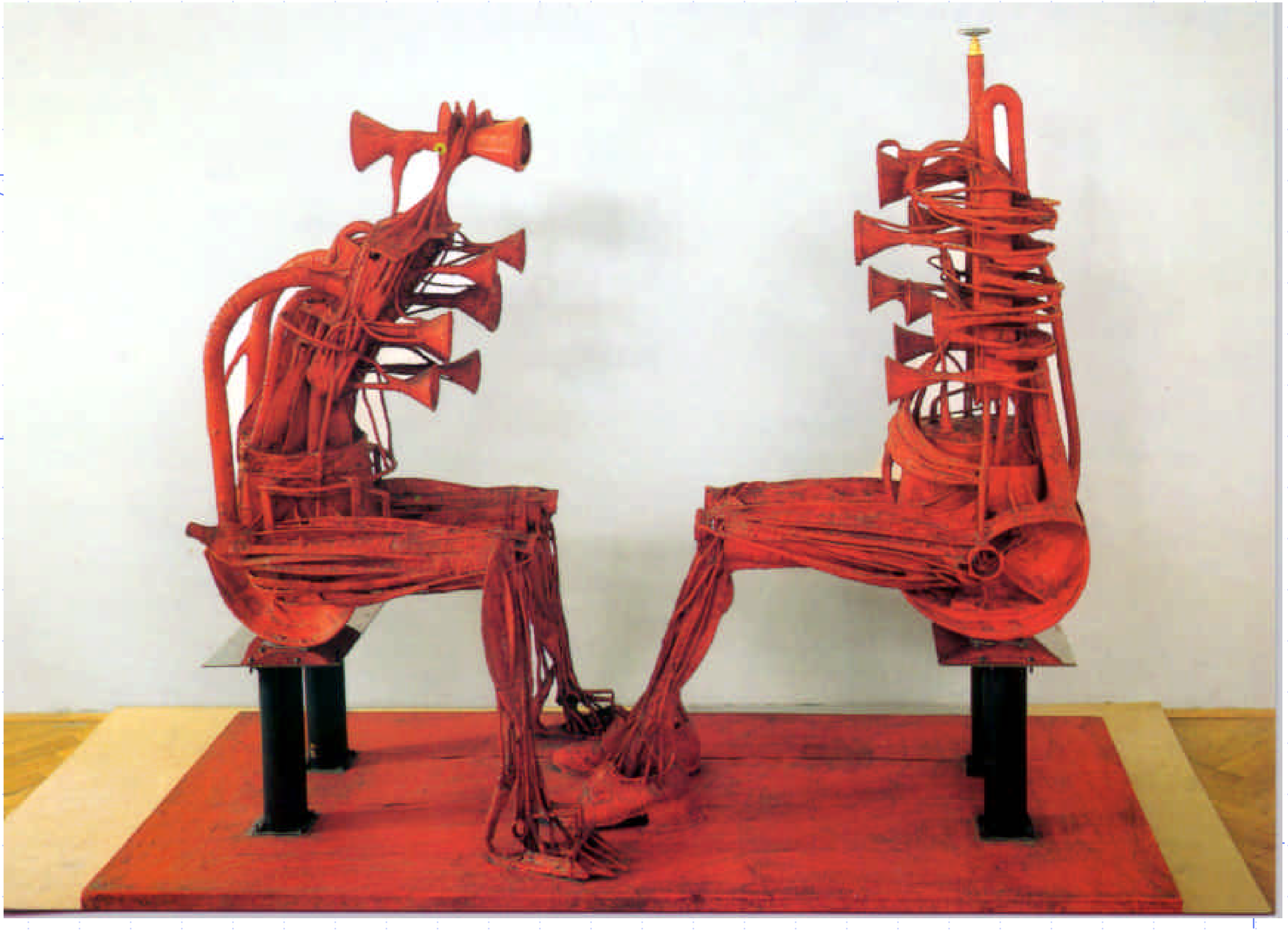
Generic Object Recognition and the Need for image Abstraction

Sven J. Dickinson
Department of Computer Science
University of Toronto

Image Processing Workshop, 2004

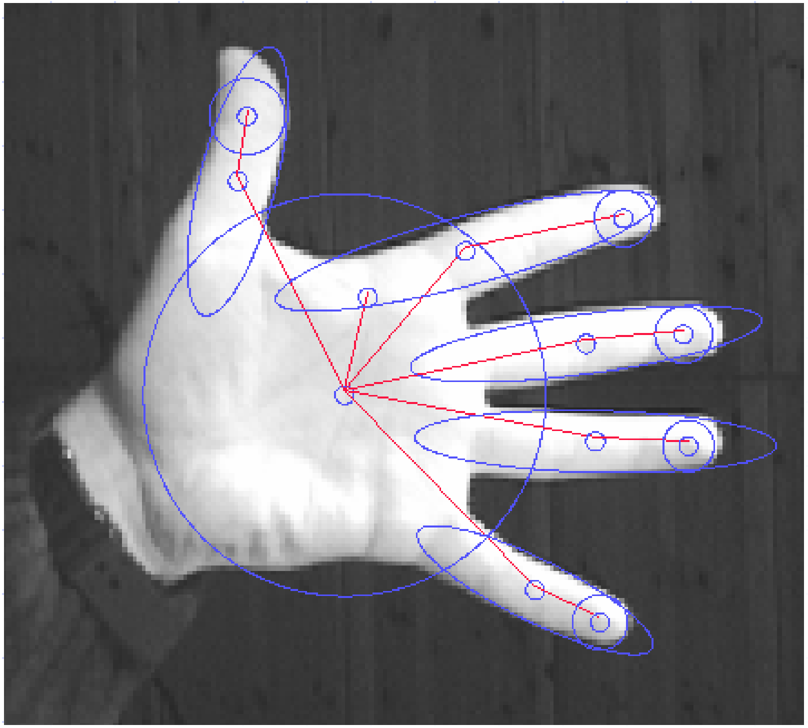


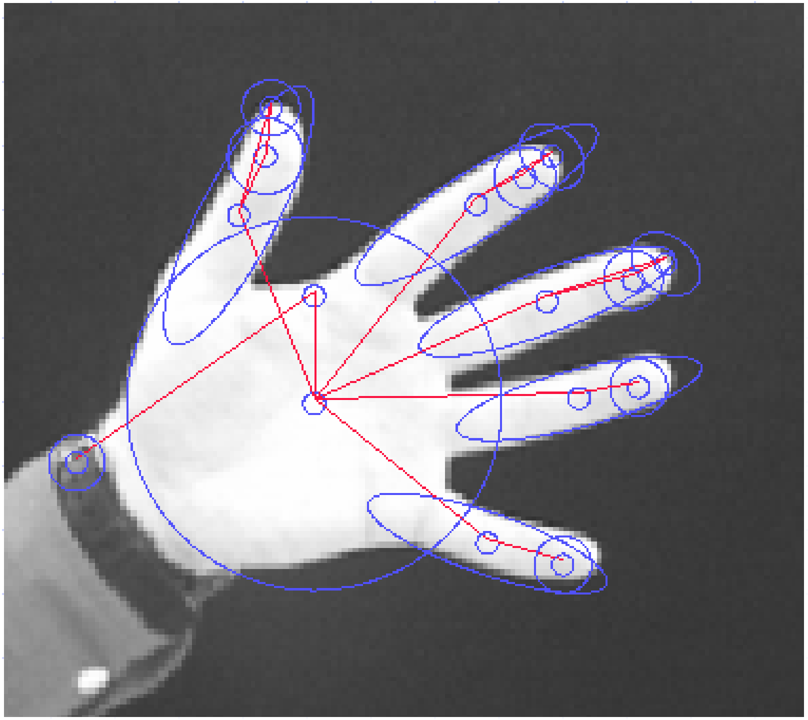
Great Dialog, Karel Nepras, 1966 (Prague National Gallery)

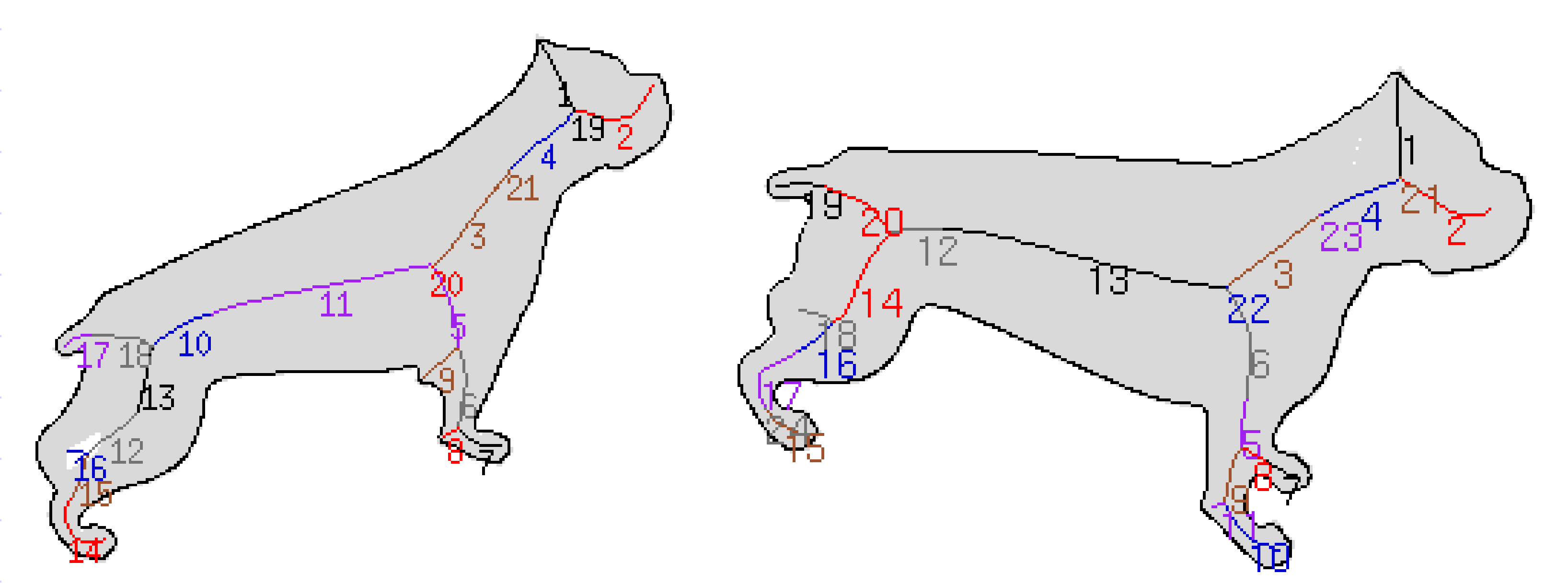
Motivation

- Most approaches to object recognition assume a **one-to-one** correspondence between image fatures and model features.
- This restriction pushes object recognition toward **exemplar-based** recognition.
- But different exemplars belonging to the same **category** may not share a single low-level feature (e.g., interest point, contour, region, etc.).
- Only at higher-levels of **abstraction** does within-class one-to-one feature correspondence occur.

Illustrations






1
19
2
4
21
3
20
11
5
10
17
18
13
9
6
12
16
15
8
7
14
1
19
20
12
13
21
4
23
2
3
14
22
18
16
6
17
15
5
8
9
11
10

Goals:

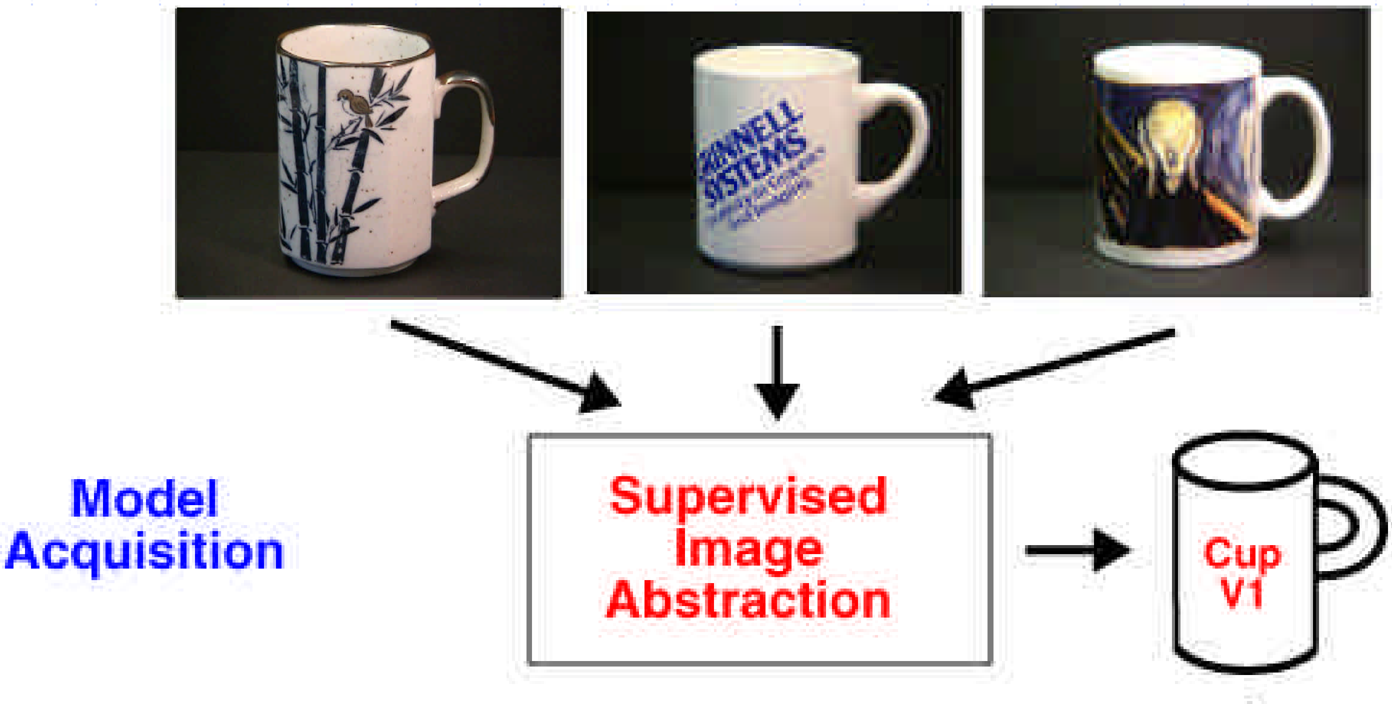
- Cognitive vision systems must be capable of **categorization.**
- **One-to-one** generic object recognition frameworks will require that low-level features be "lifted" to more abstract features.
- **Many-to-many** frameworks are more powerful and less restrictive than their one-to-one counterparts.
- Either way, **abstraction** mechanisms cannot be avoided.

