A String Matching Approach for Visual Retrieval and Classification

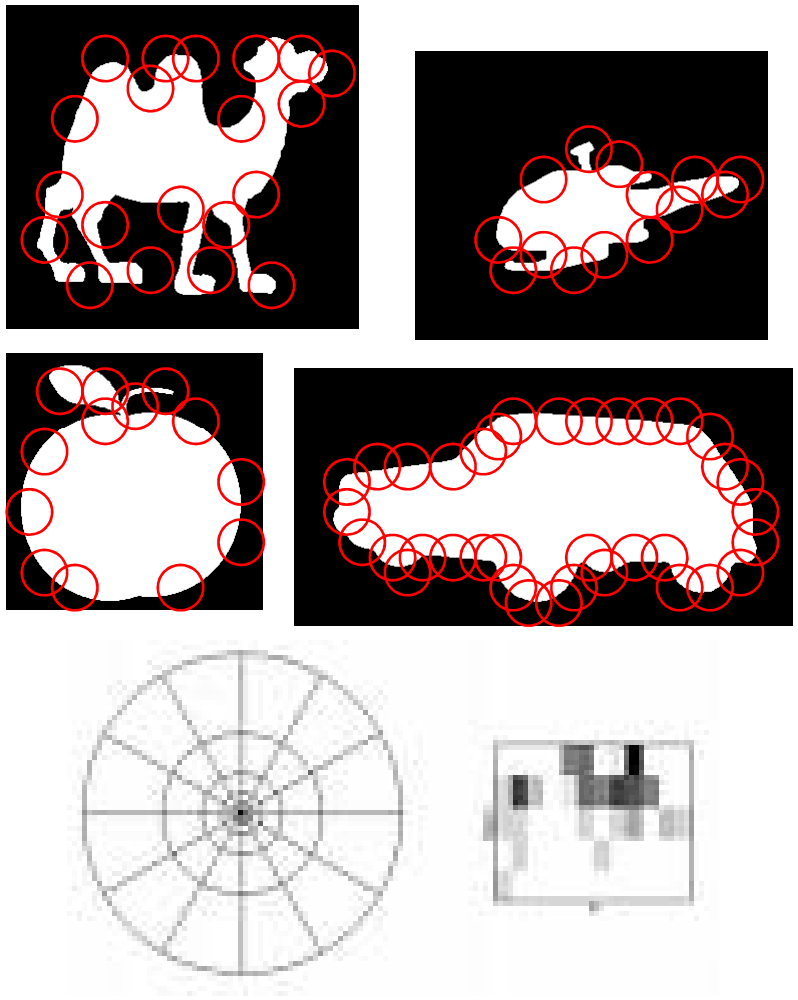
Mei-Chen Yeh* and Kwang-Ting Cheng

Learning-Based Multimedia Lab

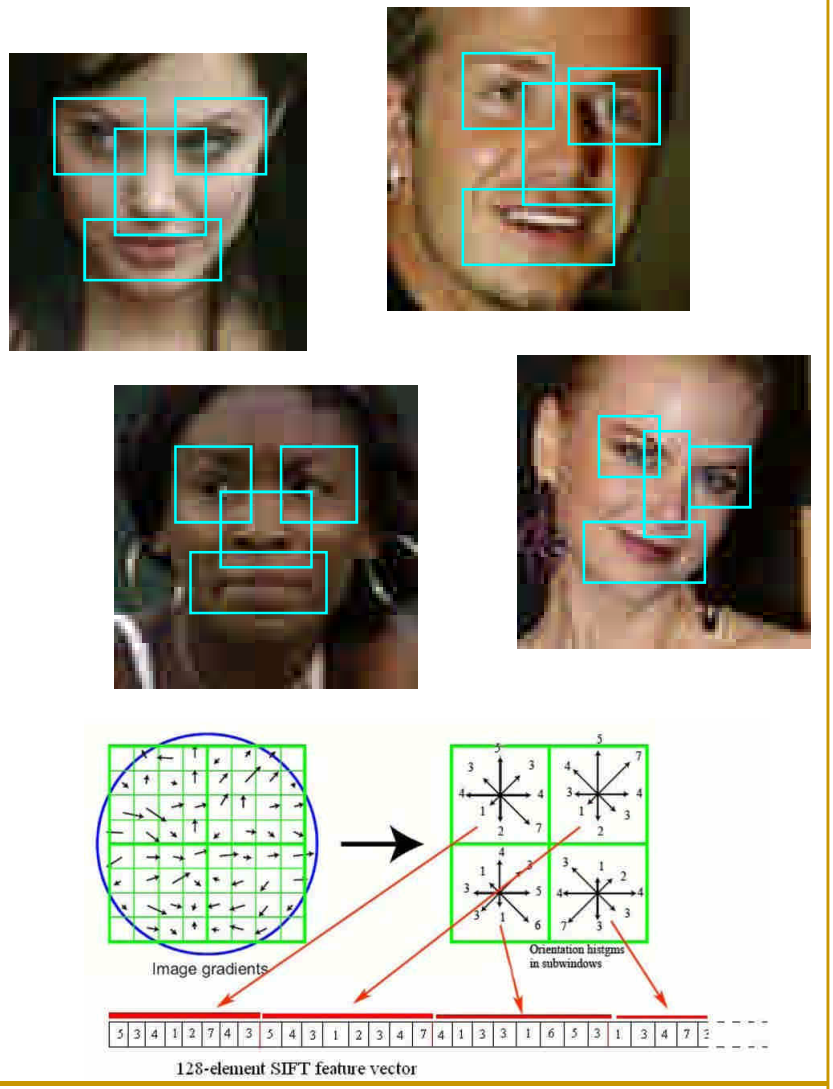
Department of Electrical and Computer Engineering

