Game-Theoretic Models of Pattern Recognition

Marcello Pelillo

Ca' Foscari University, Venice

pelillo@dsi.unive.it

Why Should We Care About Game Theory?

Answer #1:

Because it works! (well, great…)

Answer #2:

Because it allows us to naturally deal with non-Euclidean, non-metric, negative, high-order (dis)similarities

Answer #3:

Because it allows us to model in a principled way problems not formulable in terms of "simple" optimization principles

Dealing with Non-Euclidean/Non-Metric Similarities

By departing from vector-space representations one is confronted with the challenging problem of dealing with (dis)similarities that do not necessarily possess the Euclidean behavior or not even obey the requirements of a metric.

The lack of the Euclidean and/or metric properties undermines the very foundations of traditional pattern recognition theories and algorithms!

The customary approach to deal with non-(geo)metric (dis)similarities is **embedding**, which refers to any procedure that takes a set of (dis)similarities as input and produces a vectorial representation of the data as output, such that the proximities are either locally or globally preserved.

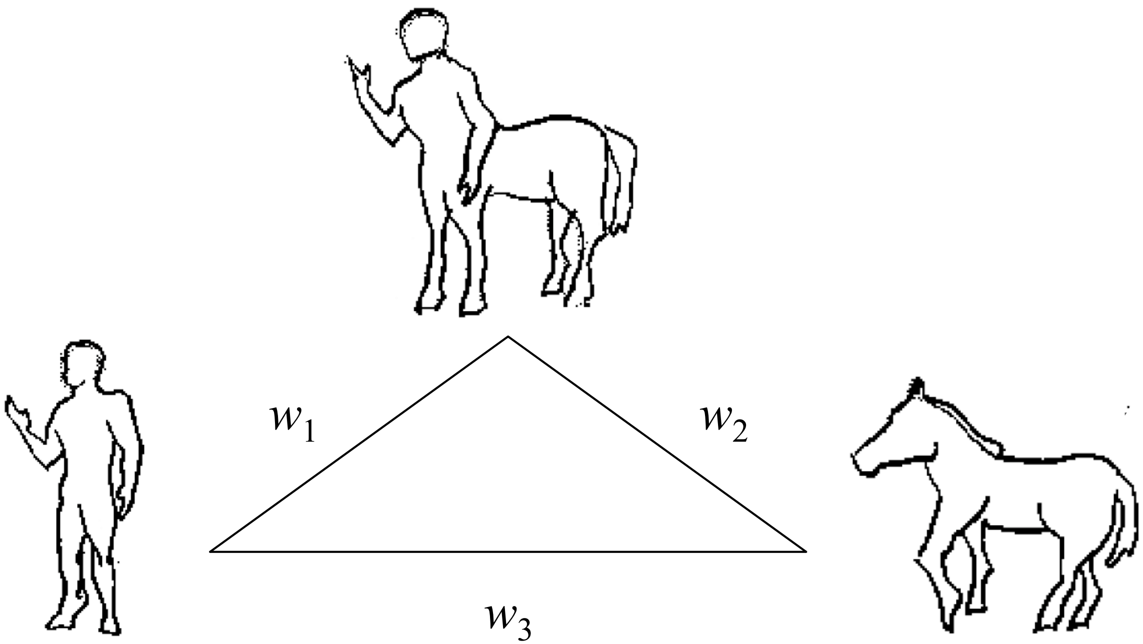
Embedding is based on the assumption that the non-(geo)metricity of similarity information can be eliminated or somehow approximated away.

When this is not the case, i.e., when there is significant information content in the non-geometricity of the data, alternative approaches are needed.

The Need for Non-Metric Similarities

«Any computer vision system that attempts to faithfully reflect human judgments of similarity is apt to devise non-metric image distance functions.»

Jacobs, Weinshall and Gdalyahu, 2000



$$w_3 > w_1 + w_2$$

Adapted from: D.W. Jacobs, D. Weinshall, and Y. Gdalyahu. Classication with non-metric distances: Image retrieval and class representation. *PAMI* 2000.

The Symmetry Assumption

«Similarity has been viewed by both philosophers and psychologists as a prime example of a symmetric relation. Indeed, the assumption of symmetry underlies essentially all theoretical treatments of similarity.

Contrary to this tradition, the present paper provides empirical evidence for asymmetric similarities and argues that **similarity should not be treated as a symmetric relation**.»

Amos Tversky

"Features of similarities," *Psychol. Rev.* (1977)

Examples of asymmetric (dis)similarities

- ✓ Kullback-Leibler divergence
- ✓ Directed Hausdorff distance
- ✓ Tversky's contrast model

The Optimization "Principle"

A classical strategy to attack pattern recognition and machine learning problems consists of formulating them in terms of optimization problems.

The main merit of optimization-based techniques is their ability of providing (by definition) a quantitative measure of a solution's quality.

In some real-world situations, however, the complexity of the problem at hand is such that no single objective function would satisfactorily capture its intricacies.

Examples include:

- ✓ Using asymmetric compatibilities in (continuous) consistency labeling problems (Hummel & Zucker, 1983)
- ✓ Integrating region- and gradient-based methods in image segmentation (Chakraborty & Duncan, 1991)
- ✓ Clustering with asymmetric affinities (Yu & Shi, 2001; Torsello, Rota Bulò & Pelillo, 2006)

Game Theory:
Beyond Optimization

Game theory was developed precisely to overcome the limitations of single-objective optimization (J. von Neumann, J. Nash, etc.).

It aims at modeling complex situations where "players" make decisions in an attempt to maximize their own (mutually conflicting) returns.

According to this perspective, the focus will shift **from optima** of objective functions **to equilibria** of (non-cooperative) "games."

Nowadays, game theory is a well-established field on its own and offers a rich arsenal of powerful concepts and algorithms.

Note: in the case of a particular class of games (i.e., doubly-symmetric games) game-theoretic criteria reduce to optimality criteria.

Outline

Part 1:

Introduction to the basic concepts of game theory

Part 2:

Evolutionary games for data clustering

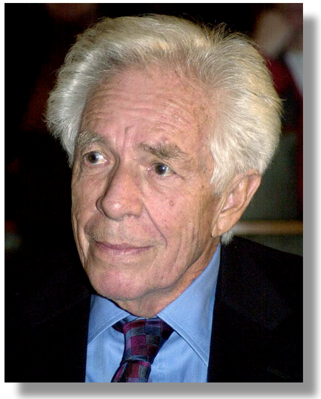
Part 3:

Contextual pattern recognition and graph transduction

Part I

Introduction to Game Theory

What is Game Theory?



"The central problem of game theory was posed by von Neumann as early as 1926 in Göttingen. It is the following: If n players, $P_1, \ldots, P_n$, play a given game Γ, how must the i^{th} player, P_i, play to achieve the most favorable result for himself?"

Harold W. Kuhn
Lectures on the Theory of Games (1953)

A few cornerstones in game theory

1921–1928: Emile Borel and John von Neumann give the first modern formulation of a mixed strategy along with the idea of finding minimax solutions of normal-form games.

1944, 1947: John von Neumann and Oskar Morgenstern publish *Theory of Games and Economic Behavior*.

1950–1953: In four papers John Nash made seminal contributions to both non-cooperative game theory and to bargaining theory.

1972–1982: John Maynard Smith applies game theory to biological problems thereby founding "evolutionary game theory."

late 1990's –: Development of algorithmic game theory…

Normal-form Games

We shall focus on finite, non-cooperative, simultaneous-move games in **normal form**, which are characterized by:

- ✓ A set of **players**: $I = \{1, 2, \ldots, n\}$ ($n \geq 2$)
- ✓ A set of **pure strategy profiles**: $S = S_1 \times S_2 \times \ldots \times S_n$ where each $S_i = \{1, 2, \ldots, m_i\}$ is the (finite) set of "pure" strategies (actions) available to the player i
- ✓ A **payoff function**: $\pi : S \rightarrow \Re^n$, $\pi(s) = (\pi_1(s),\ldots,\pi_n(s))$, where $\pi_i(s)$ (i=1…n) represents the "payoff" (or utility) that player i receives when strategy profile s is played

Each player is to choose one element from his strategy space in the absence of knowledge of the choices of the other players, and "payments" will be made to them according to the function $\pi_i(s)$.

Players' goal is to maximize their own returns.

Two Players

In the case of two players, payoffs can be represented as two m_1 x m_2 matrices (say, A for player 1 and B *for player 2*):

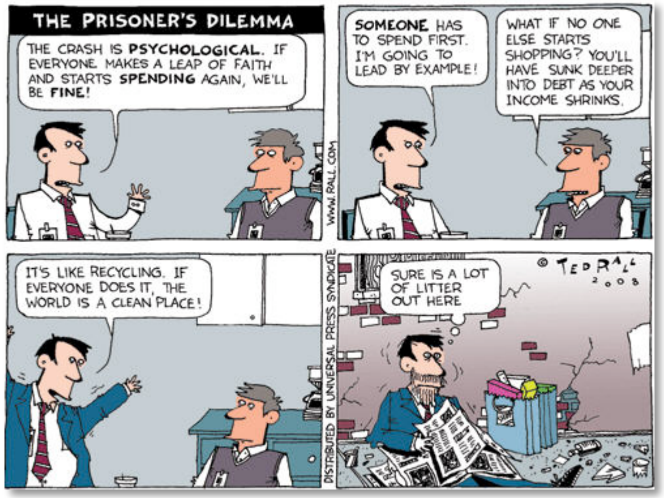
$$A = (a_{hk}) \qquad a_{hk} = \pi_1(h,k)$$

$$B = (b_{hk}) \qquad b_{hk} = \pi_2(h,k)$$

Special cases:

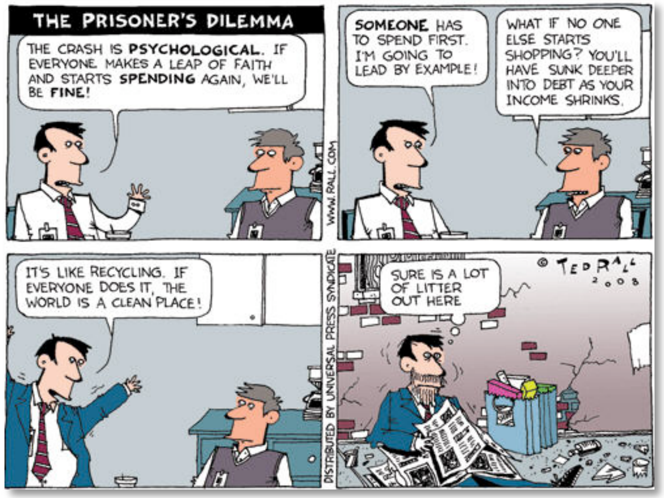
- ✓ Zero-sum games: $A + B = 0$ ($a_{hk} = -b_{hk}$ for all h and k)
- ✓ Symmetric games: $B^T = A$
- ✓ Doubly-symmetric games: $A = A^T = B^T$

An Example: Prisoner's Dilemma



		Prisoner 2	
		Confess (defect)	Deny (cooperate)
Prisoner 1	Confess (defect)	-10 , -10	-1 , -25
	Deny (cooperate)	-25 , -1	-3 , -3

How to "Solve" the Game?



Dominated strategy !

		Prisoner 2	
		Confess (defect)	Deny (cooperate)
Prisoner 1	Confess (defect)	-10 , -10	-1 , -25
	Deny (cooperate)	-25 , -1	-3 , -3

Dominated strategy !

Mixed Strategies

A **mixed strategy** for player i is a probability distribution over his set S_i of pure strategies, which is a point in the $(m_i\text{-}1)$-dimensional **standard simplex**:

$$\Delta_i = \left\{ x_i \in R^{m_i} \ : \ \forall h = 1 \ldots m_i : x_{ih} \geq 0, \text{ and } \sum_{h=1}^{m_i} x_{ih} = 1 \right\}$$

The set of pure strategies that is assigned positive probability by mixed strategy $x_i \in \Delta_i$ is called the **support** of x_i:

$$\sigma(x_i) = \left\{ h \in S_i : x_{ih} > 0 \right\}$$

A **mixed strategy profile** is a vector $x = (x_1, \ldots, x_n)$ where each component $x_i \in \Delta_i$ is a mixed strategy for player $i \in I$.

The **mixed strategy space** is the multi-simplex $\Theta = \Delta_1 \times \Delta_2 \times \ldots \times \Delta_n$

Mixed-Strategy Payoff Functions

In the standard approach, all players' randomizations are assumed to be independent.

Hence, the probability that a pure strategy profile $s = (s_1,\ldots,s_n)$ will be used when a mixed-strategy profile x is played is:

$$x(s) = \prod_{i=1}^{n} x_{is_i}$$

and the expected value of the payoff to player i is:

$$u_i(x) = \sum_{s \in S} x(s)\pi_i(s)$$

In the special case of two-players games, one gets:

$$u_1(x) = \sum_{h=1}^{m_1}\sum_{k=1}^{m_2} x_{1h}a_{hk}x_{2k} = x_1^T A x_2 \qquad u_2(x) = \sum_{h=1}^{m_1}\sum_{k=1}^{m_2} x_{1h}b_{hk}x_{2k} = x_1^T B x_2$$

where A and B are the payoff matrices of players 1 and 2, respectively.

Nash Equilibria

The Nash equilibrium concept is motivated by the idea that a theory of rational decision-making should not be a self-destroying prophecy that creates an incentive to deviate for those who believe it.

A strategy profile $x=(x_1,\ldots,x_n)\in\Theta$ is a **Nash equilibrium** if it is a best reply to itself, namely, if:

$$u_i(x_1\ldots x_i\ldots x_n) \geq u_i(x_1\ldots z_i\ldots x_n)$$

for all $i = 1\ldots n$ and all strategies $z_i \in \Delta_i$.

If strict inequalities hold for all $z_i \neq x_i$ then x is said to be a **strict Nash equilibrium**.

Theorem (Nash, 1951). Every finite normal-form game admits a mixed-strategy Nash equilibrium.

Idea of proof.

1. Define a continuous map T on Θ such that the fixed points of T are in one-to-one correspondence with Nash equilibria.
2. Use Brouwer's theorem to prove existence of a fixed point.

Finding Pure-strategy Nash Equilibria

		Player 2		
		Left	Middle	Right
Player 1	Top	3 , 1	2 , 3	10 , 2
	High	4 , 5	3 , 0	[illegible] , 4
	Low	2 , 2	5 , 4	12 , 3
	Bottom	5 , 6	4 , 5	9 , 7

Nash equilibrium!

The Complexity of Finding Nash Equilibria

"Together with factoring, *the complexity of finding a Nash equilibrium is in my opinion the most important concrete open question on the boundary of P today."*

Christos Papadimitriou

Algorithms, games, and the internet (2001)

At present, no known reduction exists from our problem to a decision problem that is *NP*-complete, nor has our problem been shown to be easier.

An intuitive stumbling block is that *every* game has at least one Nash equilibrium, whereas known *NP*-complete problems are expressible in terms of decision problems that do not always have solutions.

However, we know that NASH is "PPAD-complete" even for two players.

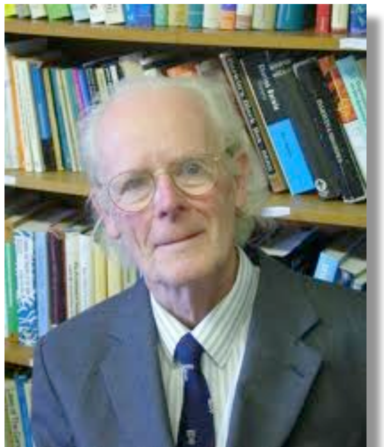
Evolution and the Theory of Games

"We repeat most emphatically that our theory is thoroughly static. A dynamic theory would unquestionably be more complete and therefore preferable.

But there is ample evidence from other branches of science that it is futile to try to build one as long as the static side is not thoroughly understood."

John von Neumann and Oskar Morgenstern

Theory of Games and Economic Behavior (1944)



"Paradoxically, it has turned out that game theory is more readily applied to biology than to the field of economic behaviour for which it was originally designed."

