in probability domains, and in fact it has served as the basis for various statistical estimation techniques developed within the theory of probabilistic functions of Markov chains [2].

## 5. The algorithm

In order to use the Baum-Eagon theorem for optimizing (3), assuming $\alpha$ fixed, we need to meet the require-

ment of having a polynomial to maximize with nonnegative coefficients in the simplex-constrained variables. To this end, we consider the following optimization program, which is proved to be equivalent to (3):

$$\begin{aligned} \max \quad & 2Tr(WY^\top Y) + \alpha(\|Y^\top E_k Y\|^2 - \|Y^\top Y\|^2) \\ \text{s.t.} \quad & Y \in \Delta_k^n \,. \end{aligned} \tag{6}$$

where $E_k$ is the $k \times k$ matrix of all 1's, and $Tr(\cdot)$ is the matrix trace function.

**Proposition 1.** *The maximizers of* (6) *are minimizers of* (3) *(assuming $\alpha > 0$ fixed) and vice versa. Moreover, the objective function of* (6) *is a polynomial with nonnegative coefficients in the variables $y_{ri}$, which are elements of $Y$.*

*Proof.* Let $P(Y)$ and $Q(Y)$ be the objective functions of (3) and (6), respectively.

To prove the second part of the proposition note that trivially every term of the polynomial $\|Y^\top Y\|^2$ is also a term of $\|Y^\top E_k Y\|^2$. Hence, $Q(Y)$ is a polynomial with nonnegative coefficients in the variables $y_{ri}$.

As for the second part, by simple algebra, we can write $Q(Y)$ in terms of $P(Y)$ as follows:

$$\begin{aligned} Q(Y) &= \alpha^{-1}\left[\|W\|^2 - P(Y)\right] + \alpha\|Y^\top E_k Y\|^2 \\ &= \alpha^{-1}\left[\|W\|^2 - P(Y)\right] + \alpha \\ &= -\alpha^{-1}P(Y) + \alpha^{-1}\|W\|^2 + \alpha \,, \end{aligned}$$

where we used the fact that $\|Y^\top E_k Y\| = 1$. Note that the removal of the constant terms from $Q(Y)$ leaves its maximizers over $\Delta_k^n$ unaffected. Therefore, maximizers of (6) are also maximizers of $-\alpha^{-1}P(Y)$ over $\Delta_k^n$ and thus minimizers of (3). This concludes the proof. □

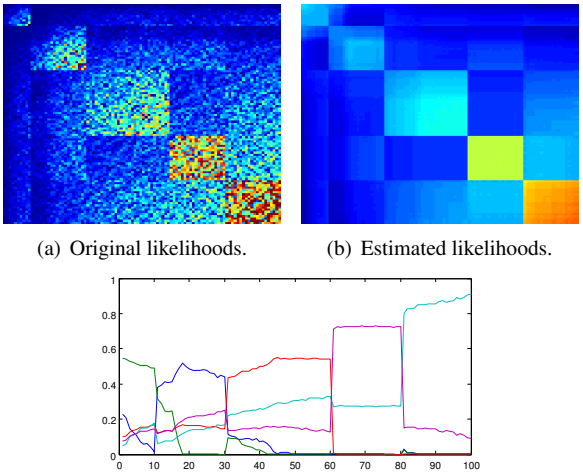
By Proposition 1, assuming $\alpha > 0$ fixed, we can use Theorem 1 to locally optimize (6). The same result guarantees that by so doing we find a solution of (3). Note that, in our case, the objective function is not a homogeneous polynomial but, as mentioned previously, this condition is not necessary [3]. By applying (5), we obtain the following updating rule for $Y = (y_{ri})$:

$$y_{ri}^{(t+1)} = y_{ri}^{(t)} \frac{\alpha n + [Y(W - \alpha Y^\top Y)]_{ri}}{\alpha n + \sum_r y_{ri}^{(t)} [Y(W - \alpha Y^\top Y)]_{ri}} \tag{7}$$

where we abbreviated $Y^{(t)}$ with $Y$.

The updating rule for the scaling factor $\alpha$, assuming $Y = Y^{(t+1)}$ fixed, can be derived from (3) by zeroing the first order derivative of the cost function with respect to $\alpha$, obtaining:

$$\alpha^{(t+1)} = \frac{Tr\left(WY^\top Y\right)}{\|Y^\top Y\|^2} \,, \tag{8}$$




(a) Original likelihoods. (b) Estimated likelihoods.

(c) Mermberships.

**Figure 1. Results on the S-Block dataset.**

which is always nonnegative. By iteratively updating $Y$ and $\alpha$ we are able to locally optimize (3). Indeed, any non-fixed iteration of (8) or (7) strictly decreases the objective function of (3).

The computational complexity of the proposed dynamics is $O(\gamma k n^2)$, where $\gamma$ is the average number of iterations required to converge (note that in our experiments we kept $\gamma$ fixed). One remarkable advantage of this dynamics is that it can be easily parallelized in order to benefit from modern multi-core processors. Additionally, it can be easily implemented with few lines of Matlab code.

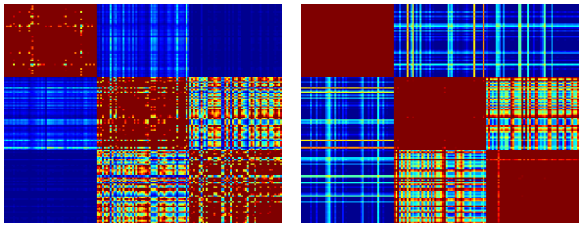
## 6. Experimental results

We performed experiments on the S100 Block Stochastic (S-Block) synthetic dataset [7], the Iris dataset (Iris), the NIST Handwritten Digits dataset (Digit) and a subset of the SCOP protein dataset (Scop) [4]. We compared our approach based on the Baum-Eagon inequality (BE) against the Copositive Factorization (CP) method [8], which has been described in Section 3, the Normalized Cuts (NCUT) [6] and the Ng-Jordan-Weiss (NJW) [5] spectral clustering approaches. Each approach has been executed 10 times and average results in terms of accuracy have been reported.

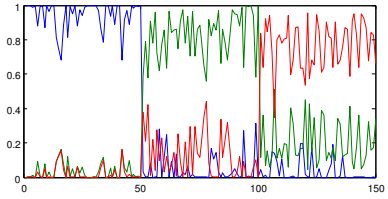
The qualitative results obtained are shown in Table 1. Our approach outperformed the competitors in the most challenging datasets, namely Digit and Scop. In the Iris dataset all approaches performed comparably well, while in the synthetic one NCUT and BE outperformed the other approaches, the latter achieving a slightly lower accuracy than the former.

| Dataset | k | n | BE | CP | NCUT | NJW |
|---|---|---|---|---|---|---|
| S-Block | 5 | 100 | .995±.013 | .556±.143 | **1.000**±.000 | .596±.010 |
| Digit | 10 | 1000 | **.700**±.042 | .430±.142 | .566±.096 | .657±.000 |
| Iris | 3 | 150 | **.993**±.000 | .991±.003 | **.993**±.000 | **.993**±.000 |
| Scop | 5 | 451 | **.706**±.001 | .703±.000 | .568±.000 | .630±.000 |

**Table 1. Clustering results on different datasets.**



(a) Original likelihoods. (b) Estimated likelihoods.




(c) Memberships.

**Figure 2. Results on the Iris dataset.**

Figure 1(b) shows the likelihoods $\alpha Y^\top Y$, estimated by our approach on the S-Block dataset. Noise has been considerably smoothed out and the block structure is now well-oulined. In Figure 1(c) we plotted also the cluster memberships of each object, i.e, matrix $Y$. Here, object indices are on the x-axis and probabilities on the y-axis, and each curve represents the profile of a cluster. As one can see, from the memberships the true cluster assignments can be clearly evinced.

In Figure 2 we present an analoguos analysis on the Iris dataset. This dataset consists of three clusters, two of which are not clearly separated. Our approach is effective also in this case, as from both the estimated likelihoods $\alpha Y^\top Y$ in Figure 2(b) and the cluster memberships in Figure 2(c), the three clusters can be clearly recognized.

## 7. Conclusion

We introduced a probabilistic framework for clustering in a similarity-based setting. The aim is to cluster objects into a given number of classes, but as opposed to conventional approaches, which induce a hard partition of the data, our algorithm provides soft assignments of objects to clusters. Our approach is based on the reasonable assumption that similarities reflect the likelihood of the objects to be in a same class in order to derive a probabilistic model for estimating the unknown cluster assignments. This reduces the clustering problem to a polynomial optimization in probability domain, which is addressed using the Baum and Eagon inquality. Experiments on both synthetic and real standard datasets show the effectiveness of our approach.


## Acknowledgements

We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250).


## References


[1] L. E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bull. Amer. Math. Soc.*, 73:360–363, 1967.

[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Ann. Math. Statistics*, 41:164–171, 1970.

[3] L. E. Baum and G. R. Sell. Growth transformations for functions on manifolds. *Pacific J. Math.*, 27:221–227, 1968.

[4] T. J. P. Hubbard, A. G. Murzin, S. E. Brenner, and C. Chothia. SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Molec. Biology*, 247:536–540, 1995.

[5] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. *Adv. in Neural Inform. Proces. Syst. (NIPS)*, pages 849–856, 2001.

[6] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 22:888–905, 2000.

[7] D. Verma and M. Meila. Comparison of spectral clustering methods. Technical report, University of Washington, 2003.

[8] R. Zass and A. Shashua. A unifying approach to hard and probabilistic clustering. In *Int. Conf. Comp. Vision (ICCV)*, volume 1, pages 294–301, 2005.

[9] R. Zass and A. Shashua. Doubly stochastic normalization for spectral clustering. *Adv. in Neural Inform. Proces. Syst. (NIPS)*, 19:1569–1576, 2006.

Computer Vision and Image Understanding 115 (2011) 984–995

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

# Graph-based quadratic optimization: A fast evolutionary approach

Samuel Rota Bulò [a,*], Marcello Pelillo [a], Immanuel M. Bomze [b]

[a] DAIS – Università Ca' Foscari Venezia, Italy
[b] ISDS, University of Vienna, Austria

ARTICLE INFO

*Article history:*
Received 14 December 2009
Accepted 17 December 2010
Available online 12 March 2011

*Keywords:*
Quadratic optimization
Population dynamics
Graph-based problems

ABSTRACT

Quadratic optimization lies at the very heart of many structural pattern recognition and computer vision problems, such as graph matching, object recognition, image segmentation, etc., and it is therefore of crucial importance to devise algorithmic solutions that are both efficient and effective. As it turns out, a large class of quadratic optimization problems can be formulated in terms of so-called "standard quadratic programs" (StQPs), which ask for finding the extrema of a quadratic polynomial over the standard simplex. Computationally, the standard approach for attacking this class of problems is to use *replicator dynamics*, a well-known family of algorithms from evolutionary game theory inspired by Darwinian selection processes. Despite their effectiveness in finding good solutions in a variety of applications, however, replicator dynamics suffer from being computationally expensive, as they require a number of operations per step which grows quadratically with the dimensionality of the problem being solved. In order to avoid this drawback, in this paper we propose a new population game dynamics (InImDyn) which is motivated by the analogy with infection and immunization processes within a population of "players." We prove that the evolution of our dynamics is governed by a quadratic Lyapunov function, representing the average population payoff, which strictly increases along non-constant trajectories and that local solutions of StQPs are asymptotically stable (i.e., attractive) points. Each step of InImDyn is shown to have a linear time/space complexity, thereby allowing us to use it as a more efficient alternative to standard approaches for solving StQPs and related optimization problems. Indeed, we demonstrate experimentally that InImDyn is orders of magnitude faster than, and as accurate as, replicator dynamics on various applications ranging from tree matching to image registration, matching and segmentation.



## 1. Introduction

Optimality is arguably one of the most pervasive and flexible meta-principles used in computer vision and pattern recognition, as it allows one to formulate real-world problems in a pure, abstract setting with solid theoretical as well as philosophical underpinnings, and permits access to the full arsenal of algorithms available in the optimization and operations research literature. On the other hand, although graphs have always been an important tool in computer vision because of their representational power and flexibility, there is now a renewed and growing interest toward explicitly formulating vision problems as graph optimization problems, and researchers are increasingly making use of sophisticated graph-theoretic concepts, results, and algorithms [18,3].

Among the variety of optimization problem families, quadratic optimization plays unquestionably a prominent role in computer vision as it naturally arises whenever abstract entities (e.g., pixels, edgels, regions, etc.) exhibit mutual pairwise interactions. The maximum clique problem, for example, which finds applications in such problems as shape and object recognition [6,2,16,45,55], stereo correspondence [26], point pattern matching [36], and image sequence analysis [48], has been successfully addressed in terms of quadratic optimization via the Motzkin–Straus theorem, a result that has recently been generalized in various ways [41,57,52] and applied to pairwise clustering problems (see Section 2). Other important applications of quadratic programming can be found in [37,21,54].

As it turns out, a large class of quadratic optimization problems can be formulated in terms of *standard quadratic programs* (StQPs), which ask for finding the extrema of a quadratic polynomial over the standard simplex. Computationally, the standard approach to solving StQPs is to use *replicator dynamics*, a class of evolutionary game-theoretic algorithms inspired by Darwinian selection processes. Indeed, there exists an intimate connection between optimization and game theory, as it can be seen that the solutions of any StQP are in one-to-one correspondence to the equilibria of a particular class of two-player games, known as partnership, or doubly-symmetric games, whereby the players' payoffs are assumed to coincide [62,25]. Interestingly, replicator dynamics also

* Corresponding author.
*E-mail address:* srotabul@dsi.unive.it (S. Rota Bulò).

1077-3142/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2010.12.004

arise independently in different branches of theoretical biology [25] and are closely related to the classical Lotka–Volterra equations from population ecology, while in population genetics they are known as selection equations [17]. Further, replicator dynamics turn out to be a special instance of a general class of dynamical systems introduced by Baum and Eagon [4] in the context of Markov chain theory and represent a special case of the well-known relaxation labeling processes for solving consistent labeling problems [49].

Although replicator dynamics have proven to be an effective technique in a variety of StQP applications [44,8,56,34,41], a typical problem associated with these algorithms is the scaling behavior with the dimensionality of the problem being solved. In particular, for a problem involving $N$ variables, the computational complexity of each replicator dynamics step is $\mathcal{O}(N^2)$, thereby hindering their use in large-scale applications, such as high-resolution image/video segmentation and matching. Previous attempts aimed at improving the computational time of the replicator dynamics can be found in works of Pelillo [42,43,47], where an exponential replicator model (a member of a larger class of "payoff-monotonic" game dynamics) has been employed in order to reduce the number of iterations needed for the algorithm to find a solution. However, despite requiring less iterations, the proposed solution still suffers from a per-step quadratic complexity.

In this paper we study a new population game dynamics, the *infection-immunization dynamics* (InImDyn), which avoids this drawback and leads to a remarkable computational gain over previous approaches. InImDyn is motivated by the analogy with infection and immunization processes within a population of "players." Intuitively, the evolutionary process can be interpreted as follows: as time passes by, an advertisement on the basis of the aggregate behavior of the population tells the agents that a certain strategy is *successful* or is *unsuccessful*. A strategy is successful if it is performing best in terms of payoff in the population, whereas it is unsuccessful if it is the worst performing strategy still alive in the population. Both variants will be taken into account: in contrast to the best-reply approach typically used in evolutionary game theory [25], which selects the strategy with highest average payoff, a successful strategy is chosen only if its *absolute* deviation from the average payoff is largest among all absolute deviations. Otherwise, the largest absolute deviation is provided by an unsuccessful strategy, and we move straight away from it by help of its *co-strategy* (to be defined below). In its most generic formulation, this phase encodes a particular selection function for infective strategies, which basically increases (decreases) the share of agents playing the successful (unsuccessful) strategy, as long as there is no barrier to the invasion. Hence, assuming that agents can gather information only about the announced strategy, they will be inclined to switch to the successful strategy, or abandon the one unsuccessful.

In the paper we prove that the evolution of our dynamics is governed by a quadratic Lyapunov function, representing the average population payoff, which strictly increases along any non-constant trajectory and that local solutions of StQPs are asymptotically stable (i.e., attractive) equilibrium points. We also show that each step of InImDyn has a linear time/space complexity, as opposed to the quadratic per-step complexity of replicator dynamics. We provide experimental evidence that the proposed algorithm is orders of magnitude faster than the standard algorithms on various graph-based computer vision applications, ranging from tree matching to image segmentation, matching and registration, while preserving the quality of the solutions found. Hence our approach can be considered an efficient and theoretically sound alternative to the replicator dynamics, that can be usefully employed in those graph-based computer vision and pattern recognition problems where computational complexity might be an issue, e.g., video and high-resolution image segmentation, matching of large graphs, clustering of large datasets, etc.

The paper is organized as follows. In Section 2 we provide a short review of various graph-based problems that lead to an StQP formulation, while in Section 3 we summarize the basic concepts and results of evolutionary graph theory and replicator dynamics. Section 4 is devoted to the description of our new class of evolutionary dynamics and Section 5 describes a specific instance which exhibits a linear time/space complexity per step. In Section 6 we report on the experimental results, and we finally draw our conclusions in Section 7. A preliminary version of this work has been presented in [51].

## 2. Quadratic formulation for graph-theoretic problems

Many graph-theoretic problems can be formulated in terms of a *standard quadratic program* (StQP), which is defined as:

$$\begin{aligned} &\text{maximize} && \mathbf{x}^\top Q\mathbf{x} \\ &\text{subject to} && \mathbf{x} \in \Delta \end{aligned}$$

where $Q \in \mathbb{R}^{n\times n}$ is a symmetric matrix, and $\Delta$ is the *standard simplex* of $\mathbb{R}^n$:

$$\Delta = \left\{ \mathbf{x} \in \mathbb{R}^n : \sum_{i=1}^{n} x_i = 1 \text{ and } x_i \geqslant 0, i = 1, \ldots, n \right\}.$$

A large class of quadratic programming problems (QPs), instances of which arise frequently in computer vision and pattern recognition, can be rewritten in terms of StQPs. In fact, consider a general QP over a bounded polyhedron

$$\begin{aligned} &\text{maximize} && \tfrac{1}{2}\mathbf{x}^\top Q\mathbf{x} + \mathbf{c}^\top\mathbf{x} \\ &\text{subject to} && \mathbf{x} \in \Delta \end{aligned} \tag{1}$$

where $M = \text{conv}\{\mathbf{v}_1, \ldots, \mathbf{v}_k\} \subseteq \mathbb{R}^n$ is the convex hull of the points $\mathbf{v}_1$, …, $\mathbf{v}_k$.

It is easy to see that the original QP in (1) can be written as the following StQP:

$$\begin{aligned} &\text{maximize} && \mathbf{y}^\top \widehat{Q}\mathbf{y} \\ &\text{subject to} && \mathbf{y} \in \Delta \end{aligned}$$

where $\widehat{Q} = \frac{1}{2}(V^\top QV + \mathbf{e}^\top V^\top \mathbf{c} + \mathbf{c}^\top V\mathbf{e})$ and $V = [\mathbf{v}_1, \ldots, \mathbf{v}_k]$.

Thus every QP over a polytope can be expressed as an StQP. Of course, this approach is practical only when the polytope is explicitly expressed in terms of its $k$ vertices (and when $k$ is not too large). This is the case of QPs over the $\ell^1$ ball, where $V = [I| - I]$, $I$ being the $n \times n$ identity matrix and $\Delta \subset \mathbb{R}^{2n}$ [11]. However, even for general QPs of the form:

$$\begin{aligned} &\text{maximize} && \tfrac{1}{2}\mathbf{x}^\top Q\mathbf{x} + \mathbf{c}^\top\mathbf{x} \\ &\text{subject to} && \mathbf{x} \in \mathbb{R}^n_+ \text{ and } A\mathbf{x} = \mathbf{b} \end{aligned}$$

we can use StQP as a relaxation without using all vertices (see [12] for details).

Now, we provide a short review of a few graph-theoretic problems that can be formulated in terms of a StQP, namely the maximum clique problem, graph/tree matching, and pairwise data clustering.

### 2.1. Maximum clique problem

Let $G = (V, E)$ be an undirected graph, where $V = \{1, \ldots, n\}$ is the set of vertices and $E \subseteq V \times V$ is the set of edges. The *order* of $G$ is the number of its vertices, and its *size* is the number of edges. Two vertices $i, j \in V$ are said to be *adjacent* if $(i, j) \in E$. The *adjacency matrix* of $G$ is the $n \times n$ symmetric matrix $A_G = (a_{ij})$ defined as $a_{ij} = 1$ if $(i, j) \in E$, and $a_{ij} = 0$ otherwise. A subset $C$ of vertices in $G$ is called

a *clique* if all its vertices are mutually adjacent. A clique is said to be *maximal* if it is not contained in any larger clique, and *maximum* if it is the largest clique in the graph. The *clique number*, denoted by $\omega(G)$, is defined as the cardinality of the maximum clique. The maximum clique problem is to find a clique whose cardinality equals the clique number. It is known to be NP-hard for arbitrary graphs and so is the problem of approximating it within a constant factor. We refer to [10] for a survey of results concerning algorithms, complexity and applications of this problem.

Based on a result due to Motzkin and Straus [33], the maximum/maximal cliques of a graph can be characterized in terms of local/global solutions of the following StQP [8]:

$$\begin{array}{ll} \text{maximize} & \mathbf{x}^\top (A_G + \frac{1}{2} I)\mathbf{x} \\ \text{subject to} & \mathbf{x} \in \Delta \end{array} \tag{2}$$

**Theorem 1.** *Let $C$ be a subset of vertices of a graph $G$ and let $\mathbf{x}^C$ be its characteristic vector, defined as $x_i^C = 1/|C|$ if $i \in C$ and $x_i^C = 0$ otherwise. Then, $C$ is a maximum (maximal) clique of $G$ if and only if $\mathbf{x}^C$ is a global (local) solution of (2). Moreover, all local (and hence global) solutions of (2) are strict.*

This result has also been generalized to the problem of finding a maximum weight clique in a vertex-weighted graph [23,7].