In This Talk

Three frameworks for many-to-many matching:

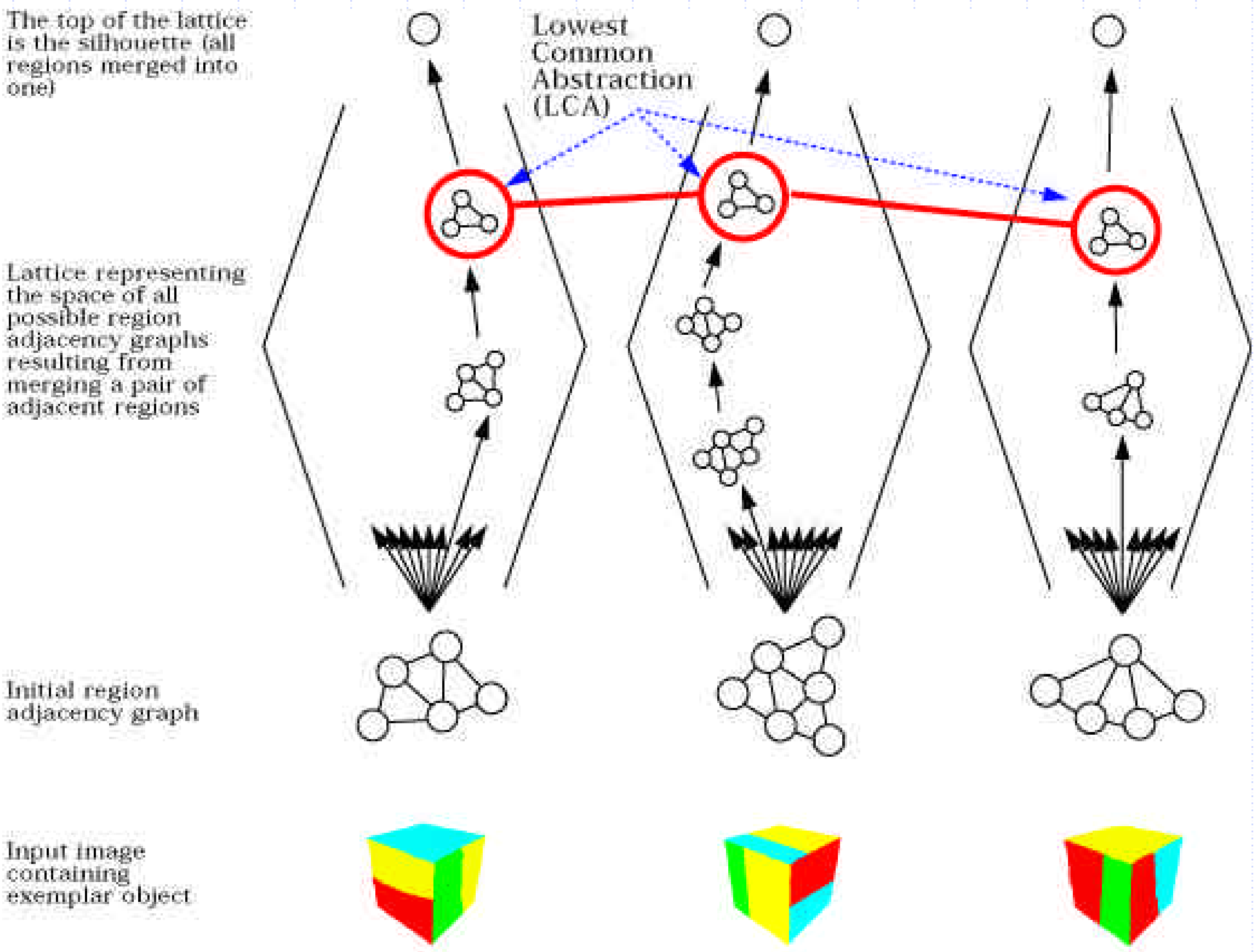
1. Model-based abstraction from examples.
2. Spectral abstractions of graph structure for matching hierarchical structures.
3. Embedding graphs to geometric spaces, where many-to-many matching is easier.

Abstraction Problem Definition



Keselman and Dickinson (CVPR 2001)

Problem Formulation



Solution Overview

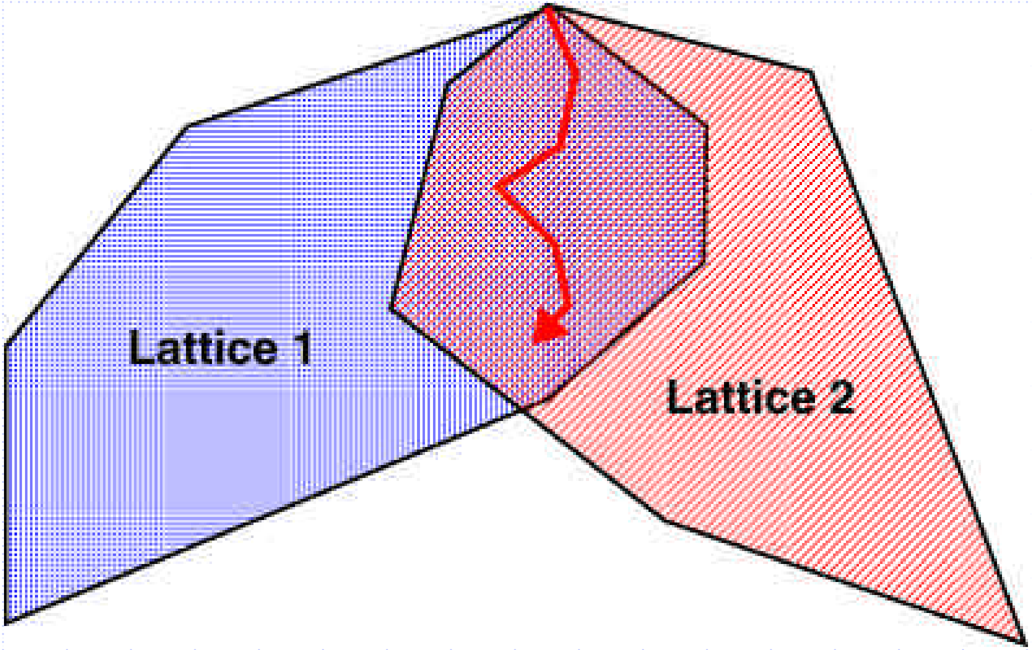
1. Restrict our problem to finding the LCA of two lattices (examples).

2. Compute the LCA of all $O(n^2)$ pairs of lattices to form an approximation to the true intersection lattice. We call this the closure graph.

3. Compute the median of the closure graph to yield the global LCA.

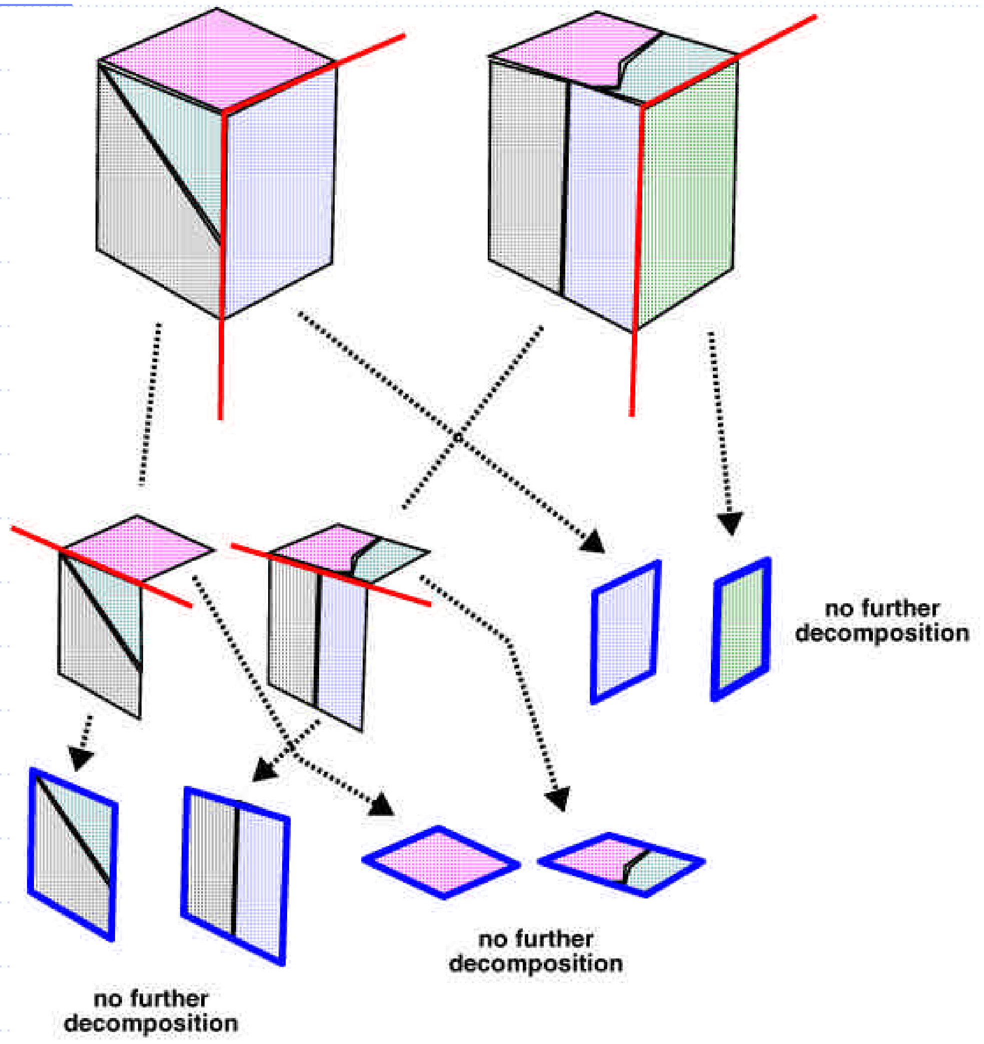
The LCA of Two Lattices

Computing the intersection of two lattices is still intractable.



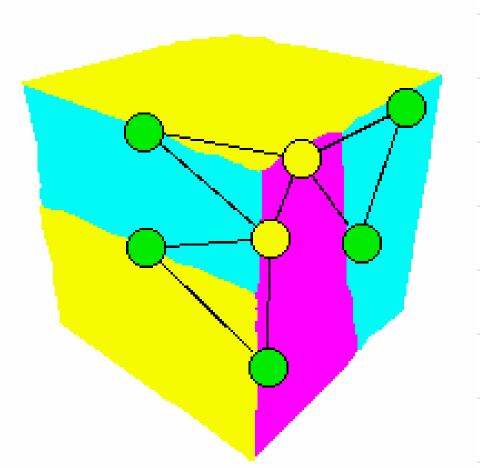
However, we do know one element of the intersection set!

Recursive Decomposition



The Search for Corresponding Cuts

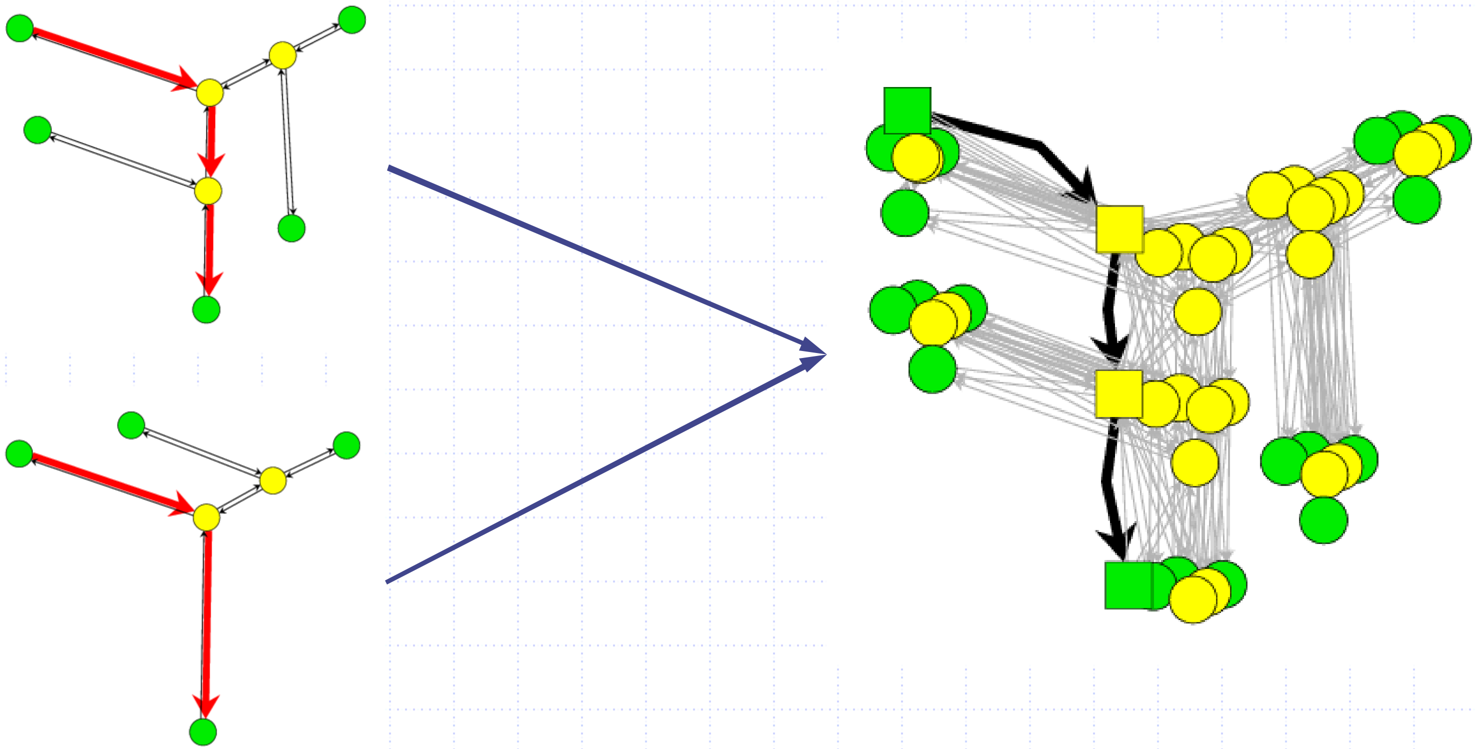
First, form a dual graph representation:



We call this the boundary segment graph.