John Maynard Smith

Evolution and the Theory of Games (1982)

Evolutionary Games

Introduced by John Maynard Smith (1973, 1974, 1982) to model the evolution of behavior in animal conflicts.

Assumptions:

- ✓ A large population of individuals belonging to the same species which compete for a particular limited resource
- ✓ This kind of conflict is modeled as a symmetric two-player game, the players being pairs of randomly selected population members
- ✓ Players do not behave “rationally” but act according to a pre-programmed behavioral pattern
- ✓ Reproduction is assumed to be asexual
- ✓ Utility is measured in terms of Darwinian fitness, or reproductive success

Evolutionary Stability

A strategy is **evolutionary stable** if it is resistant to invasion by new strategies.

Formally, assume:

- ✓ A small group of “invaders” appears in a large populations of individuals, all of whom are pre-programmed to play strategy $x \in \Delta$
- ✓ Let $y \in \Delta$ be the strategy played by the invaders
- ✓ Let ε be the share of invaders in the (post-entry) population ($0 < \varepsilon < 1$)

The payoff in a match in this bimorphic population is the same as in a match with an individual playing mixed strategy:

$$w = \varepsilon y + (1 - \varepsilon)x \in \Delta$$

hence, the (post-entry) payoffs got by the incumbent and the mutant strategies are $u(x,w)$ and $u(y,w)$, respectively.

Evolutionary Stable Strategies

Definition. A strategy $x \in \Delta$ is said to be an **evolutionary stable strategy** (ESS) if for all $y \in \Delta - \{x\}$ there exists $\delta \in (0,1)$, such that for all $\varepsilon \in (0, \delta)$ we have:

$$\underbrace{u[x, \varepsilon y + (1-\varepsilon)x]}_{\textit{incumbent}} > \underbrace{u[y, \varepsilon y + (1-\varepsilon)x]}_{\textit{mutant}}$$

Theorem. A strategy $x \in \Delta$ is an ESS if and only if it meets the following first- and second-order best-reply conditions:

1. $u(y,x) \leq u(x,x)$ for all $y \in \Delta$ (Nash condition)

2. $u(y,x) = u(x,x) \Rightarrow u(y,y) < u(x,y)$ for all $y \in \Delta - \{x\}$

Note 1. From the conditions above, we have:

✓ $\Delta^{ESS} \subseteq \Delta^{NE}$

✓ If $x \in \Delta$ is a strict Nash equilibrium, then x is an ESS

Note 2. Unlike Nash equilibria existence of ESS's is not guaranteed.

Complexity Issues

Two questions of computational complexity naturally present themselves:

- ✓ What is the complexity of determining whether a given game has an ESS (and of finding one)?
- ✓ What is the complexity of recognizing whether a given x is an ESS for a given game?

Theorem (Etessami and Lochbihler, 2004). Determining whether a given two-player symmetric game has an ESS is both NP-hard and coNP-hard.

Theorem (Nisan, 2006). Determining whether a (mixed) strategy x is an ESS of a given two-player symmetric game is coNP-hard.

Replicator Dynamics

Let $x_i(t)$ the population share playing pure strategy i at *time* t. The **state** of the population at time t is: $x(t)= (x_1(t),\ldots,x_n(t))\in\Delta$.

Replicator dynamics (Taylor and Jonker, 1978) are motivated by Darwin's principle of natural selection:

$$\frac{\dot{x}_i}{x_i} \propto \text{payoff of pure strategy i} - \text{average population payoff}$$

which yields:

$$\begin{aligned} \dot{x}_i &= x_i\left[u(e^i,x) - u(x,x)\right] \\ &= x_i\left[(Ax)_i - x^T Ax\right] \end{aligned}$$

Theorem (Nachbar, 1990; Taylor and Jonker, 1978). A point $x\in\Delta$ is a Nash equilibrium if and only if x is the limit point of a replicator dynamics trajectory starting from the interior of Δ.

Furthermore, if $x\in\Delta$ is an ESS, then it is an asymptotically stable equilibrium point for the replicator dynamics.

Doubly Symmetric Games

In a doubly symmetric (or partnership) game, the payoff matrix A is symmetric ($A = A^T$).

Fundamental Theorem of Natural Selection (Losert and Akin, 1983).

For any doubly symmetric game, the average population payoff $f(x) = x^T A x$ is strictly increasing along any non-constant trajectory of replicator dynamics, namely, $d/dt f(x(t)) \geq 0$ for all $t \geq 0$, with equality if and only if $x(t)$ is a stationary point.

Characterization of ESS's (Hofbauer and Sigmund, 1988)

For any doubly simmetric game with payoff matrix A, the following statements are equivalent:

a) $x \in \Delta^{ESS}$

b) $x \in \Delta$ is a strict local maximizer of $f(x) = x^T A x$ over the standard simplex Δ

c) $x \in \Delta$ is asymptotically stable in the replicator dynamics

Discrete-time Replicator Dynamics

A well-known discretization of replicator dynamics, which assumes non-overlapping generations, is the following (assuming a non-negative A):

$$x_i(t+1) = x_i(t)\frac{A\big(x(t)\big)_i}{x(t)^T Ax(t)}$$

which inherits most of the dynamical properties of its continuous-time counterpart (*e.g.*, the fundamental theorem of natural selection).

MATLAB implementation

```
distance=inf;
while distance>epsilon
   old_x=x;
   x = x.*(A*x);
   x = x./sum(x);
   distance=pdist([x,old_x]');
end
```

References

Texts on (classical) game theory

J. von Neumann and O. Morgerstern. *Theory of Games and Economic Behavior*. Princeton University Press (1944, 1953).

D. Fudenberg and J. Tirole. *Game Theory*. MIT Press (1991).

M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press (1994).

Texts on evolutionary game theory

J. Weibull. *Evolutionary Game Theory*. MIT Press (1995).

J. Hofbauer and K. Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University Press (1998).

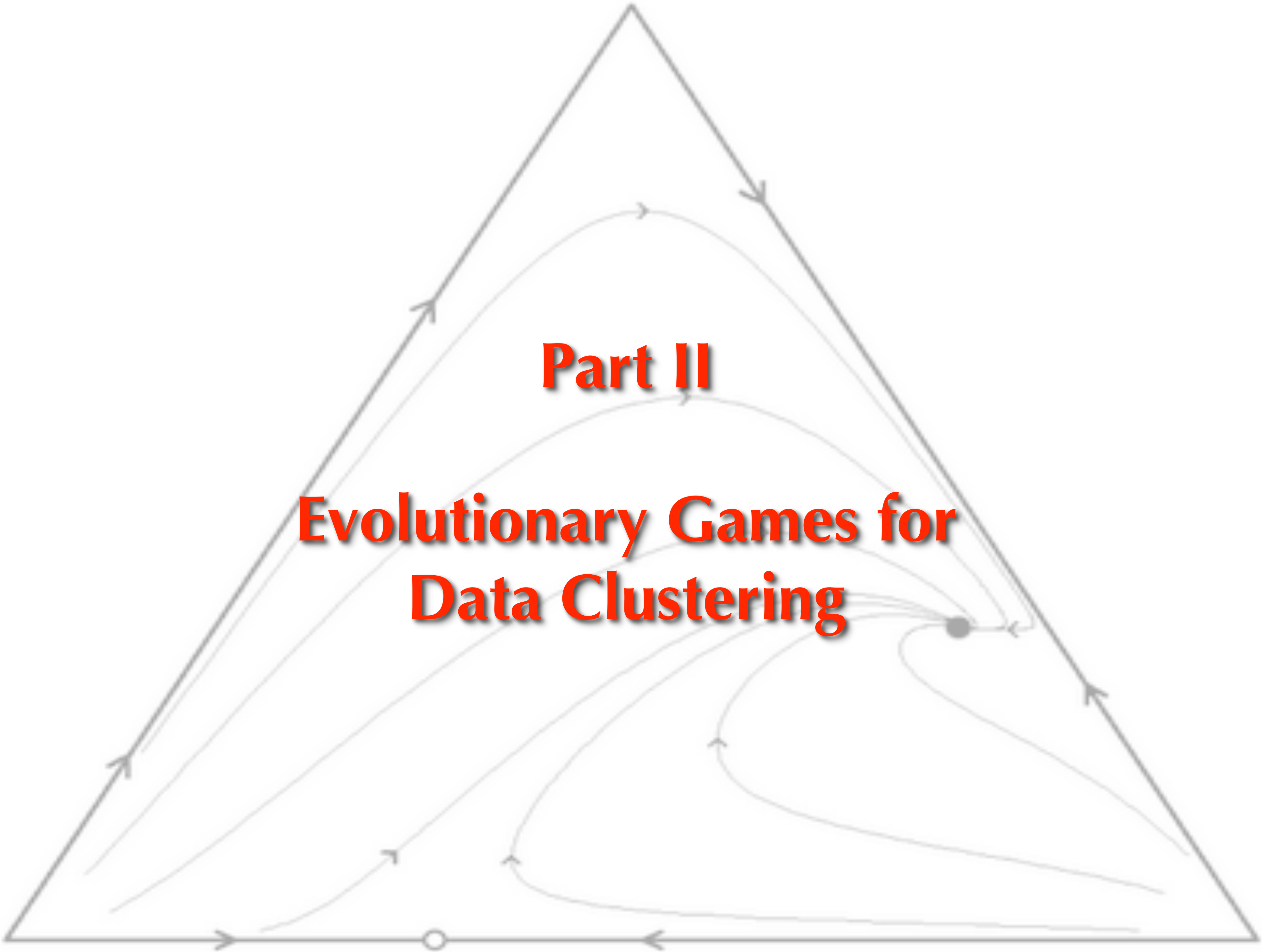
Computationally-oriented texts

N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani (Eds.) *Algorithmic Game Theory*. Cambridge University Press (2007).

Y. Shoham and K. Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press (2009).

On-line resources

http://gambit.sourceforge.net/	a library of game-theoretic algorithms
http://gamut.stanford.edu/	a suite of game generators for testing game algorithms



Part II

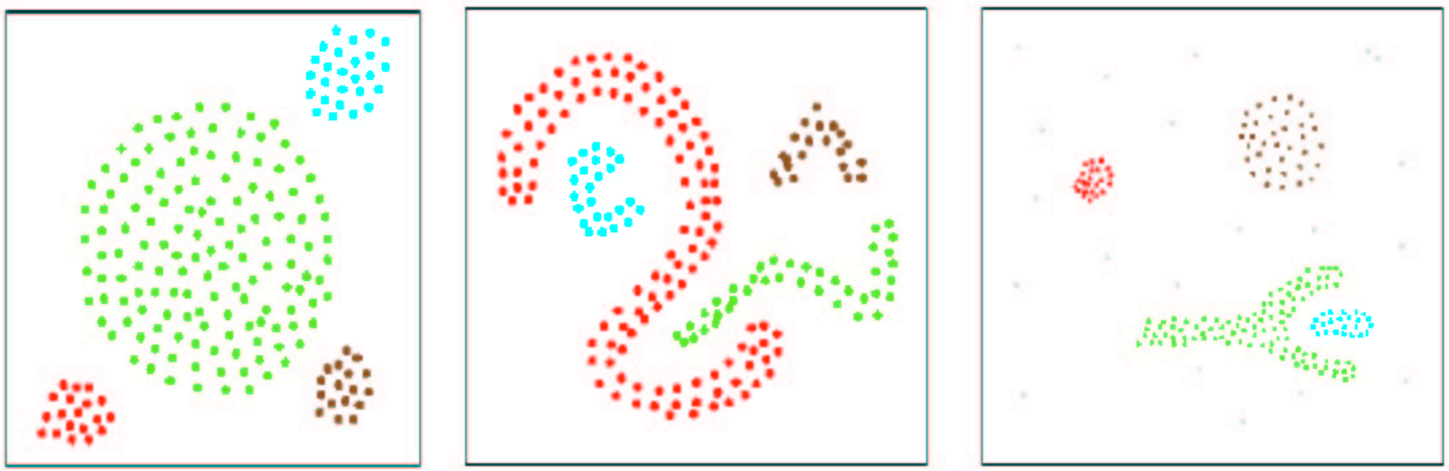
Evolutionary Games for Data Clustering

The "Classical" Clustering Problem

Given:

$$\left.\begin{array}{l}\text{- a set of } n \text{ "objects"} \\ \text{- an } n \times n \text{ matrix } A \text{ of pairwise similarities}\end{array}\right\} = \text{an edge-weighted graph}$$

Goal: <u>*Partition*</u> the input objects (the vertices of the graph) into maximally homogeneous groups (i.e., clusters).



Applications

Clustering problems abound in many areas of computer science and engineering.

A short list of applications domains:

Image processing and computer vision
Computational biology and bioinformatics
Information retrieval
Document analysis
Medical image analysis
Data mining
Signal processing
…

For a review see, e.g., A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters* 31(8):651-666, 2010.

The Need for Non-exhaustive Clusterings

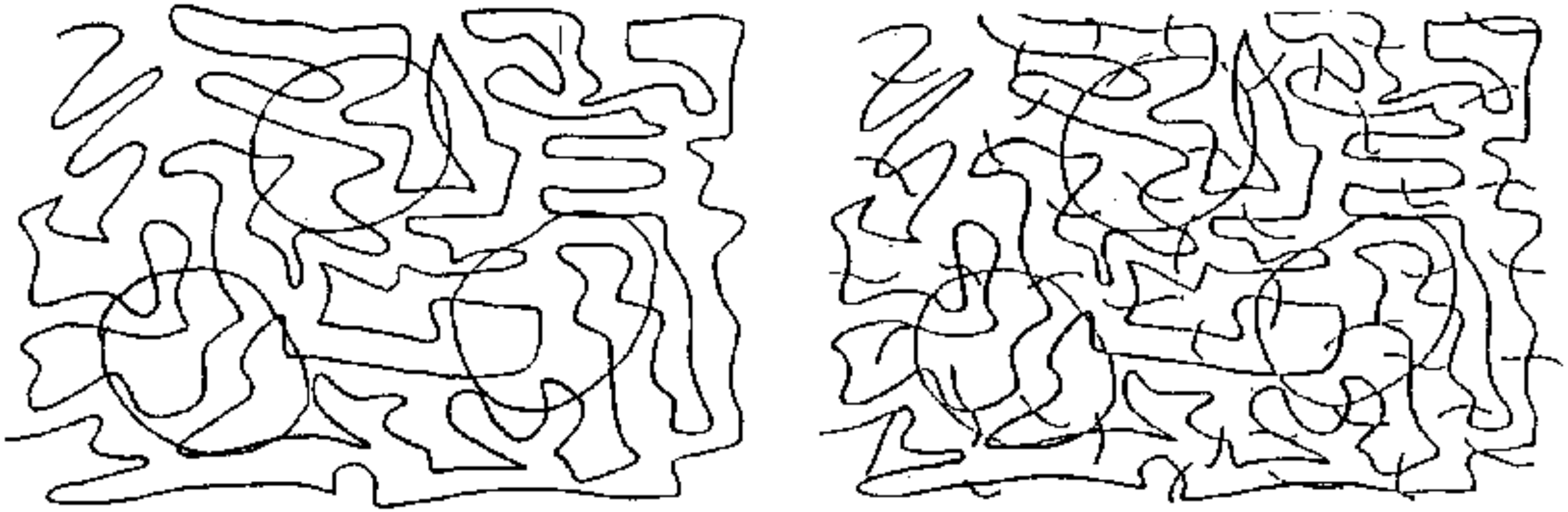


Figure 1a. Three prominent blobs are perceived immediately and with little effort. Locally, the blobs are similar to the background contours. (adopted from Mahoney (1986)

Figure 1b. Intersections were added to illustrate that the blobs are not distinguished by virtue of their intersections with the background curves.