University of California, Santa Barbara

Representation Based on Local Features

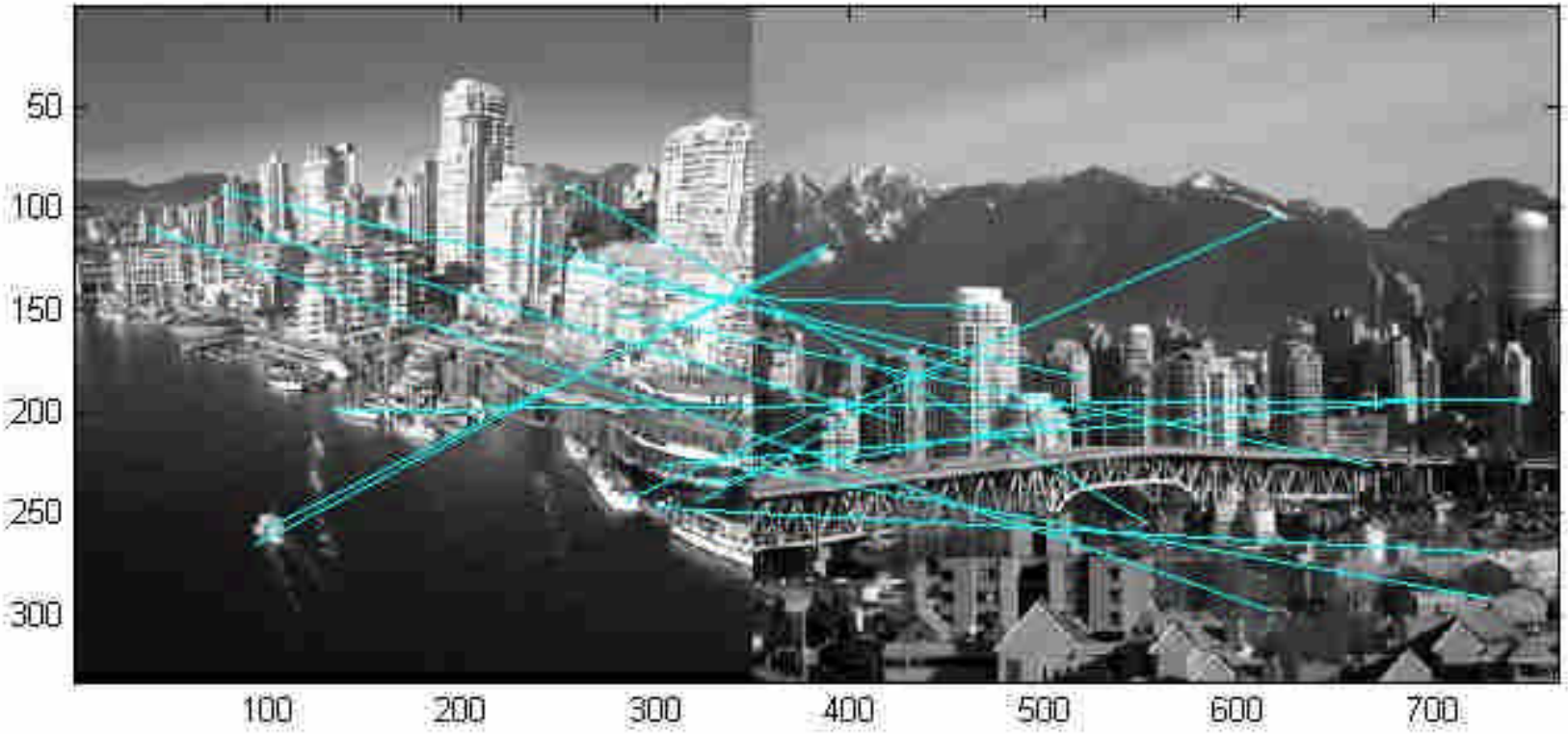


Belongie et al.; Berg. et al.; Ling & Jacobs



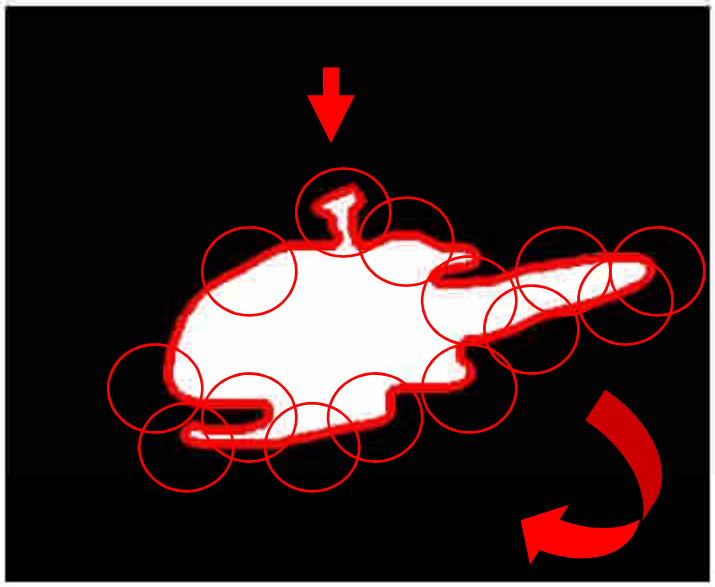
Ozkan & Duyulu; Bicego et al.; Zou et al.

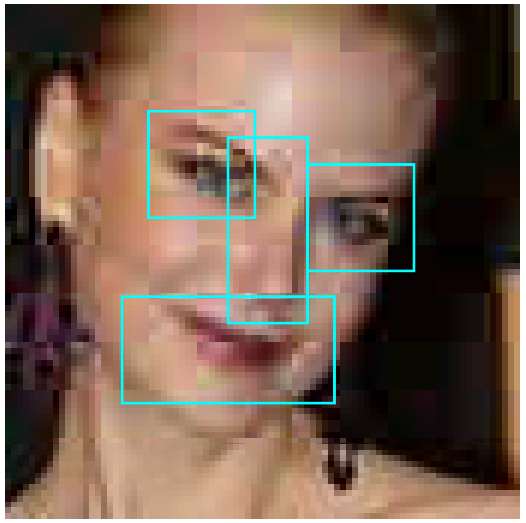
Matching



- Features are unordered
