

Visual Object Recognition

Bastian Leibe & Kristen Grauman
 Computer Vision Laboratory & Department of Computer Sciences
 ETH Zurich & University of Texas in Austin

Chicago, 14.07.2008

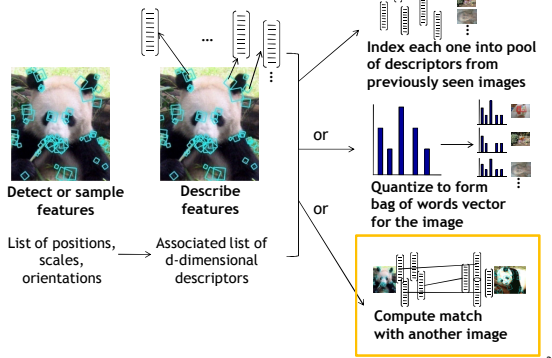


THE UNIVERSITY OF TEXAS AT AUSTIN
 Department of Computer Sciences

Outline

1. Detection with Global Appearance & Sliding Windows
2. Local Invariant Features: Detection & Description
3. Specific Object Recognition with Local Features
 – Coffee Break –
4. Visual Words: Indexing, Bags of Words Categorization
5. Matching Local Feature Sets
6. Part-Based Models for Categorization
7. Current Challenges and Research Directions

Basic flow



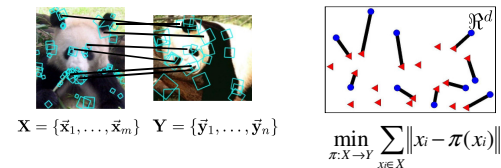
Local feature correspondences

- The matching between sets of local features helps to establish overall similarity between objects or shapes.
- Assigned matches also useful for localization



Local feature correspondences

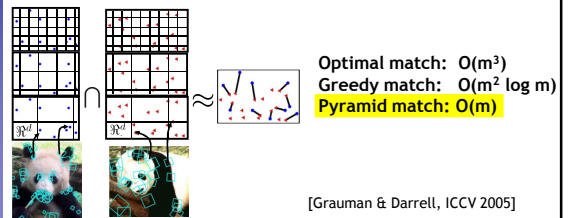
- Least cost match: minimize total cost between matched points



- Least cost *partial* match: match all of smaller set to some portion of larger set.

Pyramid match kernel (PMK)

- Optimal matching expensive relative to number of features per image (m).
- PMK is approximate partial match for efficient discriminative learning from sets of local features.



Pyramid match kernel: pyramid extraction

$\mathbf{X} = \{\vec{x}_1, \dots, \vec{x}_m\}, \vec{x}_i \in \mathbb{R}^d$

Histogram pyramid: level i has bins of size 2^i

$\Psi(\mathbf{X}) = [H_0(\mathbf{X}), \dots, H_L(\mathbf{X})]$

Visual Object Recognition Tutorial

K. Grauman

7

Pyramid match kernel: counting matches

Histogram intersection $\mathcal{I}(H(\mathbf{X}), H(\mathbf{Y})) = \sum_{j=1}^r \min(H(\mathbf{X})_j, H(\mathbf{Y})_j)$

$\mathcal{I}(H(\mathbf{X}), H(\mathbf{Y})) = 3$

Visual Object Recognition Tutorial

K. Grauman

8

Pyramid match kernel: counting new matches

Histogram intersection $\mathcal{I}(H(\mathbf{X}), H(\mathbf{Y})) = \sum_{j=1}^r \min(H(\mathbf{X})_j, H(\mathbf{Y})_j)$

$N_i = \mathcal{I}(H_i(\mathbf{X}), H_i(\mathbf{Y})) - \mathcal{I}(H_{i-1}(\mathbf{X}), H_{i-1}(\mathbf{Y}))$

matches at this level matches at previous level

Difference in histogram intersections across levels counts *number of new pairs* matched

Visual Object Recognition Tutorial

K. Grauman

9

Pyramid match kernel

histogram pyramids

$K_{\Delta}(\Psi(\mathbf{X}), \Psi(\mathbf{Y})) = \sum_{i=0}^L \frac{1}{2^i} \left(\mathcal{I}(H_i(\mathbf{X}), H_i(\mathbf{Y})) - \mathcal{I}(H_{i-1}(\mathbf{X}), H_{i-1}(\mathbf{Y})) \right)$

number of newly matched pairs at level i

measure of difficulty of a match at level i

- For similarity, weights inversely proportional to bin size (or may be learned discriminatively)
- Normalize kernel values to avoid favoring large sets

Visual Object Recognition Tutorial

K. Grauman

10

Example pyramid match

$\mathcal{I}(H_0(\mathbf{X}), H_0(\mathbf{Y})) = 2 \rightarrow N_0 = 2$
 $w_0 = 1$

