Approaches to Representing and Recognizing Objects

Visual Classification

CMSC 828J – David Jacobs

Recognition = Correspondence + Comparison?

- Correspondence
  - Search methods
    - Template matching: chamfer, distance transform, Hough, RANSAC
    - Dynamic programming for 1D (contours, actions, k-fans)
    - Assignment algorithm (Shape context)
    - Bag of words.
    - Interest points to focus attention.
- Comparison
  - Linear methods, assuming aligned images.
    - PCA, LDA, Linear combinations for lighting, mapping manifolds to low-dimensional spaces, Linear separators.
  - Complex descriptors (SIFT, Shape context)

Modeling Classes

- Convex combination of examples – Blanz and Vetter
- Learning discriminative model from examples.
  - Power can be in learning machine
  - Or in large number of examples.
  - Or in comparison method (deformations, lighting, …)
- Generative models (with parts): appearance distribution + deformation distribution. HMMs.
- Grammars (Jin and Geman)

Important Equations

$$s\begin{pmatrix}1&0&0\\0&1&0\end{pmatrix}\begin{pmatrix}1&0&0&t_x\\0&1&0&t_y\\0&0&1&t_z\end{pmatrix}\begin{pmatrix}r_{1,1}&r_{1,2}&r_{1,3}&0\\r_{2,1}&r_{2,2}&r_{2,3}&0\\r_{3,1}&r_{3,2}&r_{3,3}&0\\0&0&0&1\end{pmatrix}P$$

$$\equiv\begin{pmatrix}s_{1,1}&s_{1,2}&s_{1,3}&st_x\\s_{2,1}&s_{2,2}&s_{2,3}&st_y\end{pmatrix}P$$

where

$$(s_{1,1},s_{1,2},s_{1,3})\bullet(s_{2,1},s_{2,2},s_{2,3})=0$$

$$\|(s_{1,1},s_{1,2},s_{1,3})\|=\|(s_{2,1},s_{2,2},s_{2,3})\|$$

So if a side has surface normal, n, and we express the viewing direction with a vector v, the side is visible iff <n,v> > 0.

$$\begin{pmatrix}r_{1,1}&r_{1,2}&r_{1,3}&t_x\\r_{2,1}&r_{2,2}&r_{2,3}&t_y\\r_{3,1}&r_{3,2}&r_{3,3}&t_z\end{pmatrix}\begin{pmatrix}x\\y\\0\\1\end{pmatrix}\equiv\begin{pmatrix}r_{1,1}&r_{1,2}&t_x\\r_{2,1}&r_{2,2}&t_y\\r_{3,1}&r_{3,2}&t_z\end{pmatrix}\begin{pmatrix}x\\y\\1\end{pmatrix}$$

$$J(e)=\sum\|(m+a_ke)-x_k\|^2=-\sum e(x_k-m)(x_k-m)e+\sum\|x_k-m\|^2$$

So we need to maximize eSe subject to $\|e\|=1$

e is the leading eigenvector of S

$$J(w)=\frac{(m_1'-m_2')}{(s_1'^2-s_2'^2)}=\frac{wS_{bw}w}{wS_ww}$$

$$\begin{pmatrix}u_1^1&u_2^1&.&.&.&u_n^1\\v_1^1&v_2^1&&&&v_n^1\\u_1^2&u_2^2&&&&u_n^2\\v_1^2&v_2^2&&&&v_n^2\\.&.&&&&.\\.&.&&&&.\\.&.&&&&.\\u_1^m&u_2^m&&&&u_n^m\\v_1^m&v_2^m&.&.&.&v_n^m\end{pmatrix}=\begin{pmatrix}s_{1,1}^1&s_{1,2}^1&s_{1,3}^1&t_x^1\\s_{2,1}^1&s_{2,2}^1&s_{2,3}^1&t_y^1\\s_{1,1}^2&s_{1,2}^2&s_{1,3}^2&t_x^2\\s_{2,1}^2&s_{2,2}^2&s_{2,3}^2&t_y^2\\&.&&\\&.&&\\&.&&\\s_{1,1}^m&s_{1,2}^m&s_{1,3}^m&t_x^m\\s_{2,1}^m&s_{2,2}^m&s_{2,3}^m&t_y^m\end{pmatrix}\begin{pmatrix}x_1&x_2&.&.&.&x_n\\y_1&y_2&&&&y_n\\z_1&z_2&&&&z_n\\1&1&&&&1\end{pmatrix}$$