Separating Structure from Clutter

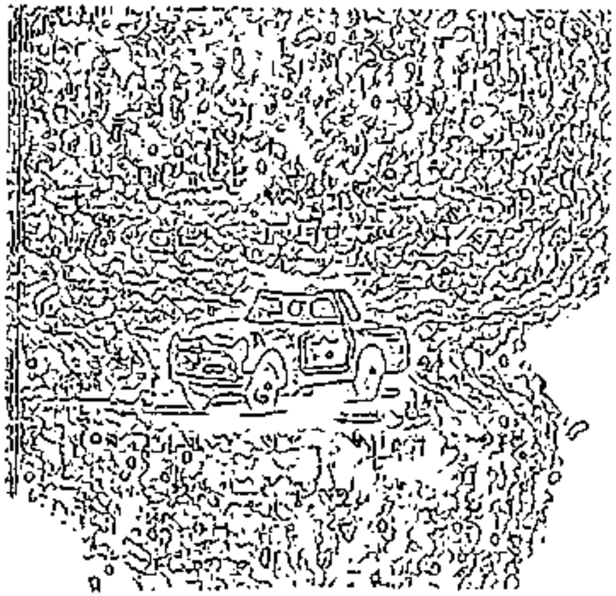
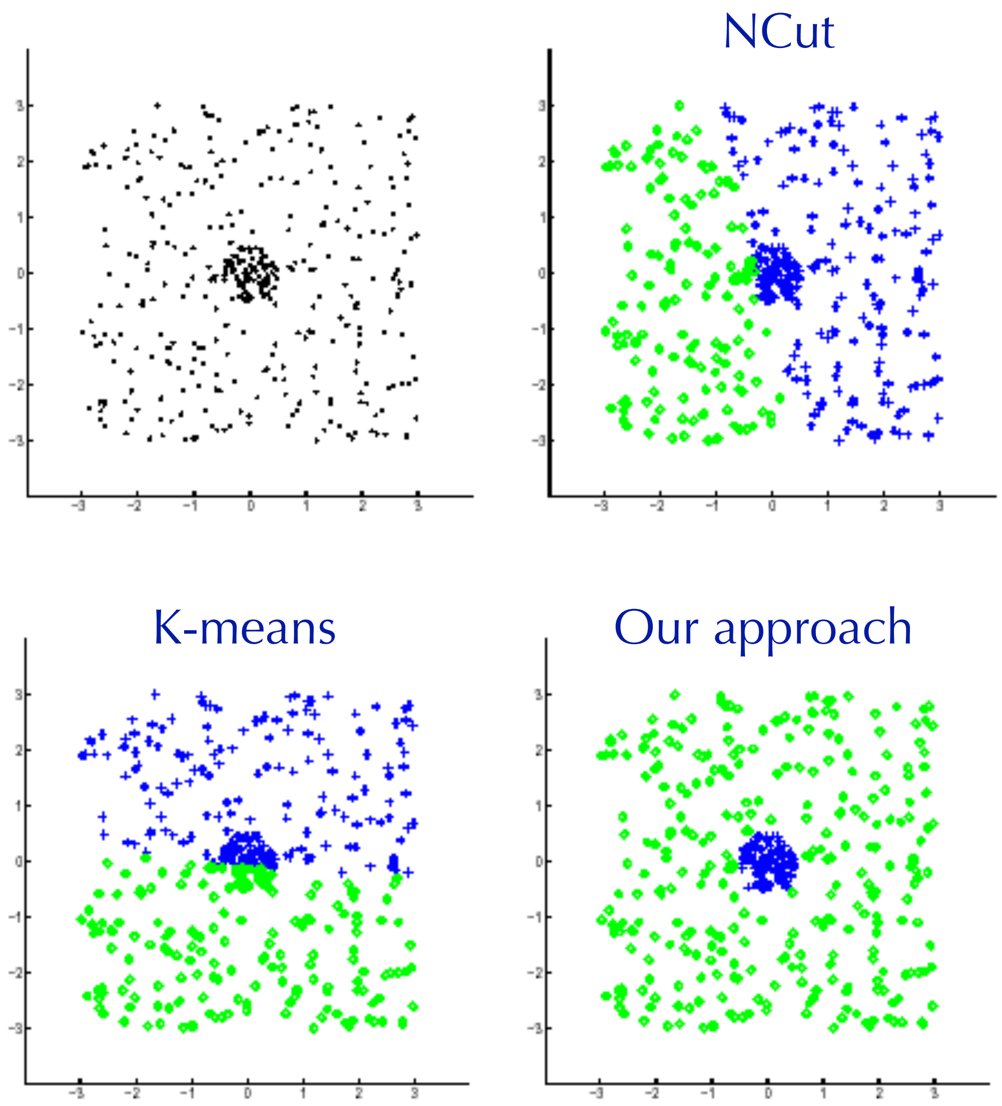


Figure 2. A circle in a background of 200 randomly placed and oriented segments The circle is still perceived immediately although its contour is fragmented.

Figure 3. An edge image of a car in a cluttered background. Our attention is drawn immediately to the region of interest. It seems that the car need not be recognized to attract our attention. The car also remains salient when parallel lines and small blobs are removed, and when the less textured region surrounding parts of the car is filled in with more texture.

Separating Structure from Clutter



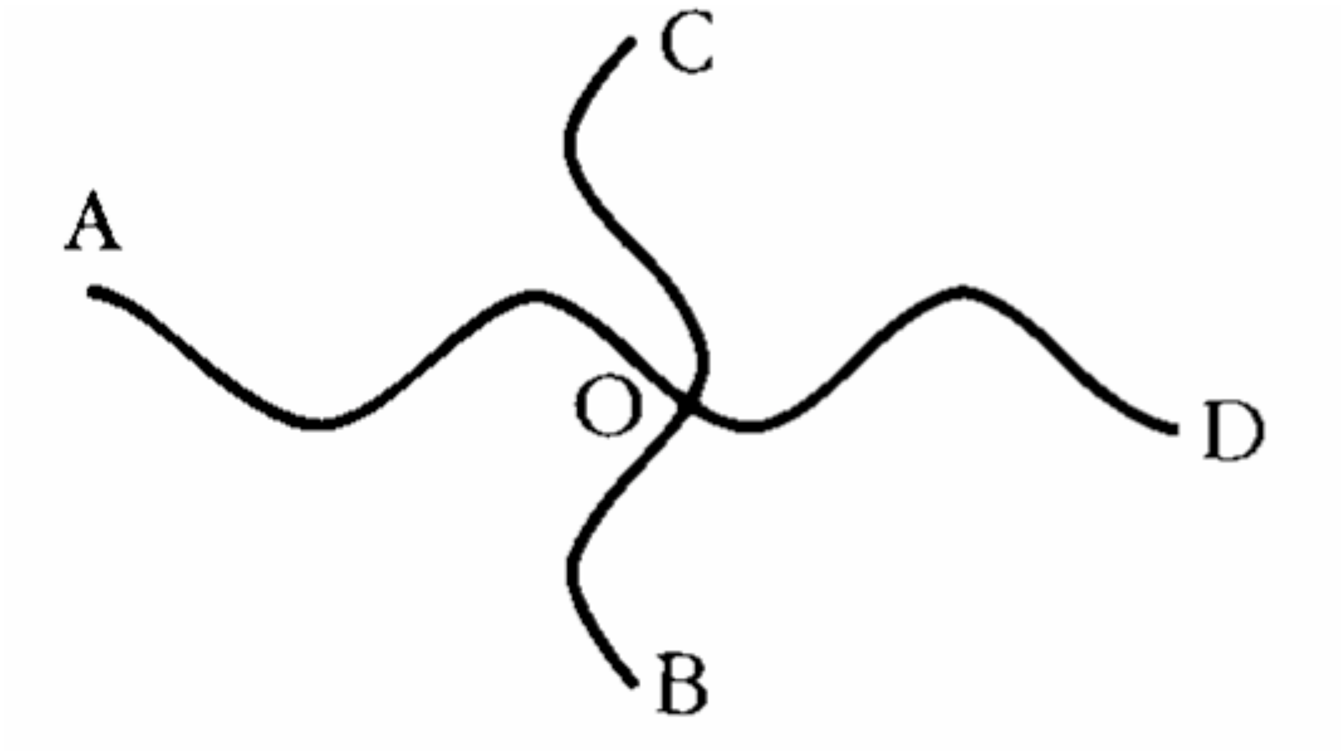
One-class Clustering

"[…] in certain real-world problems, natural groupings are found among only on a small subset of the data, while the rest of the data shows little or no clustering tendencies.

In such situations it is often more important to cluster a small subset of the data very well, rather than optimizing a clustering criterion over all the data points, particularly in application scenarios where a large amount of noisy data is encountered."

G. Gupta and J. Ghosh. Bregman bubble clustering: A robust framework for mining dense cluster. *ACM Trans. Knowl. Discov. Data* (2008).

When Groups Overlap



Does *O* belong to *AD* or to *BC* (or to none)?

The Need for Overlapping Clusters

Partitional approaches impose that each element cannot belong to more than one cluster. There are a variety of important applications, however, where this requirement is too restrictive.

Examples:

- ✓ clustering micro-array gene expression data
- ✓ clustering documents into topic categories
- ✓ perceptual grouping
- ✓ segmentation of images with transparent surfaces

References:

- ✓ N. Jardine and R. Sibson. The construction of hierarchic and non-hierarchic classifications. *Computer Journal*, 11:177–184, 1968
- ✓ A. Banerjee, C. Krumpelman, S. Basu, R. J. Mooney, and J. Ghosh. Model-based overlapping clustering. *KDD 2005*.
- ✓ K. A. Heller and Z. Ghahramani. A nonparametric Bayesian approach to modeling overlapping clusters. *AISTATS 2007*.

What is a Cluster?

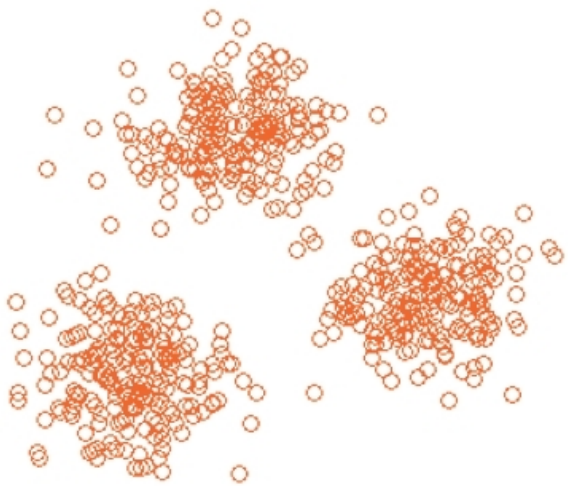
No universally accepted (formal) definition of a “cluster” but, informally, a cluster should satisfy two criteria:

Internal criterion

all “objects” *inside* a cluster should be highly similar to each other

External criterion

all “objects” *outside* a cluster should be highly dissimilar to the ones inside



The Notion of "Gestalt"

«In most visual fields the contents of particular areas "belong together" as circumscribed units from which their surrounding are excluded.»

W. Köhler, *Gestalt Psychology* (1947)

«In gestalt theory the word "Gestalt" means any segregated whole.»

W. Köhler (1929)

Data Clustering: Old *vs.* New

By answering the question "what is a cluster?" we arrive at a novel way of looking at the clustering problem.

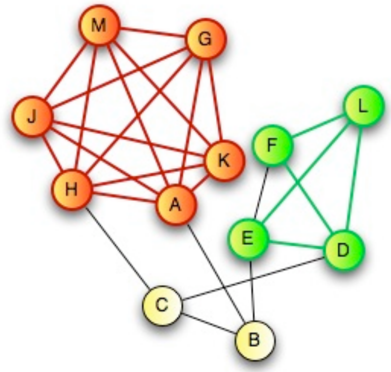
```
Clustering_old(V,A,k)
   V1,V2,...,Vk <- My_favorite_partitioning_algorithm(V,A,k)
   return V1,V2,...,Vk

------

Clustering_new(V,A)
   V1,V2,...,Vk <- Enumerate_all_clusters(V,A)
   return V1,V2,...,Vk

Enumerate_all_clusters(V,A)
   repeat
      Extract_a_cluster(V,A)
   until all clusters have been found
   return the clusters found
```

A Special Case: Binary Symmetric Affinities



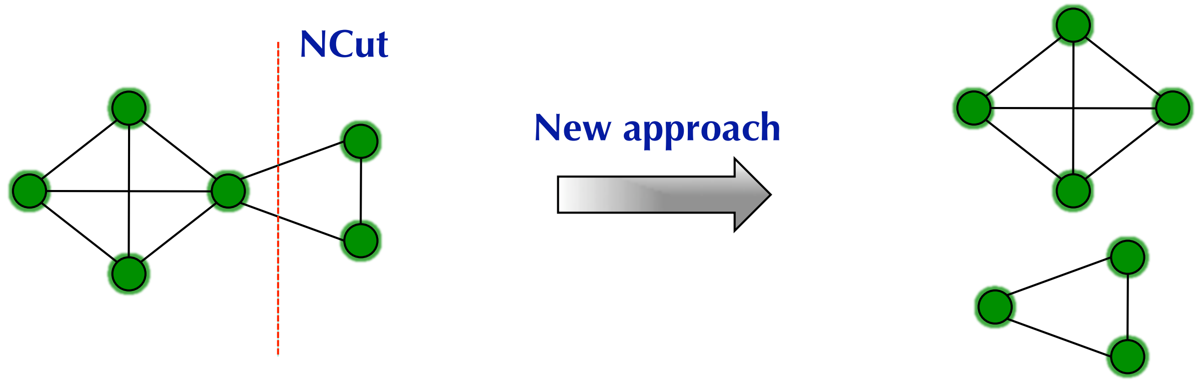
Suppose the similarity matrix is a binary (0/1) matrix.

Given an unweighted undirected graph *G=(V,E)*:

A *clique* is a subset of mutually adjacent vertices

A *maximal clique* is a clique that is not contained in a larger one

In the 0/1 case, a meaningful notion of a cluster is that of a *maximal clique*.



Advantages of the New Approach

- ✓ No need to know the number of clusters in advance (since we extract them sequentially)
- ✓ Leaves clutter elements unassigned (useful, e.g., in figure/ground separation or one-class clustering problems)
- ✓ Allows extracting overlapping clusters

Need a partition?

```
Partition_into_clusters(V,A)
  repeat
     Extract_a_cluster
     remove it from V
  until all vertices have been clustered
```

Evolutionary Stable Strategies as Clusters

Queston: How can we generalize the maximal-clique idea to edge-weighted graphs?

We claim that ESS's abstract well the main characteristics of a cluster:

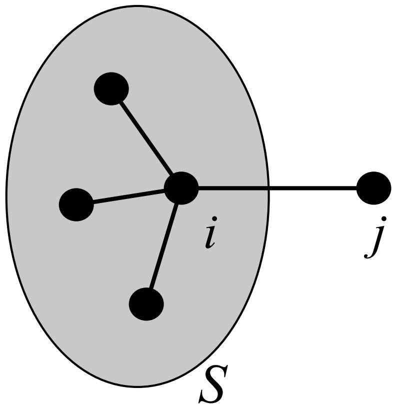
- ✓ **Internal coherency**: High mutual support of all elements within the group.
- ✓ **External incoherency**: Low support from elements of the group to elements outside the group.

Basic Definitions

Let $S \subseteq V$ be a non-empty subset of vertices, and $i \in S$.

The **(average) weighted degree** of i w.r.t. S is defined as:

$$\text{awdeg}_S(i) = \frac{1}{|S|} \sum_{j \in S} a_{ij}$$



Moreover, if $j \notin S$, we define:

$$\phi_S(i,j) = a_{ij} - \text{awdeg}_S(i)$$

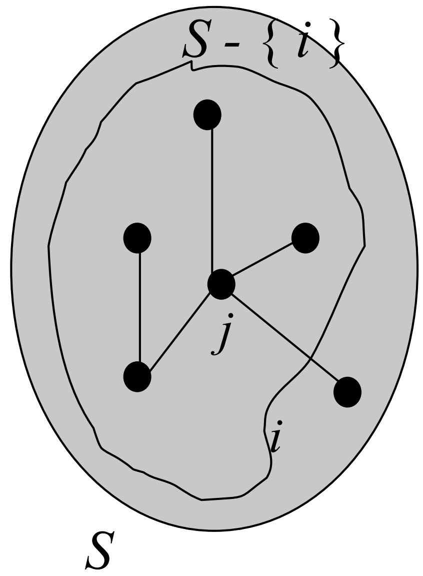
Intuitively, $\phi_S(i,j)$ measures the similarity between vertices j and i, with respect to the (average) similarity between vertex i and its neighbors in S.