Visual Object Recognition Tutorial

K. Grauman

11

Example pyramid match

$\mathcal{I}(H_1(\mathbf{X}), H_1(\mathbf{Y})) = 4 \rightarrow N_1 = 4 - 2 = 2$
 $w_1 = \frac{1}{2}$

Visual Object Recognition Tutorial

K. Grauman

12

Example pyramid match

$$\mathcal{I}(H_2(\mathbf{X}), H_2(\mathbf{Y})) = 5 \rightarrow N_2 = 5 - 4 = 1$$

$$w_2 = \frac{1}{4}$$

K. Grauman

Example pyramid match

$$K_{\Delta} = \sum_{i=0}^L w_i N_i$$

$$= 1(2) + \frac{1}{2}(2) + \frac{1}{4}(1) = 3.25$$

optimal match

$$K = \max_{\pi: \mathbf{X} \rightarrow \mathbf{Y}} \sum_{\mathbf{x}_i \in \mathbf{X}} \mathcal{S}(\mathbf{x}_i, \pi(\mathbf{x}_i))$$

$$= 1(2) + \frac{1}{2}(3) = 3.5$$

K. Grauman

Pyramid match kernel

- Forms a Mercer kernel -> allows classification with SVMs, use of other kernel methods
- Bounded error relative to optimal partial match
- Linear time -> efficient learning with large feature sets

K. Grauman, B. Leibe

Pyramid match kernel

- Forms a Mercer kernel -> allows classification with SVMs, use of other kernel methods
- Bounded error relative to optimal partial match
- Linear time -> efficient learning with large feature sets

Match [Wallraven et al.]	$O(m^2)$
Pyramid match	$O(m)$

K. Grauman, B. Leibe

Pyramid match kernel

- Forms a Mercer kernel -> allows classification with SVMs, use of other kernel methods
- Bounded error relative to optimal partial match
- Linear time -> efficient learning with large feature sets
- Use data-dependent pyramid partitions for high-d feature spaces

Code for PMK: <http://people.csail.mit.edu/jjl/libpmk/>

Matching smoothness & local geometry

- Solving for linear assignment means (non-overlapping) features can be matched independently, ignoring relative geometry.
- One alternative: simply expand feature vectors to include spatial information *before* matching.

K. Grauman, B. Leibe

Spatial pyramid match kernel

- First quantize descriptors into words, then do one pyramid match *per word* in image coordinate space.

Lazebnik, Schmid & Ponce, CVPR 2006

K. Grauman, B. Leibe

Matching smoothness & local geometry

- Use correspondence to estimate parameterized transformation, regularize to enforce smoothness

Shape context matching [Belongie, Malik, & Puzicha 2001]

Code: http://www.eecs.berkeley.edu/Research/Projects/CS/vision/shape/sc_digits.html

K. Grauman, B. Leibe

Matching smoothness & local geometry

- Let matching cost include term to penalize distortion between pairs of matched features.

Approximate for efficient solutions: Berg & Malik, CVPR 2005; Leordeanu & Hebert, ICCV 2005

Figure credit: Alex Berg

K. Grauman, B. Leibe

Matching smoothness & local geometry

- Compare “semi-local” features: consider configurations or neighborhoods and co-occurrence relationships

Correlograms of visual words [Savarese, Winn, & Criminisi, CVPR 2006]

Proximity distribution kernel [Ling & Soatto, ICCV 2007]

Hyperfeature-mesh Hyperfeatures: Agarwal & Triggs, ECCV 2006

Feature neighborhoods [Sivic & Zisserman, CVPR 2004]

Tiled neighborhood [Quack, Ferrari, Leibe, van Gool ICCV 2007]

K. Grauman, B. Leibe

Matching smoothness & local geometry

- Learn or provide explicit object-specific shape model [Next in the tutorial : part-based models]

K. Grauman, B. Leibe

Summary

- Local features are a useful, flexible representation
 - Invariance properties - typically built into the descriptor
 - Distinctive, especially helpful for identifying specific textured objects
 - Breaking image into regions/parts gives tolerance to occlusions and clutter
 - Mapping to visual words forms discrete tokens from image regions
- Efficient methods available for
 - Indexing patches or regions
 - Comparing distributions of visual words
 - Matching features

K. Grauman, B. Leibe

Outline

1. Detection with Global Appearance & Sliding Windows
2. Local Invariant Features: Detection & Description
3. Specific Object Recognition with Local Features
- *Coffee Break* –
4. Visual Words: Indexing, Bags of Words Categorization
5. Matching Local Feature Sets
6. Part-Based Models for Categorization
7. Current Challenges and Research Directions