### 2.2. Graph and tree matching

The problem of matching two structures (e.g., graphs, trees, etc.) can be cast into a maximum clique problem over an auxiliary graph, usually called *association graph*, which in turn can be approached by solving the StQP in (2).

As for graphs the following theorem holds [2].

**Theorem 2.** *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs of order $n$, and let $G = (V, E)$ be the corresponding association graph, where*

$$V = V_1 \times V_2$$

*and*

$$E = \{((i,h),(j,k)) \in V^2 : i \neq j, h \neq k \text{ and } (i,j) \in E_1 \iff (h,k) \in E_2\}.$$

*Then, $G_1$ and $G_2$ are isomorphic if and only if $\omega(G) = n$. In this case, any maximum clique of $G$ induces an isomorphism between $G_1$ and $G_2$, and vice versa. In general, maximal/maximum cliques in $G$ are in one-to-one correspondence with maximal/maximum common subgraph isomorphisms between $G_1$ and $G_2$, respectively.*

This idea also holds when the structures being matched are trees, although, in this case, we need additional constraints in order to preserve the hierarchical relations [45]. Let $i$ and $j$ be two distinct vertices of a rooted tree $T = (V,E)$, and let $i = x_0\, x_1, \ldots, x_n = j$ be the (unique) path joining them. The *path-string* of $i$ and $j$, denoted by $\mathrm{str}(i,j)$, is the string $s_1 s_2, \ldots, s_n$ on the alphabet $\{-1,+1\}$ where, for all $k = 1, \ldots, n$, $s_i = \mathrm{lev}(x_k) - \mathrm{lev}(x_{k-1})$ with $\mathrm{lev}(x) = -\mathrm{dist}(x,r)$ is the distance of a vertex $x$ from the root $r$. By convention, when $i = j$ we define $\mathrm{str}(i,j) = \varepsilon$, where $\varepsilon$ is the null string.

**Theorem 3.** *Let $T_1 = (V_1, E_1)$ and $T_2 = (V_2, E_2)$ be two (rooted) trees, and let $G = (V,E)$ be the corresponding tree association graph (TAG), where*

$$V = V_1 \times V_2$$

*and, for any two vertices $(i,h)$ and $(j,k)$ in $V$, we have*

$$((i,h),(j,k)) \in E \iff \mathrm{str}(i,j) = \mathrm{str}(h,k).$$

*Then any maximal/maximum subtree isomorphism between $T_1$ and $T_2$ induces a maximal/maximum clique in the corresponding TAG, and vice versa.*

Note that by replacing the path-string $str(i,j)$ between two nodes with the length $d(i,j)$ of the unique path along the tree joining them, it is possible to extend the previous result to the case of matching free (unrooted) trees [43].

The framework just described has also been extended in a natural way to the case of attributed and many-to-many matching problems [45,46,1], and has found applications in problems as diverse as shape matching [45,53], region-based hierarchical image matching [56], range image matching [27], image registration and region-based many-to-many image matching [1].

### 2.3. Pairwise data clustering

A further problem that can be successfully formulated in terms of StQPs is pairwise data clustering. Here, the data to be clustered is represented as an edge-weighted graph $G = (V,E,w)$, where the $n$ vertices in $V$ correspond to data points, the edges in $E$ represent neighborhood relationships, and the weight function $w : E \to \mathbb{R}$ reflects the similarity between pairs of neighboring vertices. It is customary to represent the graph $G$ with the corresponding (real-valued) similarity matrix, which is the $n \times n$ matrix $A = (a_{ij})$ defined as:

$$a_{ij} = \begin{cases} w(i,j) & \text{if } (i,j) \in E \\ 0 & \text{otherwise.} \end{cases}$$

In a recent series of papers [39,38,40,41], Pavan and Pelillo introduced a novel framework for pairwise clustering which involves finding the local solutions of the following StQP:

$$\begin{array}{ll} \text{maximize} & \mathbf{x}^\top A \mathbf{x} \\ \text{subject to} & \mathbf{x} \in \Delta. \end{array} \tag{3}$$

In particular, they showed that the local solutions of (3) are in one-to-one correspondence to *dominant sets*, a graph-theoretic notion which generalize the concept of a maximal clique to edge-weighted graphs. This result is indeed a generalization of the Motzkin–Straus theorem [33]. The theory has recently been extended to the cases of asymmetric as well as high-order similarities [57,52], for which intriguing connections to game theory exist. The dominant-set framework has proven to be successful in a variety of different applications ranging from image and video segmentation [39,41] to perceptual grouping [57], analysis of fMRI data [34,35], content-based image retrieval [59], detection of anomalous activities in video streams [24], bioinformatics [20] and human action recognition [61].

## 3. Evolutionary game theory and replicator dynamics

Evolutionary game theory originated in the early 1970s as an attempt to apply the principles and tools of (non-cooperative) game theory to biological contexts, with a view to model the evolution of animal, as opposed to human, behavior.

It considers an idealized scenario whereby pairs of individuals are repeatedly drawn at random from a large, ideally infinite, population to play a symmetric two-player game. In contrast to conventional game theory, here players are not supposed to behave rationally or to have complete knowledge of the details of the game. They act instead according to a pre-programmed behavior pattern, or pure strategy, and it is supposed that some evolutionary selection process operates over time on the distribution of behaviors. We refer the reader to [62,25] for excellent introductions to this rapidly expanding field.

Let $O = \{1, \ldots, n\}$ be the set of *pure strategies* available to the players and let $A$ be the $n \times n$ payoff or utility matrix, where $a_{ij}$ is the payoff that a player gains when playing the strategy $i$ against

an opponent playing strategy $j$.[1] A *mixed strategy* is a probability distribution $\mathbf{x} = (x_1, x_2, \ldots, x_n)^\top$ over the available strategies in $O$. Mixed strategies lie in the standard simplex $\Delta$ of the $n$-dimensional Euclidean space.

We denote by $\mathbf{e}^i$ the $i$th column of the identity matrix. The *support* of a mixed strategy $\mathbf{x} \in \Delta$, denoted by $\sigma(\mathbf{x})$, defines the set of elements with non-zero probability: $\sigma(\mathbf{x}) = \{i \in O: x_i > 0\}$. The expected payoff that a player obtains by playing the pure strategy $i$ against an opponent playing a mixed strategy $\mathbf{x}$ is

$$\pi(\mathbf{e}^i|\mathbf{x}) = (A\mathbf{x})_i = \sum_j a_{ij}x_j.$$

Hence, the expected payoff received by adopting a mixed strategy $\mathbf{y}$ is given by

$$\pi(\mathbf{y}|\mathbf{x}) = \mathbf{y}^\top A\mathbf{x}$$

while the population expected payoff is

$$\pi(\mathbf{x}) = \pi(\mathbf{x}|\mathbf{x}) = \mathbf{x}^\top A\mathbf{x}.$$

For notational compactness, in the sequel we will write $\pi(\mathbf{y} - \mathbf{x}|\mathbf{z})$ for the payoff difference $\pi(\mathbf{y}|\mathbf{z}) - \pi(\mathbf{x}|\mathbf{z})$, and $\pi(\mathbf{y} - \mathbf{x})$ for $\pi(\mathbf{y} - \mathbf{x}|\mathbf{y}) - \pi(\mathbf{y} - \mathbf{x}|\mathbf{x})$.

A mixed strategy $\mathbf{x}$ is a *(symmetric) Nash (equilibrium) strategy* if for all $\mathbf{y} \in \Delta$, we have $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) \leqslant 0$. This implies that $\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) \leqslant 0$ for all $i \in O$, which in turn implies that $\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) = 0$ for all $i \in \sigma(\mathbf{x})$. Hence, the payoff is constant across all (pure) strategies in the support of $\mathbf{x}$, while all strategies outside the support of $\mathbf{x}$ earn a payoff that is less than or equal $\pi(\mathbf{x})$. A strategy $\mathbf{x}$ is said to be an *Evolutionary Stable Strategy* (ESS) if it is a Nash strategy (*equilibrium condition*) and for all $\mathbf{y} \in \Delta$ satisfying $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$ we have $\pi(\mathbf{y} - \mathbf{x}|\mathbf{y}) < 0$ (*stability condition*). Intuitively, ESS's are strategies such that any small deviation from them will lead to an inferior payoff.

There is an intimate relation between evolutionary game theory and standard quadratic optimization, as in the case of doubly-symmetric games, where the payoff matrix $A$ is symmetric, there exists a one-to-one correspondence between ESSs and strict local maximizers of the population expected payoff $\pi(\mathbf{x}) = \mathbf{x}^\top A\mathbf{x}$ over the standard simplex, while critical points are shown to be related to Nash equilibria [62].

### 3.1. Replicator dynamics

In evolutionary game theory the assumption is made that the game is played over and over, generation after generation, and that the action of natural selection results in the evolution of the fittest strategies. If successive generations blend into each other, the evolution of behavioral patterns can be described by the following discrete-time equations [32, Appendix]:

$$x_i^{(t+1)} = x_i^{(t)} \frac{\pi(\mathbf{e}^i|\mathbf{x}^{(t)})}{\pi(\mathbf{x}^{(t)})} \tag{4}$$

for $i = 1, \ldots, n$. It is straightforward to see that the simplex $\Delta$ is invariant under dynamics (4) or, in other words, any trajectory starting in $\Delta$ will remain in $\Delta$.

A point $\mathbf{x}^{(t)}$ is said to be a *stationary* (or equilibrium) point for our dynamical systems, if $x_i^{(t+1)} = x_i^{(t)} (i = 1, \ldots, n)$. Moreover, a stationary point is said to be *asymptotically stable* if any trajectory starting in its vicinity will converge to it as $t \to \infty$. The set of stationary points of the replicator dynamics contains all the points in $\Delta$ satisfying, for all $i = 1, \ldots, n$, the condition

$$x_i^{(t)}\pi(\mathbf{e}^i - \mathbf{x}^{(t)}|\mathbf{x}^{(t)}) = 0$$

or, equivalently, $\pi(\mathbf{e}^i - \mathbf{x}^{(t)}|\mathbf{x}^{(t)}) = 0$ whenever $x_i > 0$.

The basic idea behind this model is that the fraction of $i$-strategists will grow whenever their performance, in terms of average payoff, exceeds the population's average payoff. Conversely, the share of $i$-strategists will decrease over time. The following theorem states that under replicator dynamics the population's average payoff always increases, provided that the payoff matrix is symmetric.

**Theorem 4.** *Suppose that the (nonnegative) payoff matrix $A$ is symmetric. Then, the quadratic polynomial $\pi$ defined as*

$$\pi(\mathbf{x}) = \mathbf{x}^\top A\mathbf{x}$$

*is strictly increasing along any non-constant trajectory of discrete-time (4) replicator equations. In other words, for all $t \geqslant 0$ we have $\pi(\mathbf{x}^{(t+1)}) > \pi(\mathbf{x}^{(t)})$, unless $\mathbf{x}^{(t)}$ is a stationary point. Furthermore, any such trajectory converges to a (unique) stationary point.*

The previous result is known in mathematical biology as the fundamental theorem of natural selection [17,25,62] and, in its original form, traces back to Fisher [19]. It can also be regarded as a straightforward implication of the Baum-Eagon theorem [4,5] which is valid for general polynomial functions over product of simplices. Waugh and Westervelt [60] also proved a similar result for a related class of continuous- and discrete-time dynamical systems. In the discrete-time case, however, they put bounds on the eigenvalues of $A$ in order to achieve convergence to fixed points.

The fact that all trajectories of the replicator dynamics converge to a stationary point has been proven in [29]. However, in general, not all stationary points are local maximizers of $\pi$ on $\Delta$. The vertices of $\Delta$, for example, are all stationary points for (4) whatever the landscape of $\pi$. Moreover, there may exist trajectories which, starting from the interior of $\Delta$, eventually approach a saddle point of $F$. However, a result proved by Bomze [8] asserts that all asymptotically stable stationary points of replicator dynamics correspond to (strict) local maximizers of $\pi$ on $\Delta$, and vice versa.

Another popular population game dynamics is given by the following equations:

$$x_i^{(t+1)} = x_i^{(t)} \frac{\exp(\kappa\pi(\mathbf{e}^i|\mathbf{x}))}{\sum_{j=1}^n x_j^{(t)} \exp(\kappa\pi(\mathbf{e}^j|\mathbf{x}^{(t)}))} \tag{5}$$

where $\kappa$ is a positive constant. As $\kappa$ tends to 0, the orbits of this dynamics approach those of the standard, first-order replicator model (4), slowed down by the factor $\kappa$. From a computational perspective, exponential replicator dynamics are particularly attractive as they may be considerably faster and even more accurate than the standard, first-order model (see [42,43]). Unfortunately, unlike the continuous-time case, there is no guarantee that the discrete-time exponential dynamics (5) increases the value of $\mathbf{x}^\top A\mathbf{x}$ for any fixed value of the parameter $\kappa$. For a partial result in this direction see [9]. Nevertheless, as suggested in [47] one can iteratively adapt this parameter to enforce this property. Global convergence of a variant of (5) with Armijo-like stepsize rule in a more general context is proved in [58, Section 7].

## 4. A new class of evolutionary dynamics

In this section we introduce our population game dynamics, which is motivated by the analogy with infection and immunization processes within a population of "players," and prove some

[1] Here, we shall focus on *symmetric* games, whereby the two players share the same payoff matrix (which is not necessarily symmetric). Games where the payoff matrix is symmetric are referred to as doubly-symmetric, or partnership, games [62,25].

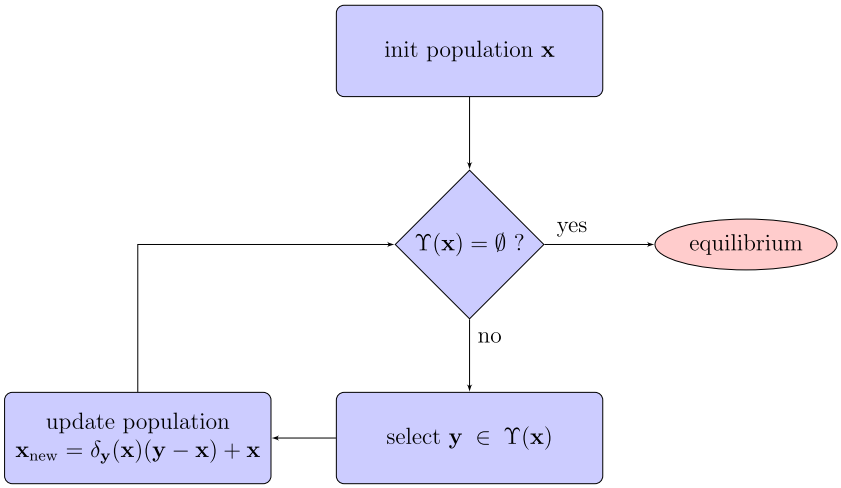



**Fig. 1.** Diagram of the evolutionary dynamics for finding a Nash equilibrium. Starting from a population $\mathbf{x}$, if there are no infective strategies for $\mathbf{x}$, i.e., if $\Upsilon(\mathbf{x}) = \emptyset$, then we have an equilibrium. Otherwise, we select an infective strategy for $\mathbf{x}$ and update the population according to Proposition 1 and reiterate the process.

properties that are instrumental in solving standard quadratic optimization problems.

Let $\mathbf{x} \in \Delta$ be the *incumbent* population state, $\mathbf{y}$ be the *mutant* population invading $\mathbf{x}$ and let $\mathbf{z} = (1-\varepsilon)\mathbf{x} + \varepsilon\mathbf{y}$ be the population state obtained by injecting into $\mathbf{x}$ a small share of $\mathbf{y}$-strategists. The *score function* of $\mathbf{y}$ versus $\mathbf{x}$ [13] is given by:

$$h_{\mathbf{x}}(\mathbf{y}, \varepsilon) = \pi(\mathbf{y} - \mathbf{x}|\mathbf{z}) = \varepsilon\pi(\mathbf{y} - \mathbf{x}) + \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}).$$

Following [15], we define the *(neutral) invasion barrier* $b_{\mathbf{x}}(\mathbf{y})$ of $\mathbf{x} \in \Delta$ against any mutant strategy $\mathbf{y}$ as the largest population share $\varepsilon_{\mathbf{y}}$ of $\mathbf{y}$-strategists such that for all smaller positive population shares $\varepsilon$, $\mathbf{x}$ earns a higher or equal payoff than $\mathbf{y}$ in the post-entry population $\mathbf{z}$. Formally:

$$b_{\mathbf{x}}(\mathbf{y}) = \inf(\{\varepsilon \in (0,1) : h_{\mathbf{x}}(\mathbf{y}, \varepsilon) > 0\} \cup \{1\}).$$

Given populations $\mathbf{x}, \mathbf{y} \in \Delta$, we say that $\mathbf{x}$ is *immune* against $\mathbf{y}$ if $b_{\mathbf{x}}(\mathbf{y}) > 0$. Trivially, a population is always immune against itself. Note that, $\mathbf{x}$ is immune against $\mathbf{y}$ if and only if either $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) < 0$ or $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$ and $\pi(\mathbf{y} - \mathbf{x}) \leqslant 0$. If $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) > 0$ we say that $\mathbf{y}$ is *infective* for $\mathbf{x}$. We denote the set of infective strategies for $\mathbf{x}$ as

$$\Upsilon(\mathbf{x}) = \{\mathbf{y} \in \Delta : \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) > 0\}.$$

Consider $\mathbf{y} \in \Upsilon(\mathbf{x})$; clearly, this implies $b_{\mathbf{x}}(\mathbf{y}) = 0$. If we allow for invasion of a share $\varepsilon$ of $\mathbf{y}$-strategists as long as the score function of $\mathbf{y}$ versus $\mathbf{x}$ is positive, at the end we will have a share of $\delta_{\mathbf{y}}(\mathbf{x})$ mutants in the post-entry population, where

$$\delta_{\mathbf{y}}(\mathbf{x}) = \inf(\{\varepsilon \in (0,1) : h_{\mathbf{x}}(\mathbf{y}, \varepsilon) \leqslant 0\} \cup \{1\}).$$

Note that if $\mathbf{y}$ is infective for $\mathbf{x}$, then $\delta_{\mathbf{y}}(\mathbf{x}) > 0$, whereas if $\mathbf{x}$ is immune against $\mathbf{y}$, then $\delta_{\mathbf{y}}(\mathbf{x}) = 0$. Since score functions are (affine-)linear, there is a simpler expression $\delta_{\mathbf{y}}(\mathbf{x}) = \min\left[\frac{\pi(\mathbf{x}-\mathbf{y}|\mathbf{x})}{\pi(\mathbf{y}-\mathbf{x})}, 1\right]$, if $\pi(\mathbf{y} - \mathbf{x}) < 0$, and $\delta_{\mathbf{y}}(\mathbf{x}) = 1$, otherwise.

**Proposition 1.** *Let $\mathbf{y} \in \Upsilon(\mathbf{x})$ and $\mathbf{z} = (1-\delta)\mathbf{x} + \delta\mathbf{y}$, where $\delta = \delta_{\mathbf{y}}(\mathbf{x})$. Then $\mathbf{y} \notin \Upsilon(\mathbf{z})$.*

The proof of this result is straightforward by linearity and can be found, e.g., in [50].

The core idea of our method is based on the fact that $\mathbf{x} \in \Delta$ is a Nash equilibrium if and only if $\Upsilon(\mathbf{x}) = \emptyset$ (we prove this in Theorem 5). Therefore, as long as we find a strategy $\mathbf{y} \in \Upsilon(\mathbf{x})$, we update the population state according to Proposition 1 in order obtain a new population $\mathbf{z}$ such that $\mathbf{y} \notin \Upsilon(\mathbf{z})$ and we reiterate this process until no infective strategy can be found, or in other words, a Nash equilibrium is reached. In Fig. 1 we schematically summarized the basic idea.

The formalization of this process provides us with a class of new dynamics which, for evident reasons, is called *Infection and Immunization Dynamics* (InImDyn):

$$\mathbf{x}^{(t+1)} = \delta_{\mathcal{S}(\mathbf{x}^{(t)})}(\mathbf{x}^{(t)})[\mathcal{S}(\mathbf{x}^{(t)}) - \mathbf{x}^{(t)}] + \mathbf{x}^{(t)}. \tag{6}$$

Here, $\mathcal{S} : \Delta \to \Delta$ is a generic *strategy selection* function which returns an infective strategy for $\mathbf{x}$ if it exists, or $\mathbf{x}$ otherwise:

$$\mathcal{S}(\mathbf{x}) = \begin{cases} \mathbf{y} & \text{for some } \mathbf{y} \in \Upsilon(\mathbf{x}) \text{ if } \Upsilon(\mathbf{x}) \neq \emptyset, \\ \mathbf{x} & \text{otherwise.} \end{cases} \tag{7}$$

By running these dynamics we aim at reaching a population state that can not be infected by any other strategy. In fact, if this is the case, then $\mathbf{x}$ is a Nash strategy, which happens if and only if it is fixed (i.e., stationary) under dynamics (6):

**Theorem 5.** *Let $\mathbf{x} \in \Delta$ be a strategy. Then the following statements are equivalent:*

(a) *$\Upsilon(\mathbf{x}) = \emptyset$: there is no infective strategy for $\mathbf{x}$;*
(b) *$\mathbf{x}$ is a Nash strategy;*
(c) *$\mathbf{x}$ is a fixed point under dynamics (6).*

**Proof.** A strategy $\mathbf{x}$ is a Nash strategy if and only if $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) \leqslant 0$ for all $\mathbf{y} \in \Delta$. This is true if and only if $\Upsilon(\mathbf{x}) = \emptyset$. Further, $\delta = 0$ implies $\mathcal{S}(\mathbf{x}) = \mathbf{x}$. Conversely, if $\mathcal{S}(\mathbf{x})$ returns $\mathbf{x}$, then we are in a fixed point. By construction of $\mathcal{S}(\mathbf{x})$ this happens only if there is no infective strategy for $\mathbf{x}$. □

The following result shows that average payoff is strictly increasing along any non-constant trajectory of the dynamics (6), provided that the payoff matrix is symmetric.