- Similarity is measured only on a feature subset

Alternative Representation?



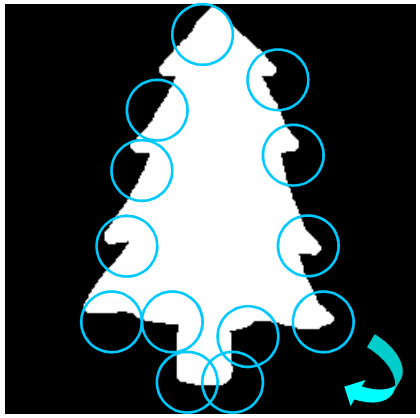


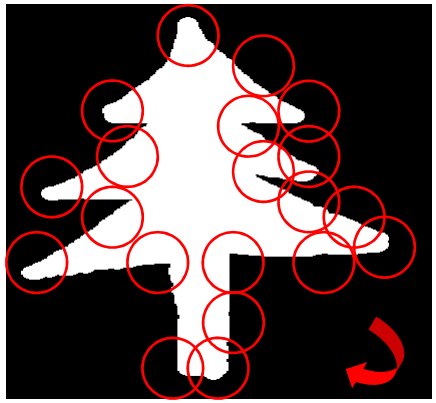
- Order carries useful information
- Incorporate order into representation

$$X = [\vec{x}_1, \vec{x}_2, ..., \vec{x}_m]$$

Globally ordered and locally unordered!