Assigning Weights to Vertices

Let $S \subseteq V$ be a non-empty subset of vertices, and $i \in S$.

The **weight** of i w.r.t. S is defined as:

$$w_S(i) = \begin{cases} 1 & \text{if } |S| = 1 \\ \sum_{j \in S-\{i\}} \phi_{S-\{i\}}(j,i)\, w_{S-\{i\}}(j) & \text{otherwise} \end{cases}$$

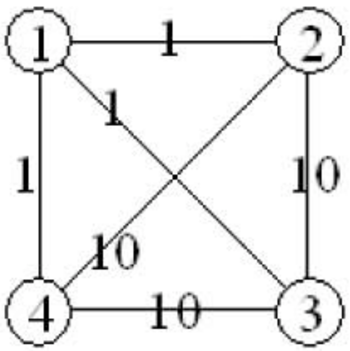


Further, the **total weight** of S is defined as:

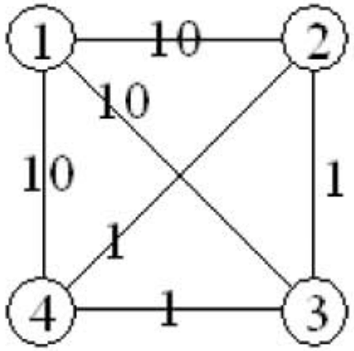
$$W(S) = \sum_{i \in S} w_S(i)$$

Interpretation

Intuitively, $w_S(i)$ gives us a measure of the overall (relative) similarity between vertex i and the vertices of S-{i} with respect to the overall similarity among the vertices in S-{i}.



$w_{\{1,2,3,4\}}(1) < 0$

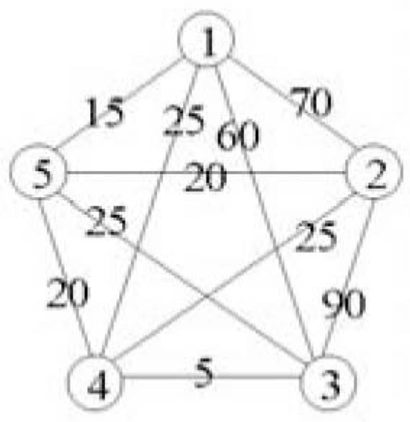


$w_{\{1,2,3,4\}}(1) > 0$

Dominant Sets

Definition (Pavan and Pelillo, 2003, 2007). A non-empty subset of vertices $S \subseteq V$ such that $W(T) > 0$ for any non-empty $T \subseteq S$, is said to be a **dominant set** if:

1. $w_S(i) > 0$, for all $i \in S$ (internal homogeneity)
2. $w_{S \cup \{i\}}(i) < 0$, for all $i \notin S$ (external homogeneity)



The set {1,2,3} is dominant.

Dominant sets ≡ clusters

The Clustering Game

Consider the following “clustering game.”

- ✓ Assume a preexisting set of objects O and a (possibly asymmetric) matrix of affinities A between the elements of O.
- ✓ Two players with complete knowledge of the setup play by simultaneously selecting an element of O.
- ✓ After both have shown their choice, each player receives a payoff, monetary or otherwise, proportional to the affinity that the chosen element has with respect to the element chosen by the opponent.

Clearly, it is in each player’s interest to pick an element that is strongly supported by the elements that the adversary is likely to choose.

Hence, in the (pairwise) clustering game:

- ✓ There are 2 players
- ✓ The objects to be clustered are the pure strategies
- ✓ The (null-diagonal) affinity matrix coincides with the similarity matrix

Dominant Sets are ESS's

Theorem (Torsello, Rota Bulò and Pelillo, 2006). Evolutionary stable strategies of the clustering game with affinity matrix A are in a one-to-one correspondence with dominant sets.

Note. Generalization of well-known Motzkin-Straus theorem from graph theory.

Dominant-set clustering

- ✓ To get a single dominant-set cluster use, e.g., replicator dynamics (but see Rota Bulò, Pelillo and Bomze, *CVIU* in press, for faster dynamics)
- ✓ To get a partition use a simple *peel-off* strategy: iteratively find a dominant set and remove it from the graph, until all vertices have been clustered
- ✓ To get overlapping clusters, enumerate dominant sets (see Bomze, 1992; Torsello, Rota Bulò and Pelillo, 2008)

Special Case: Symmetric Affinities

Given a symmetric real-valued matrix A (with null diagonal), consider the following Standard Quadratic Programming problem (StQP):

$$\begin{array}{ll} \text{maximize} & f(x) = x^T A x \\ \text{subject to} & x \in \Delta \end{array}$$

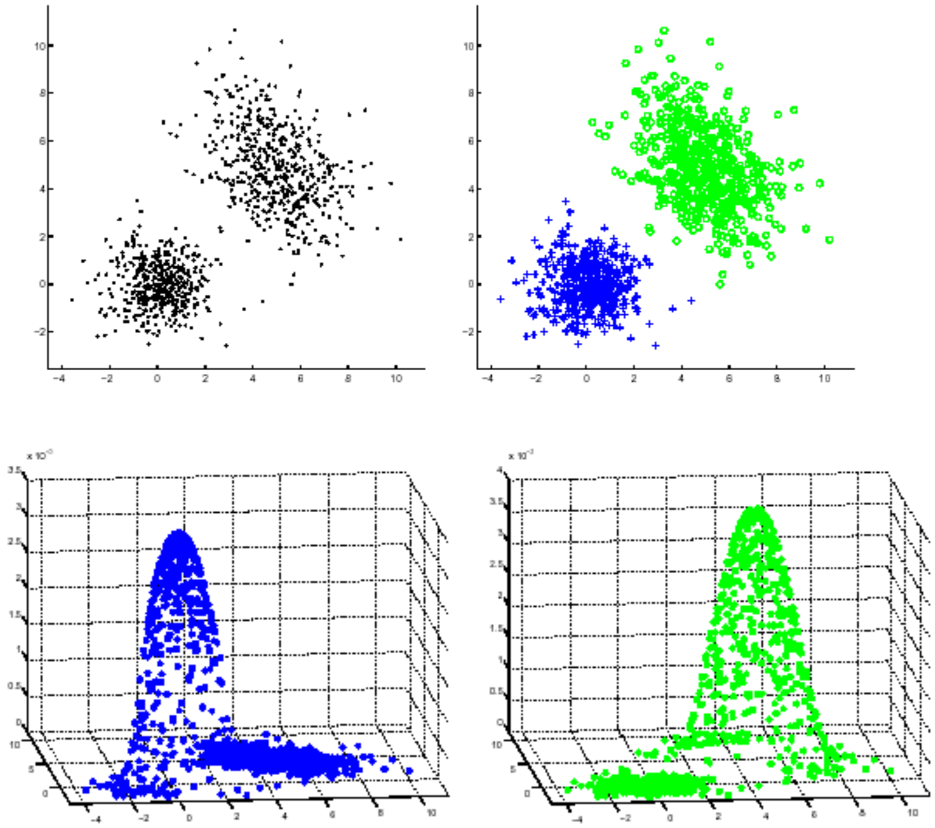
Note. The function $f(x)$ provides a measure of cohesiveness of a cluster (see Pavan and Pelillo, 2003, 2007; Sarkar and Boyer, 1998; Perona and Freeman, 1998).

ESS's are in one-to-one correspondence to (strict) local solutions of StQP

Note. In the 0/1 (symmetric) case, ESS's are in one-to-one correspondence to (strictly) maximal cliques (Motzkin-Straus theorem).

Measuring the Degree of Cluster Membership

The components of the converged vector give us a measure of the participation of the corresponding vertices in the cluster, while the value of the objective function provides of the cohesiveness of the cluster.



Application to Image Segmentation

An image is represented as an edge-weighted undirected graph, where vertices correspond to individual pixels and edge-weights reflect the "similarity" between pairs of vertices.

For the sake of comparison, in the experiments we used the same similarities used in Shi and Malik's normalized-cut paper (PAMI 2000).

To find a hard partition, the following *peel-off* strategy was used:

```
Partition_into_dominant_sets(G)
Repeat
   find a dominant set
   remove it from graph
until all vertices have been clustered
```

To find a single dominant set we used replicator dynamics (but see Rota Bulò, Pelillo and Bomze, *CVIU 2011*, for faster game dynamics).

Experimental Setup

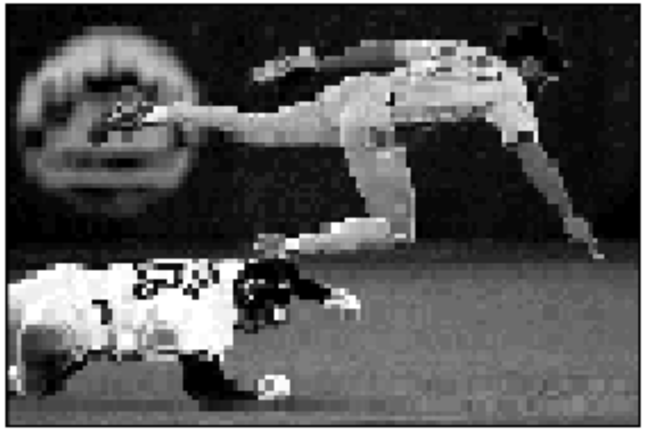
The similarity between pixels i and j was measured by:

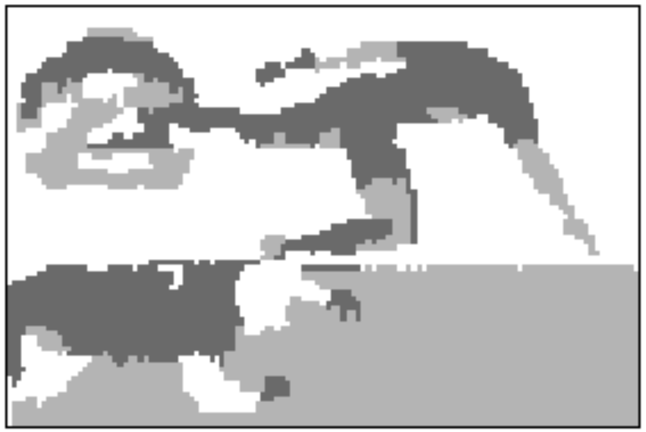
$$w(i,j) = \exp\left(\frac{-\|\mathbf{F}(i) - \mathbf{F}(j)\|_2^2}{\sigma^2}\right)$$

where σ is a positive real number which affects the decreasing rate of w, and:

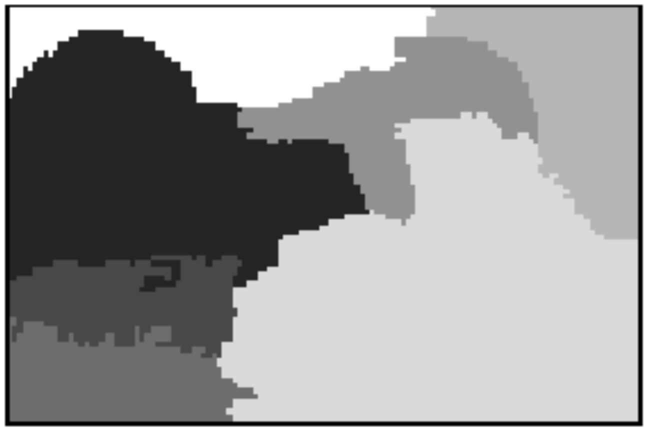
- $\mathbf{F}(i) \equiv$ (normalized) intensity of pixel i, for intensity segmentation
- $\mathbf{F}(i) = [v, vs\sin(h), vs\cos(h)](i)$, where h, s, v are the HSV values of pixel i, for color segmentation
- $\mathbf{F}(i) = [|I * f_1|, \ldots, |I * f_k|](i)$ is a vector based on texture information at pixel i, the f_i being DOOG filters at various scales and orientations, for texture segmentation

Intensity Segmentation Results



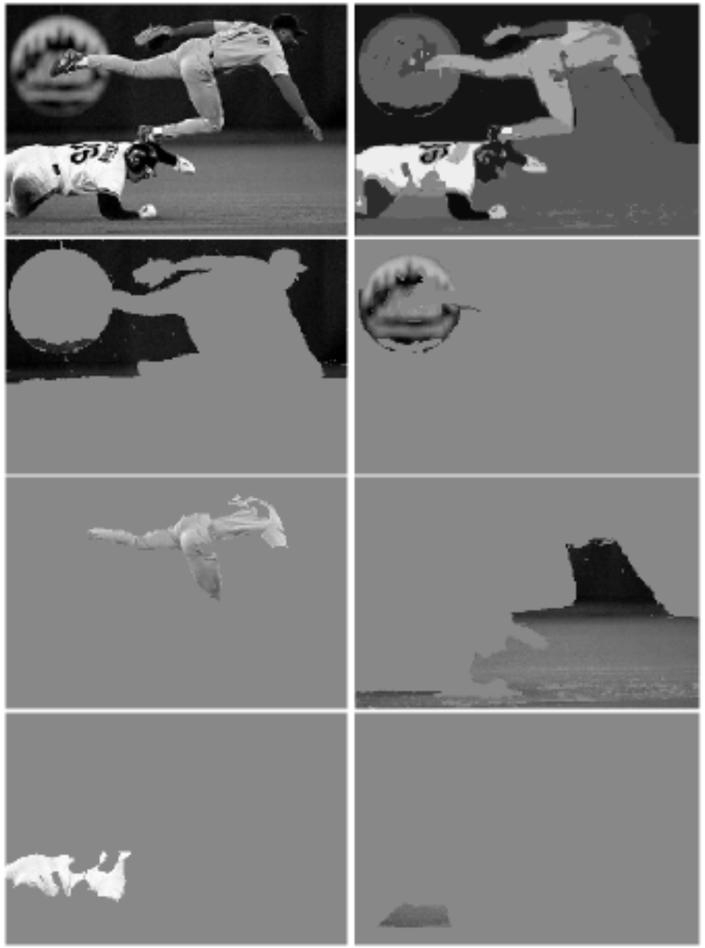


Dominant sets



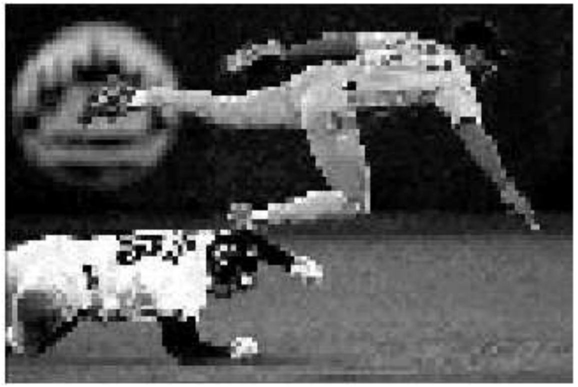
Ncut

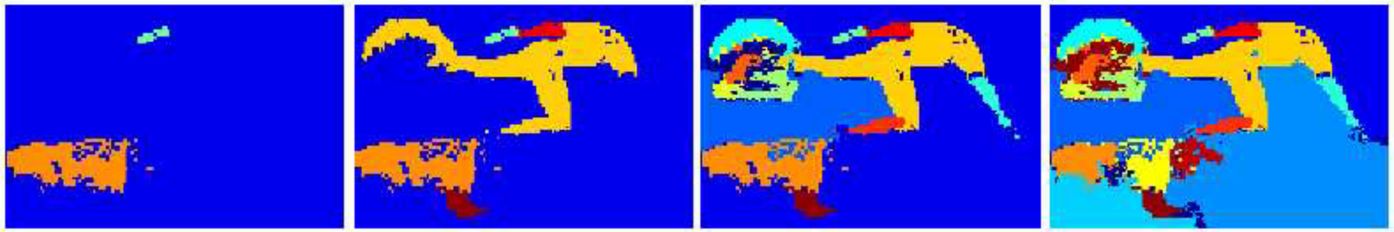
Intensity Segmentation Results



Felzenszwalb and Huttenlocher (2003).