**Theorem 6.** *Let $\{\mathbf{x}^{(t)}\}_{t \geqslant 0}$ be a trajectory of (6). Then for all $t \geqslant 0$,*

$$\pi(\mathbf{x}^{(t+1)}) \geqslant \pi(\mathbf{x}^{(t)}),$$

*with equality if and only if $\mathbf{x}^{(t)} = \mathbf{x}^{(t+1)}$, provided that the payoff matrix is symmetric.*

**Proof.** Let us write $\mathbf{x}$ for $\mathbf{x}^{(t)}$ and $\delta$ for $\delta_{\mathcal{S}(\mathbf{x})}(\mathbf{x})$.

As shown in [50], we have

$$\pi(\mathbf{x}^{(t+1)}) - \pi(\mathbf{x}^{(t)}) = \delta\left[h_{\mathbf{y}}(\mathbf{x}, \delta) + \pi(\mathbf{y} - \mathbf{x}|\mathbf{x})\right].$$

If $\mathbf{x}^{(t+1)} \neq \mathbf{x}^{(t)}$, then $\mathbf{x}$ is no Nash strategy, and $\mathbf{y} = \mathcal{S}(\mathbf{x})$ returns an infective strategy. Hence $\delta > 0$ and

$$h_{\mathbf{y}}(\mathbf{x}, \delta) + \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) \geqslant \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) > 0$$

(in fact, if $\delta < 1$, then even $h_{\mathbf{y}}(\mathbf{x}, \delta) = 0$), so that we obtain a strict increase of the population payoff. On the other hand, if $\pi(\mathbf{x}^{(t+1)}) = \pi(\mathbf{x}^{(t)})$, then the above equation implies $\delta = 0$ or $h_{\mathbf{x}}(\mathbf{x}, \delta) = \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$, due to nonnegativity of both quantities above. In particular, we have $\delta = 0$ or $\pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = 0$. In both cases, $\mathbf{y} = \mathcal{S}(\mathbf{x})$ cannot be infective for $\mathbf{x}$. Thus $\Upsilon(\mathbf{x}) = \emptyset$ and $\mathbf{x}$ must be a fixed point, according to Theorem 5. This establishes the last assertion of the theorem. □

Theorem 6 shows that by running InImDyn, under symmetric payoff function, we strictly increase the population payoff unless we are at a fixed point, i.e., have already reached Nash equilibrium. This of course holds for any selection function $\mathcal{S}(\mathbf{x})$ satisfying (7). However, the way we choose $\mathcal{S}(\mathbf{x})$ may affect the efficiency of the dynamics. The next section introduces a particular selection function that leads to a well-performing dynamics for our purposes.

## 5. A pure strategy selection function

Depending on how we choose the function $\mathcal{S}(\mathbf{x})$ in (6), we may obtain different dynamics. One in particular, which is simple and leads to nice properties, consists in allowing only infective pure strategies. This way, our equilibrium selection process closely resembles a vertex-pivoting method, as opposed to interior-point approaches like replicator dynamics or best-response dynamics [25].

Let

$$\tau_+(\mathbf{x}) = \{i \in O : \pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) > 0\}$$
$$\tau_-(\mathbf{x}) = \{i \in O : \pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) < 0\}$$
$$\tau_0(\mathbf{x}) = \{i \in O : \pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) = 0\}.$$

Given a population $\mathbf{x}$, we define the co-strategy of $\mathbf{e}^i$ with respect to $\mathbf{x}$ as

$$\overline{\mathbf{e}^i}_{\mathbf{x}} = \frac{x_i}{x_i - 1}(\mathbf{e}^i - \mathbf{x}) + \mathbf{x}.$$

Note that if $i \in \tau_+(\mathbf{x})$ or $i \in \tau_-(\mathbf{x})$ then $\mathbf{e}^i \in \Upsilon(\mathbf{x})$ or $\overline{\mathbf{e}^i}_{\mathbf{x}} \in \Upsilon(\mathbf{x})$, respectively.

Consider the strategy selection function $\mathcal{S}_{Pure}(\mathbf{x})$, which finds a pure strategy $i$ maximizing $|\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x})|$, and returns $\mathbf{e}^i, \overline{\mathbf{e}^i}_{\mathbf{x}}$ or $\mathbf{x}$ according to whether $i \in \tau_+(\mathbf{x})$, $i \in \tau_-(\mathbf{x}) \cap \sigma(\mathbf{x})$ or $i \in \tau_0(\mathbf{x})$. Let $\mathcal{M}(\mathbf{x})$ be a pure strategy such that

$$\mathcal{M}(\mathbf{x}) \in \arg\max\big(\{\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}) : i \in \tau_+(\mathbf{x})\} \cup \{\pi(\mathbf{x} - \mathbf{e}^i|\mathbf{x}) : i \in \tau_-(\mathbf{x}) \cap \sigma(\mathbf{x})\}\big).$$

Then $\mathcal{S}_{Pure}(\mathbf{x})$ can be written as

$$\mathcal{S}_{Pure}(\mathbf{x}) = \begin{cases} \mathbf{e}^i & \text{if } i = \mathcal{M}(\mathbf{x}) \in \tau_+(\mathbf{x}) \\ \overline{\mathbf{e}^i}_{\mathbf{x}} & \text{if } i = \mathcal{M}(\mathbf{x}) \in \tau_-(\mathbf{x}) \cap \sigma(\mathbf{x}) \\ \mathbf{x} & \text{otherwise.} \end{cases}$$

Note that the search space for an infective strategy is reduced from $\Delta$ to a finite set. Therefore, it is not obvious that $\mathcal{S}_{\mathcal{P}\cap\nabla}(\mathbf{x})$ is a well-defined selection function, i.e., it satisfies (7). The next theorem shows that indeed it is.

**Proposition 2.** *Let $\mathbf{x} \in \Delta$ be a population state. There exists an infective strategy for $\mathbf{x}$, i.e., $\Upsilon(\mathbf{x}) \neq \emptyset$, if and only if $\mathcal{S}_{Pure}(\mathbf{x}) \in \Upsilon(\mathbf{x})$.*

**Proof.** Let $\mathbf{y} \in \Upsilon(\mathbf{x})$. Then

$$0 < \pi(\mathbf{y} - \mathbf{x}|\mathbf{x}) = \sum_{i=1}^{n} y_i \pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}).$$

But this implies that there exists at least one infective pure strategy for $\mathbf{x}$, i.e., $\mathbf{e}^i \in \Upsilon(\mathbf{x})$ for some $i = 1, \ldots, n$. The remainder follows as in [50]. The converse trivially holds.

**Theorem 7.** *A state $\mathbf{x}$ is asymptotically stable for InImDyn with $\mathcal{S}_{Pure}$ as strategy selection function if and only if $\mathbf{x}$ is an ESS, provided that the payoff matrix is symmetric.*

**Proof.** If the payoff matrix is symmetric, every accumulation point of InImDyn with $\mathcal{S}_{\mathcal{P}\cap\nabla|}$ is a Nash equilibrium [50]. Moreover *ESSs* are strict local maximizers of $\pi(\mathbf{x})$ over $\Delta$ and vice versa [62]. If $\mathbf{x}$ is asymptotically stable, then there exists a neighborhood $U$ of $\mathbf{x}$ in $\Delta$ such that any trajectory starting in $U$ converges to $\mathbf{x}$. By Theorem 6 this implies that $\pi(\mathbf{x}) > \pi(\mathbf{y})$ for all $\mathbf{y} \in U, \mathbf{y} \neq \mathbf{x}$. Hence, $\mathbf{x}$ is a strict local maximizer of $\pi(\mathbf{x})$ and therefore $\mathbf{x}$ is an ESS. The converse follows as in [50].

This selection function exhibits the nice property of rendering the complexity per iteration of our new dynamics linear in both space and time, as opposed to the replicator dynamics, which have quadratic space/time complexity per iteration.

**Theorem 8.** *Given the iterate $\mathbf{x}^{(t)}$ and its linear transformation $A\mathbf{x}^{(t)}$, both space and time requirement of one iteration step is linear in n, the number of objects.*

**Proof.** Again abbreviate $\mathbf{x} = \mathbf{x}^{(t)}$. Now, given $A\mathbf{x}$ we can straightforwardly compute in linear time and space $\pi(\mathbf{x})$ and $\mathcal{S}_{Pure}(\mathbf{x})$. Assume that $\mathcal{S}_{Pure}(\mathbf{x}) = \mathbf{e}^i$, then the computation of $\delta_{\mathbf{e}^i}(\mathbf{x})$ has a linear complexity, since $\pi(\mathbf{x} - \mathbf{e}^i|\mathbf{x}) = (A\mathbf{x})_i - \pi(\mathbf{x})$ and $\pi(\mathbf{e}^i - \mathbf{x}) = a_{ii} - 2A\mathbf{x} + \pi(\mathbf{x})$. Moreover, $A\mathbf{x}^{(t+1)}$ can be also computed in linear time and space since

$$A\mathbf{x}^{(t+1)} = \delta_{\mathbf{e}^i}(\mathbf{x})[A_i - A\mathbf{x}] + A\mathbf{x},$$

where $A_i$ is the $i$th column of $A$. Similar arguments hold if $\mathcal{S}_{Pure}(\mathbf{x}) = \overline{\mathbf{e}^i}_{\mathbf{x}}$. Indeed,

$$\pi(\overline{\mathbf{e}^i}_{\mathbf{x}} - \mathbf{x}|\mathbf{x}) = \frac{x_i}{x_i - 1}\pi(\mathbf{e}^i - \mathbf{x}|\mathbf{x}),$$
$$\pi(\overline{\mathbf{e}^i}_{\mathbf{x}} - \mathbf{x}) = \left(\frac{x_i}{x_i - 1}\right)^2 \pi(\mathbf{e}^i - \mathbf{x}),$$

and finally,

$$A\mathbf{x}^{(t+1)} = \left(\frac{x_i}{x_i - 1}\right)\delta_{\overline{\mathbf{e}^i}_{\mathbf{x}}}(\mathbf{x})[A_i - A\mathbf{x}] + A\mathbf{x}.$$

Hence the result.

The only step of quadratic complexity is the first one, where we need to compute $A\mathbf{x}^{(0)}$. Even this can be reduced to linear complexity, if we start from a pure strategy $\mathbf{e}^i$, in which case we have $A\mathbf{x}^{(0)} = A_i$. Note that the latter is impossible, e.g., for the replicator dynamics.

The algorithmic procedure for finding an equilibrium using InImDyn with $\mathcal{S}_{Pure}$ is summarized in Algorithm 1. Note that as stopping criterion we compute the following quantity:

$$\epsilon(\mathbf{x}) = \sum_i \min\{x_i, \pi(\mathbf{x} - \mathbf{e}^i|\mathbf{x})\}^2 < \tau, \tag{8}$$

which measures the degree of violation of the Nash conditions. Indeed, $\epsilon(\mathbf{x}) = 0$ if and only if $\mathbf{x}$ is a Nash equilibrium.

**Algorithm 1.** FindEquilibrium($A, \mathbf{x}, \tau$)

**Require**: $n \times n$ symmetric payoff matrix $A$, $\mathbf{x} \in \Delta$ and tolerance $\tau$
  **while** $\epsilon(\mathbf{x}) > \tau$ **do**
    $\mathbf{y} \leftarrow \mathcal{S}_{\text{Pure}}(\mathbf{x})$
    $\delta \leftarrow 1$
    **if** $\pi(\mathbf{y} - \mathbf{x}) < 0$ **then**
      $\delta \leftarrow \min\left[\frac{\pi(\mathbf{x}-\mathbf{y}|\mathbf{x})}{\pi(\mathbf{y}-\mathbf{x})}, 1\right]$
    **end if**
    $\mathbf{x} \leftarrow \delta(\mathbf{y} - \mathbf{x}) + \mathbf{x}$
  **end while**
  **return x**

## 6. Experimental results

In order to test the effectiveness of InImDyn, we present experiments on various graph-based computer vision problems formulated in terms of StQP. Specifically, we present comparisons on tree matching [43], image registration [1] image segmentation [41] and region-based hierarchical image matching [56]. Our goal is to demonstrate the computational gain over standard replicator dynamics (RD) and its exponential counterpart (expRD).

All dynamics were started from the simplex barycenter. As for the stopping criterion, InImDyn was stopped when the violation of the Nash conditions was below a given tolerance $\tau$. In our experiments we set $\tau = 10^{-10}$. As for RD and expRD, the same criterion could not be used because it usually leads to an excessive computation time. Therefore, at the cost of a lower accuracy, we stopped them when the distance between two consecutive states was below a predefined threshold which, in the experiments, was set to $10^{-6}$, or the number of iterations exceeded a problem-dependent limit.

All the algorithms were coded in C and run on a machine equipped with 8 Intel Xeon 2.33 GHz CPUs and 8 GB RAM, except for the image segmentation experiments, which were run on a AMD Sempron 3 GHz computer with 1 GB RAM.

### 6.1. Matching free trees

In [43], RD and expRD were used for matching free trees by the free tree association graph (FTAG), with application to shape-axis matching. We repeated the same experiments as described in [43], which consisted in generating random free trees and then corrupting them by deleting a fraction of the terminal vertices in order to get trees that were isomorphic to subtrees of the original ones. Specifically, we generated a hundred 100-vertex trees using the procedure described by Wilf in [63] and used the following corruption levels: 0%, 10%, 20%, 30% and 40% (which yields 500 pairs of trees being matched).

To evaluate the quality of the solution found by the three dynamics, after convergence, we calculated the proportion of matched nodes, i.e., the ratio between the cardinality of the clique found and the order of the smaller tree and then we averaged. Fig. 2 reports the results obtained, together with the corresponding running times in logarithmic scale, for all dynamics as a function of the corruption level. As can be seen all three algorithms performed comparably well, obtaining a 100% accuracy on almost all cases, however the corresponding CPU time varied dramatically, InImDyn being orders of magnitude faster than RD and, to a lesser extent, expRD.

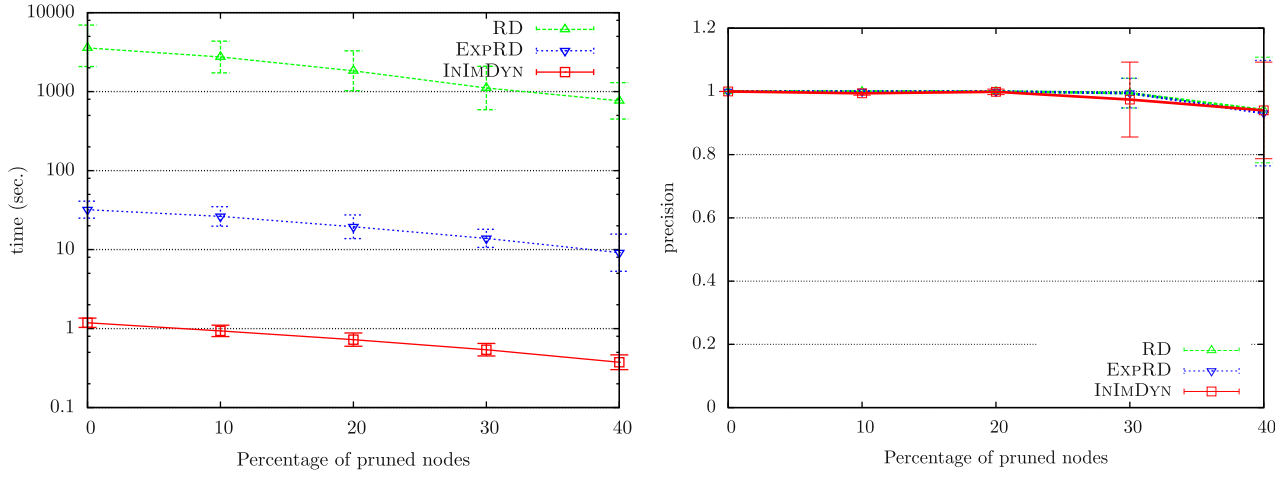
### 6.2. Image registration

Here, we perform experiments on feature-based image registration by similarity transformations. This problem can, indeed, be solved by searching ESSs equilibria of a particular two-player non-cooperative *matching game* [1], which is in turn equivalent to finding local solutions of a StQP.

We work with images from the Amsterdam Library of Object Images (ALOI) [22]. For each image we generate a new one by applying a random similarity transformation. We extract from each image a set of SIFT features [28], each of which is augmented with a scale $s$, an orientation $\theta$, a position $\mathbf{p}$ in the image and a descriptor $\mathbf{d}$ (i.e., a 128-dimensional real vector). Given two sets of features $F_1$ and $F_2$ extracted from two images to be registered, we define a two-player game, where strategies $\mathbb{A}$ correspond to feasible feature associations, i.e., subsets of $F_1 \times F_2$, and the payoff function $C : \mathbb{A} \times \mathbb{A} \to \mathbb{R}_+$ provides the compatibility between two such features. From any feature association $(i,j) \in \mathbb{A}$ we can estimate a similarity transformation. Indeed, the isotropic scale factor is given by $s_j/s_i$, the rotation angle is given by $\theta_j - \theta_i$ and the translation vector is computed as

$$\mathbf{t} = \mathbf{p}_j - \frac{s_j}{s_i} R_{\theta_j - \theta_i} \mathbf{p}_i,$$

where $R_\alpha$ is a rotation matrix with angle $\alpha$.




(a) CPU time (in seconds) (b) Percentage of correct vertex matches

**Fig. 2.** Experiments on matching 100 randomly generated 100-vertex free trees against corrupted versions of themselves. We plotted the percentage of correct vertex matches found and the CPU time in seconds (in logarithmic scale) of InImDyn, RD and expRD as a function of the tree corruption level measured in terms of number or pruned nodes.

In our experiments, $\mathbb{A}$ contains the $n$ feature associations from $F_1 \times F_2$ having the most similar descriptors (we consider several values of $n$ between 100 and 2000), each of which represents a single transformation. To measure the compatibility between two associations $a_1, a_2 \in \mathbb{A}$, we project the first point of $a_1$ with the transformation estimated from $a_2$, and measure the distance between the transformed point and the corresponding point in the first association. We then repeat the operation by reversing the role of the two associations, thereby obtaining the two distances $d_1$ and $d_2$. Accordingly, $\exp(-\min(d_1,d_2))$ is used as a measure of compatibility between $a_1$ and $a_2$.

Fig. 3 plots the average error rate in estimating the similarity transformation and the running times (in logarithmic scale) of RD, EXPRD and INIMDYN as a function of the number of feature associations in $\mathbb{A}$. As can be seen, EXPRD is more efficient than RD, but our approach turns out to be orders of magnitude faster than its competitors. Note also that, despite being much faster, the results obtained by INIMDYN are as good as the others. Interestingly, increasing the number of feature associations in $\mathbb{A}$ does not affect the quality of the solution.

Fig. 4 shows in the first row some examples of matched images from the ALOI dataset. Note that the first two images are particularly difficult, because of the presence of multiple possibly wrong sub-matches, but, as we can see, our approach finds the best solution. In the second row, we presents the registration results obtained on two challenging image pairs from [64], namely, two retina images (on the left) and two images, where one is an extremely zoomed version of the other. In both cases, our approach succeeds in finding the best solution.

### 6.3. Image segmentation

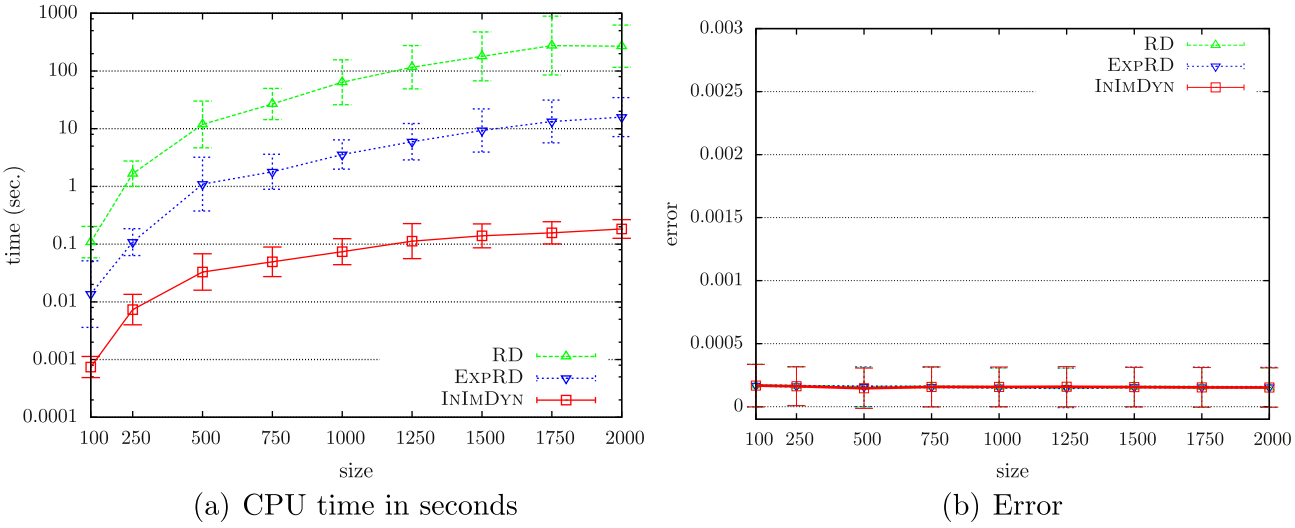
We performed image segmentation experiments over the whole Berkeley dataset [31] using the dominant-set framework as proposed in [41] (see Section 2.3). The affinity between two pixels $i$ and $j$ was computed using the standard Gaussian kernel:

$$w(i,j) = \exp(-\|C(i) - C(j)\|^2/\sigma^2),$$

where $\sigma$ is a positive real number and $C(j)$ is the color of pixel $j$ expressed as a 3-dimensional vector in the Lab color space.

Since in this application both RD and EXPRD reached the maximum allowed number of iterations, their performance in terms of computational time was indistinguishable. Hence, here we only report the results for RD. To test the behavior of the algorithms under different input sizes we performed experiments at different pixel sampling rates, namely 0.005, 0.015, 0.03 and 0.05, which roughly correspond to affinity matrices of size 200, 600, 1200 and 2000, respectively. Since the Nyström method, as opposed to the dominant set approach, needs as input the desired number of clusters, we selected an optimal one after a careful tuning phase.

