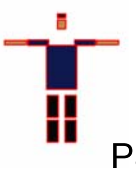



Part-based models for recognition: matching

Tuesday, January 30


Part-based models

- Today: focus on efficient matching
- Thursday: focus on representation and learning of parts

Types of recognition approaches

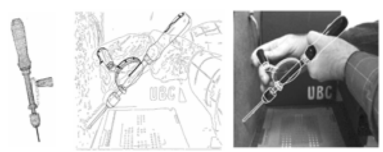
- pose consistency; geometry
- global measures of appearance
- local measures of appearance
- local part appearance and relative geometry

Particular systems may have aspects of one or more type

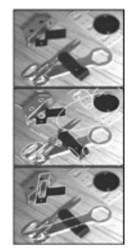
(3D) Model-based approaches

- Alignment/pose consistency: fit projected model to image data
- Index invariants and verify
- Geometry is key

[Lowe 1991]



Model — Observed edges, previous location — Projected model



[Rothwell et al. 1992]

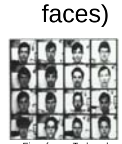
(3D) Model-based approaches

Challenges:

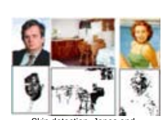
- Constructing the model
- Poor scaling with number of models
- Occlusions
- Ambiguity without strong appearance evidence
- Generic categories?

Appearance-based approaches

- Recognize by matching overall appearance
- Windowed search (multi-scale/orientation)
- Match “templates”, build classifiers
- Represent space of variation from examples, or model separately (e.g. frontal vs. profile faces)

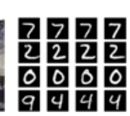


Eigenfaces, Turk and Pentland 1991



Skin detection, Jones and Rehg 1999

Pedestrians and faces, Viola and Jones, 2001, 2003



MNIST database of handwritten digits

Appearance-based approaches

Advantages:

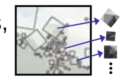
- Capture characteristic appearance properties, if they exist
- Many existing learning techniques applicable depending on feature choice

Challenges:

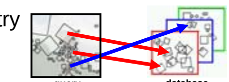
- Clutter, occlusion sensitivity
- Capturing variation for generic categories or complex objects

Local appearance approaches

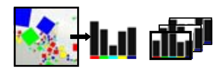
Decompose image into local parts, describe appearance of each part



• Index via voting, +/- local geometry constraints *[Schmid, Lowe, Tuytelaars, -(et al)….]*



• Compare distribution of local appearance patches ("bags of features") *[Leung, Csurka, Sivic, Lazebnik, -(et al)….]*



Local appearance approaches

Advantages:

- Local appearance often simpler and more reliable, easier to detect and learn
- Possibilities for handling occlusions and clutter
- Invariant local features (coming up) are distinctive and repeatable, especially for object instances
- With sparse set: # regions << # pixels

Local appearance approaches

Challenges:

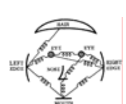
- Large-scale indexing problems (voting)
- Single feature matching assumes independence
- Sparse interest operators may bias towards particular types of regions (e.g., textured)
- How to define a feature "vocabulary"?
- Geometry gone (bags of features)
- Localization

(More on these approaches in coming weeks.)

Local appearance+geometry

Part-based models, constellation models

- Model for an object/class is set of parts and their spatial relationships
- Model fit is measured by
  - similarity of the part appearance
  - parts' geometry agreement



Fischler and Elschlager 1973

Felzenszwalb and Huttenlocher 2000

Leung, Burl, Perona 1995

Weber, Welling, Perona 2000

Local appearance+geometry

Advantages:

- Local appearance may be simpler, reliable
- Possibilities to handle clutter/occlusion
- Maintain configuration information
- Possibilities to exploit independence properties among parts for computational gain
- Capture variations of complex objects more succinctly

Local appearance+geometry

Challenges:

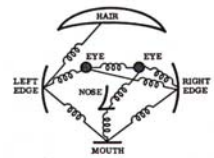
- Invariant geometric constraints
- Computational issues: correspondences, matching
- Can sparse parts scale for large number of categories?
- Constructing/learning models