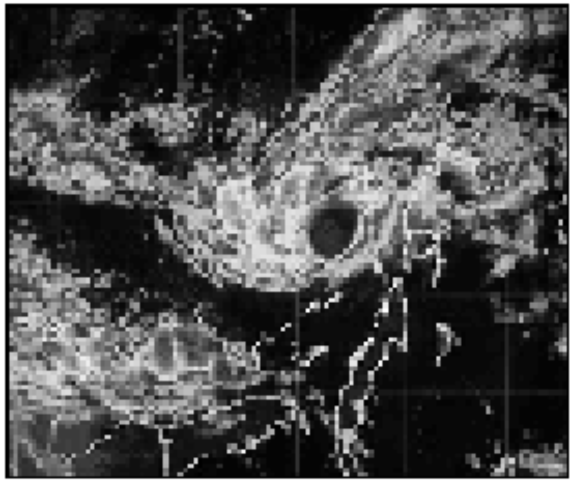
Intensity Segmentation Results

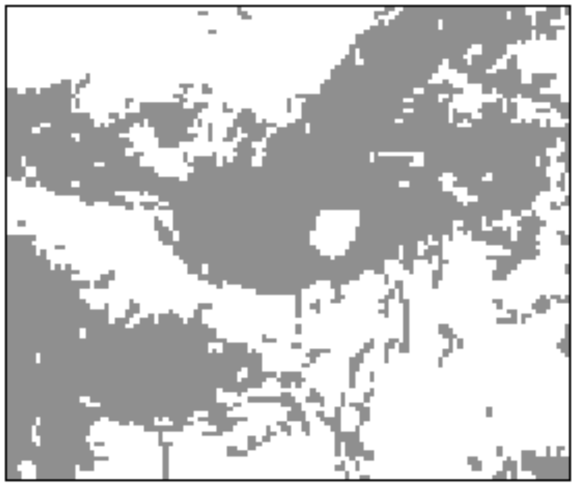




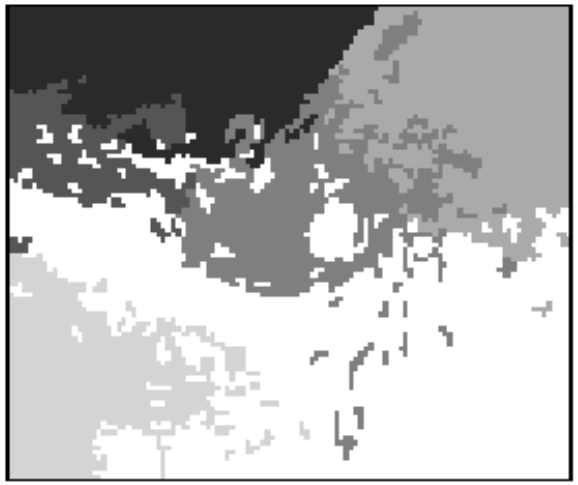
Gdalyahu, Weinshall, and Werman (2001).