A cut in the original region adjacency graph is a path through its dual boundary segment graph.

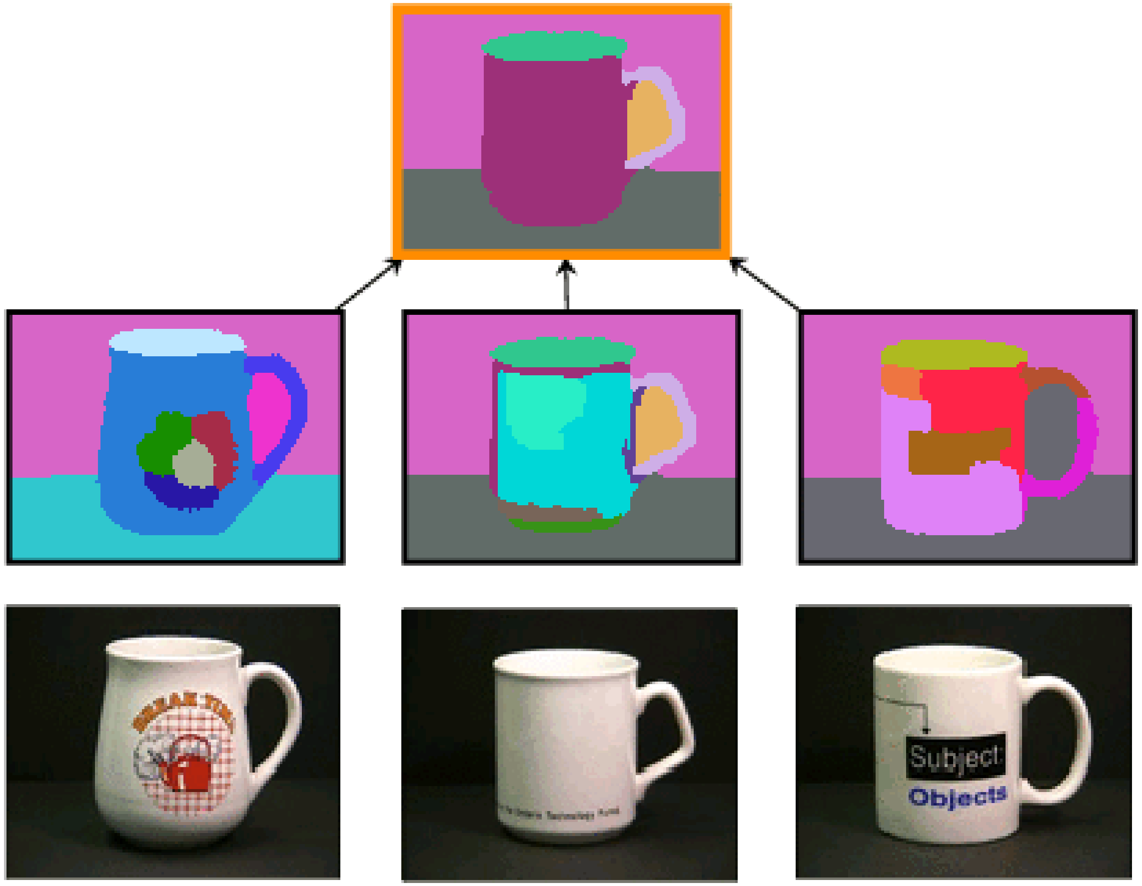
Combining the Dual Graphs

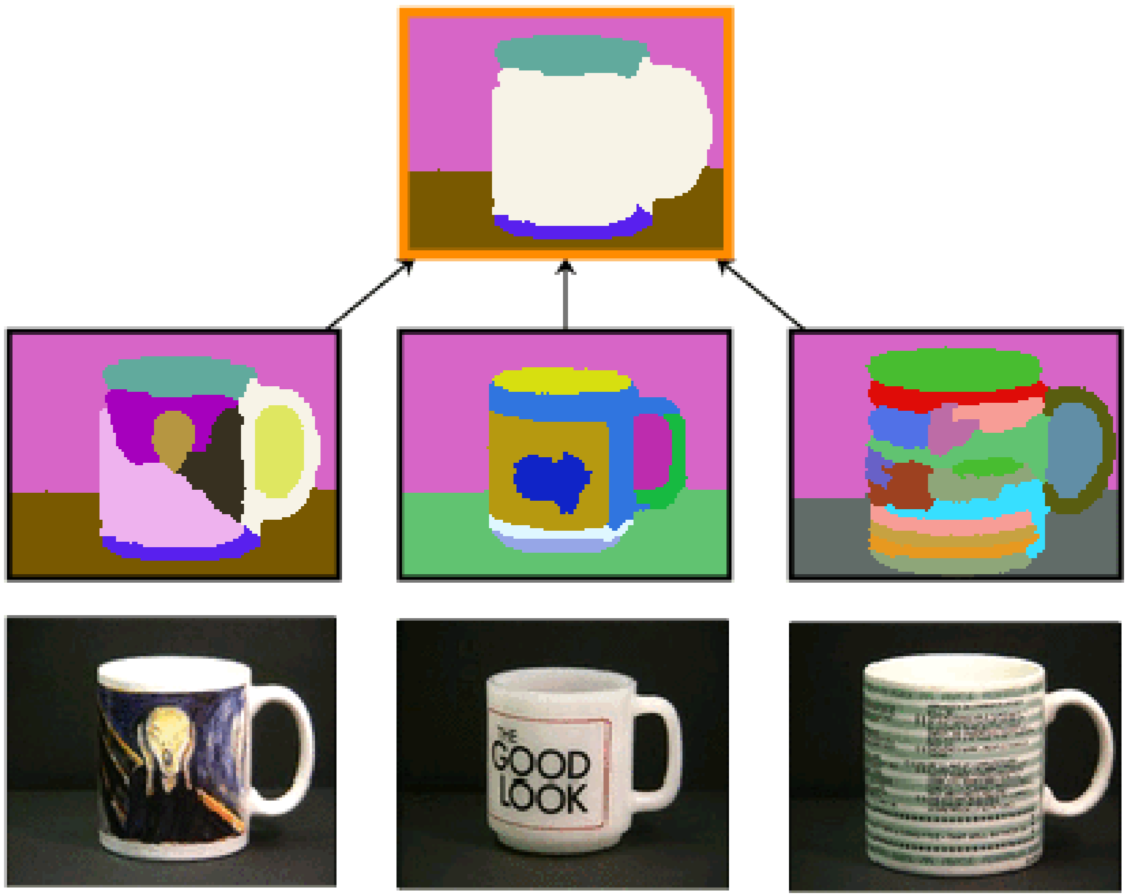


Boundary segment graphs

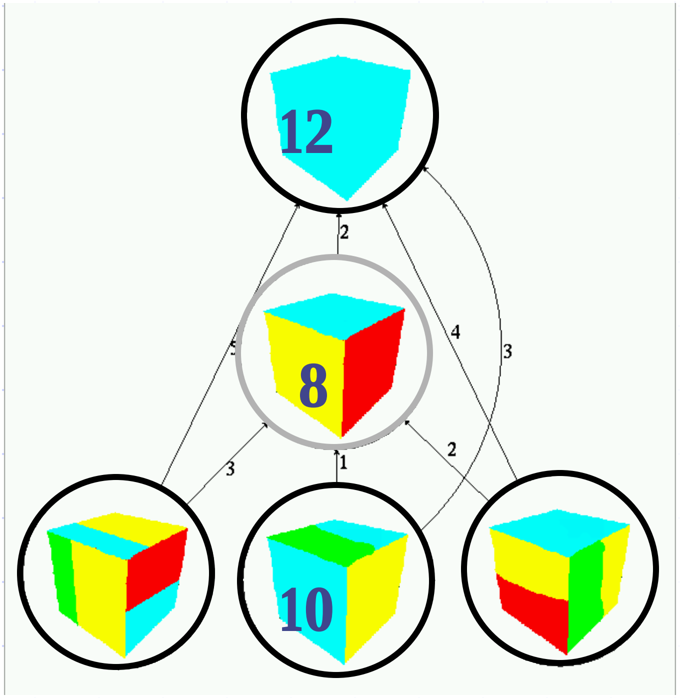
Association graph

Demonstration



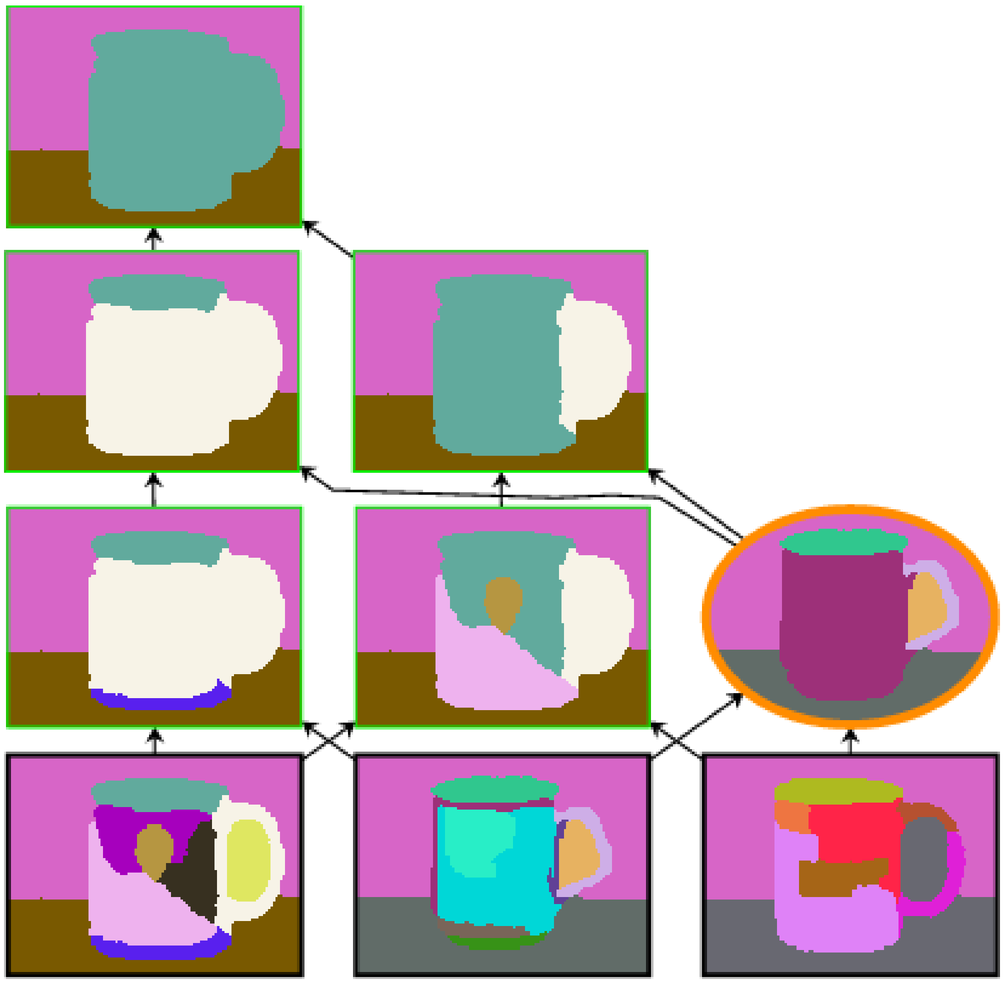

THE GOOD LOOK

The LCA of Multiple Exemplars



1. Compute closure under the pair-wise LCA operation, removing duplicates.
2. Compute edge weights as graph edit (region merge) distances.
3. Robust LCA of all inputs is node which minimizes the sum of shortest paths from initial region adjacency graphs.