Part-based models: History

- Fischler & Elschlager 1973

- Yuille '91
- Brunelli & Poggio '93
- Lades, v.d. Malsburg et al. '93
- Cootes, Lanitis, Taylor et al. '95
- Amit & Geman '95, '99
- Perona et al. '95, '96, '98, '00
- Felzenszwalb & Huttenlocher '00

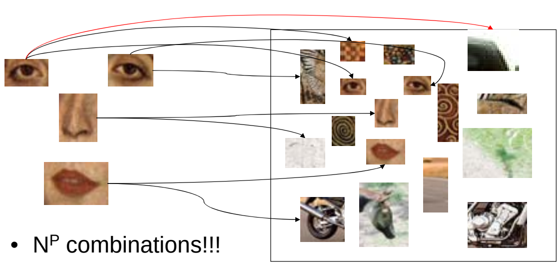
- Many papers since 2000



[This and some of following slides from R. Fergus's ICCV 2005 tutorial]

The correspondence problem

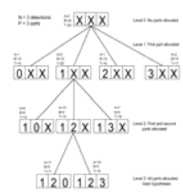
- Model with P parts
- Image with N possible locations for each part



- N^P combinations!!!

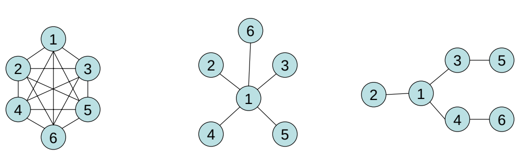
Efficient search methods

- Interpretation tree (Grimson '87)
  - Condition on assigned parts to give search regions for remaining ones
  - Branch & bound, A*



Connectivity of parts

- Complexity is given by size of maximal clique in graph

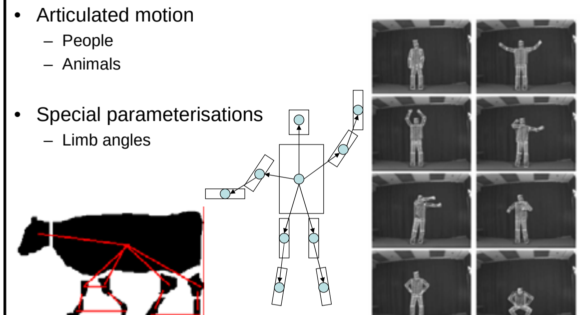


Fully connected	Star structure	Tree structure
$O(N^6)$	$O(N^2)$	$O(N^2)$

Sparser graphs cannot capture all interactions between parts, *but* reduced structure can be exploited to simplify computation.

Some class-specific graphs

- Articulated motion
  - People
  - Animals
- Special parameterisations
  - Limb angles



Images from [Kumar05, Felzenszwalb05]

Distance transforms

- Set of points, P, some distance $\| \bullet \|$

$$D_P(x) = \min_{y \in P} \| x - y \|$$

  - For each location x distance to nearest y in P
  - Think of as cones rooted at each point of P
- Commonly computed on a grid Γ using

$$D_P(x) = \min_{y \in \Gamma} (\| x - y \| + 1_P(y))$$

  - Where $1_P(y) = 0$ when $y \in P$, ∞ otherwise

[Slide from D. Huttenlocher's 2003 tutorial]

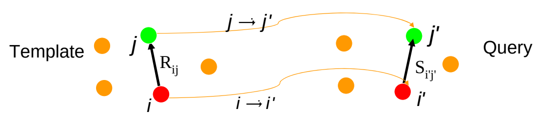
Distance transforms

Efficiently computed with dynamic programming

Useful for
- Chamfer measure
- Hausdorff distance
- Matching tree structured part-based models [Felzenszwalb and Huttenlocher 2000]: $O(N^2P) \rightarrow O(NP)$ time

Low-distortion correspondences

[Berg, Berg, and Malik CVPR 2005]

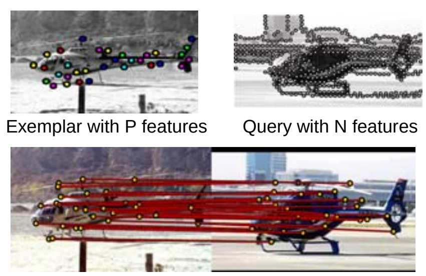


Enforce relationship constraints among corresponding features:

Measure distortion in vectors between pairs of feature points, i.e., changes in direction or length.