Matching





$$X = [\vec{x}_1, \vec{x}_2, ..., \vec{x}_m] \qquad Y = [\vec{y}_1, \vec{y}_2, ..., \vec{y}_n]$$

- Order of features is constrained
- Sequences may not be of equal lengths
- Tolerate errors
- Incorporate the ground distance
- Measure similarity based on both matched and unmatched features

Outline

- Introduction
  - Representation based on ordered features
  - Matching criteria
- Approach
- Experiments
  - Shape retrieval
  - Scene recognition
- Conclusion and future work

Approximate String Matching

$$X = [\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_m]$$

$$Y = [\vec{y}_1, \vec{y}_2, \ldots, \vec{y}_n]$$

$$d(X, Y) = \min \sum_i c_i$$

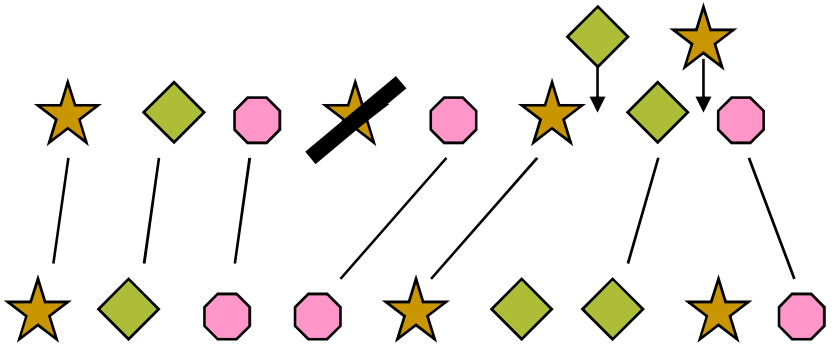
The distance is defined as the *minimal cost* of operations that transform X into Y

Edit Distance

Application dependent
Integrates the ground distance

Three operations

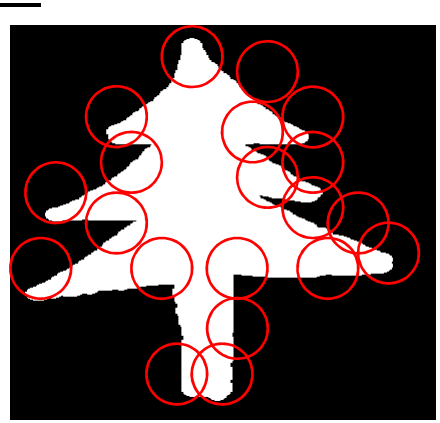
- Insertion δ (ε, *a*)
- Deletion δ (*a*, ε)
- Substitution δ (*a*, *b*)

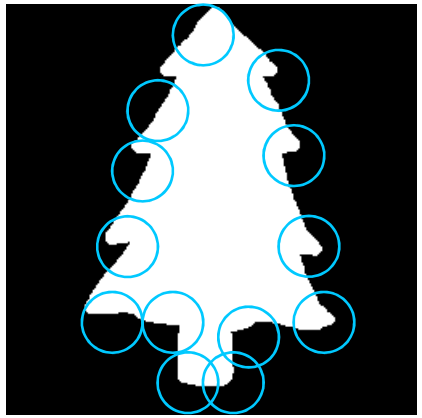


Example: Matching two shapes

- Shape Context Descriptor
- χ^2 distance *d(·,·)*

δ (ε, a) = d (**0**, a)
δ (a, ε) = d (a, **0**)
δ (a, b) = d (a, b)