Intensity Segmentation Results



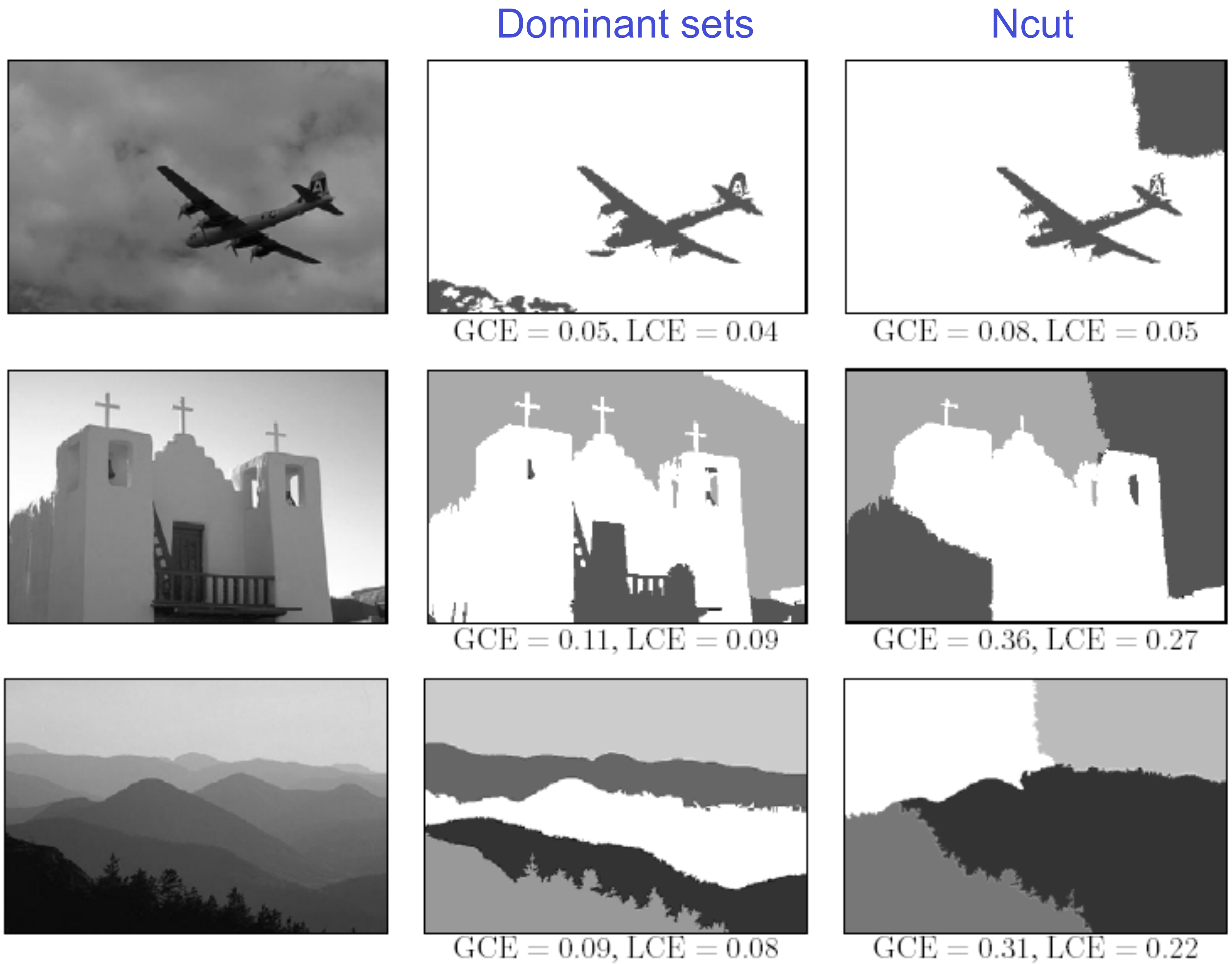


Dominant sets

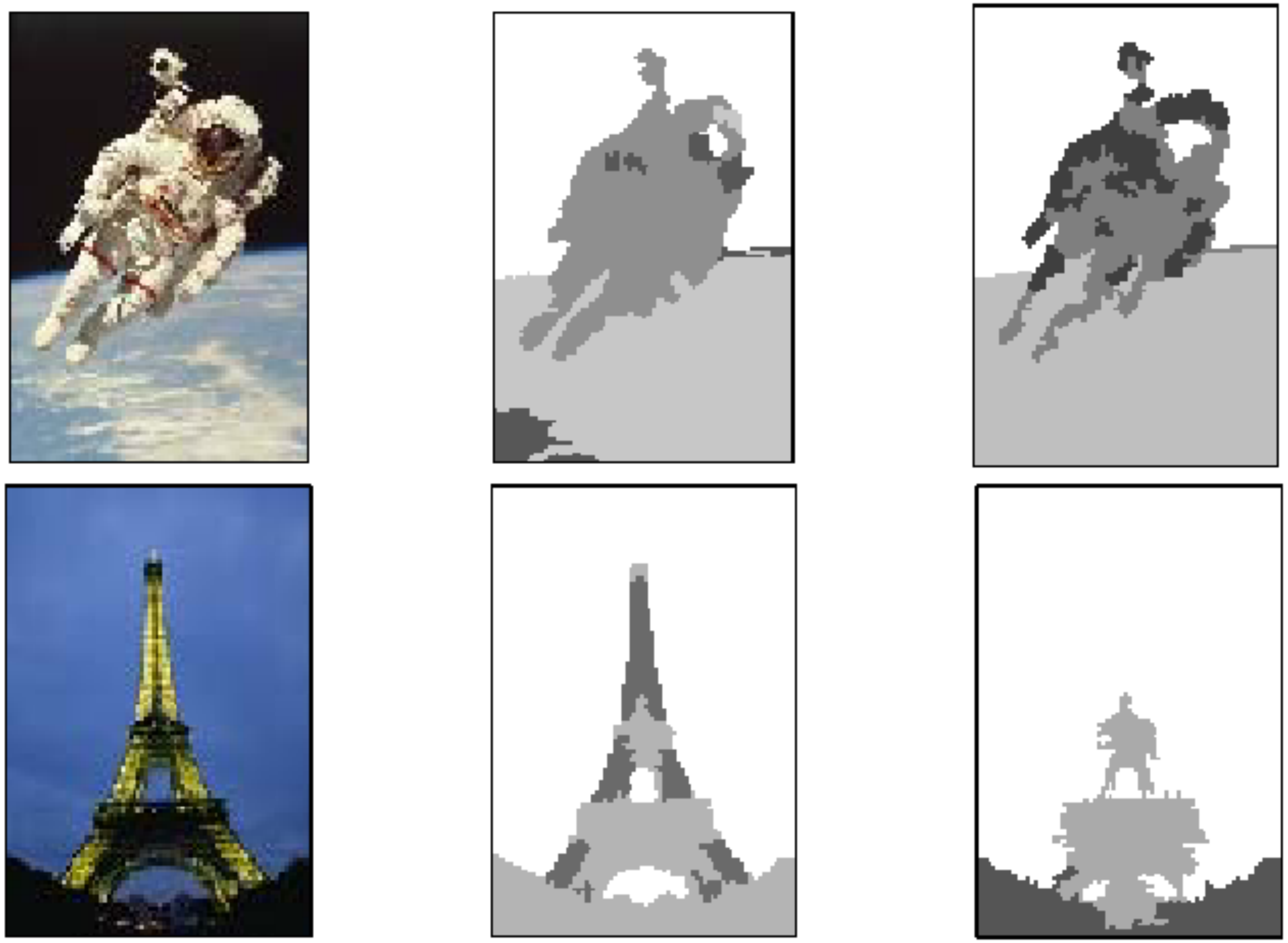


Ncut

Results on the Berkeley Dataset



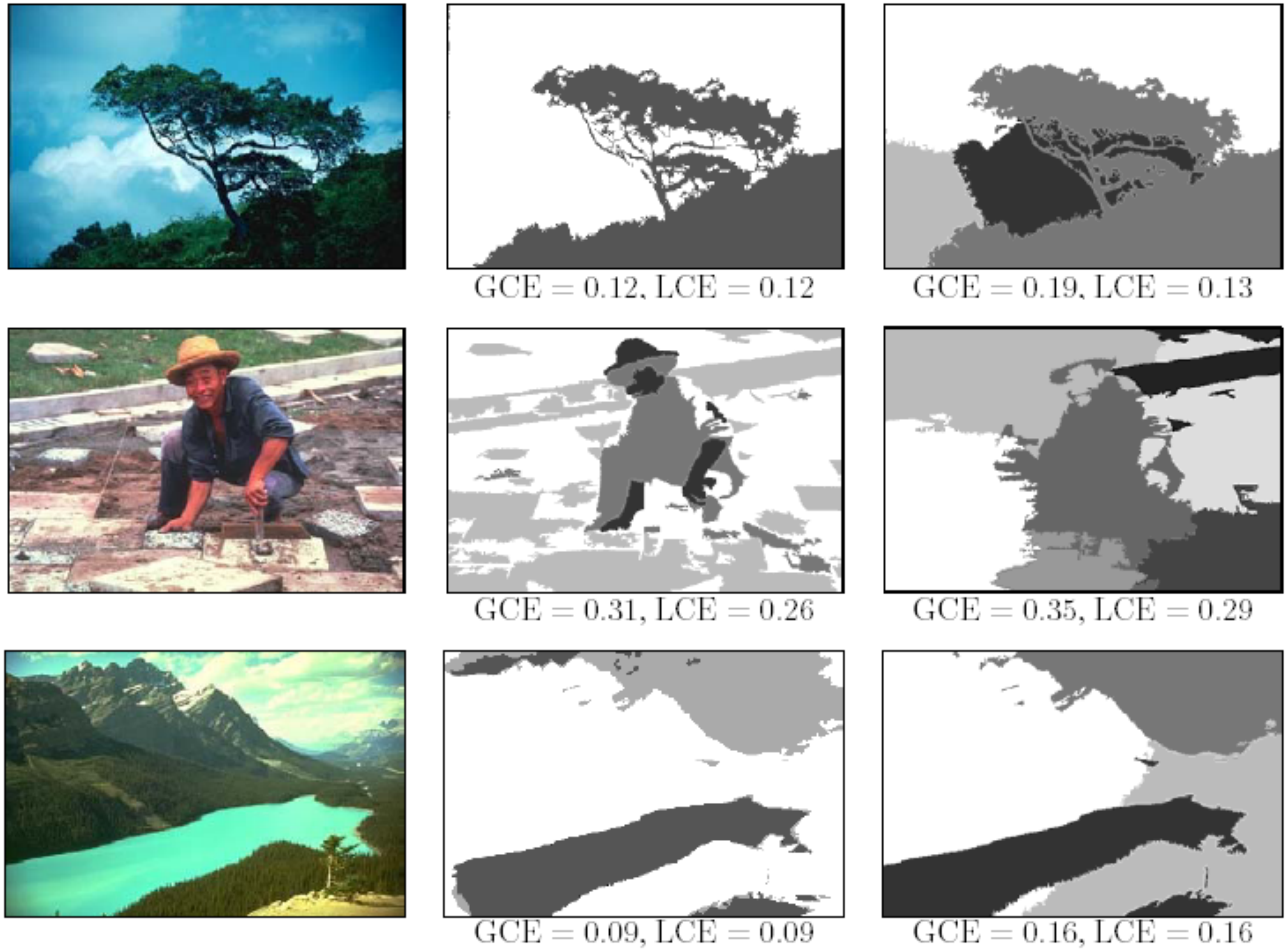
Color Segmentation Results



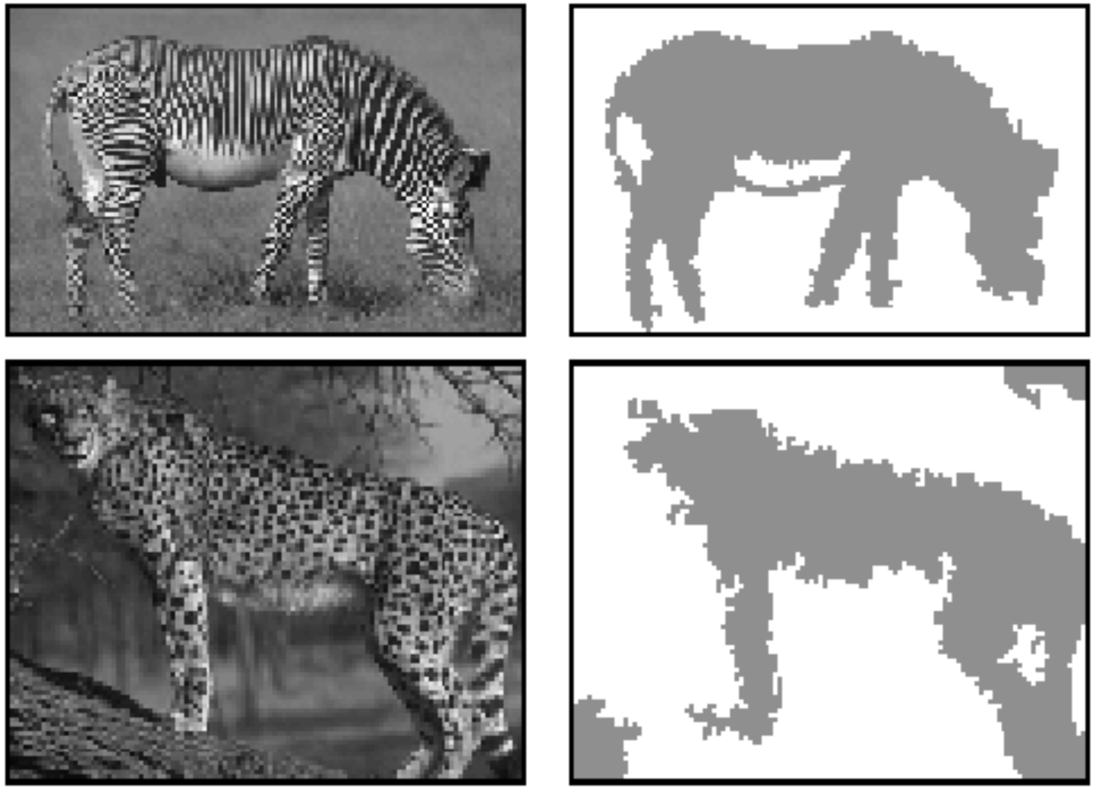
Original image Dominant sets Ncut

Results on the Berkeley Dataset

Dominant sets | Ncut

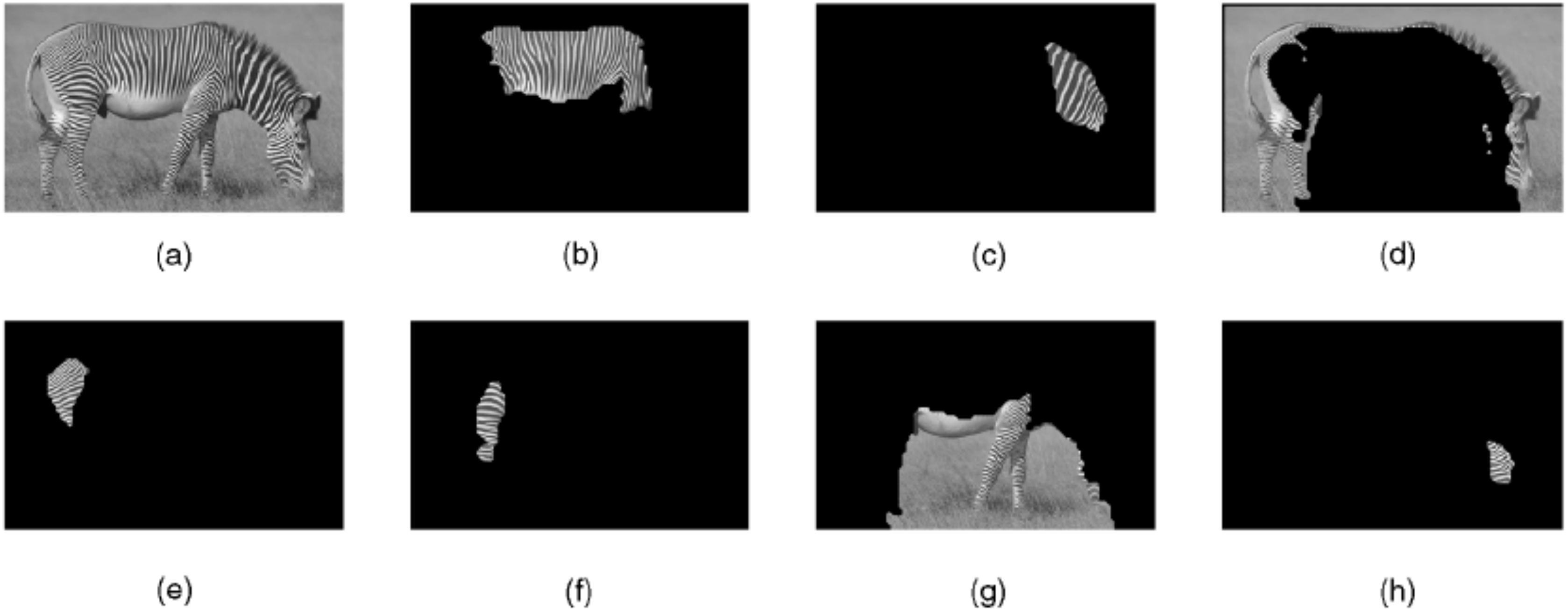


Texture Segmentation Results



Dominant sets

Texture Segmentation Results

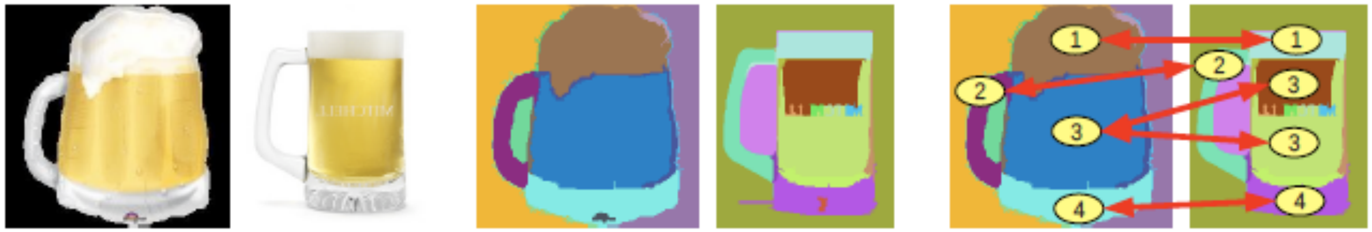


NCut

Graph Matching and Related Problems

Formulate the (graph) matching problem as a dominant-set (game-theoretic) clustering problem and use replicator game dynamics to solve it.

The framework can easily deal with many-to-many matching problems.



Idea: build an "association graph" where nodes correspond to correspondences, edges encode the matching (e.g., isomorphism) constraints, and edge-weights reflect similarities between correspondences.

References: ICCV 2009; IJCV 2012; CVPR 2012, etc.

Other Applications of Dominant-Set Clustering

Security and video surveillance

Detection of anomalous activities in video streams (*Hamid et al.*, CVPR'05; AI'09)
Detection of malicious activities in the internet (*Pouget et al.*, J. Inf. Ass. Sec. 2006)

Content-based image retrieval

Wang et al. (Sig. Proc. 2008)

Analysis of fMRI data

Neumann et al (NeuroImage 2006); *Muller et al* (J. Mag Res Imag. 2007)

Video analysis, object tracking, human action recognition

Torsello et al. (EMMCVPR'05); *Gualdi et al.* (IWVS'08); *Wei et al.* (ICIP'07)

Multiple instance learning

Erdem and Erdem (SIMBAD'11)

Feature selection

Hancock et al. (GbR'11; ICIAP'11; SIMBAD'11)

Image matching and registration

Torsello et al. (ICCV'09, IJCV 2011, CVPR'10, ECCV'10; CVPR'12)

Bioinformatics

Identification of protein binding sites (*Zauhar and Bruist*, 2005)
Clustering gene expression profiles (*Li et al*, 2005)
Tag Single Nucleotide Polymorphism (SNPs) selection (*Frommlet*, 2010)

In a nutshell…

The dominant-set (ESS) approach:

- ✓ makes no assumption on the underlying (individual) data representation
- ✓ makes no assumption on the structure of the affinity matrix, being it able to work with asymmetric and even negative similarity functions
- ✓ does not require *a priori* knowledge on the number of clusters (since it extracts them sequentially)
- ✓ leaves clutter elements unassigned (useful, e.g., in figure/ground separation or one-class clustering problems)
- ✓ allows principled ways of assigning out-of-sample items *(NIPS'04)*
- ✓ allows extracting overlapping clusters *(ICPR'08)*
- ✓ generalizes naturally to hypergraph clustering problems, i.e., in the presence of high-order affinities, in which case the clustering game is played by more than two players *(NIPS'09; PAMI'12 in press)*
- ✓ extends to hierarchical clustering *(ICCV'03: EMMCVPR'09)*

References

M. Pavan and M. Pelillo. A new graph-theoretic approach to clustering and segmentation. *CVPR 2003*.

M. Pavan and M. Pelillo. Dominant sets and hierarchical clustering. *ICCV 2003*.

M. Pavan and M. Pelillo. Efficient out-of-sample extension of dominant-set clusters. *NIPS 2004*.

A. Torsello, S. Rota Bulò and M. Pelillo. Grouping with asymmetric affinities: A game-theoretic perspective. *CVPR 2006*.

M. Pavan and M. Pelillo. Dominant sets and pairwise clustering. *PAMI 2007*.

A. Torsello, S. Rota Bulò and M. Pelillo. Beyond partitions: Allowing overlapping groups in pairwise clustering. *ICPR 2008*.

S. Rota Bulò and M. Pelillo. A game-theoretic approach to hypergraph clustering. *NIPS 2009; PAMI 2012 (in press)*.

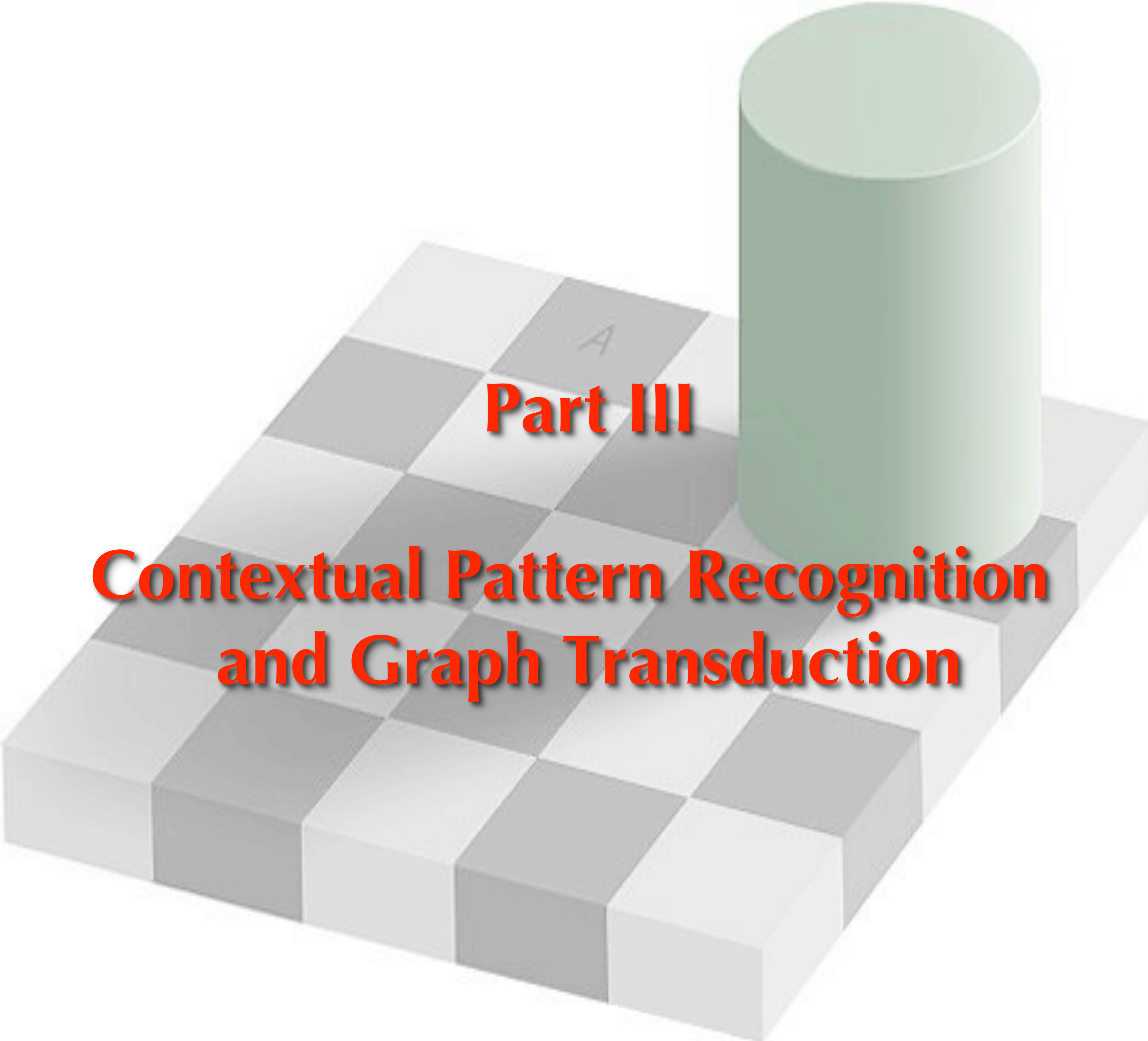
M. Pelillo. What is a cluster? Perspectives from game theory. *NIPS 2009 Workshop on "Clustering: Science or Art?"* (talk available on videolectures.net).

S. Rota Bulò, M. Pelillo and I. M. Bomze. Graph-based quadratic optimization: A fast evolutionary approach. *CVIU 2011*.

A. Albarelli *et al.* Matching as a non-cooperative game. *ICCV 2009*.

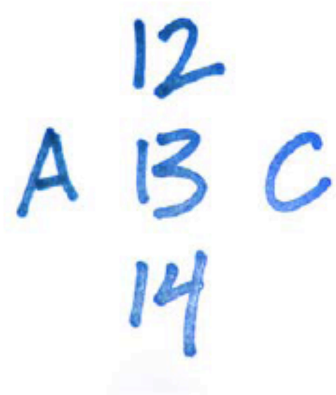
Part III

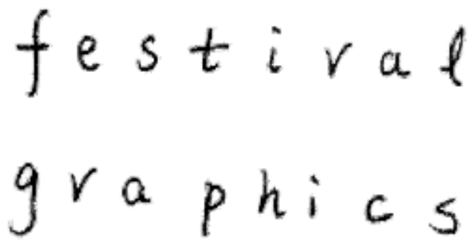
Contextual Pattern Recognition and Graph Transduction



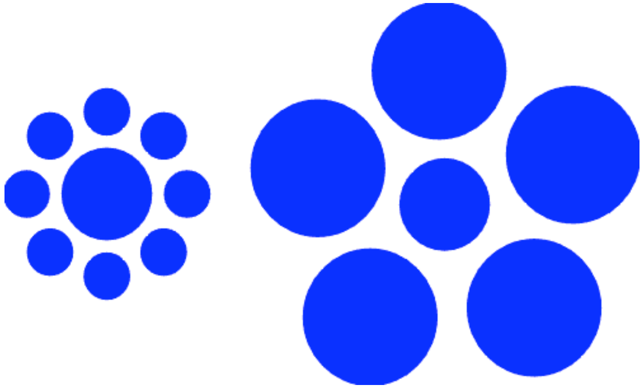
Context helps …

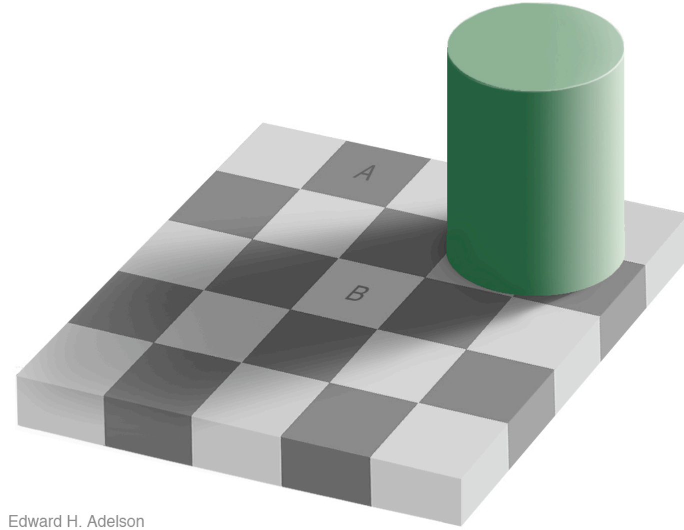
c	→	cat
	→	circus
i	→	sin
	→	fine
e	→	red
	→	read

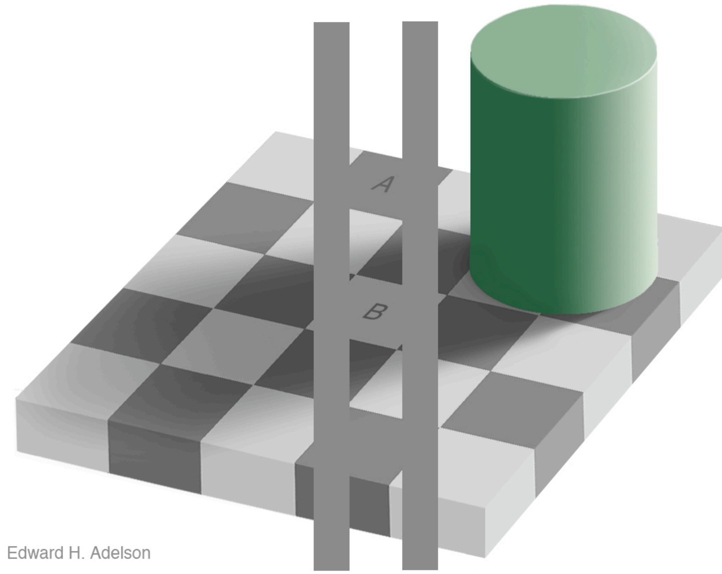




… but can also deceive!