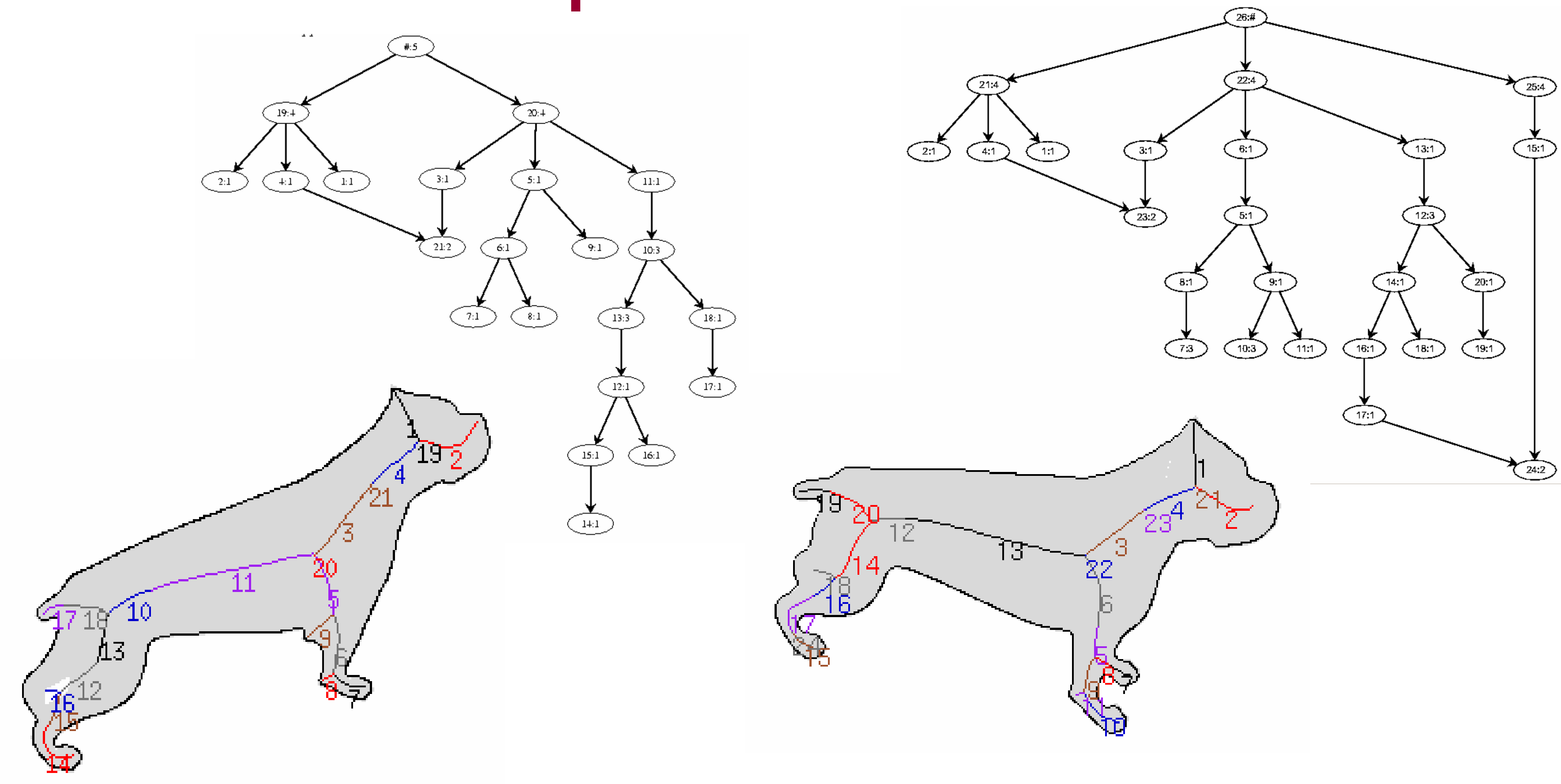
Demonstration



Matching Spectral Abstractions of Graph Structure

- Image features and their relations can be conveniently represented by labeled graphs.
- When features are multi-scale, or when part/whole relations exist between features, resulting graphs can be represented as directed acyclic graphs.
- Object recognition can therefore be formulated as hierarchical graph matching.
- Using spectral graph theory, we embed discrete graphs into low-dimensional continuous spaces.

Shock Graphs



Siddiqi, Shokoufandeh, Dickinson, Zucker (ICCV ‘98, IJCV ‘99)

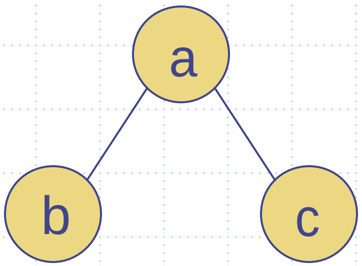
The Eigenspace of a Graph

The eigenvalues of a graph's adjacency matrix encode the connectivity structure in the graph.

But, are they unique? No, but cospectral graphs are not that common.

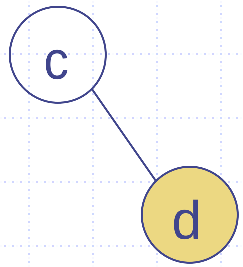
And are they stable to noise and minor structural perturbation? Yes! Let's have a look.

Perturbing a Graph



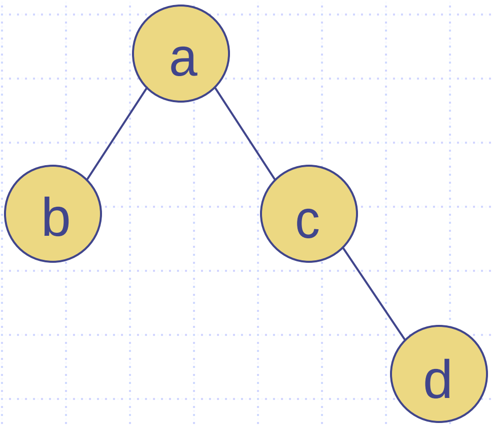
G (original)

+



E (noise)

=



H (perturbed)

	a	b	c	
a	0	1	1	0
b	-1	0	0	0
c	-1	0	0	0
	0	0	0	0

$\Psi(A_G)$

+

	a	b	c	d
a	0	0	0	0
b	0	0	0	0
c	0	0	0	1
d	0	0	-1	0

A_E

=

	a	b	c	
a	0	1	1	0
b	-1	0	0	0
c	-1	0	0	1
	0	0	-1	0

A_H

Establishing Stability

Theorem (Wilkinson, 1965):

If A and $A + E$ are n £ n symmetric matrices, then for all k 2 {1,···,n}, λ_1 ¸ λ_2 ¸ ··· λ_n:

$$?_k(A) + ?_k(E) \leq ?_k(A+E) \leq ?_k(A) + ?_1(E).$$

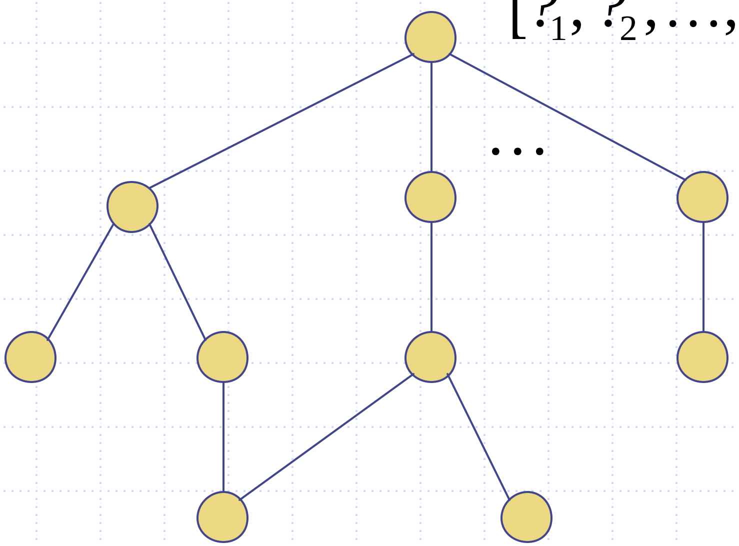
For H (perturbed graph) and G (original graph), the above theorem yields (after manipulation):

$$\left| ?_k(A_H) - ?_k(\Psi(A_G)) \right| \leq \left| ?_1(A_E) \right|$$

The eigenvalues of a graph are therefore stable under minor perturbations in graph structure.

The Eigenvalues are Stable Now What?

$[?_1, ?_2, \ldots, ?_n]$

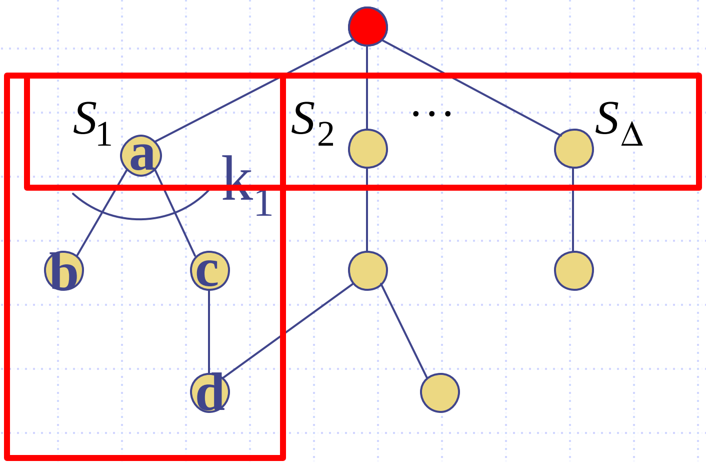


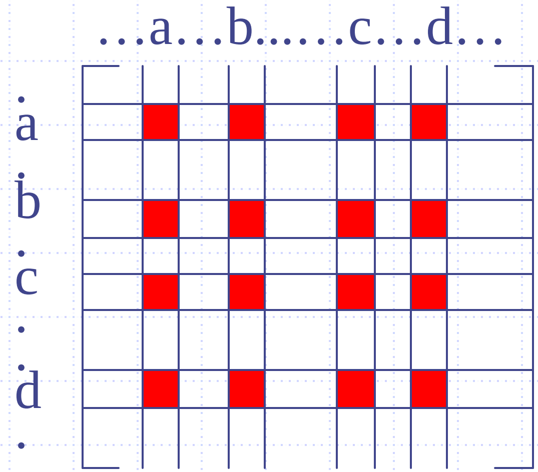
We *could* compute the graph's eigenvalues, sort them, and let them become the components of a vector assigned to the graph.

But:

1. Dimensionality grows with size of graph.
2. Eigenvalues are global! Therefore, can't accommodate occlusion or clutter.

Forming a Structural Signature



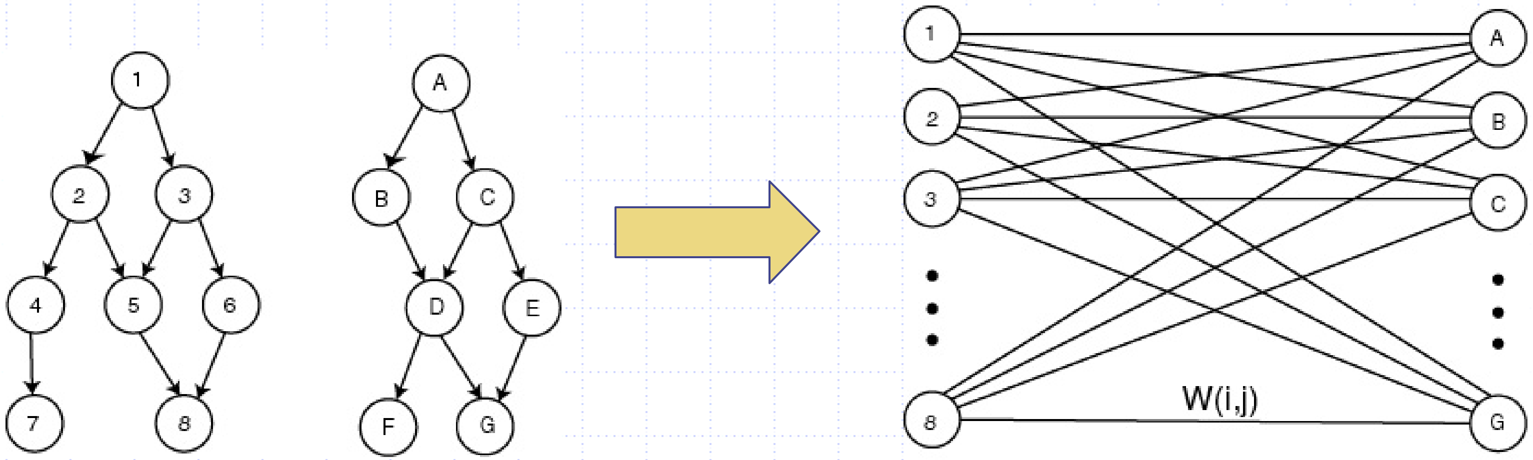


$$V = [S_1, S_2, S_3, \ldots, S_\Delta], \quad S_1 \geq S_2 \geq S_3 \geq \ldots S_\Delta \qquad S_i = |?_1| + |?_2| + \ldots |?_{k_i}|$$

Why Sum the k largest Eigenvalues?

1. Summing reduces dimensionality.
2. Largest eigenvalues most informative.
3. Sums are "normalized" according to richness (k_i) of branching structure.

Matching using the Spectral Abstraction

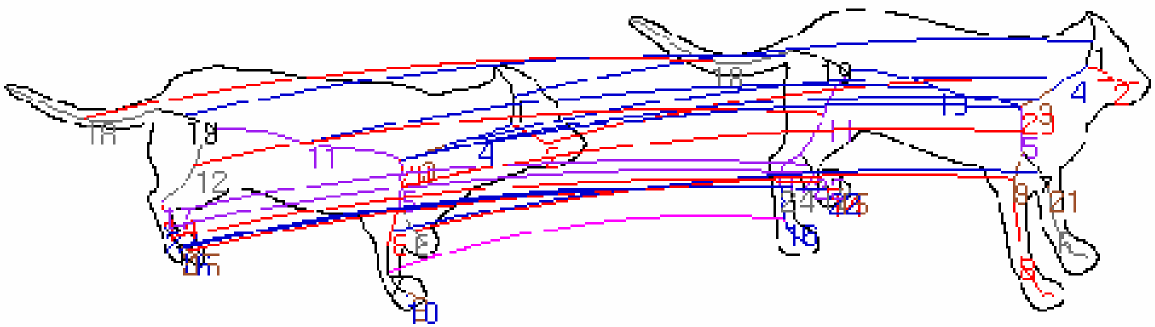


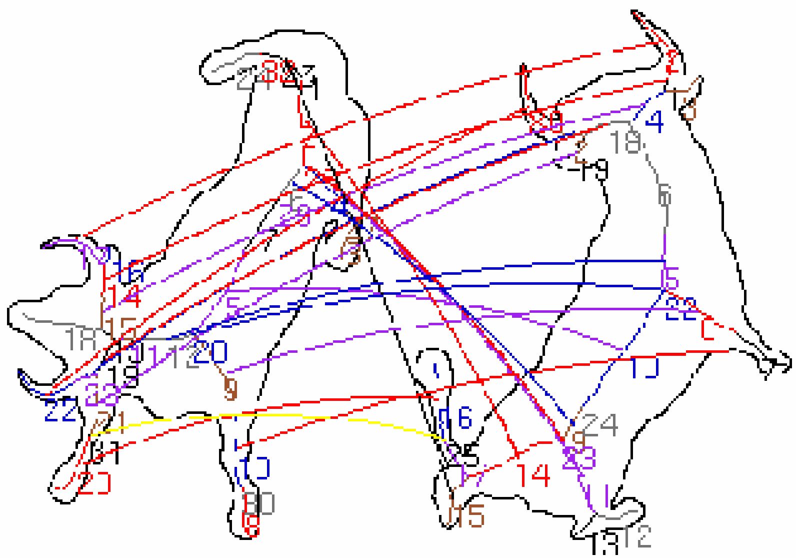
Consider a bipartite graph matching formulation, in which the edges in the query and model graphs are discarded.