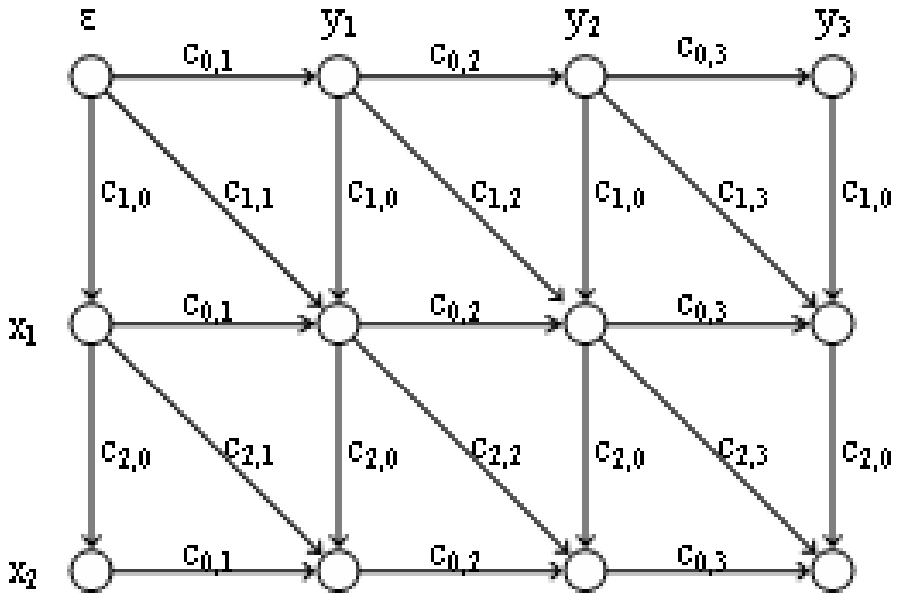




Edit Distance (cont.)

- Metric if each operation cost is also a metric
- Preserve the ordering
- The edit distance reflects
  - The similarity of the corresponding features by substitution
  - The dissimilarity of unmatched features by insertion/deletion
- Easy to incorporate the ground distance
- No need to create a visual alphabet

Computing the edit distance



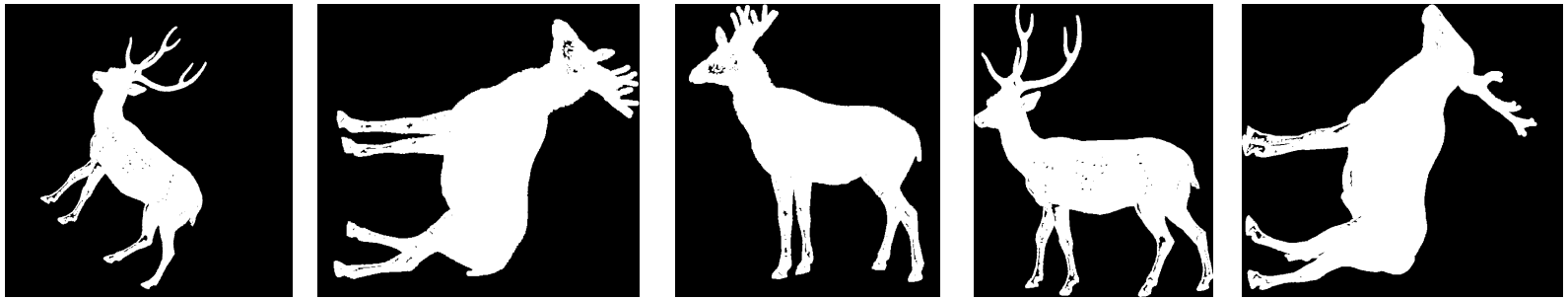
$$D(i,j) = \min\{ D(i-1,j-1) + \delta(x_i, y_j),$$
$$D(i-1,j) + \delta(x_i, \varepsilon),$$
$$D(i,j-1) + \delta(\varepsilon, y_j)\}.$$