In Fig. 5 we report the average computational times (in logarithmic scale) per image obtained with the three approaches as a function of the sampling rate. Note how the computational gain of




(a) CPU time in seconds (b) Error

**Fig. 3.** Experiments on image registration under similarity transformation. We plotted the error in the estimation of the transformation and the CPU time in seconds (in logarithmic scale) of INIMDYN, RD and EXPRD as a function of the number of associations in $\mathbb{A}$.

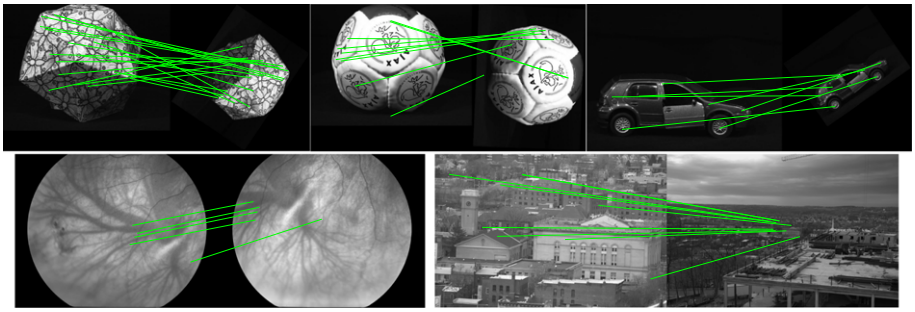


**Fig. 4.** Examples of matched correspondences for image registration. In the first row, three examples of images from the ALOI dataset. In the second row, two challenging image registration instances from [64].

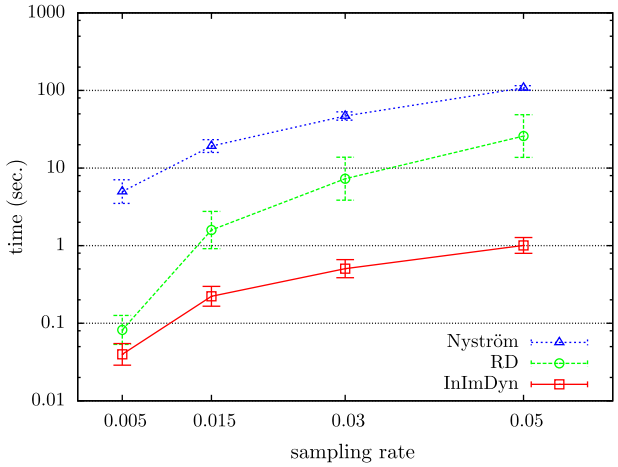



**Fig. 5.** Average CPU times (in logarithmic scale) for image segmentation over the Berkeley dataset at varying pixel sampling rates.

InImDyn over both competitors is remarkable and it clearly increases at larger sampling rates.

As for the quality of the segmentation results, we report in Fig. 6 the average precision/recall obtained in the experiment with the different sampling rates. As can be seen, all the approaches perform equivalently. In particular, as expected, RD and InImDyn achieved precisely the same results. Of course, better results might be obtained by incorporating more information (e.g., texture and contours [30]), but achieving state-of-the-art segmentation performances was beyond the scope of this work. Fig. 7 shows three segmentation examples from the Berkeley dataset. From left to right we find the original image and the segmentations obtained by our approach, RD and the Nyström method. As we can see the three approaches achieve approximately the same results, but our approach is order of magnitude faster.

### 6.4. Region-based hierarchical image matching

In [56] the authors present an approach to region-based hierarchical image matching, aimed at identifying the most similar regions in two images, according to a similarity measure defined in terms of geometric and photometric properties. To this end, each image is mapped into a tree of recursively embedded regions, obtained by a multiscale segmentation algorithm. In this way the image matching problem is cast into a tree matching problem, that is solved recursively through a set of sub-matching problems, each of which is then attacked using standard replicator dynamics (see [56] for details). Given that, typically, hundreds of sub-matching problems are generated by a single image matching instance, it is of primary importance to have at one's disposal a fast matching

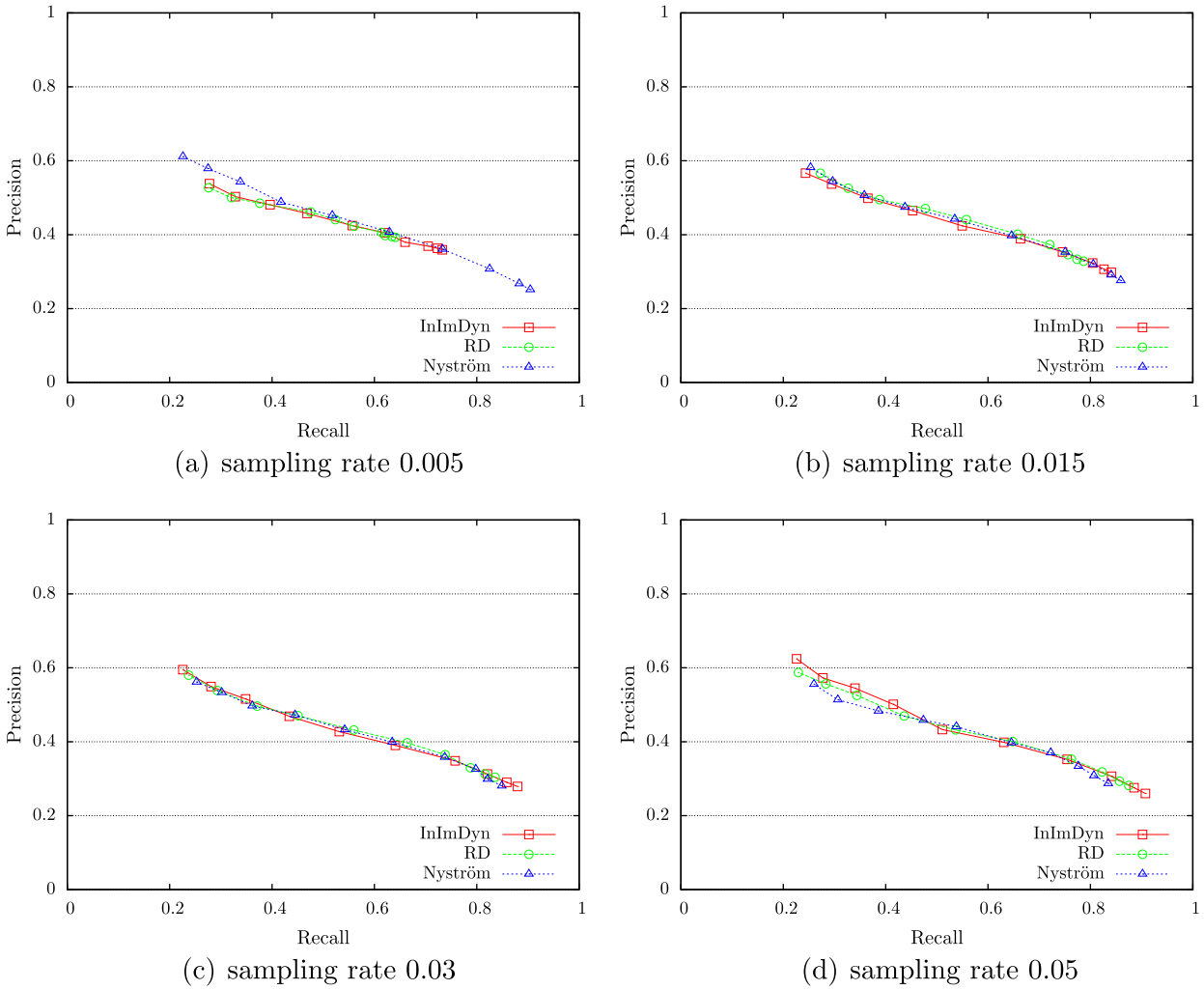



**Fig. 6.** Precision/recall plots obtained on the Berkeley Image Database for various sampling rates.

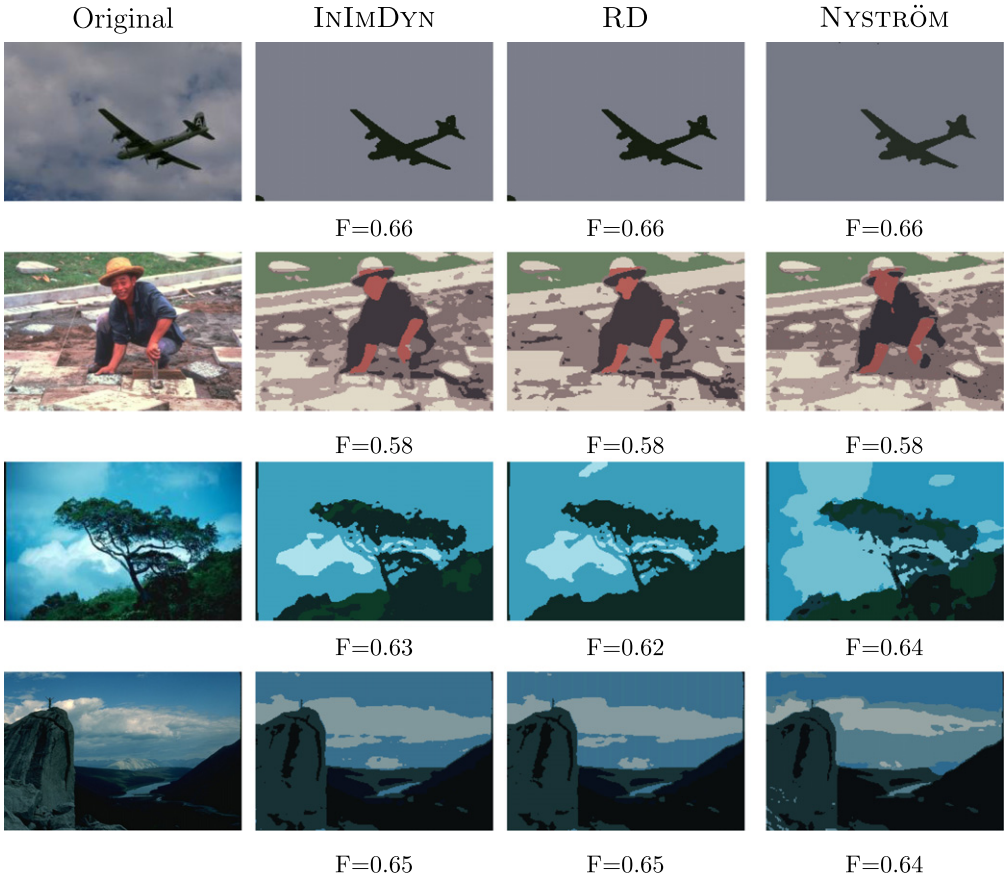



**Fig. 7.** Examples of color-based image segmentation. From left to right we find the original image and the segmentations obtained by our approach, RD and the Nyström method. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

**Fig. 8.** Examples of region-based hierarchical image matching. Each row presents a pair of images that have been matched (left) and the best matched regions that have been obtained (right).

algorithm. This makes our solution particularly appealing for this application.

We compared the running time of InImDyn and RD over a set of images taken from the original paper [56] (for the same reason as before, we do not report the results of expRD). Fig. 8 shows an example of image pairs used in the experiments and the most similar regions that have been matched: each row presents a pair of images that have been matched (left) and the best matched regions that have been obtained (right).

Fig. 9 shows the average computation times (in seconds) needed by RD and InImDyn to solve the set of sub-matching problems generated from 10 image matching instances. Since each image matching problem generated sub-matching problems of different sizes, we grouped the instances having approximately the same size together. We plotted the average running time within each group (in logarithmic scale) as a function of the instance sizes and reported the standard deviations as error bars. Again, as can be seen, InImDyn turned out to be orders of magnitude faster than RD.

## 7. Conclusions

Many computer vision and pattern recognition problems can be cast into standard quadratic programs (StQPs). Having efficient and effective algorithms for this problem at one's disposal is thus very

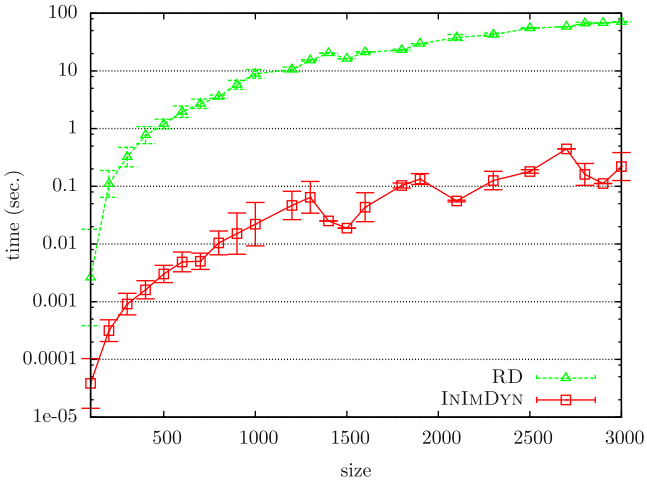



**Fig. 9.** Average CPU times (in logarithmic scale) at varying instance sizes for solving sub-matching problems, which arise in hierarchical image matching.

important. In this paper we introduced a new population game dynamics for solving StQPs and reviewed some application where this class of quadratic optimization problems arises in connection to graph-theoretical problems. Our dynamics is inspired by evolutionary game-theoretic principles and exhibits linear space and time complexity per iteration, as opposed to the quadratic one of the replicator dynamics, which is the standard approach to solving StQPs. The dynamical behavior of our dynamics is governed by a quadratic Lyapunov function which strictly increases along non-constant trajectories, thereby making it a viable alternative to replicator dynamics for solving StQPs. We demonstrate the effectiveness of our approach over various graph-theoretic problems arising in computer vision. The results show that our dynamics is dramatically faster than standard approaches, while preserving the quality of the solution found. Future works include extending the proposed algorithm over a multi-population setting, with applications to multi-StQPs [14], as well as studying variations of the proposed dynamics.

## Acknowledgments

We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250).

## References

[1] A. Albarelli, A. Torsello, S. Rota Bulò, M. Pelillo, Matching as a non-cooperative game, in: Int. Conf. Comput. Vis. (ICCV), 2009.
[2] H. Barrow, R.M. Burstall, Subgraph isomorphism, matching relational structures and maximal cliques, Inf. Process. Lett. 4 (4) (1976) 83–84.
[3] M. Basu, H. Bunke, A. Del Bimbo (Eds.), Syntactic and Structural Pattern Recognition, 2005 (Special Issue on IEEE Trans. Pattern Anal. Machine Intell., vol. 27, no. 7).
[4] L.E. Baum, J.A. Eagon, An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology, Bull. Am. Math. Soc. 73 (1967) 360–363.
[5] L.E. Baum, G.R. Sell, Growth transformations for functions on manifolds, Pacific J. Math. 27 (1968) 221–227.
[6] R.C. Bolles, R.A. Cain, Recognizing and locating partially visible objects: the local-feature-focus method, Int. J. Robot. Res. 1 (n) (1982) 57–82.
[7] I. Bomze, M. Pelillo, V. Stix, Approximating the maximum weight clique using replicator dynamics, IEEE Trans. Neural Networks 11 (2000) 1228–1241.
[8] I.M. Bomze, Evolution towards the maximum clique, J. Global Optimiz. 10 (2) (1997) 143–164.
[9] I.M. Bomze, Portfolio selection via replicator dynamics and projections of indefinite estimated covariances, Dyn. Cont., Discr. Impulsive Syst. B 12 (2005) 527–564.
[10] I.M. Bomze, M. Budinich, P.M. Pardalos, M. Pelillo, The maximum clique problem, in: Handbook of Combinatorial Optimization (Supplement Volume A), Kluwer Academic Publishers, Boston, MA, 1999, pp. 1–74.
[11] I.M. Bomze, F. Frommlet, M. Rubey, Improved SDP bounds for minimizing quadratic functions over the $\ell^1$-ball, Optim. Lett. 1 (2007) 49–59.
[12] I.M. Bomze, M. Locatelli, F. Tardella, Efficient and cheap bounds for (standard) quadratic optimization. Tech. rep., University "La Sapienza" of Rome, 2005.
[13] I.M. Bomze, B.M. Pötscher, Game Theoretical Foundations of Evolutionary Stability, Springer, 1989.
[14] I.M. Bomze, W. Schachinger, Multi-standard quadratic optimization problems: interior point methods and cone programming reformulation, Comput. Optim. Appl. 45 (2) (2010) 237–256.
[15] I.M. Bomze, J.W. Weibull, Does neutral stability imply Lyapunov stability?, Games Econ Behav. 11 (1995) 173–192.
[16] R.T. Chin, C.R. Dyer, Model-based recognition in robot vision, Comput. Surveys 18 (1) (1986) 67–108.
[17] J.F. Crow, M. Kimura, An Introduction to Population Genetics Theory, Harper & Row, New York, 1970.
[18] S. Dickinson, M. Pelillo, R. Zabih (Eds.), Graph Algorithms in Computer Vision, 2001 (Special Issue of IEEE Trans. Pattern Anal. Machine Intell., vol. 23, no. 10).
[19] R.A. Fisher, The Genetical Theory of Natural Selection, Oxford University Press, London, UK, 1930.
[20] F. Florian, Tag SNP selection based on clustering according to dominant sets found using replicator dynamics, Adv. Data Anal. Classif. 4 (2010) 65–83.
[21] H.K. Fung, S. Rao, C.A. Floudas, O.A. Prokopyev, P.M. Pardalos, F. Rendl, Computational comparison studies of quadratic assignment like formulations for the in silico sequence selection problem in de novo protein design, J. Comb. Optim. 10 (1) (2005) 41–60.
[22] J.M. Geusebroek, G.J. Burghouts, A.W.M. Smeulders, The amsterdam library of object images, Int. J. Comput. Vis. 61 (1) (2005) 103–112.
[23] L.E. Gibbons, D.W. Hearn, P.M. Pardalos, M.V. Ramana, Continuous characterizations of the maximum clique problem, Math. Oper. Res. 22 (1997) 754–768.
[24] R. Hamid, A. Johnson, S. Batta, A. Bobick, C. Isbell, G. Coleman, Detection and explanation of anomalous activities: representing activities as bags of event n-grams, in: IEEE Conf. Comput. Vis. Patt. Recogn. (CVPR), vol. 1, 2005, pp. 20–25.
[25] J. Hofbauer, K. Sigmund, Evolutionary Games and Population Dynamics, Cambridge University Press, 1998.
[26] R. Horaud, T. Skordas, Stereo correspondence through feature grouping and maximal cliques, IEEE Trans. Pattern Anal. Machine Intell. 11 (11) (1989) 1168–1180.
[27] Y. Liu, Replicator dynamics in the iterative process for accurate range image matching, Int. J. Comput. Vis. 83 (1) (2009) 30–56.
[28] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 20 (2003) 91–110.
[29] Y. Lyubich, G.D. Maistrowskii, Y.G. Ol'khovskii, Selection-induced convergence to equilibrium in a single-locus autosomal population, Probl. Inf. Transm. 16 (1980) 66–75.
[30] J. Malik, S. Belongie, T. Leung, J. Shi, Contour and texture analysis for image segmentation, Int. J. Comput. Vis. 43 (1) (2001) 7–27.
[31] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Int. Conf. Comput. Vis. (ICCV), vol. 2, July 2001, pp. 416–423.
[32] J. Maynard Smith, Evolution and the Theory of Games, Cambridge University Press, 1982.
[33] T.S. Motzkin, E.G. Straus, Maxima for graphs and a new proof of a theorem of Turán, Can. J. Math. 17 (1965) 533–540.
[34] K. Müller, J. Neumann, M. Grigutsch, D.Y. von Cramon, G. Lohmann, Detecting groups of coherent voxels in functional MRI data using spectral analysis and replicator dynamics, J. Magn. Reson. Imag. 26 (6) (2007) 1642–1650.
[35] J. Neumann, D.Y. von Cramon, B.U. Forstmann, S. Zysset, G. Lohmann, The parcellation of cortical areas using replicator dynamics in fMRI, NeuroImage 32 (1) (2006) 208–219.
[36] H. Ogawa, Labeled point pattern matching by delaunay triangulation and maximal cliques, Pattern Recogn. 19 (1) (1986) 35–40.
[37] P.M. Pardalos, W.A. Chaovalitwongse, L.D. Iasemidis, J.C. Sackellares, D.S. Shiau, P.R. Carney, O.A. Prokopyev, V.A. Yatsenko, Seizure warning algorithm based on optimization and nonlinear dynamics, Math. Program. 101 (2) (2004) 365–385.
[38] M. Pavan, M. Pelillo, Dominant sets and hierarchical clustering, in: Int. Conf. Comput. Vis. (ICCV), vol. 1, 2003, pp. 362–369.
[39] M. Pavan, M. Pelillo, A new graph-theoretic approach to clustering and segmentation, in: IEEE Conf. Comput. Vis. Patt. Recogn. (CVPR), vol. 1, 2003, pp. 145–152.
[40] M. Pavan, M. Pelillo, Efficient out-of-sample extension of dominant-set clusters, Adv. Neural Inform. Process. Syst. (NIPS) 17 (2005) 1057–1064.
[41] M. Pavan, M. Pelillo, Dominant sets and pairwise clustering, IEEE Trans. Pattern Anal. Machine Intell. 29 (1) (2007) 167–172.
[42] M. Pelillo, Replicator equations, maximal cliques, and graph isomorphism, Neural Comput. 11 (8) (1999) 1933–1955.
[43] M. Pelillo, Matching free trees, maximal cliques, and monotone game dynamics, IEEE Trans. Pattern Anal. Machine Intell. 24 (11) (2002) 1535–1541.
[44] M. Pelillo, Replicator dynamics in combinatorial optimization, in: C.A. Floudas, P.M. Pardalos (Eds.), Encyclopedia of Optimization, second ed., Springer, 2009, pp. 3279–3291.
[45] M. Pelillo, K. Siddiqi, S.W. Zucker, Matching hierarchical structures using association graphs, IEEE Trans. Pattern Anal. Machine Intell. 21 (11) (1999) 1105–1120.