Hierarchical structure is seemingly lost, but can be encoded in the edge weights:

$$W(i, j) = e^{-\left(a_1 d_{struct}(i,j) + a_2 d_{geom}(i,j)\right)}$$

Sample Matches



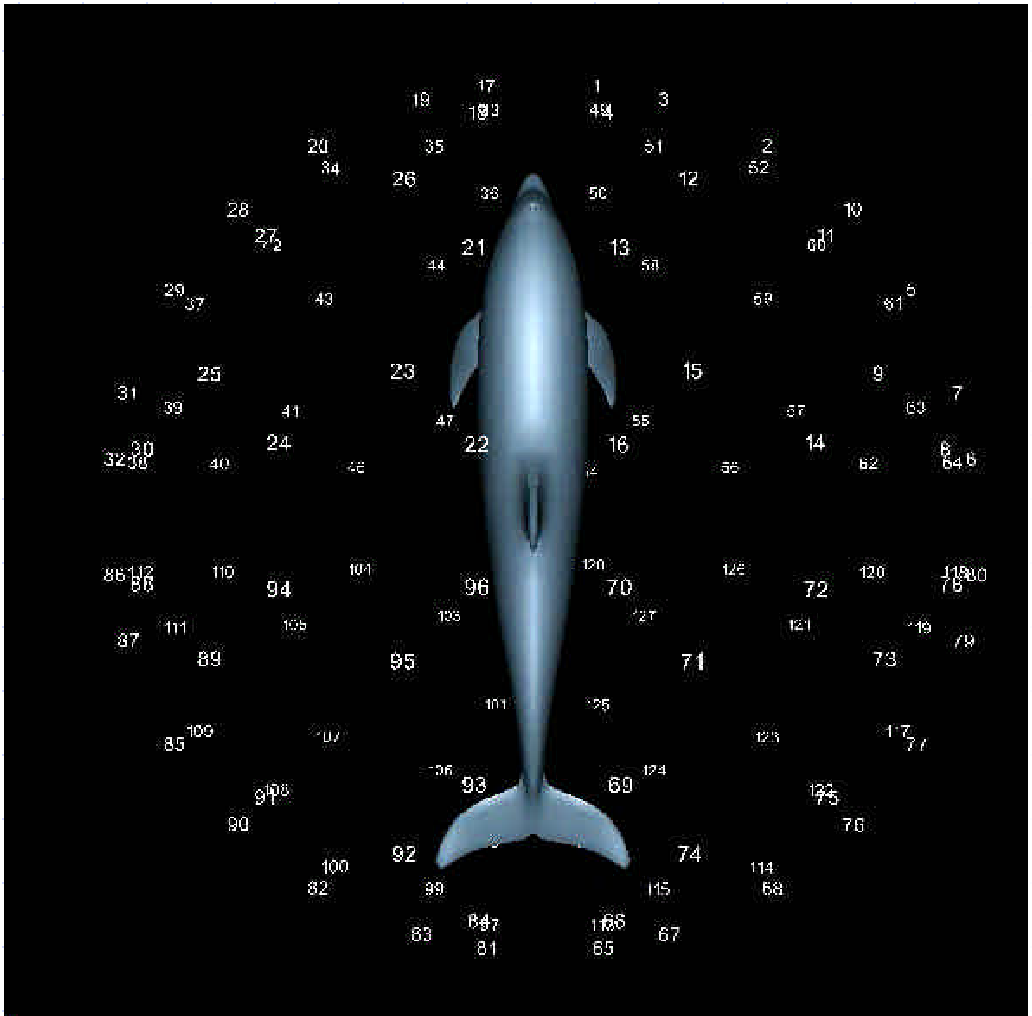


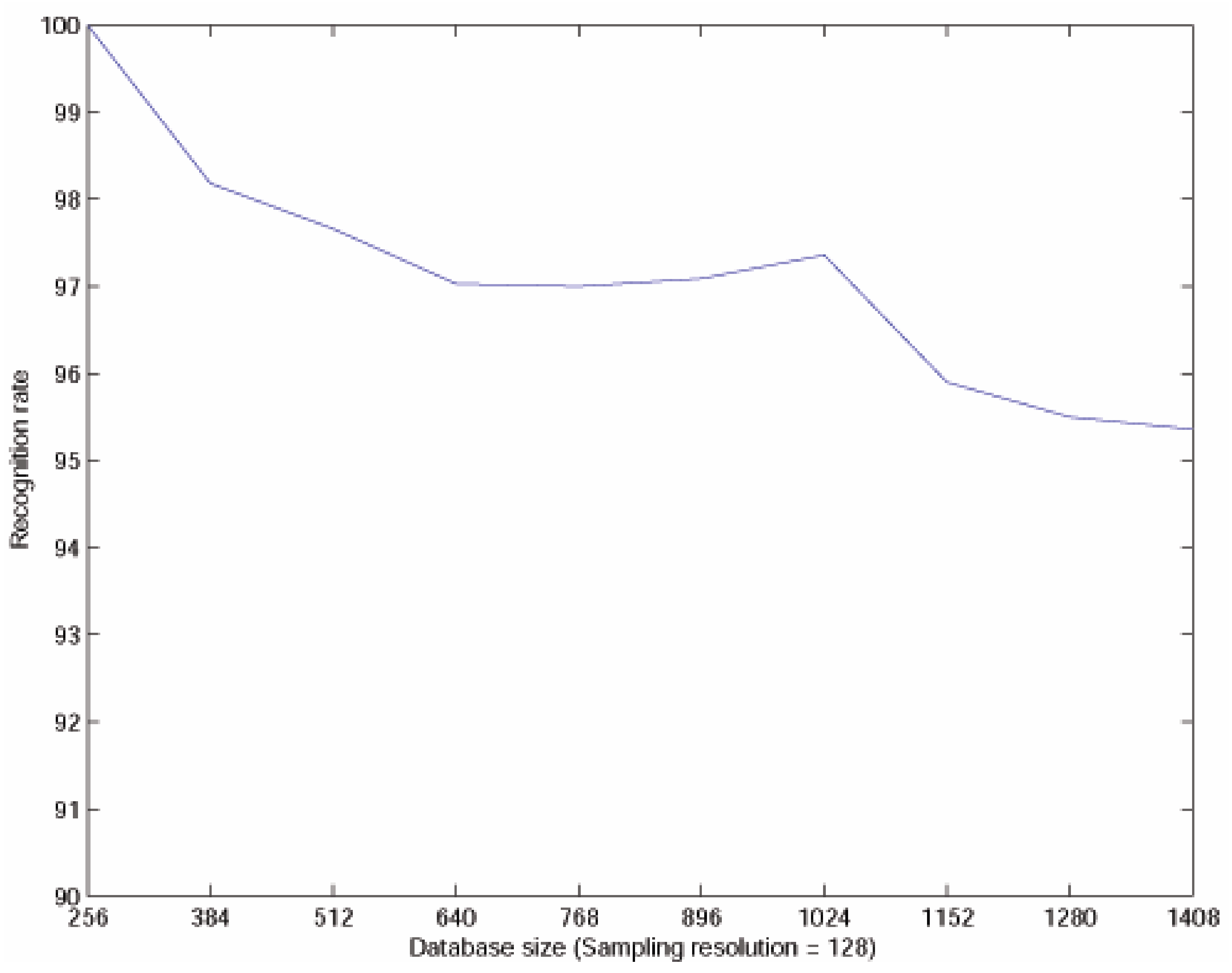
View-Based 3-D Recognition

- Shock graphs are computed from a regularly tessellated viewing sphere centered around a CAD model of a 3-D object.
- The TSV of every node of every shock graph is stored in a database supporting nearest-neighbour search.
- A query shock graph returns the 50 model shock graphs receiving the most votes.
- These 50 candidates are verified using our shock graph matching algorithm.
- We conducted over 25,000 trials, varying number of objects, sampling resolution, and degree of occlusion.

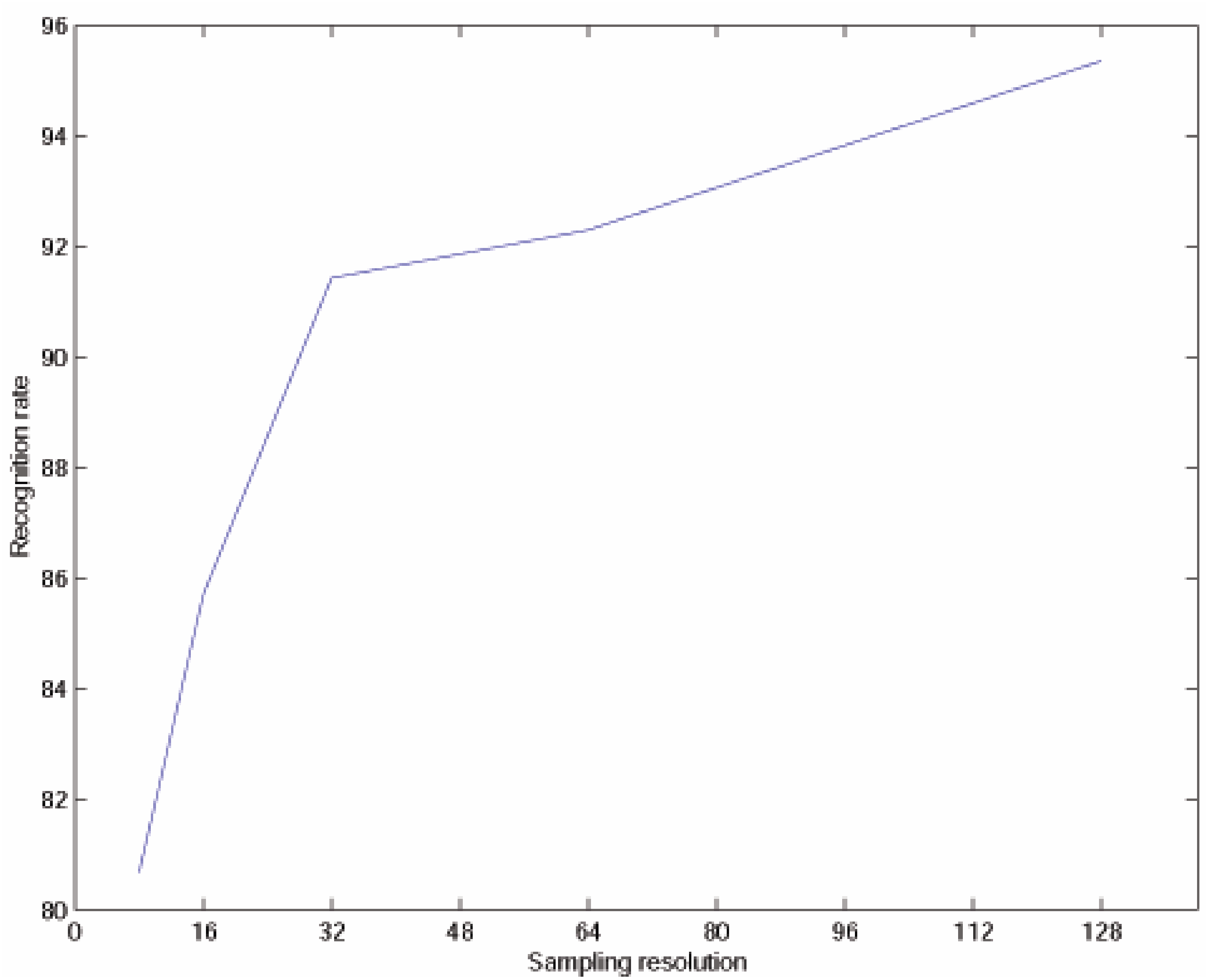
Macrini, Shokoufandeh, Dickinson, Siddiqi, Zucker (ICPR ‘02)