Computation **O(*mn*)**

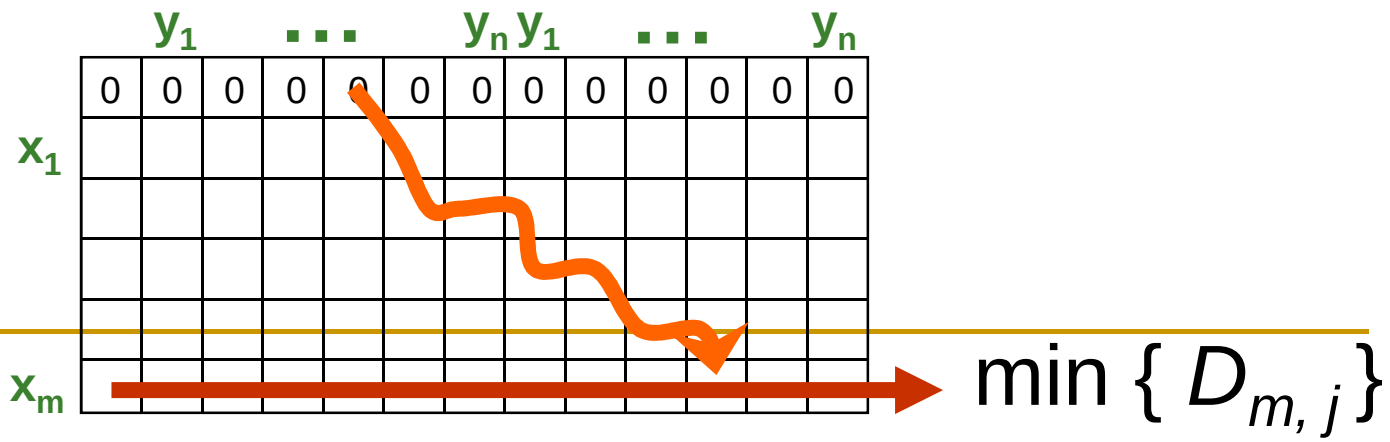
Space **O(min(*m*, *n*))**

Alignment of Two Sequences

- Cyclic sequences



- Search for a *pattern* X in a *duplicate string* Y-Y



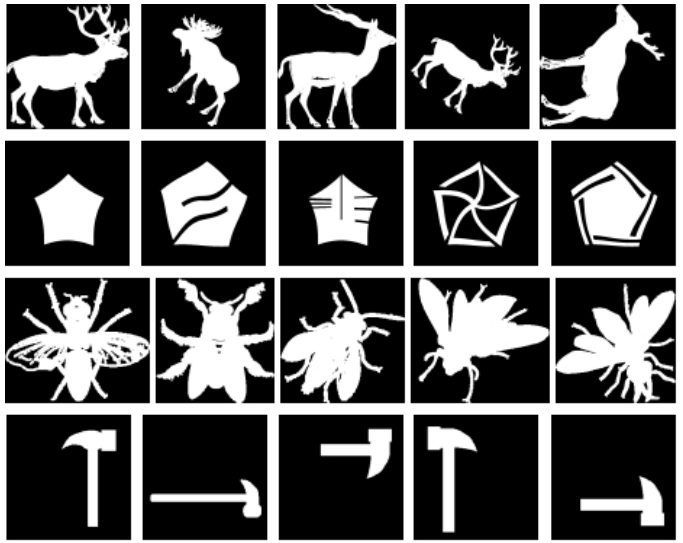
Outline

- Introduction
  - Representation based on ordered features
  - Matching criteria
- Approach
- Experiments
  - Shape retrieval
  - Scene recognition
- Conclusion and future work

Shape Retrieval

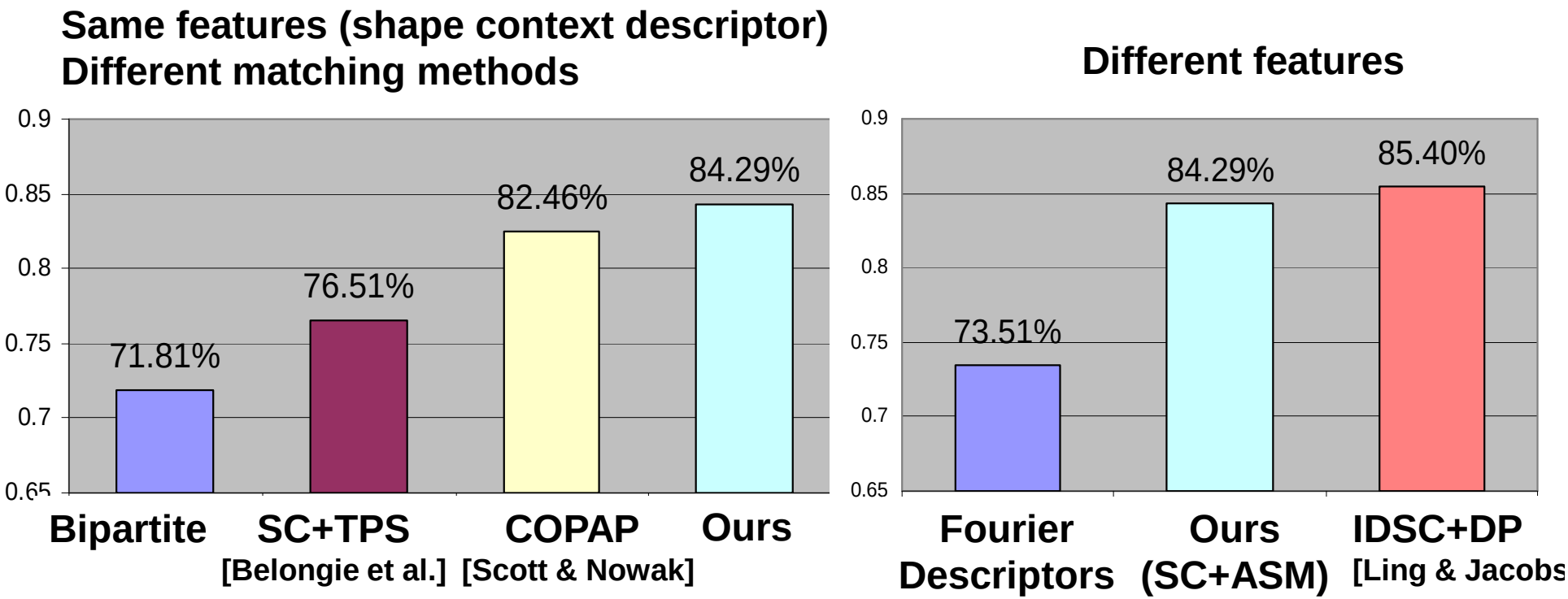
- MPEG-7 Core Experiment CE-Shape-1 part B
  - 1400 shapes, 70 categories
  - Bull's eye test