[46] M. Pelillo, K. Siddiqi, S.W. Zucker, Many-to-many matching of attributed trees using association graphs and game dynamics, in: Visual Forms, 2001, pp. 583–593.
[47] M. Pelillo, A. Torsello, Payoff-monotonic game dynamics and the maximum clique problem, Neural Comput. 18 (5) (2006) 1215–1258.
[48] B. Radig, Image sequence analysis using relational structures, Pattern Recogn. 17 (1) (1984) 161–167.
[49] A. Rosenfeld, R. Hummel, S.W. Zucker, Scene labeling by relaxation operations, IEEE Trans. Syst. Man & Cybern. 6 (1976) 420–433.
[50] S. Rota Bulò, I.M. Bomze, Infection and immunization: a new class of evolutionary game dynamics, Games Econ. Behav. 71 (2011) 193–211.
[51] S. Rota Bulò, I.M. Bomze, M. Pelillo, Fast population game dynamics for dominant sets and other quadratic optimization problems, in: Int. Work. Struct. Synt. Patt. Recogn, 2010, pp. 275–285.
[52] S. Rota Bulò, M. Pelillo, A game-theoretic approach to hypergraph clustering, in: Adv. Neural Inform. Process. Syst. (NIPS), vol. 22, 2009, pp. 1571–1579.
[53] S. Rota Bulò, A. Torsello, M. Pelillo, A game-theoretic approach to partial clique enumeration, Image Vis. Comput. 27 (7) (2009) 911–922.
[54] S. Sabesan, N. Chakravarthy, K. Tsakalis, P.M. Pardalos, L.D. Iasemidis, Measuring resetting of brain dynamics at epileptic seizures: application of global optimization and spatial synchronization techniques, J. Comb. Optim. 17 (1) (2009) 74–97.
[55] P. Suetens, P. Fua, A.J. Hanson, Computational strategies for object recognition, Comput. Surveys 24 (1) (1992) 5–61.
[56] S. Todorovic, N. Ahuja, Region-based hierarchical image matching, Int. J. Comput. Vis. 78 (1) (2008) 47–66.
[57] A. Torsello, S. Rota Bulò, M. Pelillo, Grouping with asymmetric affinities: a game-theoretic perspective, in: IEEE Conf. Comput. Vis. Patt. Recogn. (CVPR), 2006, pp. 292–299.
[58] P. Tseng, I.M. Bomze, W. Schachinger, A first-order interior-point method for linearly constrained smooth optimization, Math. Program. 127 (2) (2011) 399–424.
[59] M. Wang, Z.L. Ye, Y. Wang, S.X. Wang, Dominant sets clustering for image retrieval, Signal Process. 88 (11) (2008) 2843–2849.
[60] F.R. Waugh, R.M. Westervelt, Analog neural networks with local competition dynamics and stability, Phys. Rev. E 47 (6) (1993) 4524–4536.
[61] Q.D. Wei, W.M. Hu, X.Q. Zhang, G. Luo, Dominant sets-based action recognition using image sequence matching, in: Int. Conf. Image Process. (ICIP), vol. 6, 2007, pp. 133–136.
[62] J.W. Weibull, Evolutionary Game Theory, Cambridge University Press, 1995.
[63] H. Wilf, The uniform selection of free trees, J. Algorithms 2 (1981) 204–207.
[64] G. Yang, C.V. Stewart, M. Sofka, C.L. Tsai, Registration of challenging image pairs: initialization, estimation, and decision, IEEE Trans. Pattern Anal. Machine Intell. 29 (11) (2007) 1973–1989.

# A Graph-Based Approach to Feature Selection

Zhihong Zhang and Edwin R. Hancock

Department of Computer Science, University of York, UK

**Abstract.** In many data analysis tasks, one is often confronted with very high dimensional data. The feature selection problem is essentially a combinatorial optimization problem which is computationally expensive. To overcome this problem it is frequently assumed either that features independently influence the class variable or do so only involving pairwise feature interaction. To tackle this problem, we propose an algorithm consisting of three phases, namely, i) it first constructs a graph in which each node corresponds to each feature, and each edge has a weight corresponding to mutual information (MI) between features connected by that edge, ii) then perform dominant set clustering to select a highly coherent set of features, iii) further selects features based on a new measure called multidimensional interaction information (MII). The advantage of MII is that it can consider third or higher order feature interaction. By the help of dominant set clustering, which separates features into clusters in advance, thereby allows us to limit the search space for higher order interactions. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard data-sets.

## 1 Introduction

High-dimensional data pose a significant challenge for pattern recognition. The most popular methods for reducing dimensionality are variance based subspace methods such as PCA. However, the extracted PCA feature vectors only capture sets of features with a significant combined variance, and this renders them relatively ineffective for classification tasks. Hence it is crucial to identify a smaller subset of features that are informative for classification and clustering. The idea underpinning feature selection is to select the features that are most relevant to classification while reducing redundancy. Mutual information provides a principled way of measuring the mutual dependence of two variables, and has been used by a number of researchers to develop information theoretic feature selection criteria. For example, Batti [1] has developed the Mutual Information-Based Feature Selection (MIFS) criterion, where the features are selected in a greedy manner. Given a set of existing selected features $S$, at each step it locates the feature $x_i$ that maximize the relevance to the class $I(x_i; C)$. The selection is regulated by a proportional term $\beta I(x_i; S)$ that measures the overlap information between the candidate feature and existing features. The parameter $\beta$ may significantly affect the features selected, and its control remains an open problem. Peng et al [7] on the other hand, use the so-called Maximum-Relevance

X. Jiang, M. Ferrer, and A. Torsello (Eds.): GbRPR 2011, LNCS 6658, pp. 205–214, 2011.

Minimum-Redundancy criterion (MRMR), which is equivalent to MIFS with $\beta = \frac{1}{n-1}$. Yang and Moody's [9] Joint Mutual Information (JMI) criterion is based on conditional MI and selects features by checking whether they bring additional information to an existing feature set. This method effectively rejects redundant features. Kwak and Choi [5] improve MIFS by developing MIFS-U under the assumption of a uniform distribution of information for input features. It calculates the MI based on a Parzen window, which is less computationally demanding and also provides better estimates.

However, there are two limitations for the above MI feature selection methods. Firstly, they assume that each individual relevant feature should be dependent with the target class. This means that if a single feature is considered to be relevant it should be correlated with the target class, otherwise the feature is irrelevant [2]. So only a small set of relevant features is selected, and larger feature combinations are not considered. The second weakness is that most of the methods simply consider pairwise feature dependencies, and do not check for third or higher order dependencies between the candidate features and the existing features. To overcome the above problem, we introduce the so called multidimensional interaction information (MII) $I(F;C) = I(f_1, \ldots, f_m; C)$ to select the optimal subset of features. The main reason for using $I(F;C)$ as feature selection criterion is that: because $I(F;C)$ is a measure of the reduction of uncertainty in class $C$ due to the knowledge of feature vector $F = \{f_1, \ldots, f_m\}$, selecting features that maximize $I(F;C)$, from an information theoretic perspective, translates into selecting those features that contain the maximum information about class $C$.

Although an MII based on the second-order feature dependence assumption can be used to select features that both maximize class-separability and simultaneously minimize dependencies between feature pairs, there is no reason to assume that the final optimal feature subset formed by features that only exhibit pairwise interactions. In particular the approach neglects the fact that third or higher order dependencies feature combinations may determine the optimal feature subset.

The primary reason for using the approximation $\widehat{I}(F;C)$ for feature selection instead of directly using multidimensional interaction information $I(F;C)$ is that $I(F;C)$ requires estimation of the joint probability distribution for features using large training samples. To overcome this problem, in this paper, we propose a graph-based approach to feature selection. In this feature selection scheme, the original features are clustered into different dominant-sets based on dominant-set clustering and each dominant-set just includes a small set of features. Therefore, for each dominant set, we do not need to use the approximation $\widehat{I}(F;C)$. Instead we can directly use the multidimensional interaction information $I(F;C)$ criterion for feature selection. Using the Parzen window for probability distribution estimation, we then apply a greedy strategy to incrementally select the feature that maximizes the multidimensional mutual information between the already selected features and the output class set.

## 2 Dominant-Set Clustering Algorithm

**Concept of Dominant Set:** The dominant set[6], is a combinational concept in graph theory that generalizes the notion of a maximal complete subgraph from simple graphs to edge-weighted graphs. In fact, dominant sets turn out to be equivalent to maximal cliques. The definition of the dominant set simultaneously emphasizes internal homogeneity and together with external inhomogeneity. Thus it is can be used as a general definition of a "cluster". To provide an example, assume there are $N$ training samples, each having 5 feature vectors. In order to capture the dominant features from these 5 features (represented as $F_1, \ldots, F_5$), we construct a graph $G = (V, E)$ with node-set $V$, edge-set $E \subseteq V \times V$ and edge weight matrix $W$ whose elements are in the interval $[0, 1]$. Each vertex represents a feature and the edge between two features represents their pairwise relationship. The weight on the edge reflects the degree of relevance between two features. Therefore, we represent the graph $G$ with the corresponding edge-weight or weighted relevance matrix. In our example, in Fig. 1, features $\{F_1, F_2, F_3\}$ form the dominant set, since the edge weights "internal" to that set (0.6, 0.7 and 0.9) are larger than the sum of those between the internal and external features (which is between 0.05 and 0.25).

For the graph $G = (V, E)$ above, we can locate the dominant set by finding the solutions of a quadratic program that maximizes the functional

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{W}\mathbf{x} \; . \tag{1}$$

subject to $\mathbf{x} \in \triangle$, where $\triangle = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \geq 0 and \sum_{i=1}^{n} x_i = 1\}$ and $\mathbf{W}$ is the relevance weight matrix between features. The dominant set corresponds in the strict sense with solutions of the quadratic program. Let $u$ denote a strict local solution of the above program. It has been proved by [6] that $\sigma(u) = \{i|u_i > 0\}$ is equivalent to a dominant set of the graph represented by the edge-weight matrix $\mathbf{W}$. In addition, the local maximum of $f(u)$ indicates the "cohesiveness" of the corresponding cluster. The replicator equation can be used to solve the program using the iterative update equation:

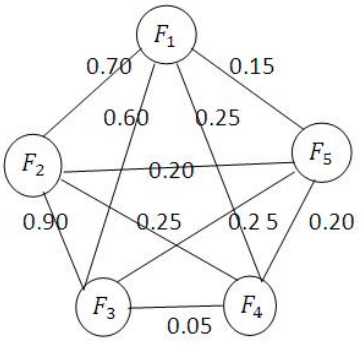



**Fig. 1.** The subset of features $\{F_1, F_2, F_3\}$ is dominant

$$x_i(t+1) = x_i(t)\frac{(\mathbf{W}\mathbf{x}(t))_i}{\mathbf{x}(t)^T\mathbf{W}\mathbf{x}(t)} \ . \tag{2}$$

where $x_i(t)$ is correspondent to the $i-th$ feature vector at iteration $t$ of the update process.

**Dominant-Set Clustering Algorithm:** Pavan et al have demonstrated that the concept of a dominant set provides an effective framework for iterative pairwise clustering. Consider a set of features represented by an undirected edge-weighted graph with no self-loops. Let the graph be denoted by $G = (V, E, \omega)$ where $V = 1, \ldots, n$ is the vertex set, $E \subseteq V \times V$ is the edge set, and $\omega$ is the weight function. Each vertex represents a feature and the weight residing on the edge between two nodes represents the pairwise affinity of the corresponding features. To cluster the features into coherent groups, a dominant set of the weighted graph is iteratively located, and then removed from the graph. This process is repeated until the node-set of the graph is empty. The main property of a dominant set is that the overall similarity among the internal features is greater than that between the external features and the internal features.

## 3 Feature Selection Using Dominant-Set Clustering

In this paper we aim to utilize the dominant-set clustering algorithm for feature selection. Using a graph representation of the features, there are three steps to the algorithm, namely a) computing the relevance matrix $\mathbf{W} = (\mathbf{w}_{ij})_{n\times n}$ based on the mutual information between feature vectors, b) dominant-set clustering to cluster the feature vectors and c) selecting the optimal feature set from each dominant set using the multidimensional interaction information (MII) criterion. Fig. 2 shows a schematic view of the proposed method for feature selection. In the remainder of this paper we describe these elements of our feature selection algorithm in more detail.

**Computing the Relevance Matrix:** In accordance with Shannon's information theory [8], the uncertainty of a random variable $Y$ can be measured by the entropy $H(Y)$. For two variables $X$ and $Y$, the conditional entropy $H(Y|X)$ measures the remaining uncertainty about $Y$ when $X$ is known. The mutual information (MI) represented by $I(X;Y)$ quantifies the information gain about $Y$ provided by variable $X$. The relationship between $H(Y)$, $H(Y|X)$ and $I(X;Y)$ is $I(X;Y) = H(Y) - H(Y|X)$.

As defined by Shannon, the initial uncertainty for the random variable $Y$ is expressed as:

$$H(Y) = -\sum_{y\in Y} P(y)\log P(y) \ . \tag{3}$$

where $P(y)$ is the prior probability density function over $Y$. The remaining uncertainty in the variable $Y$ if the variable $X$ is known is defined by the conditional entropy $H(Y|X)$

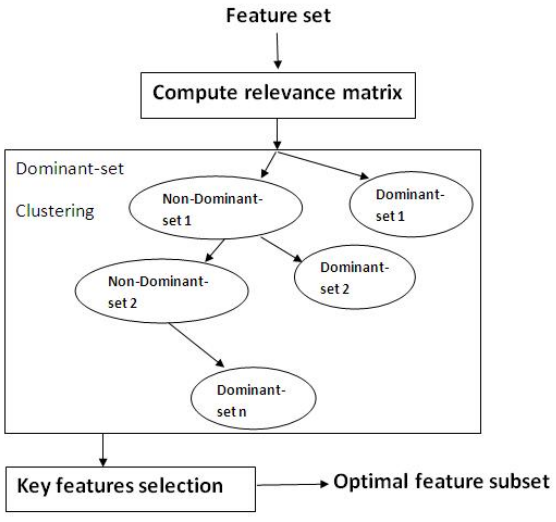



**Fig. 2.** The flowchart of our approach for feature selection

$$H(Y|X) = -\int_x p(x)\{\sum_{y\in Y} p(y|x)\log p(y|x)\}dx \ . \tag{4}$$

where $p(y|x)$ denotes the posterior probability for variable $Y$ given another random variable $X$. After observing the variable vector $x$, the amount of additional information gain is given by the mutual information (MI)

$$I(X;Y) = H(Y) - H(Y|X) = \sum_{y\in Y}\int_x p(y,x)log\frac{p(y,x)}{p(y)p(x)}dx \ . \tag{5}$$

From the above definition, we can see that mutual information quantifies the information which is shared by two variables $X$ and $Y$. When the $I(X;Y)$ is large, this implies that variable $X$ and variable $Y$ are closely related, otherwise, when $I(X;Y)$ is equal to 0, this means that two variables are totally unrelated. Therefore, in our feature selection scheme, the relevance of pairs of feature vectors is computed using mutual information. Suppose there are $N$ training samples, each having $K$ feature vectors. The $k^{th}$ feature vector for the $l^{th}$ training sample is $f_k^l$, so we can represent the $k^{th}$ feature vector for the $N$ training samples as the long vector $F_k = \{f_k^1, f_k^2, \ldots, f_k^N\}$. The entropy of the feature vector $F_k$ where $(k = 1, 2, \ldots, K)$ can be computed using Equation (3). For two feature vectors $F_{k1}$ and $F_{k2}$, their mutual information $I(F_{k1}, F_{k2})$ can be computed by Equation (5). The relevance degree between two feature vectors $F_{k1}$ and $F_{k2}$ can be defined as [10]:

$$\mathbf{W}(F_{k1}, F_{k2}) = \frac{2I(F_{k1}, F_{k2})}{H(F_{k1}) + H(F_{k2})} \ . \tag{6}$$

where $k1, k2 \in K$ and the higher the value of $\mathbf{W}(F_{k1}, F_{k2})$ the more relevant are the features $F_{k1}$ and $F_{k2}$. Otherwise, if $\mathbf{W}(F_{k1}, F_{k2}) = 0$, the two features are totally unrelated. In addition, for the above computation, we use Parzen-Rosenblatt window method to estimate the probability density function of random variables $F_{k1}$ and $F_{k2}$ [7]. The Parzen probability density estimation formula is given by: $p(x) = \frac{1}{N}\phi(\frac{x-x_i}{h})$, where $\phi(\frac{x-x_i}{h})$ is the window function and $h$ is the window width. Here, we use a Gaussian as the window function, so $\phi(\frac{x-x_i}{h}) = \frac{1}{(2\pi)^{\frac{d}{2}} h^d |\Sigma|^{\frac{1}{2}}}$ $\exp(\frac{(x-x_i^T)\Sigma^{-1}(x-x_i)}{-2h^2})$, where $\Sigma$ is the covariance of $(x - x_i)$, $d$ is the length of vector $x$. When $d = 1$, $p(x)$ estimates the marginal density and when $d = 2$, $p(x)$ estimates the joint density of variables such as $F_{k1}$ and $F_{k2}$.

**Dominant-set Clustering:** As illustrated in Fig. 2, the dominant-set clustering algorithm commences from the relevance matrix and iteratively bi-partitions the features into a dominant set and a non-dominant set. It therefore produces the dominant-set progressively and hierarchically. The clustering process stops when all the features are grouped into one of the dominant-sets.

**Selecting Key Features:** The multidimensional interaction information between feature vector $F = \{f_1, \ldots, f_m\}$ and class variable $C$ is:

$$I(F;C) = I(f_1, \ldots, f_m; C) = \sum_{f_1,\ldots,f_m} \sum_{c \in C} P(f_1, \ldots, f_m; c) \times \log \frac{P(f_1, \ldots, f_m; c)}{P(f_1, \ldots, f_m)P(c)} \ . \tag{7}$$

The main reason for using $I(F;C)$ as a feature selection criterion is that: because $I(F;C)$ is a measure of the reduction of uncertainty in class $C$ due to knowledge of the feature vector $F = \{f_1, \ldots, f_m\}$, from an information theoretic perspective selecting features that maximize $I(F;C)$ translates into selecting those features that contain the maximum information about class $C$. In practice and as noted in the introduction, locating a feature subset that maximizes $I(F;C)$ presents two problems: 1) it requires an exhaustive "combinatorial" search over the feature space, and 2) it demands large training sample sizes to estimate the higher order joint probability distribution in $I(F;C)$ with a high dimensional kernel [5]. Bearing these obstacles in mind, most of the existing related papers approximate $I(F;C)$ based on the assumption of lower-order dependencies between features. For example, the first-order class dependence assumption includes only first-order interactions. That is it assumes that each feature independently influences the class variable, so as to select the *mth* feature, $f_m$, $P(f_m|f_1, \ldots, f_{m-1}, C) = P(f_m|C)$. A second-order feature dependence assumption is proposed by Guo and Nixon [4] to approximate $I(F;C)$, and this is arguably the most simple yet effective evaluation criterion for selecting features. The approximation is given as

$$I(F;C) \approx \widehat{I}(F;C) = \sum_i I(f_i;C) - \sum_i \sum_{j>i} I(f_i;f_j) + \sum_i \sum_{j>i} I(f_i;f_j|C) \ . \tag{8}$$

By using $\widehat{I}(F;C)$ instead of $I(F;C)$, it is possible to locate a subset of informative features by implementing a greedy "pick-one-feature-at-a-time" selection procedure. Given $K$ features, out of which $m$ are to be selected ($m < K$), this involves two steps: 1) select the first feature $f'_{max}$ that maximizes $I(f';C)$, and 2) select $m-1$ subsequent features that maximize the criterion in Equation (8), i.e., select the second feature $f''_{max}$ that maximizes $I(f'';C) - I(f'';f'_{max}) + I(f'';f'_{max}|C)$, select the third feature $f'''_{max}$ that maximizes $I(f''';C) - I(f''';f'_{max}) - I(f''';f''_{max}) + I(f''';f'_{max}|C) + I(f''';f''_{max}|C)$ and so on.

Although an MII based on the second-order feature dependence assumption can select features that maximize class-separability and simultaneously minimize dependencies between feature pairs, there is no reason to assume that the final optimal feature subset is formed by pairwise interactions between features. Infact, it neglects the fact that third or higher order dependencies can be lead to an optimal feature subset.

The primary reason for using the approximation $\widehat{I}(F;C)$ for feature selection instead of directly using multidimensional interaction information $I(F;C)$ is that $I(F;C)$ requires estimation of the joint probability distribution of features using a large training sample. Consider the joint distribution $P(F) = P(f_1,\ldots,f_m)$, by the chain rule of probability

$$P(f_i,\ldots,f_m) = P(f_1)P(f_2|f_1) \times P(f_3|f_2,f_1)\cdots P(f_m|f_1,f_2\ldots f_{m-1}) \ , \quad (9)$$

$$P(F;C) = P(f_1,\ldots f_m;C) = P(C)p(f_1|C)P(f_2|f_1,C)P(f_3|f_1,f_2,C) \times P(f_4|f_1,f_2,f_3,C)\cdots P(f_i|f_1,\ldots,f_m,C) \ . \quad (10)$$

In our feature selection scheme, the original features are clustered into different dominant-sets based on dominant-set clustering and each dominant-set just includes a small set of features. Therefore, for each dominant set, we do not need to use the approximation $\widehat{I}(F;C)$. Instead, we can directly use the multidimensional interaction information $I(F;C)$ criterion for feature selection. Using Parzen windows for probability distribution estimation, we then apply the greedy strategy to select the feature that maximizes the multidimensional mutual information between the features and the output class set. As a result the first feature $f^{'}_{max}$ maximizes $I(f^{'},C)$, the second selected feature $f^{''}_{max}$ maximizes $I(f^{''},f^{'},C)$, the third feature $f^{'''}_{max}$ maximizes $I(f^{'''},f^{''},f^{'},C)$, and so on. For each dominant set, we repeat this procedure until $|S| = k$.