Sampling the Viewing Sphere

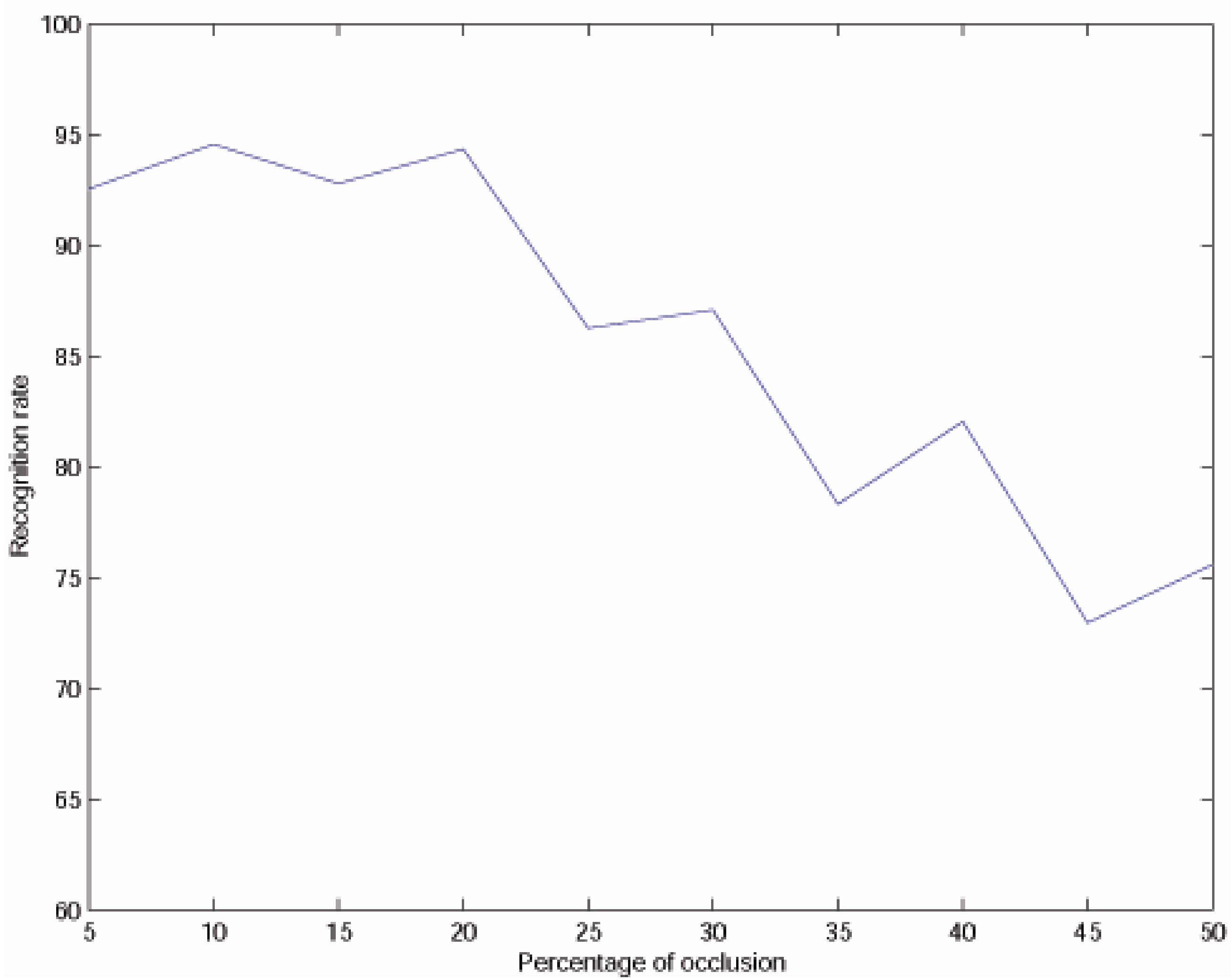




Recognition performance as a function of increasing number of objects (with 128 views per new object).

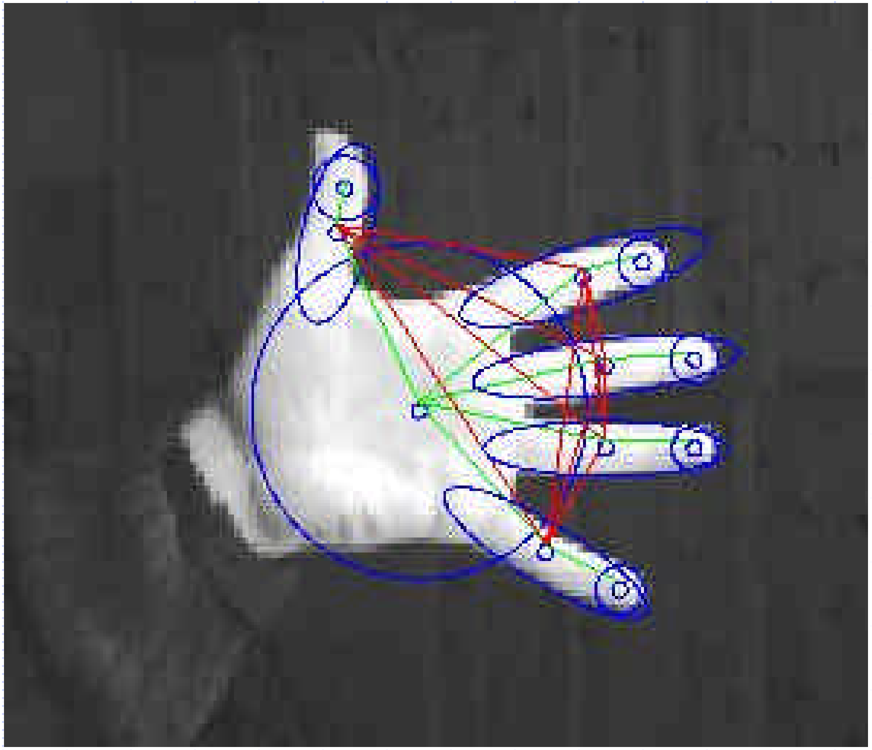


Recognition performance as a function of sampling resolution (for a fixed number of objects (11)).



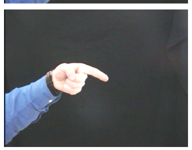
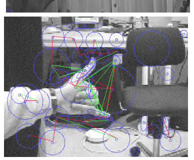
Recognition performance as a function of degree of occlusion (missing data) for occluded queries.

Matching Blob and Ridge Abstractions

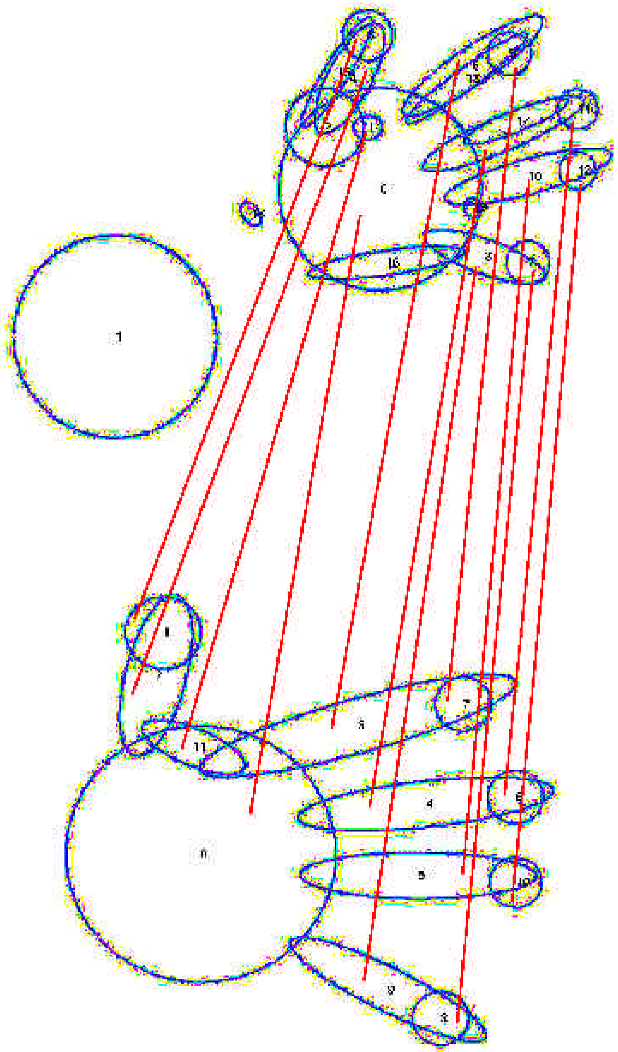


Shokoufandeh, Dickinson, Jönsson, Bretzner, Lindeberg (ECCV ‘02)

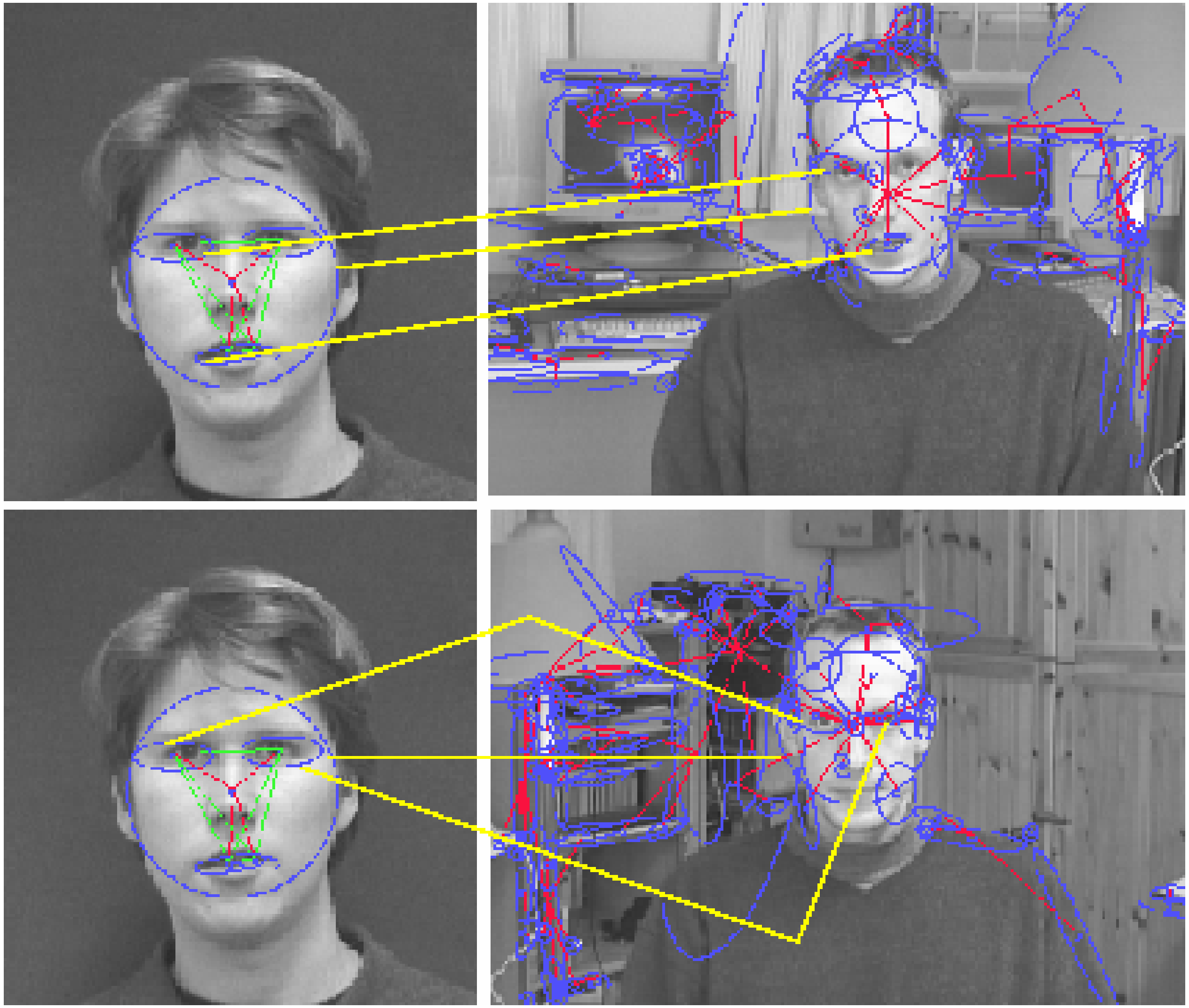
Example: Gesture Recognition

							
	**12.35**	5.73	6.02	7.35	10.25	11.15	9.12
	4.05	**6.88**	5.22	3.27	2.84	4.18	5.95
	5.37	3.13	**8.40**	4.47	7.56	4.21	2.72
	15.18	9.02	5.44	13.19	10.18	**15.95**	13.22
	**21.84**	11.01	12.17	15.88	9.21	17.75	16.37
	**10.43**	3.41	4.19	4.00	7.26	5.69	4.96