[Figure from Alex Berg]

Low-distortion correspondences

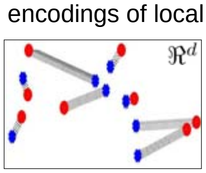


- Exemplar(s) define model
- Formulate as Integer Quadratic Programming problem
- $O(N^P)$ in general
- Use approximation that takes $O(P^2\ N \log(N))$ time

[Figures from Alex Berg]

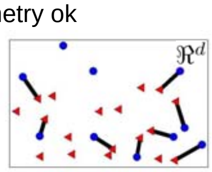
Approximate matching cost

- O(P) approximations for $O(P^3)$ optimal assignment problem $\min_{\pi: \mathbf{A} \rightarrow \mathbf{B}} \sum_{a \in \mathbf{A}} D(a, \pi(a))$
- No configuration geometry, but encodings of local geometry ok



Bijective matching :

[Indyk and Thaper 2003] comparing color histograms for fast nearest-neighbor retrieval



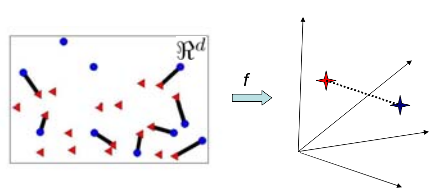
Partial matching :

[Grauman and Darrell 2005] partial match kernel for efficient learning from sets of features

Geometric embeddings

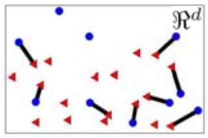
- Map inputs to a space where an inexpensive distance approximates an expensive one

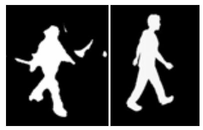
$$D_m(A,B) <= \| f(A) - f(B) \| < C \cdot D_m(A,B)$$



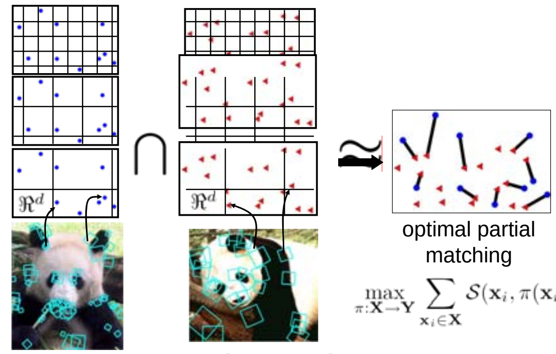
Approximate partial matching

Compare sets by computing a *partial matching* between their features.





Pyramid match overview



optimal partial matching

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

$\mathbf{X} = \{\vec{\mathbf{x}}_1, \ldots, \vec{\mathbf{x}}_m\}$ $\mathbf{Y} = \{\vec{\mathbf{y}}_1, \ldots, \vec{\mathbf{y}}_n\}$

Pyramid match overview

Pyramid match measures similarity of a partial matching between two sets:

- Place multi-dimensional, multi-resolution grid over point sets
- Consider points matched at finest resolution where they fall into same grid cell
- Approximate optimal similarity with worst case similarity within pyramid cell

No explicit search for matches!

Pyramid match

Number of newly matched pairs at level *i*

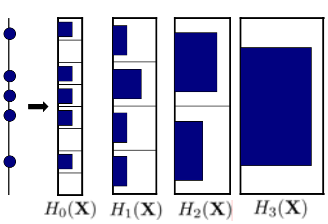
Approximate partial match similarity $K_\Delta = \sum_{i=0}^{L} w_i N_i$

Measure of difficulty of a match at level *i*

[Grauman and Darrell, ICCV 2005]

Pyramid extraction

$\mathbf{X} = \{\vec{\mathbf{x}}_1, \ldots, \vec{\mathbf{x}}_m\}, \quad \vec{\mathbf{x}}_i \in \Re^d$