What do you see?

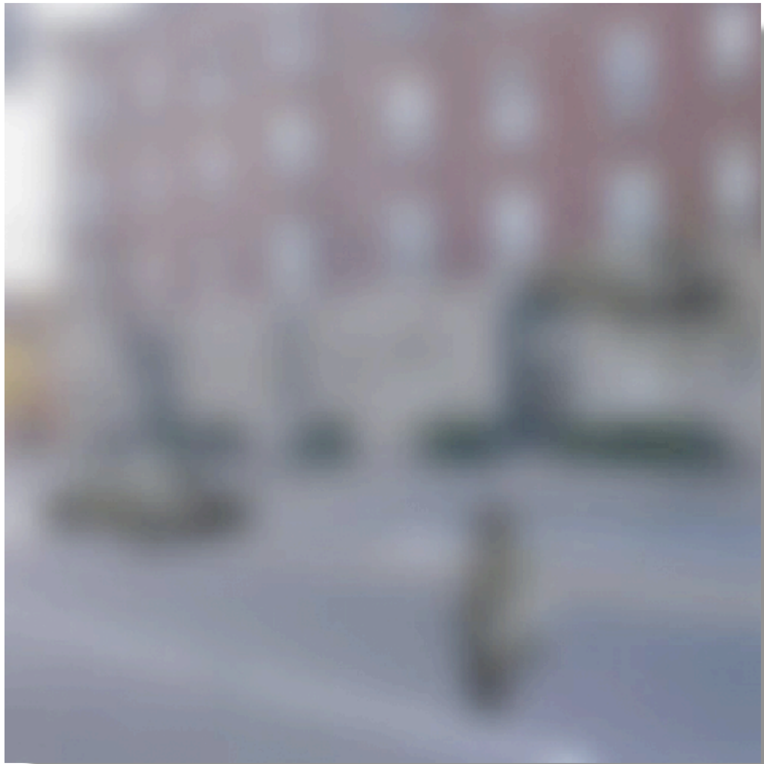
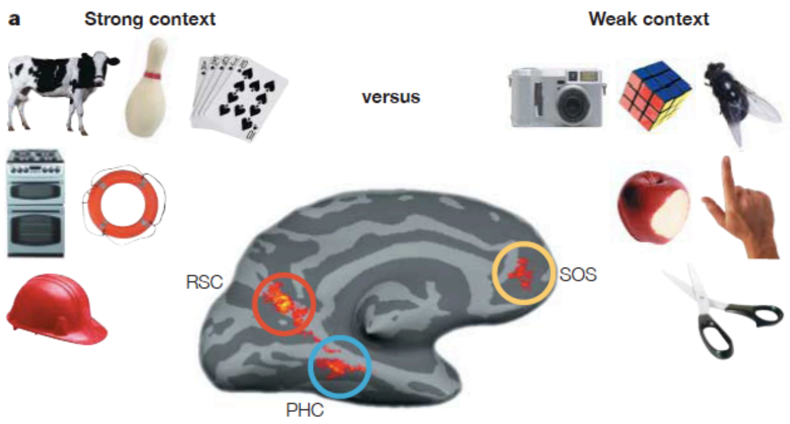


Figure 2. The strength of context. The visual system makes assumptions regarding object identities according to their size and location in the scene. In this picture, observers describe the scene as containing a car and pedestrian in the street. However, the pedestrian is in fact the same shape as the car, except for a 90° rotation. The atypicality of this orientation for a car within the context defined by the street scene causes the car to be recognized as a pedestrian.

From: A. Oliva and A. Torralba, “The role of context in object recognition”, *Trends in Cognitive Sciences*, 2007.

Context and the Brain



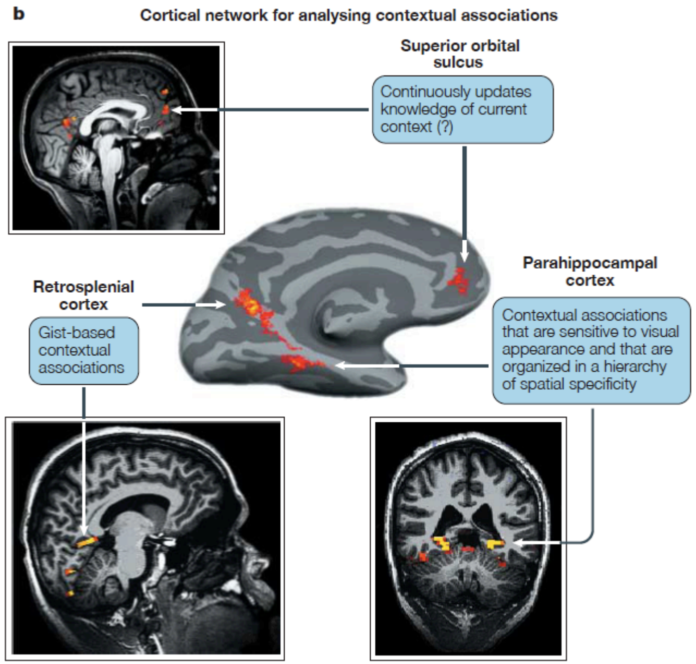


Figure 3 | **Cortical areas involved in processing context. a** | A functional magnetic resonance imaging (fMRI) statistical activation map representing the difference between perceiving objects that are strongly associated with a specific context and perceiving objects that are not associated with a unique context. This is a medial view of the left hemisphere, shown using a precise computer reconstruction where the sulci have been exposed by 'inflation'. The parahippocampal cortex (PHC) is circled in blue; the retrosplenial cortex (RSC) is circled in red; the superior orbital sulcus (SOS) is circled in yellow. Note that in all experimental conditions, subjects viewed similar looking colour photographs of meaningful, everyday common objects that were equally recognizable. Consequently, activation due to low-level processes was presumably subtracted out, and the differential activation map shown here represents only processes that are related to the level of contextual association. **b** | The cortical network for contextual associations among visual objects, suggested on the basis of existing evidence. Other types of context might involve additional regions (for example, hippocampus for navigation[125] and Broca's area for language-related context). Modified, with permission, from REF. 12 © (2003) Elsevier Science.

From: M. Bar, "Visual objects in context", *Nature Reviews Neuroscience*, August 2004.

The (Consistent) Labeling Problem

A **labeling problem** involves:

- ✓ A set of n **objects** $B = \{b_1,\ldots,b_n\}$
- ✓ A set of m **labels** $\Lambda = \{1,\ldots,m\}$

The goal is to label each object of B with a label of Λ.

To this end, two sources of information are exploited:

- ✓ Local measurements which capture the salient features of each object viewed in isolation
- ✓ Contextual information, expressed in terms of a real-valued n^2 x m^2 matrix of **compatibility coefficients** $R = \{r_{ij}(\lambda, \mu)\}$.

The coefficient $r_{ij}(\lambda, \mu)$ measures the strenght of compatibility between the two hypotheses: "b_i is labeled λ" and "b_j is labeled μ".

Relaxation Labeling Processes

The initial local measurements are assumed to provide, for each object $b_i \in B$, an m-dimensional (probability) vector:

$$p_i^{(0)} = \left(p_i^{(0)}(1), \cdots, p_i^{(0)}(m)\right)^T$$

with $p_i^{(0)}(\lambda) \geq 0$ and $\sum_\lambda \, p_i^{(0)}(\lambda) = 1$. Each $p_i^{(0)}(\lambda)$ represents the initial, non-contextual degree of confidence in the hypothesis "b_i is labeled λ".

By concatenating vectors $p_1^{(0)}, \ldots, p_n^{(0)}$ one obtains an (initial) **weighted labeling assignment** $p^{(0)} \in \Re^{nm}$.

The space of weighted labeling assignments is

$$\mathrm{IK} = \underbrace{\Delta \times \ldots \times \Delta}_{m \text{ times}}$$

where each Δ is the standard simplex of $\Re^n$. Vertices of IK represent unambiguous labeling assignments

A **relaxation labeling process** takes the initial labeling assignment $p^{(0)}$ as input and iteratively updates it taking into account the compatibility model R.

Relaxation Labeling Processes

In a now classic 1976 paper, Rosenfeld, Hummel, and Zucker introduced heuristically the following update rule (assuming a non-negative compatibility matrix):

$$p_i^{(t+1)}(\lambda) = \frac{p_i^{(t)}(\lambda) q_i^{(t)}(\lambda)}{\sum_\mu p_i^{(t)}(\mu) q_i^{(t)}(\mu)}$$

where

$$q_i^{(t)}(\lambda) = \sum_j \sum_\mu r_{ij}(\lambda,\mu) p_i^{(t)}(\mu)$$

quantifies the support that context gives at time t to the hypothesis "b_i is labeled with label λ".

See (Pelillo, 1997) for a rigorous derivation of this rule in the context of a formal theory of consistency.

Hummel and Zucker's Consistency

In 1983, Bob Hummel and Steve Zucker developed an elegant theory of consistency in labeling problem.

By analogy with the unambiguous case, which is easily understood, they define a weighted labeling assignment $p \in$ IK **consistent** if:

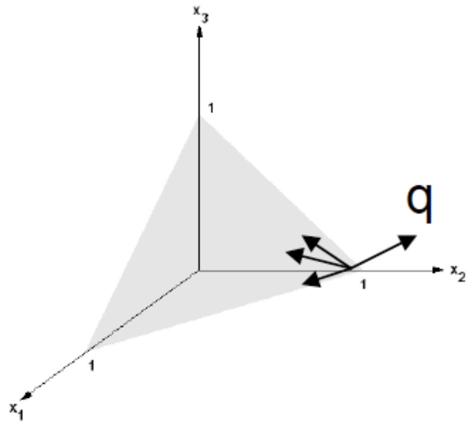
$$\sum_{\lambda} p_i(\lambda) q_i(\lambda) \geq \sum_{\lambda} v_i(\lambda) q_i(\lambda) \qquad i = 1 \ldots n$$

for all labeling assignments $v \in$ IK.

If strict inequalities hold for all $v \neq p$, then p is said to be **strictly consistent**.

Geometrical interpretation.

The support vector q points away from all tangent vectors at p (it has null projection in IK).



Generalization of classical constraint satisfaction problems!

Applications

Since their introduction in the mid-1970's relaxation labeling algorithms have found applications in virtually all problems in computer vision and pattern recognition:

- ✓ Edge and curve detection and enhancement
- ✓ Region-based segmentation
- ✓ Stereo matching
- ✓ Shape and object recognition
- ✓ Grouping and perceptual organization
- ✓ Graph matching
- ✓ Handwriting interpretation
- ✓ …

Further, intriguing similarities exist between relaxation labeling processes and certain mechanisms in the early stages of biological visual systems (see Zucker, Dobbins and Iverson, 1989, for physiological and anatomical evidence).

Relaxation Labeling as a Noncooperative Game

As observed by Miller and Zucker (1991) the consistent labeling problem is equivalent to a noncoperative (polymatrix) game.

Indeed, in such formulation we have:

- ✓ Objects = players
- ✓ Labels = pure strategies
- ✓ Weighted labeling assignments = mixed strategies
- ✓ Compatibility coefficients = payoffs

and:

- ✓ Consistent labeling = Nash equilibrium
- ✓ Strictly consistent labeling = strict Nash equilibrium

Further, the RHZ update rule corresponds to discrete-time multi-population "replicator dynamics" used in evolutionary game theory (see Part I).

Semi-Supervised Learning

Unsupervised learning

- Learning with unlabeled data $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$

Supervised learning

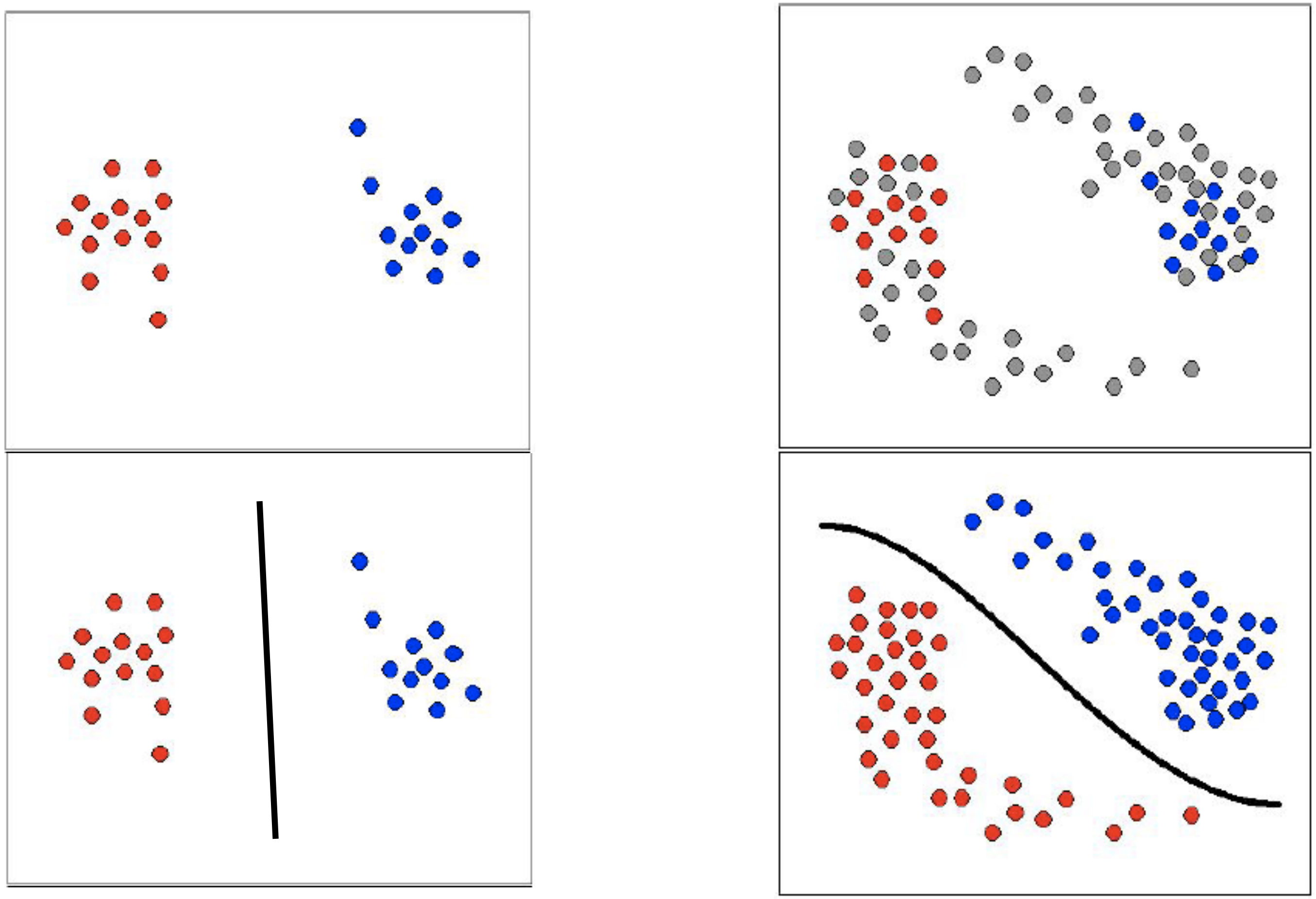
- Learning with labeled data $\{(\mathbf{x}_1, y_1, \ldots, \mathbf{x}_n, y_n)\}$
- Finding a mapping from the feature space to the label space $f : \mathcal{X} \to \mathcal{Y}$

Semi-supervised learning

- Learning with labeled and unlabeled data
    - labeled data: $\{(\mathbf{x}_1, y_1), \ldots, \mathbf{x}_\ell, y_\ell)\}$
    - unlabeled data: $\{\mathbf{x}_{\ell+1}, \ldots, \mathbf{x}_n\}$

Can we find a better classifier from both labeled and unlabeled data?