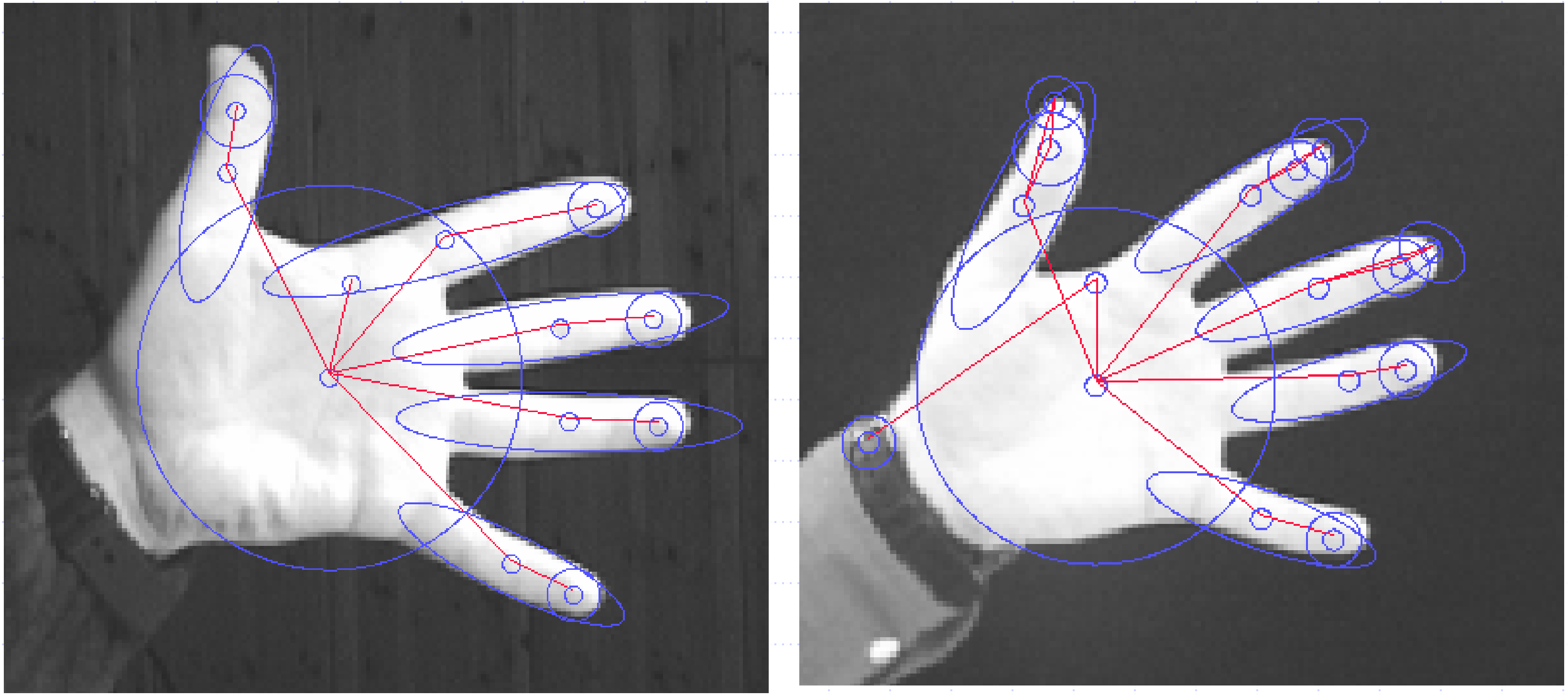
Blob Correspondence



Example: Face Detection



Many-to-Many Matching using Graph Embedding

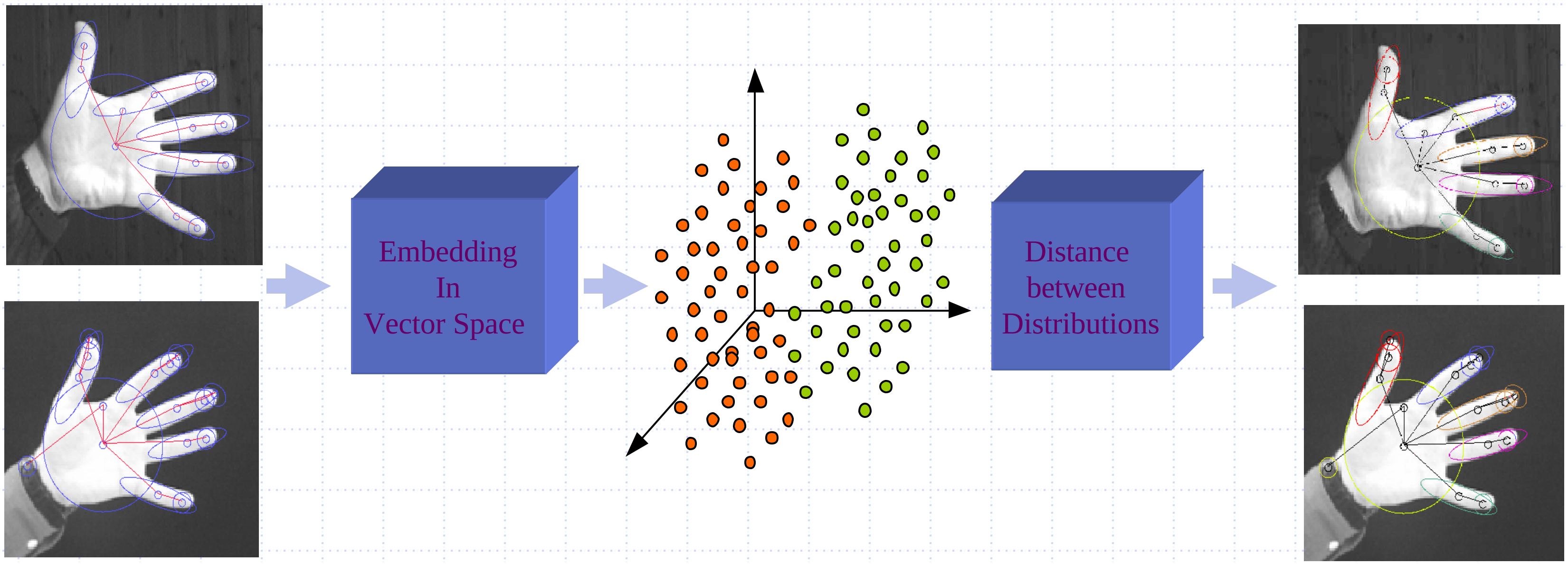


Demirci, Shokoufandeh, Keselman, Dickinson, Bretzner (Scale-Space '03)

Keselman, Shokoufandeh, Demirci, Dickinson (CVPR ‘03)

Demirci, Shokoufandeh, Dickinson, Keselman, Bretzner (ECCV '04)

Matching Graphs as Weighted Point Sets in Euclidean Spaces



Embedding Graphs into Euclidean Spaces

- To embed a graph into a Euclidean space, we view the graph as a metric space (distances on pairs of nodes are defined by lengths of shortest paths) and use existing algorithms for embedding metric spaces into Euclidean spaces.
- The embedding will be **low-distortion** (distances between pairs of graph nodes will change little).

Embedding Graphs into Euclidean Spaces

- To embed a graph into a Euclidean space, we view the graph as a metric space (distances on pairs of nodes are defined by lengths of shortest paths) and use existing algorithms for embedding metric spaces into Euclidean spaces.
- The embedding will be **low-distortion** (distances between pairs of graph nodes will change little).
- Terminology:
  - Metric space: arbitrary set of points with a distance function d satisfying: $d(x,x)=0$; $d(x,y)>0$, if $x \neq y$; $d(x,y)=d(y,x)$ [symmetry] $d(x,z) \leq d(x,y)+d(x,z)$ [the triangle inequality].
  - A Euclidean space is also a metric space: $d(x,y) = ||x-y||$.

Low Distortion Embedding of Graphs into Euclidean Spaces

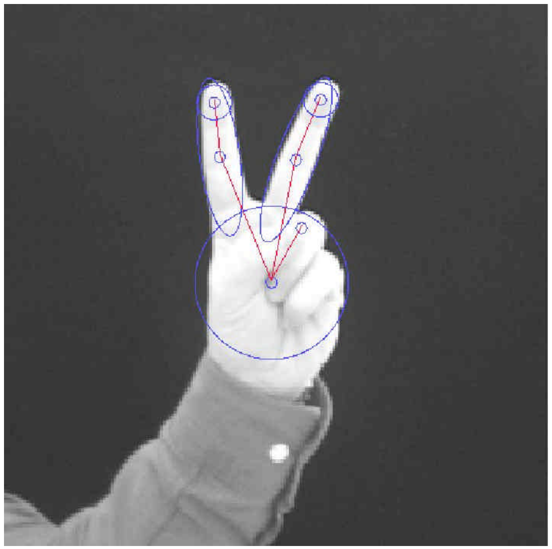
To embed a metric space into a Euclidean space with low distortion, we use existing algorithms:

- Embed the metric space into a tree metric with low distortion (Agarwala et al., 1999).
- Embed tree metric into a Euclidean space with low distortion (Matousek, 1999).

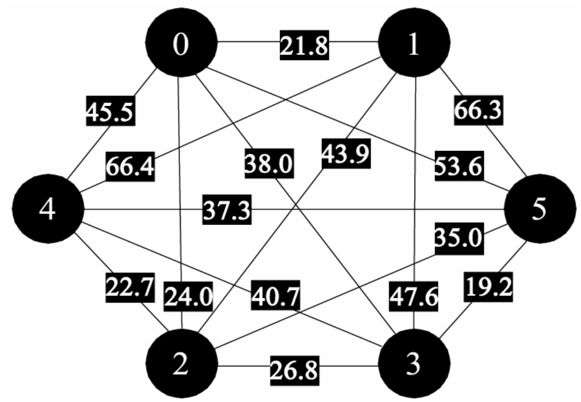
Low Distortion Embedding of Graphs into Euclidean Spaces

- To embed a metric space into a Euclidean space with low distortion, we use existing algorithms:
  - Embed the metric space into a tree metric with low distortion (Agarwala et al., 1999).
  - Embed tree metric into a Euclidean space with low distortion (Matousek, 1999).
- Terminology:
  - Tree metric space: metric that can be realized by a tree (points of the original metric space can be mapped into nodes of a tree so that original distances are shortest path distances on the tree).
  - Not every metric can be realized by a tree (e.g., consider 3 points, with distance of 1 between each pair).
  - Not every metric tree can be embedded without distortion into a Euclidean space (e.g., consider a rooted tree with 3 leaves, with distance of 1 between the root and the leaves).

Step 1: Construct a Metric Tree

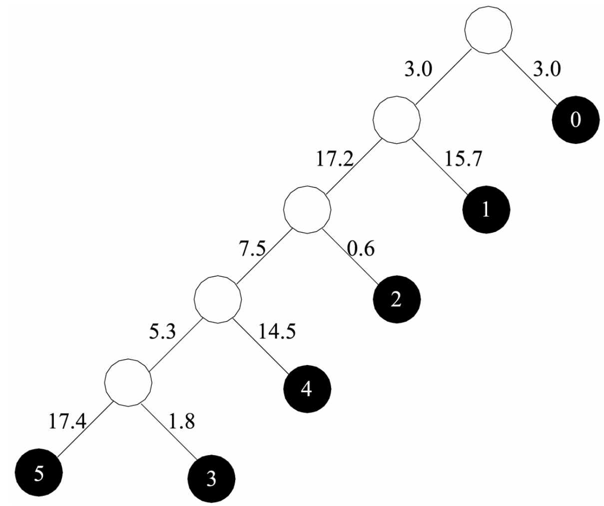


original graph



complete graph with edge weights equal to Euclidean distance between region centroids

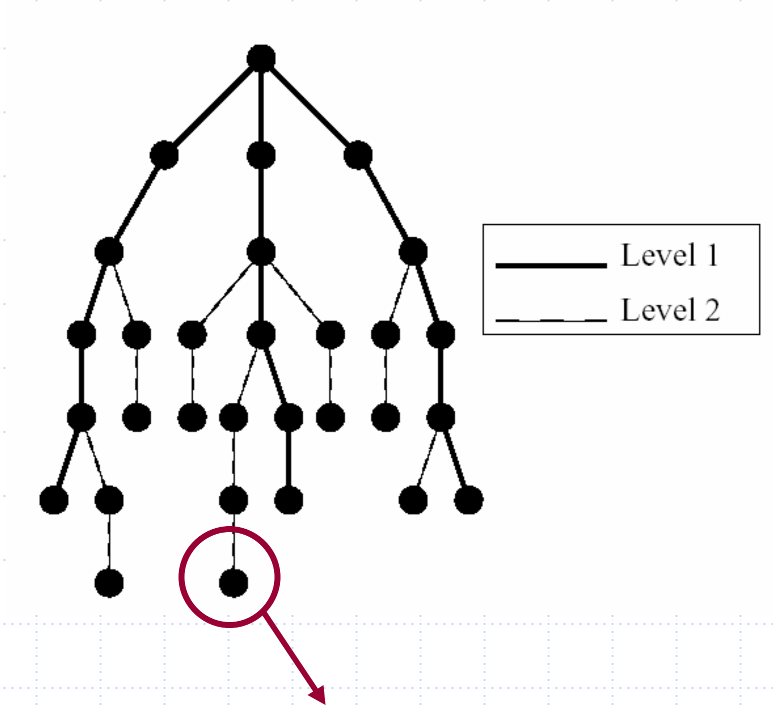
note that any distance function can be chosen



resulting low-distortion tree metric with additional vertices (Agarwala et al., 1999).

For example, D(3,5) above equals weight of edge (3,5) in graph to left. But, D(0,1) above (21.7) is slightly less than edge (0,1) (21.8) at left. Distances are preserved with low distortion.

Construction of Matousek's Embedding (Matousek, 1999).



Example: will traverse 3 edges of the 2nd level-1 path, and 3 edges of the 4th level-2 path $[0, w_2, 0, 0, 0, 0, w_7, 0, 0, 0]$

- these 10 (3+7) paths represent the components of a 10-dimensional vector, our embedding: the 3 level-1 paths are the first 3 coordinates, and the 7 level-2 paths are the remaining 7 coordinates.
- each vertex is characterized by its path from the root
- non-zero components of the vector correspond to the length of traversed path segments
- coordinates are chosen with distortion at most $\sqrt{\log \log |V'|}$
- constructed in linear time

Encoding Scale-Space Features

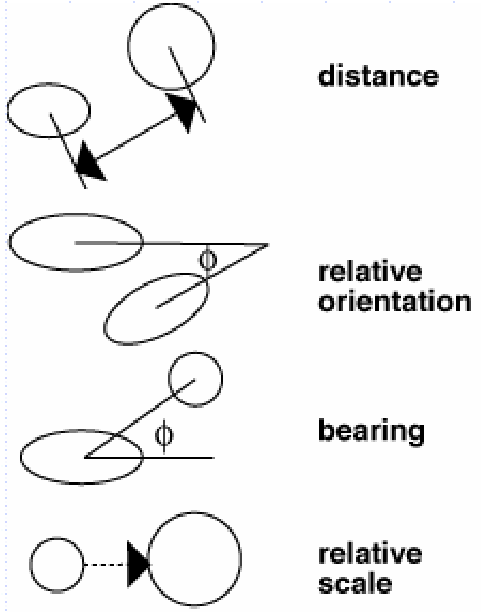
Problem: undirected graph embedding has failed to account for the hierarchical, non-metric relations common to scale-space structures.

Solution: move the non-metric relational information into the nodes by computing, as node attributes, two sets of distributions on the incoming and outgoing relations.