$$\frac{\#\{\text{correct hits in top 40}\}}{\#\{\text{total possible hits}\}}$$



- Representation
  - Uniformly sample 100 points
  - 60-d (5 bins, 12 orientations) shape context descriptor
  - χ^2 distance, $\delta(\varepsilon, a) = \delta(a, \varepsilon) = 0.5$

Shape Retrieval

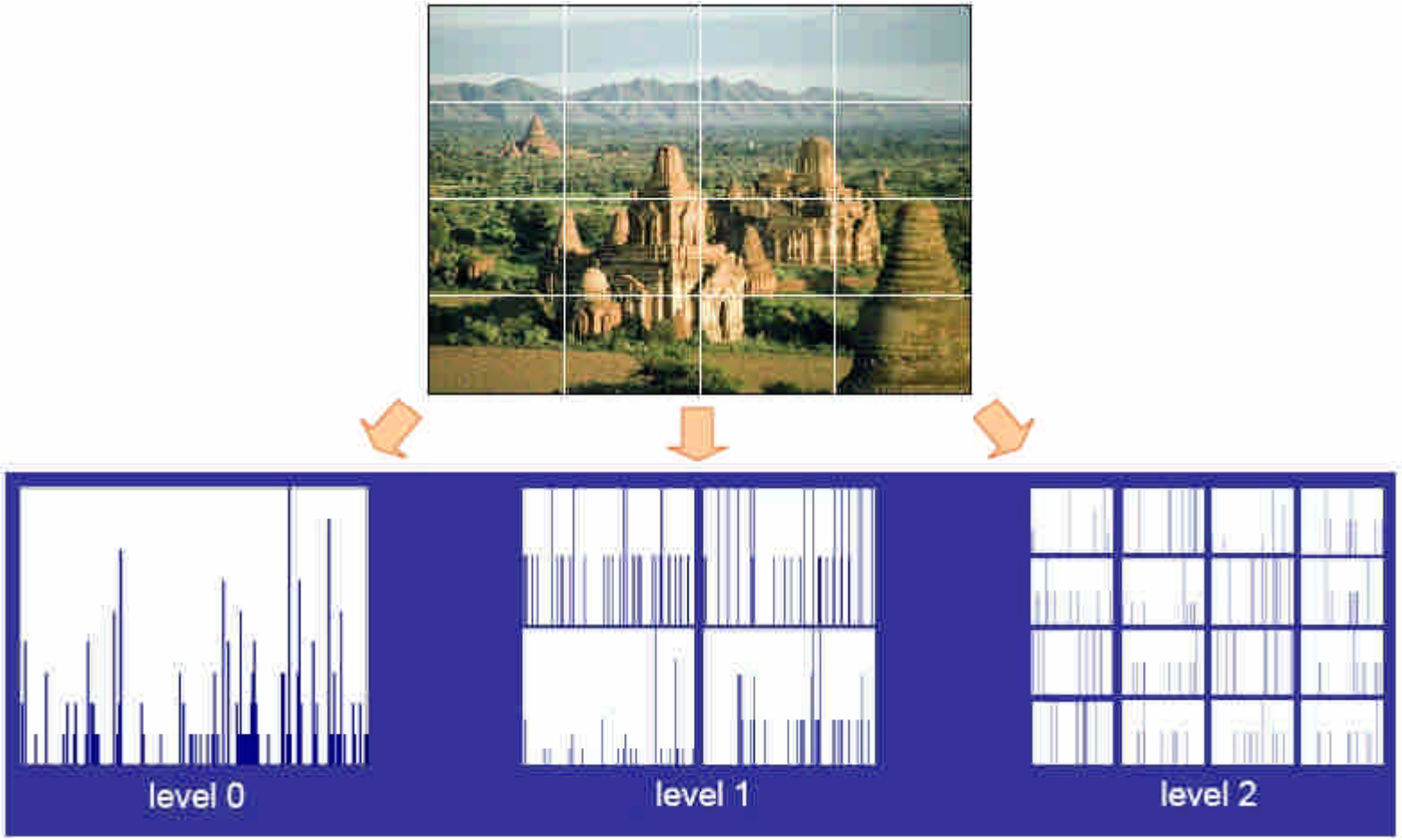


- Find the alignment during matching
- No need for any transformation model

Scene Recognition

- Scene dataset [Lazebnik et al.]
  - 15 categories, 4485 images, 200-400 images per category
  - 100 images for training, rest for testing
- Representation
  - Spatial pyramid representation
  - Harris-Hessian-Laplace detector + SIFT features
  - 200 visual words
- Classification
  - SVM with specified kernel values

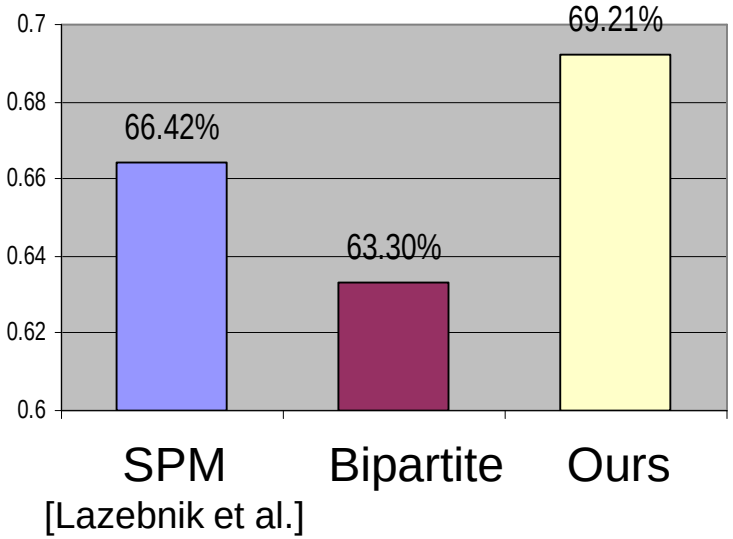
Spatial Pyramid Representation



- Level-2 partitioning (16 bags-of-features) achieves the best performance
- Each image is represented by 16 bags of features

Scene Recognition

Using the L-2 partitioning alone and the same ground distance χ^2

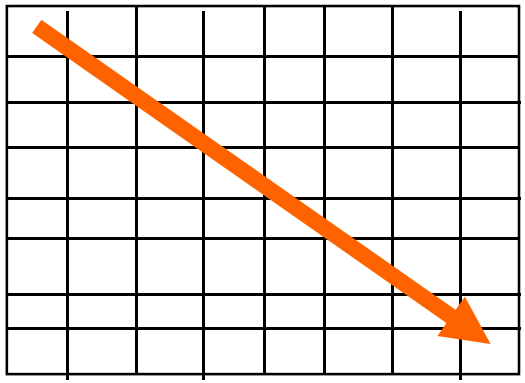


[Lazebnik et al.]

	SPM	Bipartite	Ours
L0 (1x1)	58.84±0.78		
L1 (2x2)	64.48±0.67		
L2 (4x4)	66.42±0.62	63.30±0.42	69.21±0.66
All Levels	67.95±0.50		

Spatial Pyramid Matching : bag-to-bag matching

Our matching method could allow matching across bags



86.74% suburb (90.85%)

76.73% coast (76.69%)

95.48% forest (94.82%)

48.71% bedroom (56.72%)

78.69% highway (78.13%)

55.87% inside city (63.70%)

35.21% industrial (48.34%)

67.74% mountain (70.80%)

52.65% open country (62.03%)

58.52% office (68.17%)

73.91% street (73.96%)

65.63% tall building (65.86%)

46.55% kitchen (48.00%)

53.76% living room (52.06%)

77.72% store (79.30%)

back

Conclusion

- A globally ordered and locally unordered representation for visual data
- Approximate String Matching for measuring the similarity between such representations
  - Order is considered
  - Naturally integrates the ground distance between features
  - Similarity is derived based on both matched and unmatched features

Future Work

- Image registration by using correspondences found by implementing this approach.
- Video retrieval and video copy detection by exploring the local alignment ability of string matching approach.

Thank you

More information:
http://lbmedia.ece.ucsb.edu