## 4 Experiments and Comparisons

The data sets used to test the performance of our proposed algorithm are the benchmark data sets from the NIPS 2003 feature selection challenge and the UCI Machine Learning Repository. Table. 1 summarizes the properties of these data-sets. Using the feature selection algorithm outlined above, we make a comparison between our proposed feature selection method (referred to as the DS*plus*MII method) (which utilises the multidimensional interaction information

**Table 1.** Summary of UCI and NIPS benchmark data sets

| Data-set | From | Examples | Features | Classes |
|---|---|---|---|---|
| Madelon | NIPS | 2000 | 500 | 2 |
| Breast cancer | UCI | 699 | 10 | 2 |
| Pima | UCI | 768 | 8 | 2 |

**Table 2.** The experiment results on three data-sets

| Method | Madelon | Breast cancer | Pima |
|---|---|---|---|
| MII | $\{f_{476}, f_{49}, f_{178}, f_{131}, f_{491}, f_{299}, f_{283}, f_{121}, f_{425}, f_7, f_{385}, f_{216}, f_{458}, f_{237}, f_{310}, f_{366}, f_{98}, f_{499}, f_{54}, f_{346}, f_{198}, f_{368}\}$ | $\{f_3, f_7\}$ | $\{f_2, f_8, f_6, f_7\}$ |
| DS*plus*MII | $\{f_{476}, f_{379}, f_{49}, f_{330}, f_{412}, f_{137}, f_{11}, f_{256}, f_{135}, f_{56}, f_{138}, f_{283}, f_{324}, f_{425}, f_{467}, f_{62}, f_{455}, f_{472}, f_{208}, f_{206}, f_{169}, f_{424}\}$ | $\{f_3, f_7\}$ | $\{f_2, f_8, f_6, f_1\}$ |

(MII) criterion and dominant-sets for feature selection) and the use of multidimensional interaction information (MII) using the second-order approximation, see Equation (8).

The experimental results shown in Table. 2 demonstrate that at small dimensionality, i.e. with the Breast cancer data-set(10 features and 699 examples) and the Pima data-set (8 features and 768 examples), the feature subset selected using our proposed method (i.e. DS*plus*MII ) is consistent at least to some degree with those obtained using MII with second-order approximation. However, at higher dimensionality (e.g. the Madelon data set with 500 features and 2000 examples), there is a significant difference between the selected feature subsets. There are three reasons for this. The first reason is that dominant-set clustering focuses on the information-contribution of each feature, so the most informative features can be extracted. The second reason is that the multidimensional interaction information (MII) criterion is applied to each dominant set for feature selection, and can consider the effects of third and higher order dependecies between the features and the class. As a result the optimal feature combination can be located so as to guarantee the optimal feature subset. The third and final reason is that multidimensional interaction information (MII) by second-order approximation simply checks for pair-wise dependencies between features and the class, and so only limited feature subsets can be obtained. When the database is large, our proposed method DS*plus*MII shows its advantage.

To illustrate the dominant-set clustering process for feature extraction in more detail, we list the dominant sets for Pima and Breast cancer data-set in Table. 3. By inspection, we can see that the first dominant set includes most of the important features. For example, in the Breast cancer data-set, the final selected

**Table 3.** The dominant sets for Breast cancer and Pima data-set

| Dominant-sets | Breast cancer | Pima |
|---|---|---|
| Dominant set 1 | $\{f_3,\ f_4,\ f_6,\ f_7$ $f_9,\ f_5,\ f_8\}$ | $\{f_5,\ f_2,\ f_4,\ f_8$ $f_3,\ f_6\}$ |
| Dominant set 2 | $\{f_1,\ f_2,\ f_{10}\}$ | $\{f_7,\ f_1\}$ |

**Table 4.** J value comparisons for two methods on three data sets

| Method | Madelon | Breast cancer | Pima |
|---|---|---|---|
| MII | 1.0867 | 3.7939 | 1.3867 |
| DS*plus*MII | 1.1082 | 3.7939 | 1.3977 |

features $\{f_3, f_7\}$ are all from the first dominant-set, the second dominant-set provides no further information relevant to the classication process. For the Pima data-set, most of the final selected informative features are also from the first dominant set. This reveals the advantage of our dominant-set based feature extraction method. It focuses on the information-contribution of each feature which is capable capturing the greatest number of informative features at a low computation cost. Additionally, it also indicates that not all of the dominant-sets located by dominant-set clustering are significant. It is for this reason that we utilize the multidimensional interaction information (MII) criterion for further feature selection.

After obtaining the discriminating features, we compute a scatter separability criterion to evaluate the quality of the selected feature subset. This is a well known measure of class separability introduced by Devijiver and Kittler [3], and given by

$$J(Y) = \frac{|S_w + S_b|}{|S_w|} = \prod_{k=1}^{d}(1 + \lambda_k)\ . \tag{11}$$

where $Y$ denotes the feature set, $tr(S)$ is the sum of the diagonal elements of $S$, $\lambda_k$, $k = 1 \ldots d$, are the eigenvalues of matrix $S_w^{-1}S_b$, and $S_w$ and $S_b$ are the between and within class scatter matrices.

In Table. 4, we compare the the performance of the two methods. At small dimensionality there is little difference between the two methods. However, at higher dimensionality, the features selected by our proposed DS*plus*MII method are superior to the features selected by MII based on the second-order approximation. This means that our proposed DS*plus*MII feature selection method can guarantee the optimal feature subset, as it not only focuses on the information-contribution of each feature but also considers its contribution to class.

## 5 Conclusions

This paper has presented a new graph theoretic approach to feature selection. The proposed feature selection method offers two major advantages. First,

dominant-set clustering can capture the most informative features. Second, the MII criteria takes into account high-order feature interactions, overcoming the problem of overestimated redundancy. As a result the features associated with the greatest amount of joint information can be preserved.

## References

1. Battiti, R.: Using Mutual Information for Selecting Features in Supervised Neural Net Learning. IEEE Transactions on Neural Networks 5(4), 537–550 (2002)
2. Cheng, H., Qin, Z., Qian, W., Liu, W.: Conditional Mutual Information Based Feature Selection. In: IEEE International Symposium on Knowledge Acquisition and Modeling, pp. 103–107 (2008)
3. Devijver, P., Kittler, J.: Pattern Recognition: A Statistical Approach, vol. 761. Prentice-Hall, London (1982)
4. Guo, B., Nixon, M.: Gait Feature Subset Selection by Mutual Information. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 39(1), 36–46 (2008)
5. Kwak, N., Choi, C.: Input Feature Selection by Mutual Information Based on Parzen Window. IEEE TPAMI 24(12), 1667–1671 (2002)
6. Pavan, M., Pelillo, M.: A New Graph-Theoretic Approach to Clustering and Segmentation. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1. IEEE, Los Alamitos (2003)
7. Peng, H., Long, F., Ding, C.: Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1226–1238 (2005)
8. Shannon, C.: A Mathematical Theory of Communication. ACM SIGMOBILE Mobile Computing and Communications Review 5(1), 3–55 (2001)
9. Yang, H., Moody, J.: Feature Selection Based on Joint Mutual Information. In: Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis, pp. 22–25 (1999)
10. Zhang, F., Zhao, Y., Fen, J.: Unsupervised Feature Selection based on Feature Relevance. In: International Conference on Machine Learning and Cybernetics, vol. 1, pp. 487–492. IEEE, Los Alamitos (2009)

# A Hypergraph-Based Approach to Feature Selection

Zhihong Zhang and Edwin R. Hancock*

Department of Computer Science, University of York, UK
{zhihong,erh}@cs.york.ac.uk

**Abstract.** In many data analysis tasks, one is often confronted with the problem of selecting features from very high dimensional data. The feature selection problem is essentially a combinatorial optimization problem which is computationally expensive. To overcome this problem it is frequently assumed that either features independently influence the class variable or do so only involving pairwise feature interaction. To overcome this problem, we draw on recent work on hyper-graph clustering to extract maximally coherent feature groups from a set of objects using high-order (rather than pairwise) similarities. We propose a three step algorithm that, namely, i) first constructs a graph in which each node corresponds to each feature, and each edge has a weight corresponding to the interaction information among features connected by that edge, ii) perform hypergraph clustering to select a highly coherent set of features, iii) further selects features based on a new measure called the multidimensional interaction information (MII). The advantage of MII is that it incorporates third or higher order feature interactions. This is realized using hypergraph clustering, which separates features into clusters prior to selection, thereby allowing us to limit the search space for higher order interactions. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard data-sets.

**Keywords:** Hypergraph clustering, Multidimensional interaction information(MII).

## 1 Introduction

High-dimensional data pose a significant challenge for pattern recognition. The most popular methods for reducing dimensionality are variance based subspace methods such as PCA. However, the extracted PCA feature vectors only capture sets of features with a significant combined variance, and this renders them relatively ineffective for classification tasks. Hence, it is crucial to identify a smaller subset of features that are informative for classification and clustering. The idea underpinning feature selection is to a) reduce the dimensionality of the feature space, b) speed up and reduce the cost of a learning algorithm, c) obtain

* Edwin Hancock is supported by the EU FET project SIMBAD and by a Royal Society Wolfson Research Merit Award.

A. Berciano et al. (Eds.): CAIP 2011, Part I, LNCS 6854, pp. 228–235, 2011.

the feature subset which is most relevant to classification. In practice, however, optimal feature selection requires $2^n$ feature subset evaluations, where $n$ is the original number of features and many problems related to feature selection are shown to be NP-hard [2]. Traditional feature selection methods address this issue by partitioning the original feature set into distinct clusters formed by similar features [3]. However, all of the above methods are weakened by only considering pairwise relations. In some applications higher-order relations are more appropriate to the classification task on hand, and approximating them in terms of pairwise interactions can lead to a substantial loss of information.

To overcome the above problem, in this paper, we propose a hypergraph-based approach to feature selection. Hypergraph clustering is capable of detecting high-order feature similarities. In this feature selection scheme, the original features are clustered into different groups based on hypergraph clustering and each group includes just a small set of features. In addition, for each group, a new feature selection criterion referred to as multidimensional interaction information (MII) $I(F;C)$ is applied to feature selection. In contrast to existing feature selection criterion, MII is sensitive to the relations between feature combinations and can be used to seek third or ever higher order dependencies between the relevant features. However, the limitations of the MII criterion are that it requires an exhaustive "combinatorial" search over the feature space and demands estimation of the joint probability distribution for features using large training samples. So most existing works use MII based on the second-order feature dependence assumption [1]. Since hypergraph clustering separates features into clusters in advance, this allows us to limit the search space for higher order interactions directly using the MII criterion $I(F;C)$ for feature selection. Using the Parzen window for probability distribution estimation, we apply a greedy strategy to incrementally select the features that maximize the multidimensional mutual information between the current selected features and the output class set.

## 2 Hypergraph Clustering Algorithm

**Concept of hypergraph:** A hypergraph is defined as a triplet $H = (V, E, s)$, where $V = \{1, \ldots, n\}$ is the node-set, $E$ is a set of non-empty subsets of $V$ or hyperedges and $s$ is a weight function which associates a real value with each edge. A hypergraph is a generalization of a graph. Unlike graph edges which consisting pairs of vertices, hyperedges are arbitrarily sized sets of vertices. Examples of a hypergraph are shown in Fig. 1. For the hypergraph, the vertex set is $V = \{v_1, v_2, v_3, v_4, v_5\}$, where each vertex represents a feature, and the hyperedge set is $E = \{e_1 = \{v_1, v_3\}, e_2 = \{v_1, v_2\}, e_3 = \{v_2, v_4, v_5\}, e_4 = \{v_3, v_4, v_5\}\}$. The number of vertices constituting each hyperedge represent the order of the relationship between features.

**Hypergraph Clustering Algorithm:** Let $H = (V, E, s)$ be a hypergraph clustering problem. We can locate the hypergraph cluster by finding the solutions of the following non-linear optimization problem that maximizes the functional

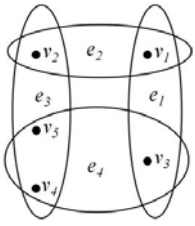



**Fig. 1.** Hypergraph example

$$f(\mathbf{x}) = \sum_{e \in E} s(e) \prod_{i \in e} x_i \ . \tag{1}$$

subject to $\mathbf{x} \in \triangle$, where $\triangle = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \geq 0, \sum_{i=1}^{n} x_i = 1\}$ and $s$ is a weight function which associates a real value with each edge. The local maximum of $f(x)$ can be solved using the Baum-Eagon inequality and leads to the iteratively updated lambda:

$$z_i = \frac{x_i \partial_i f(x)}{\sum_{j=1}^{n} x_j \partial_j f(x)}, i = 1, \ldots, n \ . \tag{2}$$

where $f(x)$ is a homogeneous polynomial in the variables $x_i$ and $z = \mathcal{M}(x)$ is a growth transformation of $x$. The Baum-Eagon inequality $f(\mathcal{M}(x)) > f(x)$ provides an effective iterative means for maximizing polynomial functions in probability domains.

## 3 Feature Selection Using Hypergraph Clustering

In this paper we aim to utilize the hypergraph clustering algorithm for feature selection. Using a hypergraph representation of the features, there are three steps to the algorithm, namely a) computing the relevance matrix **S** based on the interaction information among feature vectors, b) hypergraph clustering to cluster the feature vectors and c) selecting the optimal feature set from each cluster using the multidimensional interaction information (MII) criterion. In the remainder of this paper we describe these elements of our feature selection algorithm in more detail.

**Computing the Relevance Matrix:** In accordance with Shannon's information theory, the uncertainty of a random variable $Y$ can be measured by the entropy $H(Y)$. For two variables $X$ and $Y$, the conditional entropy $H(Y|X)$ measures the remaining uncertainty about $Y$ when $X$ is known. The mutual information (MI) represented by $I(X;Y)$ quantifies the information gain about $Y$ provided by variable $X$. The relationship between $H(Y)$, $H(Y|X)$ and $I(X;Y)$ is $I(X;Y) = H(Y) - H(Y|X)$. As defined by Shannon, the initial uncertainty for the random variable $Y$ is expressed as: $H(Y) = -\sum_{y \in Y} P(y) \log P(y)$, where

$P(y)$ is the prior probability density function over $y \in Y$. The remaining uncertainty in the variable $Y$ if the variable $X$ is known is defined by the conditional entropy $H(Y|X) = -\int_x p(x)\{\sum_{y\in Y} p(y|x)\log p(y|x)\}dx$, where $p(y|x)$ denotes the posterior probability for variable $y \in Y$ given another random variable $x \in X$. After observing the variable vector $x$, the amount of additional information gain is given by the mutual information (MI) $I(X;Y) = \sum_{y\in Y}\int_x p(y,x)log\frac{p(y,x)}{p(y)p(x)}dx$.

From the above definition, we can see that mutual information quantifies the information which is shared by two variables $X$ and $Y$. When the $I(X;Y)$ is large, this implies that variable $x \in X$ and variable $y \in Y$ are closely related, otherwise, when $I(X;Y)$ is equal to 0, this means that two variables are totally unrelated. Analogically, the conditional mutual information of $X$ and $Y$, denoted as $I(X;Y|Z) = H(X|Z) - H(X|Y,Z)$, represents the quantity of information shared by $X$ and $Y$ when $Z$ is known. The conditioning on a third random variable may either increase or decrease the original mutual information.That is, the difference between the conditional mutual information and the simple mutual information, referred to as the Interaction Information is:

$$I(X;Y;Z) = I(X;Y|Z) - I(X;Y)\ . \tag{3}$$

The interaction information measures the influence of the variable $Z$ on the amount of information shared between variables $\{Y,X\}$, the value can be positive, negative, or zero. A zero value means that the relation between $X$ and $Y$ is entirely because of $Z$. A positive value means that $X$ and $Y$ are independent of each other. However, when combined with $Z$, $X$ and $Y$ are correlated with each other. A negative value indicates that $Z$ can account for or explain the correlation between $X$ and $Y$. The extension of interaction information to $n$ variables is defined recursively,

$$I(\{X_1,\ldots,X_n\}) = I(\{X_1,\ldots,X_{n-1}\}|X_n) - I(\{X_1,\ldots,X_{n-1}\})\ . \tag{4}$$

In our feature selection scheme, the high-order relevance of features is computed using interaction information. Suppose there are $N$ training samples, each having $K$ feature vectors. The $k^{th}$ feature vector for the $l^{th}$ training sample is $f_k^l$, and so we can represent the $k^{th}$ feature vector for the $N$ training samples as the long vector $F_k = \{f_k^1, f_k^2, \ldots, f_k^N\}$. For three feature vectors $F_{k1}$, $F_{k2}$ and $F_{k3}$, their interaction information $I(F_{k1}, F_{k2}, F_{k3})$ can be computed by Equation (3). The relevance degree among three feature vectors $F_{k1}$, $F_{k2}$ and $F_{k3}$ can be defined as

$$\mathbf{S}(F_{k1}, F_{k2}, F_{k3}) = \frac{3I(F_{k1}, F_{k2}, F_{k3})}{H(F_{k1}) + H(F_{k2}) + H(F_{k3})}\ . \tag{5}$$

where $k1, k2, k3 \in K$ and the higher the value of $\mathbf{S}(F_{k1}, F_{k2}, F_{k3})$ the more relevant are the features $F_{k1}$, $F_{k2}$ and $F_{k3}$. Otherwise, if $\mathbf{S}(F_{k1}, F_{k2}, F_{k3}) = 0$, the three features are totally unrelated. In addition, for the above computation, we use Parzen-Rosenblatt window method to estimate the probability density function of random variables $F_{k1}$, $F_{k2}$ and $F_{k3}$. The Parzen probability density

estimation formula is given by: $p(x) = \frac{1}{N}\phi(\frac{x-x_i}{h})$, where $\phi(\frac{x-x_i}{h})$ is the window function and $h$ is the window width. Here, we use a Gaussian as the window function, so $\phi(\frac{x-x_i}{h}) = \frac{1}{(2\pi)^{\frac{d}{2}}h^d|\Sigma|^{\frac{1}{2}}}\exp(\frac{(x-x_i^T)\Sigma^{-1}(x-x_i)}{-2h^2})$, where $\Sigma$ is the covariance of $(x - x_i)$, $d$ is the length of vector $x$. When $d = 1$, $p(x)$ estimates the marginal density and when $d = 3$, $p(x)$ estimates the joint density of variables such as $F_{k1}$, $F_{k2}$ and $F_{k3}$.

**Hypergraph Clustering:** the hypergraph clustering algorithm commences from the relevance matrix and iteratively bi-partitions the features into a foreground cluster and a background cluster. It locates the foreground cluster progressively and hierarchically. The clustering process stops when all the features are grouped into either the foreground or background cluster.

**Selecting Key Features:** The multidimensional interaction information between feature vector $F = \{f_1, \ldots, f_m\}$ and class variable $C$ is:

$$I(F;C) = \sum_{f_1,\ldots,f_m}\sum_{c\in C} P(f_1,\ldots,f_m;c) \times \log\frac{P(f_1,\ldots,f_m;c)}{P(f_1,\ldots,f_m)P(c)} \ . \qquad (6)$$

The main reason for using $I(F;C)$ as a feature selection criterion is that since $I(F;C)$ is a measure of the reduction of uncertainty in class $C$ due to knowledge of the feature vector $F = \{f_1, \ldots, f_m\}$, from an information theoretic perspective selecting features that maximize $I(F;C)$ translates into selecting those features that contain the maximum information about class $C$. In practice, and as noted in the introduction, locating a feature subset that maximizes $I(F;C)$ presents two problems: 1) it requires an exhaustive "combinatorial" search over the feature space, and 2) it demands large training sample sizes to estimate the higher order joint probability distribution in $I(F;C)$ with a high dimensional kernel [6]. Bearing these obstacles in mind, most of the existing related papers approximate $I(F;C)$ based on the assumption of lower-order dependencies between features. For example, the first-order class dependence assumption includes only first-order interactions. That is, it assumes that each feature independently influences the class variable, so as to select the *mth* feature, $f_m$, $P(f_m|f_1,\ldots,f_{m-1},C) = P(f_m|C)$. A second-order feature dependence assumption is proposed by Guo and Nixon [5] to approximate $I(F;C)$, and this is arguably the most simple yet effective evaluation criterion for selecting features. The approximation is given as

$$I(F;C) \approx \widehat{I}(F;C) = \sum_i I(f_i;C) - \sum_i\sum_{j>i} I(f_i;f_j) + \sum_i\sum_{j>i} I(f_i;f_j|C) \ . \ (7)$$

Although an MII based on the second-order feature dependence assumption can select features that maximize class-separability and simultaneously minimize dependencies between feature pairs, there is no reason to assume that the final optimal feature subset is formed by pairwise interactions between features. Infact, it neglects the fact that third or higher order dependencies can be lead to an optimal feature subset.

The primary reason for using the approximation $\widehat{I}(F;C)$ for feature selection instead of directly using multidimensional interaction information $I(F;C)$ is that $I(F;C)$ requires estimation of the joint probability distribution of features using a large training sample. Consider the joint distribution $P(F) = P(f_1, \ldots, f_m)$, by the chain rule of probability

$$P(f_i, \ldots, f_m) = P(f_1)P(f_2|f_1) \times P(f_3|f_2, f_1) \cdots P(f_m|f_1, f_2 \ldots f_{m-1}) \ , \quad (8)$$

$$P(F;C) = P(f_1, \ldots f_m; C) = P(C)p(f_1|C)P(f_2|f_1, C)P(f_3|f_1, f_2, C) \times P(f_4|f_1, f_2, f_3, C) \cdots P(f_i|f_1, \ldots, f_m, C) \ . \quad (9)$$

In our feature selection scheme, the original features are clustered into different groups based on hypergraph clustering and each cluster just includes a small set of features. Therefore, for each cluster, we do not need to use the approximation $\widehat{I}(F;C)$. Instead, we can directly use the multidimensional interaction information $I(F;C)$ criterion for feature selection. Using Parzen windows for probability distribution estimation, we then apply the greedy strategy to select the feature that maximizes the multidimensional mutual information between the features and the output class set. As a result the first feature $f^{'}_{max}$ maximizes $I(f^{'}, C)$, the second selected feature $f^{''}_{max}$ maximizes $I(f^{''}, f^{'}, C)$, the third feature $f^{'''}_{max}$ maximizes $I(f^{'''}, f^{''}, f^{'}, C)$, and so on. For each cluster, we repeat this procedure until $|S| = k$.