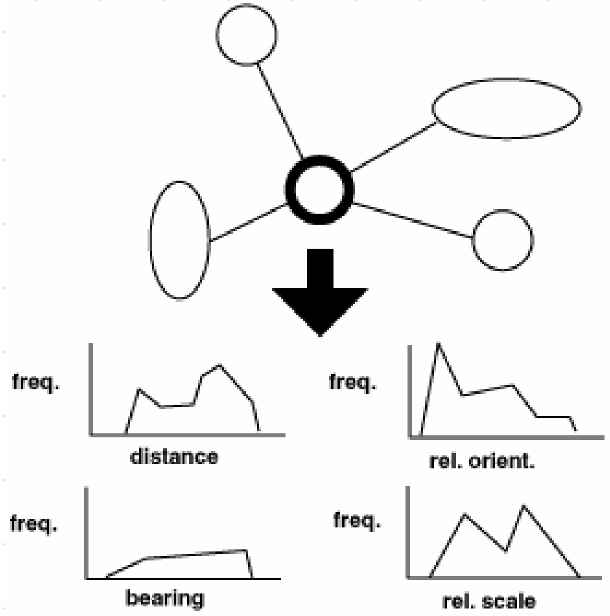
(Shokoufandeh, Dickinson, Jönsson, Bretzner, Lindeberg (ECCV '02)

Scale-Space "Context" as a Node Attribute

Example Relations



Relational Context



1. Two relational contexts of a node can be computed, based on outgoing and incoming edges, respectively.
2. A node's relations to its parents, its children, and its siblings all help to define its location in the scale space.

Aligning the Distributions (Embeddings)

Problem:

Two point distributions must be mapped into the same Euclidean space before being matched (registration step).

Solution:

- Project the two distributions onto the first K (minimum dimension of two distributions) right singular vectors of their covariance matrices.
- This equalizes their dimensions while losing minimal information.
- This is a global transformation and serves only to initialize an iterative procedure that will align the two distributions in the presence of noise and occlusion.

Matching the Distributions using the Earth Mover's Distance (EMD)

- Evaluates dissimilarity between two multi-dimensional distributions in some feature space by lifting geometric distance from individual features to full distributions (Cohen and Guibas, 1999).
- Analogous to moving dirt from piles to holes.
- Extended EMD formulation allows one point set to undergo a transformation.

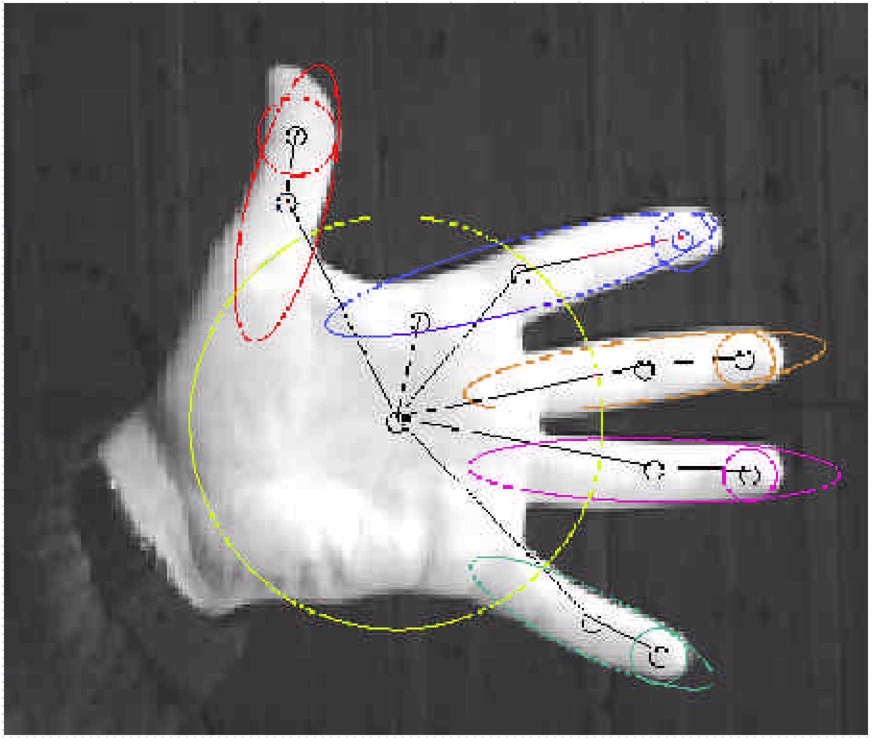
Matching the Distributions using the Earth Mover's Distance (EMD)

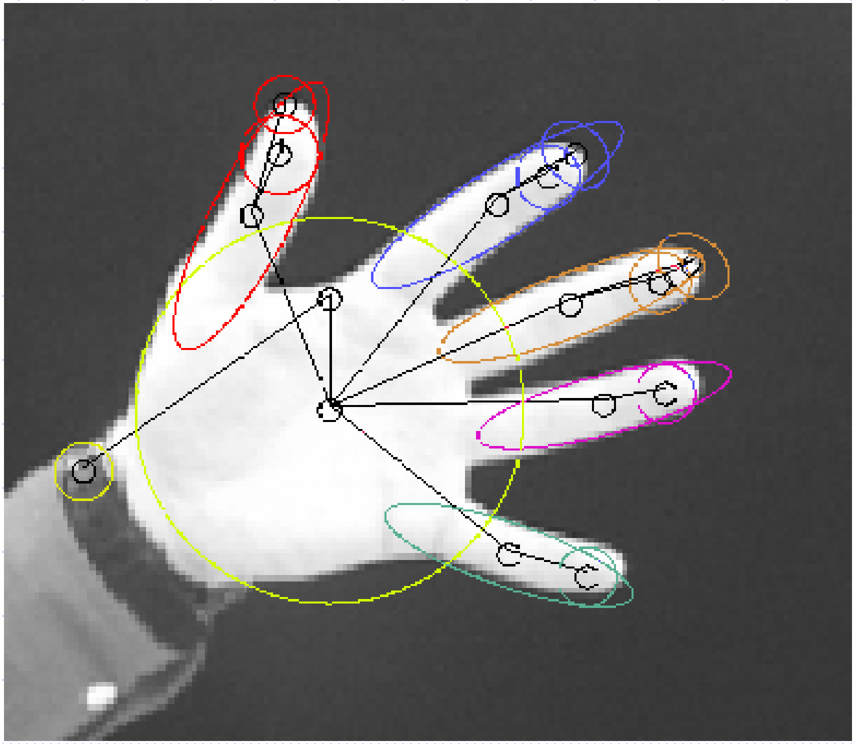
- Evaluates dissimilarity between two multi-dimensional distributions in some feature space by lifting geometric distance from individual features to full distributions (Cohen and Guibas, 1999).
- Analogous to moving dirt from piles to holes.
- Extended EMD formulation allows one point set to undergo a transformation (Cohen and Guibas, 1999).
- An optimal transformation is computed via an iterative EM-like algorithm called FT (**F**low/**T**ransformation).
- The optimal transformation depends on the initial one.

Final Algorithm

- Given two vertex-labeled graphs,
  - find low-distortion embeddings of the graphs into low-dimensional Euclidean spaces.
  - register one distribution with respect to the other (using projection) to minimize the original EMD between them.
  - apply the FT iteration to minimize the extended EMD.
- Points matched when minimizing the EMD yield a weighted, many-to-many matching of nodes.

Demonstration





corresponding feature groups are indicated by the same color

Experiments

COIL-20 (Columbia University Image Library) database consisting of 72 views per object.

Experiments (cont'd)

For each view:

- compute the multi-scale blob decomposition,
- construct a tree metric for the complete edge-weighted blob graph,
- embed each tree into a Euclidean space with low distortion.

Experiments (cont'd)

- For the 72 views of each object, every second view serves as a query view, with remaining 36 views added to the database.
- Compute the distance between each query view and each database view.
- Ideally, for view *i* of object *j*, recognition trial is correct if closest view is $v_{i+1,j}$ or $v_{i-1,j}$.

Results

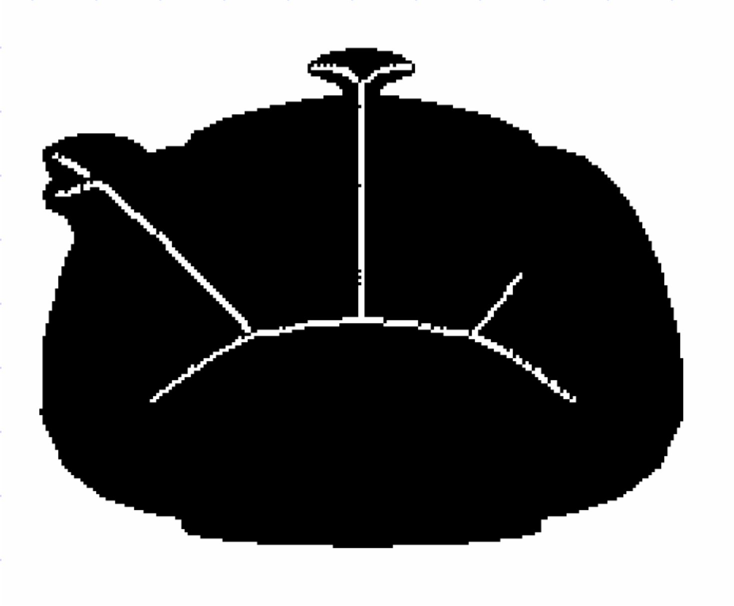
- In all but 10.7% of the experiments, the closest match selected by our algorithm was a neighboring view.
- Among the mismatches, the closest view belonged to the correct object 80% of the time.
- These results ignore the effects of symmetry, and can be considered worst-case.

Non-Hierarchical Graphs

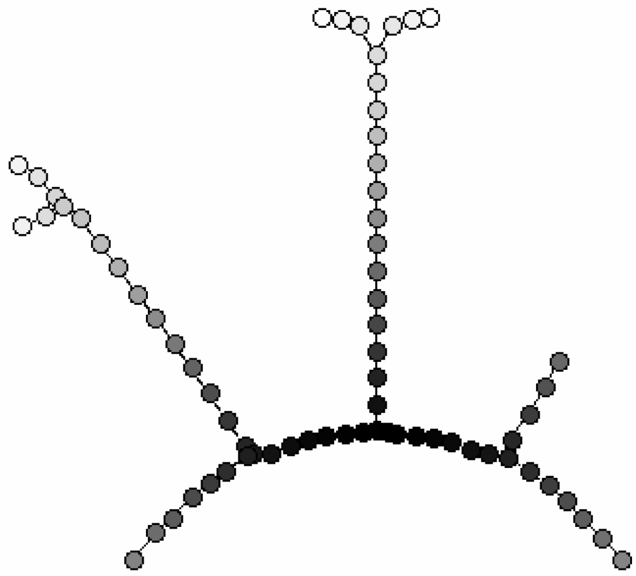
- Non-hierarchical graphs can be easily accommodated within the framework, and differ only in that their relational distributions are not oriented.
- As in the hierarchical case, node feature values map to point attribute vectors in the embedded space.

Experiments

silhouette recognition based on skeleton graph matching



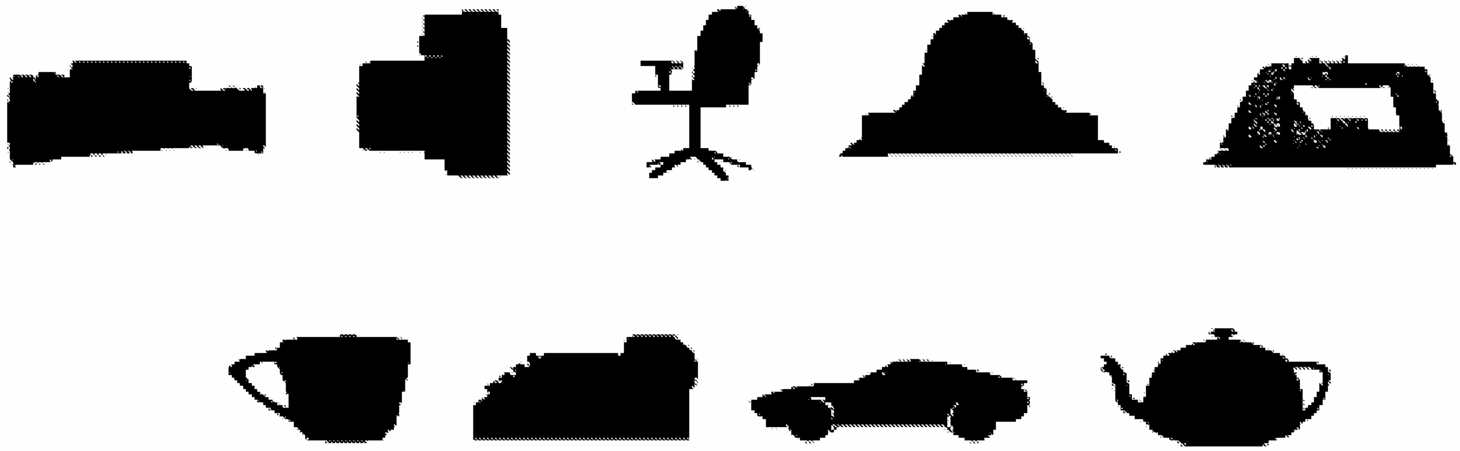
silhouette and its medial axis



medial axis tree - edges encode Euclidean distance between nodes while darker nodes reflect larger radii

Database

9 objects, 180 views per objects, 1620 silhouettes



Results

- 19 equidistant views per object were selected as query views

- for each query, closest view in the database found by linear search.

- In only 5.74% of the trials, the closest view was not a neighboring view.

Example Correspondence

Occlusion Experiments

- Each view, in turn, is used as a query.

- Each query was perturbed by deleting a randomly selected, connected subset of the skeleton points whose size was between 5% and 25% of the total points.

- Average correct recognition performance was 90%.

- These results are conservative and don't account for symmetry.

Conclusions

- The assumption of one-to-one correspondence is a restrictive one that's appropriate for exemplar-based recognition.
- Within-class variation, scale, articulation, and segmentation errors require a more general framework that supports many-to-many feature matching.
- Graphs capture both feature properties and relational information, leading to a many-to-many graph matching formulation.

Conclusions (cont'd)

- Model-based abstraction can be formulated as finding a many-to-many mapping between regions of exemplars belonging to a single class.
- Hierarchical graph matching can be formulated as matching low-dimensional spectral descriptions of entire subgraphs, yielding a many-to-many mapping.
- Many-to-many, directed and undirected graph matching can be reformulated as a geometric point distribution matching problem.