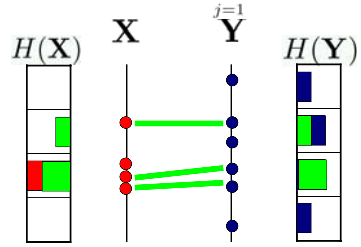


Histogram pyramid: level i has bins of size 2^i

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

Counting matches

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



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

Counting new matches

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

$$N_i = \underbrace{\mathcal{I}\left(H_i(\mathbf{X}), H_i(\mathbf{Y})\right)}_{\text{matches at this level}} - \underbrace{\mathcal{I}\left(H_{i-1}(\mathbf{X}), H_{i-1}(\mathbf{Y})\right)}_{\text{matches at previous level}}$$

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

Pyramid match

$$K_\Delta\left(\overbrace{\Psi(\mathbf{X}), \Psi(\mathbf{Y})}^{\text{histogram pyramids}}\right) = \sum_{i=0}^{L} \frac{1}{2^i}\left(\underbrace{\mathcal{I}\left(H_i(\mathbf{X}), H_i(\mathbf{Y})\right) - \mathcal{I}\left(H_{i-1}(\mathbf{X}), H_{i-1}(\mathbf{Y})\right)}_{\text{number of newly matched pairs at level } i}\right)$$

$\frac{1}{2^i}$: measure of difficulty of a match at level *i*

- For similarity, weights inversely proportional to bin size
- For cost, weights proportional: $w_i = 2^i$
- Normalize values to avoid favoring large sets

Efficiency

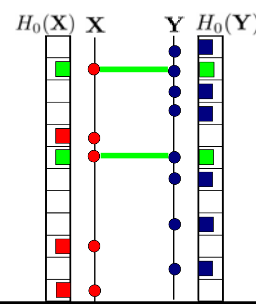
Pyramid match complexity $O(dmL)$

d	feature dimension
m	set size
$L = \log(D)$	number of pyramid levels
D	range of feature values

for uniformly-sized bins

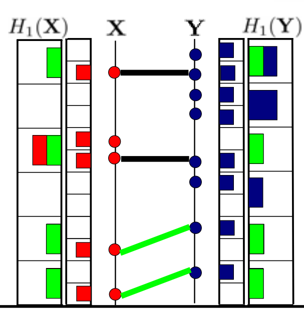
Example pyramid match

$$\mathcal{I}\left(H_0(\mathbf{X}), H_0(\mathbf{Y})\right) = 2 \longrightarrow \begin{array}{l} N_0 = 2 \\ w_0 = 1 \end{array}$$



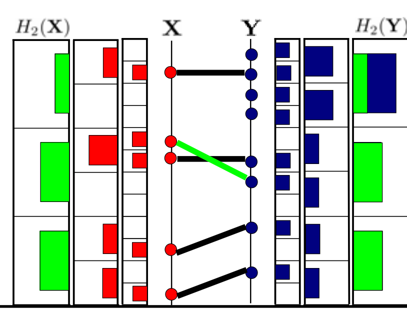
Example pyramid match

$$\mathcal{I}\left(H_1(\mathbf{X}), H_1(\mathbf{Y})\right) = 4 \longrightarrow \begin{array}{l} N_1 = 4 - 2 = 2 \\ w_1 = \frac{1}{2} \end{array}$$



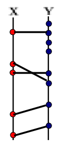
Example pyramid match

$$\mathcal{I}\left(H_2(\mathbf{X}), H_2(\mathbf{Y})\right) = 5 \longrightarrow \begin{array}{l} N_2 = 5 - 4 = 1 \\ w_2 = \frac{1}{4} \end{array}$$



Example pyramid match

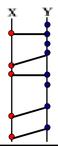
pyramid match



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

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

optimal match



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

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

Pyramid match properties

- Linear time matching
- Mercer kernel
- Bounded approximation error relative to optimal partial matching cost
- Sub-linear time hashing over matching

Coming up

- Thursday, Feb 1
  - Weber et al. and Fergus et al. papers on constellation models
  - Demo?
- Next week: Invariant local features
  - Demo?
- Project plans:
  - Find your partner
  - We'll talk about proposal scope the week after next