## 4 Experiments and Comparisons

The data sets used to test the performance of our proposed algorithm are the benchmark data sets from the UCI Machine Learning Repository. Table. 1 summarizes the properties of these data-sets. Using the feature selection algorithm outlined above, we make a comparison between our proposed feature selection method (referred to as the HG*plus*MII method) (which utilizes the multidimensional interaction information (MII) criterion and hypergraph clustering for feature selection) and the use of multidimensional interaction information (MII) using the second-order approximation (see Equation (7)).

The experimental results shown in Table. 2 demonstrate that our proposed method (i.e. HG*plus*MII ) can achieve higher degree of dimensionality reduction, as it selects a smaller feature subset compared with those obtained using MII with second-order approximation. There are three reasons for this. The first reason is that hypergraph clustering simultaneously considers the information-contribution of each feature and the correlation between features, so the structural information concealed in the data can be effectively identified. The second

**Table 1.** Summary of UCI benchmark data sets

| Data-set | Examples | Features | Classes |
|---|---|---|---|
| Australian | 690 | 14 | 2 |
| Breast cancer | 699 | 10 | 2 |
| Pima | 768 | 8 | 2 |

reason is that the multidimensional interaction information (MII) criterion is applied to each cluster for feature selection, and can consider the effects of third and higher order dependencies between the features and the class. As a result the optimal feature combination can be located so as to guarantee the optimal feature subset. The third and final reason is that second-order approximation to multidimensional interaction information (MII) simply checks for pair-wise dependencies between features and the class, and so only limited feature subsets can be obtained.

**Table 2.** The experiment results on three data-sets

| Method | Australian | Breast cancer | Pima |
|---|---|---|---|
| MII | $\{f_8,\ f_{14},\ f_5, f_{13}\}$ | $\{f_3,\ f_8,\ f_7\}$ | $\{f_2, f_8, f_6, f_7\}$ |
| HG*plus*MII | $\{f_8,\ f_9,\ f_5\}$ | $\{f_3,\ f_7,\ f_9\}$ | $\{f_2, f_6, f_1\}$ |

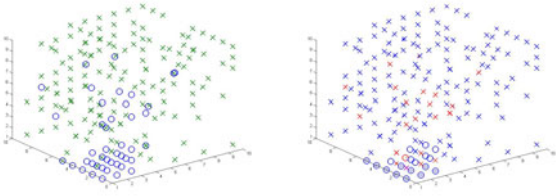
After obtaining the discriminating features, we compute a scatter separability criterion to evaluate the quality of the selected feature subset. This is a well known measure of class separability introduced by Devijiver and Kittler [4], and given by $J(Y) = \frac{|S_w+S_b|}{|S_w|} = \prod_{k=1}^{d}(1+\lambda_k)$, where $Y$ denotes the feature set, $\lambda_k$, $k = 1 \ldots d$, are the eigenvalues of matrix $S_w^{-1}S_b$, and $S_w$ and $S_b$ are the between and within class scatter matrices.

**Table 3.** J value comparisons for two methods on three data sets

| Method | Australian | Breast cancer | Pima |
|---|---|---|---|
| MII | 2.2832 | 5.0430 | 1.3867 |
| HG*plus*MII | 2.3010 | 5.1513 | 1.3942 |

In Table. 3, we compare the the performance of the two methods. We find that the effective feature subsets can be obtained using our proposed HG*plus*MII method, e.g., for dataset Australian and Pima, it can achieve a higher discriminability power based on fewer features.This means that our feature selection method can guarantee the optimal feature subset, as it not only achieves higher degree of the dimensionality reduction but also obtains better discriminability power.

After obtaining the discriminating features, we apply a variational EM algorithm to learn Gaussian mixture model on the selected feature subset for the purpose of classification. For the Breast Cancer dataset, we visualize the classification results using the selected feature subset. The classification accuracy achieved using the selected feature subset is 96.3% which is superior to the accuracy of 95.4% achieved by RD-based method [7]. The classification results are shown in Fig. 2. The left hand panel is the data with correct labeling, and the right hand panel is the classification results with the misclassified data highlighted. Because of the unsupervised nature of the variational EM algorithm and



(a) Original data (b) Classification result

**Fig. 2.** Classification result visualized on 3rd, 7th and 9th features

the Gaussian mixture model, the classification accuracy of 96.3% demonstrates the adequate class separability provided by the selected feature subset.

## 5 Conclusions

This paper has presented a new graph theoretic approach to feature selection. The proposed feature selection method offers two major advantages. First, hypergraph clustering simultaneously considers the significance of both the features and the correlation between features, and therefor the structural information concealed in the data can be more effectively utilized. Second, the MII criteria takes into account high-order feature interactions with the class, overcoming the problem of overestimated redundancy. As a result the features associated with the greatest amount of joint information can be preserved.

## References

1. Balagani, S., Phoha, V.: On the Feature Selection Criterion Based on an Approximation of Multidimensional Mutual Information. IEEE TPAMI 32(7), 1342–1343 (2010)
2. Blum, L., Rivest, L.: Training a 3-Node Neural Network is NP-complete. Neural Networks 5(1), 117–127 (1992)
3. Covões, T., Hruschka, E., de Castro, L., Santos, Á.: A Cluster-based Feature Selection Approach. In: Corchado, E., Wu, X., Oja, E., Herrero, Á., Baruque, B. (eds.) HAIS 2009. LNCS, vol. 5572, pp. 169–176. Springer, Heidelberg (2009)
4. Devijver, A., Kittler, J.: Pattern Recognition: A Statistical Approach, vol. 761. Prentice-Hall, London (1982)
5. Guo, B., Nixon, S.: Gait Feature Subset Selection by Mutual Information. IEEE TSMC, Part A: Systems and Humans 39(1), 36–46 (2008)
6. Kwak, N., Choi, H.: Input Feature Selection by Mutual Information Based on Parzen Window. IEEE TPAMI 24(12), 1667–1671 (2002)
7. Zhang, F., Zhao, Y.J., Fen, J.: Unsupervised Feature Selection based on Feature Relevance. In: ICMLC, vol. 1, pp. 487–492 (2009)

# Mutual Information Criteria for Feature Selection

Zhihong Zhang and Edwin R.Hancock

Department of Computer Science,University of York,UK

**Abstract.** In many data analysis tasks, one is often confronted with very high dimensional data. The feature selection problem is essentially a combinatorial optimization problem which is computationally expensive. To overcome this problem it is frequently assumed either that features independently influence the class variable or do so only involving pairwise feature interaction. In prior work [18], we have explained the use of a new measure called multidimensional interaction information (MII) for feature selection. The advantage of MII is that it can consider third or higher order feature interaction. Using dominant set clustering, we can extract most of the informative features in the leading dominant sets in advance, limiting the search space for higher order interactions. In this paper, we provide a comparison of different similarity measures based on mutual information. Experimental results demonstrate the effectiveness of our feature selection method on a number of standard data-sets.

## 1 Introduction

High-dimensional data pose a significant challenge for pattern recognition. The most popular methods for reducing dimensionality are variance based subspace methods such as PCA. However, the extracted PCA feature vectors only capture sets of features with a significant combined variance, and this renders them relatively ineffective for classification tasks. Hence it is crucial to identify a smaller subset of features that are informative for classification and clustering. The idea underpinning feature selection is to a) reduce the dimensionality of the feature space, b) speed up and reduce the cost of a learning algorithm, c) obtain the feature subset which is most relevant to classification. Mutual information provides a principled way of measuring the mutual dependence of two variables, and has been used by a number of researchers to develop information theoretic feature selection criteria. For example, Batti [1] has developed the Mutual Information-Based Feature Selection (MIFS) criterion, where the features are selected in a greedy manner. Given a set of existing selected features $S$, at each step it locates the feature $x_i$ that maximize the relevance to the class $I(x_i; C)$. The selection is regulated by a proportional term $\beta I(x_i; S)$ that measures the overlap information between the candidate feature and existing features. The parameter $\beta$ may significantly affect the features selected, and its control remains an open problem. Peng et al [11] on the other hand, use the so-called Maximum-Relevance Minimum-Redundancy criterion (MRMR), which is equivalent to MIFS with

$\beta = \frac{1}{n-1}$. Yang and Moody's [15] Joint Mutual Information (JMI) criterion is based on conditional MI and selects features by checking whether they bring additional information to an existing feature set. This method effectively rejects redundant features. Kwak and Choi [8] improve MIFS by developing MIFS-U under the assumption of a uniform distribution of information for input features. It calculates the MI based on a Parzen window, which is less computationally demanding and also provides better estimates.

However, there are two limitations for the above MI feature selection methods. Firstly, they assume that each individual relevant feature should be dependent with the target class. This means that if a single feature is considered to be relevant it should be correlated with the target class, otherwise the feature is irrelevant [3]. So only a small set of relevant features is selected, and larger feature combinations are not considered. The second weakness is that most of the methods simply consider pairwise feature dependencies, and do not check for third or higher order dependencies between the candidate features and the existing features. To overcome the above problem, Zhang and Hancock [18] introduce the so called multidimensional interaction information (MII) $I(F;C) = I(f_1,\ldots,f_m;C)$ to select the optimal subset of features. The main reason for using $I(F;C)$ as feature selection criterion is that: because $I(F;C)$ is a measure of the reduction of uncertainty in class $C$ due to the knowledge of feature vector $F = \{f_1,\ldots,f_m\}$, selecting features that maximize $I(F;C)$, from an information theoretic perspective, translates into selecting those features that contain the maximum information about class $C$.

In prior work[18], we have proposed a graph-based method to feature selection. In this feature selection scheme, the original features are clustered into different clusters based on dominant-set clustering and each cluster just includes a small set of features. As dominant set clustering can group most of the informative features into the leading dominant set based on suitable similarity measure, this allows us to limit the search space for further feature selection. The similarity measure used for clustering is based on mutual information. We compare the similaity measure with other two well known alternative measures of similarity, namely Pearson's correlation coefficient ($\rho$) which based on distance and the Least square regression error ($e$) is made. Using the Parzen window for probability distribution estimation, we then apply a greedy strategy to incrementally select the features that maximizes the multidimensional mutual information between the already selected features and the output class set.

## 2 Dominant-Set Clustering Algorithm

There are several different methods for clustering features, well-known examples are: k-means algorithm [9] is built for all sample, but requires a user to supply the number of clusters in advance. In addition, it can not detect clusters of arbitrary shapes. The Self Organizing Map(SOM) [14] is a type of artificial neural network which can produce a low-dimensional space for the input data objects using a neighborhood function to cluster nodes. As same with k-means

algorithm, it does not explicitly optimize any measure of the total dissimilarity to locate clusters. Again, it requires the number of clusters as user input. In this paper, we use dominant set clustering which is suitable for both subspace and high dimensional data clustering. In addition, it does not require the user to provide the number of clusters and can also handle outliers efficiently. Most importantly, it can group most of the informative features into cluster based on a suitable similarity measure.

### 2.1 Concept of Dominant Set

The dominant set[10], is a combinational concept in graph theory that generalizes the notion of a maximal complete subgraph from simple graphs to edge-weighted graphs. In fact, dominant sets turn out to be equivalent to maximal cliques. The definition of the dominant set simultaneously emphasizes internal homogeneity and together with external inhomogeneity. Thus it is can be used as a general definition of a "cluster". To provide an example, assume there are $N$ training samples, each having 5 feature vectors. In order to capture the dominant features from these 5 features (represented as $F_1, \ldots, F_5$), we construct a graph $G = (V, E)$ with node-set $V$, edge-set $E \subseteq V \times V$ and edge weight matrix $W$ whose elements are in the interval $[0, 1]$. Each vertex represents a feature and the edge between two features represents their pairwise relationship. The weight on the edge reflects the degree of relevance between two features. Therefore, we represent the graph $G$ with the corresponding edge-weight or weighted relevance matrix. In our example, in Fig. 1, features $\{F_1, F_2, F_3\}$ form the dominant set, since the edge weights "internal" to that set (0.6, 0.7 and 0.9) are larger than the sum of those between the internal and external features (which is between 0.05 and 0.25).

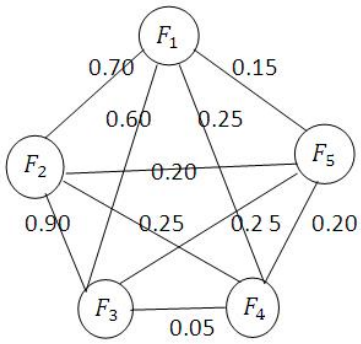



**Fig. 1.** The subset of features $\{F_1, F_2, F_3\}$ is dominant

For the graph $G = (V, E)$ above, we can locate the dominant set by finding the solutions of a quadratic program that maximizes the functional

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{W}\mathbf{x} \ . \tag{1}$$

subject to $\mathbf{x} \in \triangle$, where $\triangle = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \geq 0 and \sum_{i=1}^{n} x_i = 1\}$ and $\mathbf{W}$ is the relevance weight matrix between features. The dominant set corresponds in the strict sense with solutions of the quadratic program. Let $u$ denote a strict local solution of the above program. It has been proved by [10] that $\sigma(u) = \{i|u_i > 0\}$ is equivalent to a dominant set of the graph represented by the edge-weight matrix $\mathbf{W}$. In addition, the local maximum of $f(u)$ indicates the "cohesiveness" of the corresponding cluster. The replicator equation can be used to solve the program using the iterative update equation:

$$x_i(t+1) = x_i(t)\frac{(\mathbf{W}\mathbf{x}(t))_i}{\mathbf{x}(t)^T\mathbf{W}\mathbf{x}(t)} \ . \tag{2}$$

where $x_i(t)$ corresponded to the $i-th$ feature vector at iteration $t$ of the update process.

### 2.2 Dominant-Set Clustering Algorithm

Pavan et al have demonstrated that the concept of a dominant set provides an effective framework for iterative pairwise clustering. Consider a set of features represented by an undirected edge-weighted graph with no self-loops. Let the graph be denoted by $G = (V, E, \omega)$ where $V = 1, \ldots, n$ is the vertex set, $E \subseteq V \times V$ is the edge set, and $\omega$ is the weight function. Each vertex represents a feature and the weight residing on the edge between two nodes represents the pairwise affinity of the corresponding features. To cluster the features into coherent groups, a dominant set of the weighted graph is iteratively located, and then removed from the graph. This process is repeated until the node-set of the graph is empty. The main property of a dominant set is that the overall similarity among the internal features is greater than that between the external features and the internal features.

## 3 Feature Similarity Measure

There are different similarity measure methods that can be used for clustering and different methods may lead to different cluster results. As a result, we need to carefully select the most suitable measure to use. In general, the Euclidean distance is widely used as the distance or similarity measure for clustering [7]. However, Euclidean distance only accounts for a data which follows a particular distribution [16], it is not effective to reflect functional similarity such as positive and negative correlation and interdependency. Rao [12] introduced two approaches to measure the linear dependency between variables, namely, a)Pearson's correlation coefficient ($\rho$), b) Least square regression error ($e$).

**Pearson's correlation coefficient** ($\rho$)**:** The Correlation coefficient ($\rho$) between two random variables $x$ and $y$ is defined as:

$$\rho(x,y) = \frac{cov(x,y)}{\sqrt{var(x)var(y)}} \ . \tag{3}$$

where $var()$ denotes the variance of a variable and $cov(x,y)$ is the covariance between two random variables. From the above definition, we can see that Pearson's correlation coefficient quantifies the linear dependency between two variables $x$ and $y$. When the $\rho(x,y)$ is large (i.e. 1 or -1), this implies that variable $x$ and variable $y$ are closely related, otherwise, when $\rho(x,y)$ is equal to 0, this means that two variables are totally unrelated. As a result, the method can be used to detect positive and negative correlation. However, there are two limitations which unsuit the utility of Pearson coefficient to used for dominant set clustering. First, it is not robust to outliers and as a result it may assign a high similarity score to a pair of dissimilar features. Second, as it is sensitive to rotation and invariant to scaling, the two pairs of variables having different variances may give the same value of the similarity measure.

**Least square regression error** ($e$)**:** The dependency of two variables $x$ and $y$ can be modeled by the linear model, $y = a + bx$. As a result, the degree of dependency between them can be measure by the error in predicting $y$ from the linear model. The parameters of the model $a$ and $b$ can be learned by minimizing the mean square error as follows:

$$e(x,y)^2 = \frac{1}{n}\sum (e(x,y)_i)^2 \ . \tag{4}$$

where $e(x,y)_i = y_i - a - bx_i$, $a = \bar{y}$, $b = \frac{cov}{(x,y)var(x)}$ and $e(x,y) = var(y)(1 - \rho(x,y)^2)$. From this definition, we can see that the least square regression error ($e$) quantifies the amount of variance of $y$ unexplained by the linear model. As with Pearson's correlation coefficient ($\rho$), it is sensitive to rotation and scaling.

## 4 Feature Selection Using Dominant-Set Clustering

In this paper we aim to utilize the dominant-set clustering algorithm for feature selection. Using a graph representation of the features, there are three steps to the algorithm, namely a) computing the relevance matrix $\mathbf{W} = (\mathbf{w}_{ij})_{n\times n}$ based on the mutual information between feature vectors, b) dominant-set clustering to cluster the feature vectors and c) selecting the optimal feature set from leading dominant set using the multidimensional interaction information (MII) criterion. In the remainder of this paper we describe these elements of our feature selection algorithm in more detail.

### 4.1 Computing the Similarity Matrix

Instead of using the Euclidean distance, Pearson's correlation coefficient ($\rho$) or the least square regression error ($e$), our similarity measure employs an mutual

information measure to evaluate the interdependence of features. The use of this mutual information measure allows dominant set clustering to discover the informative features and group them into cluster. In accordance with Shannon's information theory [13], the uncertainty of a random variable $Y$ can be measured by the entropy $H(Y)$. For two variables $X$ and $Y$, the conditional entropy $H(Y|X)$ measures the remaining uncertainty about $Y$ when $X$ is known. The mutual information (MI) represented by $I(X;Y)$ quantifies the information gain about $Y$ provided by variable $X$. The relationship between $H(Y)$, $H(Y|X)$ and $I(X;Y)$ is $I(X;Y) = H(Y) - H(Y|X)$.

As defined by Shannon, the initial uncertainty for the random variable $Y$ is expressed as:

$$H(Y) = -\sum_{y \in Y} P(y) \log P(y) \ . \tag{5}$$

where $P(y)$ is the prior probability density function over $Y$. The remaining uncertainty in the variable $Y$ if the variable $X$ is known is defined by the conditional entropy $H(Y|X)$

$$H(Y|X) = -\int_x p(x) \{ \sum_{y \in Y} p(y|x) \log p(y|x) \} dx \ . \tag{6}$$

where $p(y|x)$ denotes the posterior probability for variable $Y$ given another random variable $X$. After observing the variable vector $x$, the amount of additional information gain is given by the mutual information (MI)

$$I(X;Y) = H(Y) - H(Y|X) = \sum_{y \in Y} \int_x p(y,x) log \frac{p(y,x)}{p(y)p(x)} dx \ . \tag{7}$$

From the above definition, we can see that mutual information quantifies the information which is shared by two variables $X$ and $Y$. When the $I(X;Y)$ is large, this implies that variable $X$ and variable $Y$ are closely related, otherwise, when $I(X;Y)$ is equal to 0, this means that two variables are totally unrelated. Therefore, in our feature selection scheme, the relevance of pairs of feature vectors is computed using mutual information. Suppose there are $N$ training samples, each having $K$ feature vectors. The $k^{th}$ feature vector for the $l^{th}$ training sample is $f_k^l$, so we can represent the $k^{th}$ feature vector for the $N$ training samples as the long vector $F_k = \{f_k^1, f_k^2, \ldots, f_k^N\}$. The entropy of the feature vector $F_k$ where $(k = 1, 2, \ldots, K)$ can be computed using Equation (3). For two feature vectors $F_{k1}$ and $F_{k2}$, their mutual information $I(F_{k1}, F_{k2})$ can be computed by Equation (5). The relevance degree between two feature vectors $F_{k1}$ and $F_{k2}$ can be defined as [17]:

$$\mathbf{W}(F_{k1}, F_{k2}) = \frac{2I(F_{k1}, F_{k2})}{H(F_{k1}) + H(F_{k2})} \ . \tag{8}$$

where $k1, k2 \in K$ and the higher the value of $\mathbf{W}(F_{k1}, F_{k2})$ the more relevant are the features $F_{k1}$ and $F_{k2}$. Otherwise, if $\mathbf{W}(F_{k1}, F_{k2}) = 0$, the two features are

totally unrelated. In addition, for the above computation, we use the Parzen-Rosenblatt window method to estimate the probability density function of random variables $F_{k1}$ and $F_{k2}$ [11]. The Parzen probability density estimation formula is given by: $p(x) = \frac{1}{N}\phi(\frac{x-x_i}{h})$, where $\phi(\frac{x-x_i}{h})$ is the window function and $h$ is the window width. Here, we use a Gaussian as the window function, so $\phi(\frac{x-x_i}{h}) = \frac{1}{(2\pi)^{\frac{d}{2}} h^d |\Sigma|^{\frac{1}{2}}} \exp(\frac{(x-x_i^T)\Sigma^{-1}(x-x_i)}{-2h^2})$, where $\Sigma$ is the covariance of $(x - x_i)$, $d$ is the length of vector $x$. When $d = 1$, $p(x)$ estimates the marginal density and when $d = 2$, $p(x)$ estimates the joint density of variables such as $F_{k1}$ and $F_{k2}$.