$$f(x) = g * h = \int g(x - x_0)h(x_0)dx_0$$

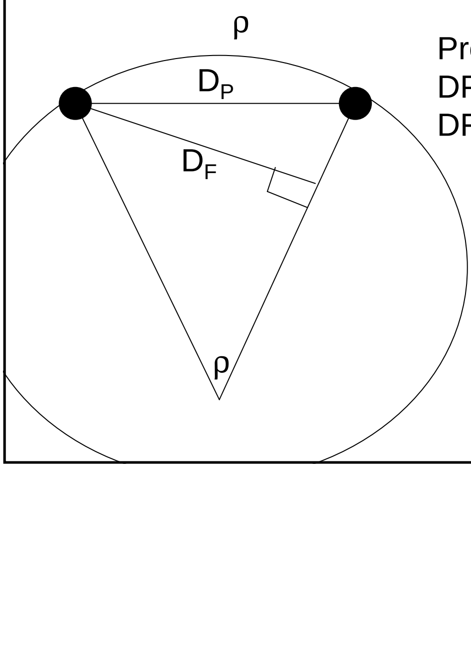
$$f \otimes g = T^{-1} F * G$$

$$i = \max(0, \lambda(\vec{l} \bullet \hat{n}))$$

$$r = k * l = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} (K_{nm} L_{nm}) h_{nm} \approx \sum_{n=0}^{2} \sum_{m=-n}^{n} (K_{nm} L_{nm}) h_{nm}$$

$$\begin{pmatrix} u_1 & u_2 & . & . & . & u_n \\ v_1 & v_2 & & & & v_n \end{pmatrix} = s \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_1 & x_2 & . & . & . & x_n \\ y_1 & y_2 & & & & y_n \end{pmatrix}$$

$$\begin{pmatrix} a & b \\ -b & a \end{pmatrix} \begin{pmatrix} x_1 & x_2 & . & . & . & x_n \\ y_1 & y_2 & & & & y_n \end{pmatrix}$$



Procrustes Distances is r.
DP = 2 sin (r/2)
DF = sin r.

$$\lambda \| J(x) - I(v(x)) \|_{L_2} + \|v\|_g$$

$$x' = f(x,y) = a_1 + a_2 x + a_3 y + \sum_{i=1}^{n} w_i U\left(\left|P_i - (x,y)\right|\right)$$

$$U(r) = -r^2 \log(r)$$

$$\min_y \sum_i \sum_j \left(\left\|x_i - x_j\right\| - \left\|y_i - y_j\right\|\right)^2 \quad x \in R^n, y \in R^d, d < n$$

$$E(W) = \sum_i \left|X_i - \sum_j W_{ij} X_j\right|^2 = X^T (I - W)^T (I - W) X \equiv X^T M X$$

$$0 = I_t + \nabla I \cdot [\frac{\partial x}{\partial t} \ \frac{\partial y}{\partial t}]$$

$$\left[\begin{array}{cc} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{array} \right] \left[\begin{array}{c} u \\ v \end{array} \right] = - \left[\begin{array}{c} \sum I_x I_t \\ \sum I_y I_t \end{array} \right]$$

Markov: P(qt = Sj | qt-1= Si, qt-2=Sk, …) = P(qt = Sj | qt-1 = Si).

Important Algorithms

- Distance transform
- Hough transform
- RANSAC
- Histogram Equalization
- Elastic matching
- Grassfire algorithm
- MDS
- ISOMAP
- LLE
- HMMs: likelihood of sequence, Baum-Welch
- Naïve Bayes
- Perceptron
- SVM

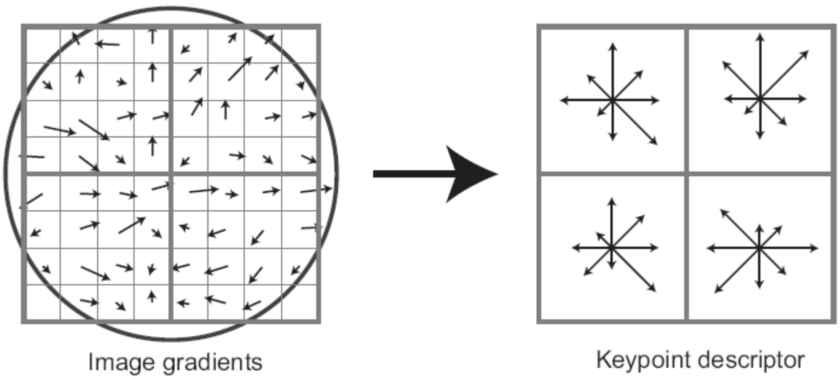


Figure 7: A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location, as shown on the left. These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 subregions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array.

Discussion

- What is the state of recognition?
  - Do things work?
  - Are we on the right track?
- What are the best future directions?
- Important open questions?
  - Combination of sources of variability
  - Generalization

Discussion – the class

- What topics did you like?
- Pace?
- Debates?
- Suggestions for the future