Unlabeled Points Can Help…



Adapted from: O. Duchene, J.-Y. Audibert, R. Keriven, J. Ponce, and F. Ségonne. Segmentation by transduction. *CVPR 2008*.

Graph Transduction

Given a set of data points grouped into:

- ✓ labeled data: $\{(\mathbf{x}_1, y_1), \ldots, \mathbf{x}_\ell, y_\ell)\}$
- ✓ unlabeled data: $\{\mathbf{x}_{\ell+1}, \ldots, \mathbf{x}_n\}$ $\qquad \ell \ll n$

Express data as a graph *G=(V,E)*

- ✓ *V* : nodes representing labeled and unlabeled points
- ✓ *E* : pairwise edges between nodes weighted by the similarity between the corresponding pairs of points

Goal: Propagate the information available at the labeled nodes to unlabeled ones in a "consistent" way.

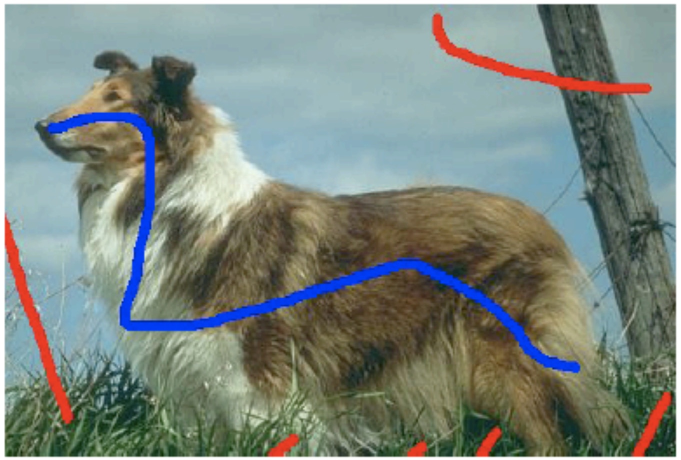
Cluster assumption:

- ✓ The data form distinct clusters
- ✓ Two points in the same cluster are expected to be in the same class

An Application:
Interactive Image Segmentation

Segmentation by transduction: “Given a set of user-supplied seeds representative of each region to be segmented in an image, generate a segmentation of the entire image that is consistent with the seeds.”

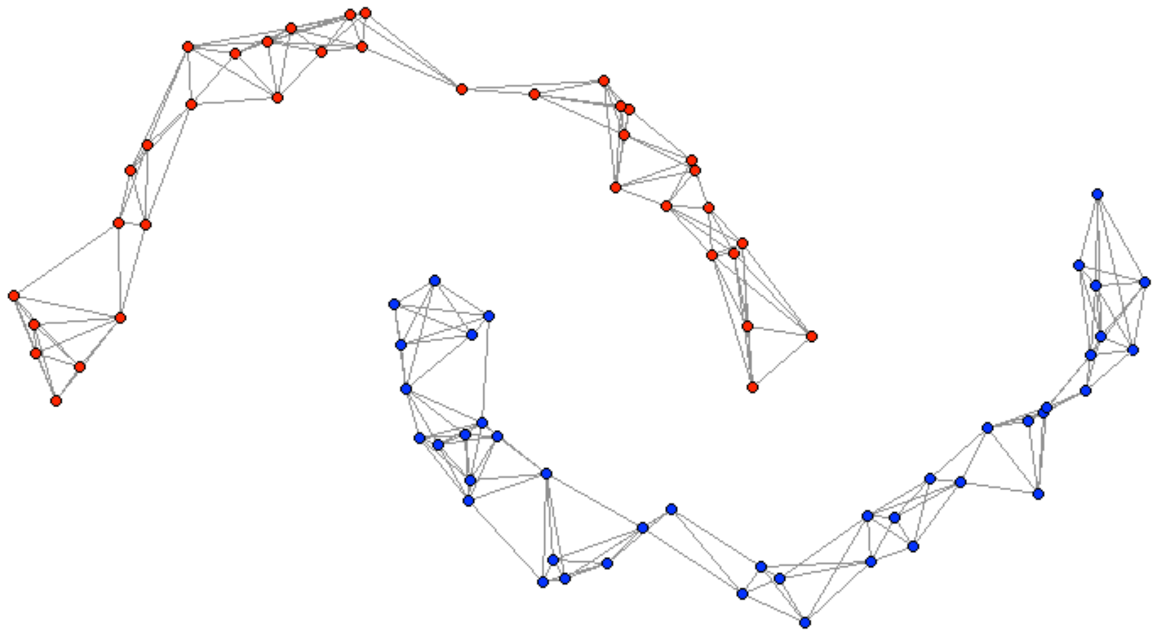
From: O. Duchene, J.-Y. Audibert, R. Keriven, J. Ponce, and F. Ségonne. Segmentation by transduction. *CVPR 2008*.



A Special Case: Unweighted Undirected Graphs

A simple case of graph transduction in which the graph G is an unweighted undirected graph:

- ✓ An edge denotes perfect similarity between points
- ✓ The adjacency matrix of G is a 0/1 matrix



The cluster assumption: Each node in a connected component of the graph should have the same class label.

A Special Case: Unweighted Undirected Graphs

This toy problem can be formulated as a (binary) **constraint satisfaction problem** (CSP) as follows:

- ✓ The set of variables: $V = \{v_1, \ldots, v_n\}$
- ✓ Domains: $D_{v_i} = \begin{cases} \{y_i\} & \text{for all } 1 \le i \le l \\ Y & \text{for all } l+1 \le i \le n \end{cases}$
- ✓ Binary constraints: $\forall i,j$: if $a_{ij} = 1$, then $v_i = v_j$
  e.g. for a 2-class problem $R_{ij} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

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

Question: How to generalize this to the case of real-valued (soft) constraints?

Idea: Use consistency criterion of relaxation labeling (= Nash equilibrium)

The Graph Transduction Game

Assume:

- ✓ the players participating in the game correspond to the vertices of the graph
- ✓ the set of strategies available to each player denote the possible hypotheses about its class membership

- labeled players $\mathcal{I}_\ell = \{\mathcal{I}_{\ell|1}, \ldots, \mathcal{I}_{\ell|c}\}$
- unlabeled players $\mathcal{I}_u$

Labeled players choose their strategies at the outset:

- ✓ each player $i \in \mathcal{I}_{l|k}$ always play its k^{th} pure strategy.

The transduction game is in fact played among the unlabeled players to choose their memberships.

By assuming that only pairwise interactions are allowed, we obtain a non-cooperative (polymatrix) game that can be solved used standard relaxation labeling / replicator algorithms.

Defining the Payoffs

If the fixed choices of labeled players are considered, the payoff function is:

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

But how to specify partial payoff matrices?

If $A = (A_{ij})$ represent partial payoff matrices in block form, we define

$$A = I_c \otimes W$$

e.g., for a 3-class problem:

$$A_{ij} = \begin{bmatrix} w_{ij} & 0 & 0 \\ 0 & w_{ij} & 0 \\ 0 & 0 & w_{ij} \end{bmatrix}$$

We end up with a generalization of the binary CSP
for the toy transduction problem!

Example Results: Symmetric Similarities

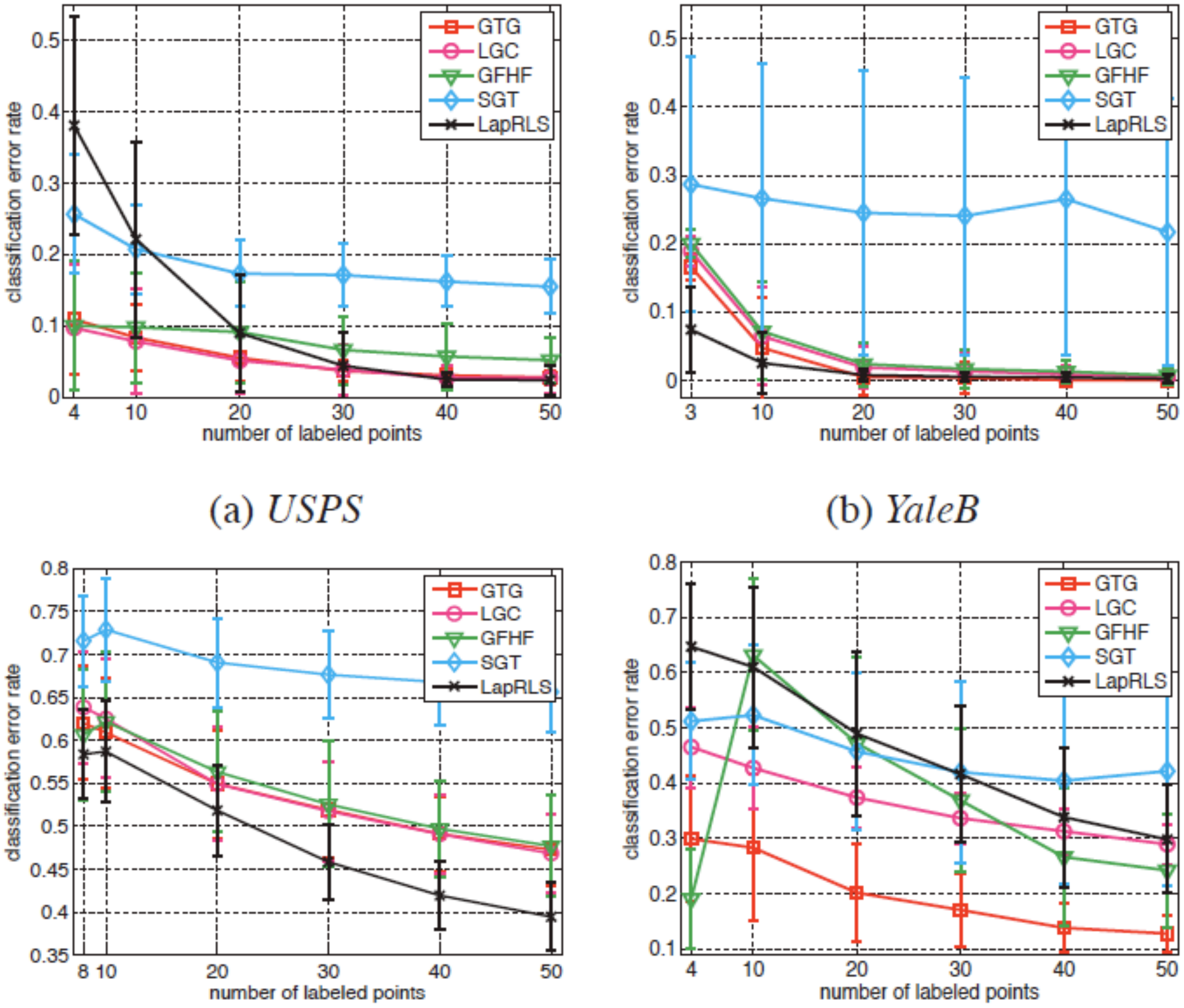
Data sets used: *USPS, YaleB, Scene, 20-news*

	USPS	*YaleB*	*Scene*	*20-news*
# objects	3874	1755	2688	3970
# dimensions	256	1200	512	8014
# classes	4	3	8	4

Methods compared:

- ✓ *Gaussian fields and harmonic functions* (GFHF) (Zhu *et al.*, 2003)
- ✓ *Spectral Graph Transducer* (SGT) (Joachims, 2003)
- ✓ *Local and global consistency* (LGC) (Zhou *et al.*, 2004)
- ✓ *Laplacian Regularized Least Squares* (LapRLS) (Belkin *et al.*, 2006)

Example Results: Symmetric Similarities



(a) *USPS* (b) *YaleB*

(c) *Scene* (d) *20-news*

In short…

Graph transduction can be formulated as a (polymatrix) non-cooperative game (i.e., a consistent labeling problem).

The proposed game-theoretic framework is inherently a **multiclass approach** and can naturally cope with **symmetric, negative and asymmetric similarities** (none of the existing techniques is able to deal with all three types of similarities).

Experimental results on standard datasets show that our approach is not only more general but also competitive with standard approaches.

A. Erdem and M. Pelillo. Graph transduction as a noncooperative game. *Neural Computation* 24:3 (March 2012).

Extensions

The approach described here can be naturally extended along several directions:

- ✓ Using more powerful algorithms than "plain" replicator dynamics (e.g., Porter et al., 2008; Rota Bulò and Bomze, 2010)
- ✓ Dealing with high-order interactions (i.e., hypergraphs) (e.g., Agarwal et al., 2006; Rota Bulò and Pelillo, 2009)
- ✓ From the "homophily" to the "Hume" similarity principle?
- ✓ Introducing uncertainty in "labeled" players

References

J. T. Howson. Equilibria of polymatrix games. *Management Science* (1972).

A. Rosenfeld, R. A. Hummel, and S. W. Zucker. Scene labeling by relaxation operations. *IEEE Trans. Syst. Man. Cybern* (1976).

R. A. Hummel and S. W. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. Pattern Anal. Machine Intell.* (1983).

M. Pelillo. The dynamics of nonlinear relaxation labeling processes. *J. Math. Imaging and Vision* (1997).

J. Kittler and J. Illingworth. Relaxation labeling algorithms – A review. *Image and Vision Computing* (1986).

D. A. Miller and S. W. Zucker. Copositive-plus Lemke algorithm solves polymatrix games. *Operation Research Letters* (1991).

D. A. Miller and S. W. Zucker. Efficient simplex-like methods for equilibria of nonsymmetric analog networks. *Neural Computation (*1992).

S. W. Zucker. Relaxation labeling: 25 years and still iterating. In: L. S. Davis (Ed.) *Foundations of Image Understanding*. Kluwer Academic Publisher (2001).

A. Erdem and M. Pelillo. Graph transduction as a noncooperative game. *Neural Computation* 24:3 (March 2012).

Other Applications of Game Theory to Pattern Recognition and Computer Vision

Connections to MRF's and learning automata

Yu and Berthod (CVIU, 1995)

Sastry, Phansalkar, and Thathachar (SMC 1994)

Integration of visual cues for image analysis

Duncan et al. 1992, 1994*; Chakraborty and Duncan* (PAMI 1999)

Coalitional games for feature selection and tracking

Cohen, Dror, and Ruppin, 2007*; Dowdall, Pavlidis, and Tsiamyrtzis*, 2007

Graph/image matching, in/outlier detection, detection of anomalous behavior

Torsello et al. (IJCV 2011, ICCV'09, CVPR'10)

Integration of classifiers/predictors through regret minimization

Cesa-Bianchi and Lugosi (2006)