### 4.2 Dominant-set Clustering

The dominant-set clustering algorithm commences from the relevance matrix and iteratively bi-partitions the features into a dominant set and a non-dominant set. It therefore produces the dominant-set progressively and hierarchically. The clustering process stops when all the features are grouped into one of the dominant-sets. We can formulate the dominant-set clustering algorithm in the following: a) Initialize $\mathbf{W}^t$ by the similarity matrix $\mathbf{W}$, where $t = 1$. b) Calculate the local solution of Equation(1) by Equation(2): $u^t$ and $f(u^t)$. c) Get the dominant set: $DS^t = \sigma(u^t)$. d) Split out $DS^t$ from $\mathbf{W^t}$ and get a new similarity matrix $\mathbf{W^{t+1}}$. e) If $\mathbf{W^{t+1}}$ is not empty, $\mathbf{W^t} = \mathbf{W^{t+1}}$ and $t = t + 1$, then go to step b; else exit

### 4.3 Selecting Key Features

In accordance with Shannon's information theory [13], the uncertainty of a random variable $Y$ can be measured by the entropy $H(Y)$. For two variables $X$ and $Y$, the conditional entropy $H(Y|X)$ measures the remaining uncertainty about $Y$ when $X$ is known. The mutual information (MI) represented by $I(X;Y)$ quantifies the information gain about $Y$ provided by variable $X$. The relationship between $H(Y)$, $H(Y|X)$ and $I(X;Y)$ is $I(X;Y) = H(Y) - H(Y|X)$.

As defined by Shannon, the initial uncertainty for the random variable $Y$ is expressed as:

$$H(Y) = -\sum_{y \in Y} P(y) \log P(y) \ . \tag{9}$$

where $P(y)$ is the prior probability density function over $Y$. The remaining uncertainty in the variable $Y$ if the variable $X$ is known is defined by the conditional entropy $H(Y|X)$

$$H(Y|X) = -\int_x p(x)\{\sum_{y \in Y} p(y|x) \log p(y|x)\}dx \ . \tag{10}$$

where $p(y|x)$ denotes the posterior probability for variable $Y$ given another random variable $X$. After observing the variable vector $x$, the amount of additional information gain is given by the mutual information (MI)

$$I(X;Y) = H(Y) - H(Y|X) = \sum_{y\in Y}\int_x p(y,x)log\frac{p(y,x)}{p(y)p(x)}dx \ . \tag{11}$$

In addition, for the above computation, we use Parzen-Rosenblatt window method to estimate the probability density function of random variables $F_{k1}$ and $F_{k2}$ [11]. The Parzen probability density estimation formula is given by: $p(x) = \frac{1}{N}\phi(\frac{x-x_i}{h})$, where $\phi(\frac{x-x_i}{h})$ is the window function and $h$ is the window width. Here, we use a Gaussian as the window function, so $\phi(\frac{x-x_i}{h}) = \frac{1}{(2\pi)^{\frac{d}{2}}h^d|\Sigma|^{\frac{1}{2}}}\exp(\frac{(x-x_i^T)\Sigma^{-1}(x-x_i)}{-2h^2})$, where $\Sigma$ is the covariance of $(x-x_i)$, $d$ is the length of vector $x$. When $d=1$, $p(x)$ estimates the marginal density and when $d=2$, $p(x)$ estimates the joint density of variables such as $F_{k1}$ and $F_{k2}$.

The multidimensional interaction information between feature vector $F = \{f_1,\ldots,f_m\}$ and class variable $C$ is:

$$I(F;C) = I(f_1,\ldots,f_m;C) = \sum_{f_1,\ldots,f_m}\sum_{c\in C}P(f_1,\ldots,f_m;c) \times \log\frac{P(f_1,\ldots,f_m;c)}{P(f_1,\ldots,f_m)P(c)} \ . \tag{12}$$

The main reason for using $I(F;C)$ as a feature selection criterion is that: because $I(F;C)$ is a measure of the reduction of uncertainty in class $C$ due to knowledge of the feature vector $F = \{f_1,\ldots,f_m\}$, from an information theoretic perspective selecting features that maximize $I(F;C)$ translates into selecting those features that contain the maximum information about class $C$. In practice and as noted in the introduction, locating a feature subset that maximizes $I(F;C)$ presents two problems: 1) it requires an exhaustive "combinatorial" search over the feature space, and 2) it demands large training sample sizes to estimate the higher order joint probability distribution in $I(F;C)$ with a high dimensional kernel [8]. Bearing these obstacles in mind, most of the existing related papers approximate $I(F;C)$ based on the assumption of lower-order dependencies between features. For example, the first-order class dependence assumption includes only first-order interactions. That is it assumes that each feature independently influences the class variable, so as to select the $mth$ feature, $f_m$, $P(f_m|f_1,\ldots,f_{m-1},C) = P(f_m|C)$. A second-order feature dependence assumption is proposed by Guo and Nixon [5] to approximate $I(F;C)$, and this is arguably the most simple yet effective evaluation criterion for selecting features. The approximation is given as

$$I(F;C) \approx \widehat{I}(F;C) = \sum_i I(f_i;C) - \sum_i\sum_{j>i} I(f_i;f_j) + \sum_i\sum_{j>i} I(f_i;f_j|C) \ . \tag{13}$$

By using $\widehat{I}(F;C)$ instead of $I(F;C)$, it is possible to locate a subset of informative features by implementing a greedy "pick-one-feature-at-a-time" selection procedure. Given $K$ features, out of which $m$ are to be selected ($m < K$), this involves two steps: 1) select the first feature $f'_{max}$ that maximizes $I(f';C)$, and 2) select $m-1$ subsequent features that maximize the criterion in Equation (8), i.e., select the second feature $f''_{max}$ that maximizes $I(f'';C) - I(f'';f'_{max}) + I(f'';f'_{max}|C)$, select the third feature $f'''_{max}$ that maximizes $I(f''';C) - I(f''';f'_{max}) - I(f''';f''_{max}) + I(f''';f'_{max}|C) + I(f''';f''_{max}|C)$ and so on.

Although an MII based on the second-order feature dependence assumption can select features that maximize class-separability and simultaneously minimize dependencies between feature pairs, there is no reason to assume that the final optimal feature subset is formed by pairwise interactions between features. In fact, it neglects the fact that third or higher order dependencies can be lead to an optimal feature subset.

The primary reason for using the approximation $\widehat{I}(F;C)$ for feature selection instead of directly using multidimensional interaction information $I(F;C)$ is that $I(F;C)$ requires estimation of the joint probability distribution of features using a large training sample. Consider the joint distribution $P(F) = P(f_1,\ldots,f_m)$, by the chain rule of probability

$$P(f_i,\ldots,f_m) = P(f_1)P(f_2|f_1) \times P(f_3|f_2,f_1)\cdots P(f_m|f_1,f_2\ldots f_{m-1}) \ , \quad (14)$$

$$P(F;C) = P(f_1,\ldots f_m;C) = P(C)p(f_1|C)P(f_2|f_1,C)P(f_3|f_1,f_2,C) \times P(f_4|f_1,f_2,f_3,C)\cdots P(f_i|f_1,\ldots,f_m,C) \ . \quad (15)$$

In our feature selection scheme, the original features are clustered into different dominant-sets based on dominant-set clustering and each dominant-set just includes a small set of features. Therefore, for each dominant set, we do not need to use the approximation $\widehat{I}(F;C)$. Instead, we can directly use the multidimensional interaction information $I(F;C)$ criterion for feature selection. Using Parzen windows for probability distribution estimation, we then apply the greedy strategy to select the feature that maximizes the multidimensional mutual information between the features and the output class set. As a result the first feature $f'_{max}$ maximizes $I(f',C)$, the second selected feature $f''_{max}$ maximizes $I(f'',f',C)$, the third feature $f'''_{max}$ maximizes $I(f''',f'',f',C)$, and so on. For each dominant set, we repeat this procedure until $|S| = k$.

## 5 Classification

After finding the discriminating features, we apply the variational EM (VBEM) algorithm [2] to fit a mixture of Gaussians model to the selected feature subset. After learning the mixture model, we use the a posteriori probability, see Equation(16), to classify sample. Given a sample, we first compute its selected feature vector $b$ through feature selection. Then we compute its a posteriori probabilities $r_c$, the mean vectors $\hat{b}_c$, and the precision matrices $\Lambda_c$, where $c \in c_1,\ldots,c_l$ and

$l$ is the number of class for the data. For example, in binary class, if $r_{c_1} > r_{c_2}$ then the sample is classified as class $c_1$. Otherwise, the sample is classified as $c_2$. The posterior probabilities are given by

$$r_{nk} \propto \pi_k |\Lambda_k|^{\frac{-1}{2}} \exp\{\frac{1}{2}(x_n - \mu_k)^T \Lambda_k (x_n - \mu_k)\} \ . \tag{16}$$

where $k = 1, \ldots, K$ is the mixture component, $n = 1, \ldots, N$ denotes the data index. Model parameters $\pi_k$, $\mu_k$ and $\Lambda_k$ are respectively a priori probability, the mean of selected feature vectors and precision matrices of the $k^{th}$ component. In the variational Bayesian EM (VBEM) algorithm, all of these model parameters are characterized by hyper-parameters, which take into account the uncertainty in the parameter estimation. The parameters $r_{nk}$ are called posteriori probability because they represent the responsibility the $k^{th}$ component takes in explaining the $n^{th}$ observation. The posteriori probability can be arranged into a matrix $R = (r_{nk})$ and will have to satisfy the following conditions:

$$0 \leq r_{nk} \leq 1 \ . \tag{17}$$

## 6 Experiments and Comparisons

The data sets used to test the performance of our proposed algorithm are the benchmark data sets from the NIPS 2003 feature selection challenge and the UCI Machine Learning Repository. Table. 1 summarizes the properties of these data-sets. Our proposed feature selection method (referred to as the DS*plus*MII method) (which utilizes the multidimensional interaction information (MII) criterion and dominant-set clustering for feature selection) involves grouping a set of informative features into cluster from the original feature set by dominant-set clustering and then applying MII criterion into the cluster for feature selection. In order to examine the performance of our proposed method DS*plus*MII, we need to know how meaningful the cluster obtained based on mutual information is and what more useful information they contain. In view of this, we should first examine how discriminative the features in the leading dominant set. Next, we could use the extracted features for classification to check the performance. Our proposed scheme for evaluation and comparison can be outlined as follows: a) the study of the cluster performance obtained by different similarity measure methods(i.e., the Pearson's correlation coefficient ($\rho$) and Least square regression error ($e$) ). b) the study of classification results based on the selected feature subset captured by MII in the dominant sets and compared with other MI-based criterion methods(i.e., the MRMR algorithm [11] and the MIFS algorithm [1]).

### 6.1 Cluster Performance Evaluation using Different Similarity Measures

As we mentioned before, our proposed algorithm is capable of grouping informative features in the leading dominant set by dominant set clustering based on

**Table 1.** Summary of UCI and NIPS benchmark data sets

| Data-set | Examples | Features | Classes |
|---|---|---|---|
| Madelon | 2000 | 500 | 2 |
| Breast cancer | 699 | 10 | 2 |
| Pima | 768 | 8 | 2 |
| Australian | 690 | 14 | 2 |

**Table 2.** J value comparisons of dominant set using different feature similarity measure

| Data-set | Similarity Measure:MI | Similarity Measure: ($\rho$) | Similarity Measure: ($e$) |
|---|---|---|---|
| Madelon | 1.1082 | 1.0024 | 1.0094 |
| Breast cancer | 5.1513 | 5.1513 | 5.1513 |
| Pima | 1.3716 | 1.3716 | 1.0177 |
| Australian | 2.2546 | 2.2006 | 1.2090 |

a suitable similarity measure. Different similarity measures will lead to different clustering results, which means that an unsuitable similarity measure may group less informative features into a cluster. Therefore, we should carefully select which similarity measure to use. Here, we study the clustering results obtained by using three different similarity measures for dominant-set clustering(DS). In order to examine the discriminability of the features grouped in the leading dominant set, we will use the multidimensional interaction information (MII) criterion. Then, a criterion function is used to measure the discrimination of the selected key features. This is a well known measure of class separability introduced by Devijiver and Kittler [4], and given by

$$J(Y) = \frac{|S_w + S_b|}{|S_w|} = \prod_{k=1}^{d} (1 + \lambda_k) \ . \tag{18}$$

where $Y$ denotes the feature set, $\lambda_k$, $k = 1 \ldots d$, are the eigenvalues of matrix $S_w^{-1} S_b$, and $S_w$ and $S_b$ are the between and within class scatter matrices. Table. 2 shows the comparative cluster results of our mutual information based similarity measure with other two similarity measures in terms of the measured $J$ value. The subset obtained by our mutual information based similarity measure is more discriminative, giving the highest $J$ value.

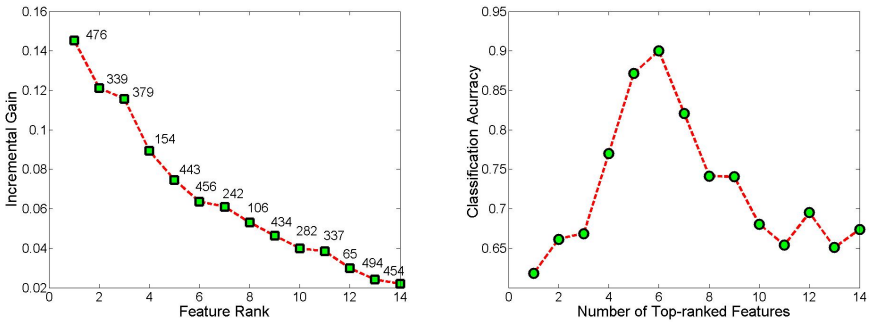
### 6.2 Classification Results using Selected Feature Subset

After obtaining the discriminating features, we apply a variational Bayesian EM(VBEM) algorithm to learn a Gaussian mixture model on the selected feature subset for the purpose of classification. We compare classification results from our proposed feature selection method (referred to as the DS$plus$MII method) (which utilizes the multidimensional interaction information (MII) criterion and dominant-sets for feature selection) with those obtained using k-means algorithm

[9] and alternative existing MI-based criterion methods, namely a) Maximum-Relevance Minimum-Redundancy (MRMR), b) Mutual Information Based Feature Selection (MIFS).

Based on the feature subsets selected by our proposed DS*plus*MII method, We first examine the classification performance using different sized feature subsets by selecting the top $k$ features ranked by their incremental gain. In the classification performance evaluation process, we employ a posteriori probability, see Equation(16), to perform classification, we got the classification accuracy by the percentage of the data, which are predicted correctly. For the purpose of comparison, we repeated the feature selection process using the k-means algorithm, MRMR algorithm and MIFS algorithm.

The Madelon data set is a 2 classes problem originally proposed in the NIPS'2003 feature selection challenge [6]. The data points grouped into 32 clusters placed on the vertices of a five dimensional hypercubes. As a result, there are only 5 informative features, but 15 redundant features and 480 probes. In Fig. 2, we present the top 14 features ranked by the incremental gain calculated by MII. The classification accuracies obtained on different feature subsets are shown in the right hand side of Fig. 2. From the figure, it is clear that using the leading 6 features (476, 339, 379, 154, 443, 456), we achieve 90% classification accuracy. Because of the unsupervised nature of the VBEM algorithm and the gaussian mixture model, the classification accuracy of 90% demonstrates the adequate separability provided by the selected feature subset. For comparison, we also visualize the classification results of using the feature subset obtained by MRMR;

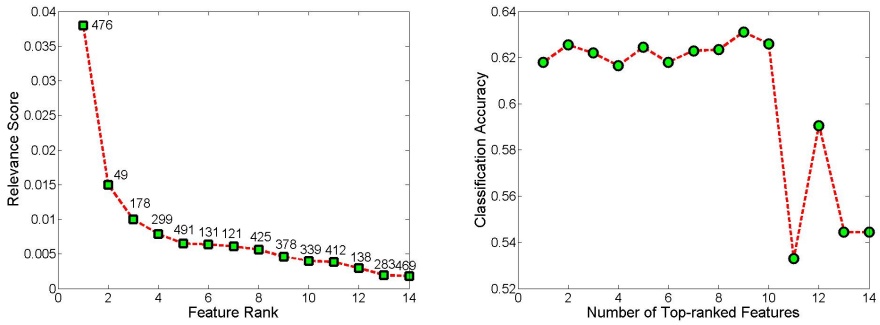



(a) Top-ranked Features by Incremental Gain

(b) Classification result

**Fig. 2.** The result on Madelon data set for our algorithm. The values of the Incremental gain for the top 14 features are presented in the left part along with the feature indices, while the classification accuracies are plotted in the right part

In Fig. 3, the top-ranked features ranked by MRMR are presented in the left hand part, and the classification accuracies using the top-ranked features




(a) Top-ranked Features by Relevance Score

(b) Classification result

**Fig. 3.** The result on Madelon data set using MRMR for feature ranking. The values of the relevance score for the top 14 features are presented in the left part along with the feature indices, while the classification accuracies are plotted in the right part

incrementally are presented in the right hand part. The best result is about 63.1% using 9 features, which is much worse than the result of our algorithm as shown in Fig. 2. The poor classification performance may be explained by our observation that most of the selected top features are not in the 1st dominant set and ranked very low by DS*plus*MII. On the other hand, we find that for MRMR there is a tendency to overestimate the redundancy between features, since they neglect the conditional redundancy term $I(x_i, S|C)$. As a result some important features can be discarded, which in turn leads to information loss.

**Table 3.** The classification accuracy on the top features selected by different methods in the Breast Cancer data set

| No.of Features Selected | DS*plus*MII | MRMR | MIFS |
|---|---|---|---|
| 2 | 88.84% | 88.84% | 88.84% |
| 3 | 96.3% | 87.98% | 84.4% |
| 4 | 96.3% | 87.55% | 82.51% |

**Table 4.** The classification accuracy on the top features selected by different methods in the Pima data set

| No.of Features Selected | DS*plus*MII | MRMR | MIFS |
|---|---|---|---|
| 2 | 74.09% | 74.09% | 74.09% |
| 3 | 75.91% | 75.91% | 75.91% |
| 4 | 72.79% | 70.31% | 70.31% |

**Table 5.** The classification accuracy on the top features selected by different methods in the Australian data set

| No.of Features Selected | DS*plus*MII | MRMR | MIFS |
|---|---|---|---|
| 3 | 83.77% | 68.84% | 64.35% |
| 4 | 83.77% | 69.13% | 64.35% |
| 5 | 83.77% | 69.28% | 83.62% |

The experimental results in Table. 3, 4 and 5 show that DS*plus*MII is, by and large, superior to the other feature clustering and feature selection methods by selecting a smaller set of discriminative features than the others as reflected by the classification results. As shown by the results, DS*plus*MII outperforms MIFS and MRMR algorithms in all cases except in the Pima dataset, in which all the four methods yield a comparable classification rate. It is interesting to note that the performance achieves a 96.3% when using the 3 features selected by DS*plus*MII and maintain at the same accuracy even when more features are selected(see Table. 3). Similaly, 83.77% is achieved when 3 features are selected by DS*plus*MII and its performance remains at this level even when more features are selected(see Table. 5). This implies that the discriminative information exists in a small set of features which can be used to fit the mixture Gaussian models to the data. In addition, in breast cancer, we find out that the leading 4 selected features are all from the first dominant set found by dominant set clustering. This again supports the fact that the first dominant set captures the greatest number of informative features. From Table. 4, it is clear that using the leading three features, then all the four methods achieve 75.91% classification accuracy, which is higher than that obtained using other sized feature subsets. Using fewer or more features both deteriorate the accuracy. This implies that classification of samples is based on a very few of the most important features.

## 7 Conclusions

This paper has presented a new graph theoretic approach to feature selection. The proposed feature selection method offers two major advantages. First, dominant-set clustering can capture the most informative features based on MI-based similarity measure. Second, the MII criteria takes into account high-order feature interactions, overcoming the problem of overestimated redundancy. As a result the features associated with the greatest amount of joint information can be preserved.

## References

1. Battiti, R.: Using Mutual Information for Selecting Features in Supervised Neural Net Learning. IEEE Transactions on Neural Networks 5(4), 537–550 (2002)
2. Bishop, C.: Pattern Recognition and Machine Learning, vol. 4. Springer New York (2006)

3. Cheng, H., Qin, Z., Qian, W., Liu, W.: Conditional Mutual Information Based Feature Selection. In: IEEE International Symposium on Knowledge Acquisition and Modeling. pp. 103–107 (2008)
4. Devijver, P., Kittler, J.: Pattern Recognition: A Statistical Approach, vol. 761. Prentice-Hall London (1982)
5. Guo, B., Nixon, M.: Gait Feature Subset Selection by Mutual Information. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 39(1), 36–46 (2008)
6. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.: Feature extraction, foundations and applications (2006)
7. Jiang, D., Tang, C., Zhang, A.: Cluster analysis for gene expression data: a survey. IEEE Transactions on Knowledge and Data Engineering 16(11), 1370–1386 (2004)
8. Kwak, N., Choi, C.: Input Feature Selection by Mutual Information Based on Parzen Window. IEEE TPAMI 24(12), 1667–1671 (2002)
9. MacQueen, J., et al.: Some Methods for Classification and Analysis of Multivariate Observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. vol. 1, pp. 281–297. California, USA (1967)
10. Pavan, M., Pelillo, M.: A New Graph-Theoretic Approach to Clustering and Segmentation. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. vol. 1 (2003)
11. Peng, H., Long, F., Ding, C.: Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1226–1238 (2005)
12. Rao, C.: {Linear statistical Inference and Its Applications} (1965)
13. Shannon, C.: A Mathematical Theory of Communication. ACM SIGMOBILE Mobile Computing and Communications Review 5(1), 3–55 (2001)
14. Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E., Golub, T.: Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. Proceedings of the National Academy of Sciences of the United States of America 96(6), 2907 (1999)
15. Yang, H., Moody, J.: Feature Selection Based on Joint Mutual Information. In: Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis. pp. 22–25 (1999)
16. Yu, J., Tian, Q., Amores, J., Sebe, N.: Toward robust distance metric analysis for similarity estimation. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. vol. 1, pp. 316–322. IEEE (2006)
17. Zhang, F., Zhao, Y., Fen, J.: Unsupervised Feature Selection based on Feature Relevance. In: International Conference on Machine Learning and Cybernetics. vol. 1, pp. 487–492 (2009)
18. Zhang, Z., Hancock, E.: A graph-based approach to feature selection. Graph-Based Representations in Pattern Recognition pp. 205–214 (2011)