Lecture on Constellation Model.


Deformable Template (DT)

Basic features connected by springs. (geometric constraints)

This DT is matched to the image. The features have appearance models that match features extracted from the image.

Simplest Model

Template \( \{ X_i : i = 1 \text{ to } N \} \)

Data Feature \( \{ Y_{ia} : a = 1 \text{ to } M \} \)

Transformation on the data T (e.g. affine)

Matching variables \( V_{ia} = 1 \) if \( i \) matches \( a \)

Constraint \( \sum_a V_{ia} = 1 \), for each \( i \).
(8) \[ E[I, V] = \sum_{i, a} V_{ia} M(T(x_i), y_a) + E[I, V] \]

Here: \( M(\cdot, \cdot) \) is a similarity measure.

- \( M(T(x_i), y_a) = |T(x_i) - y_a|^2 \)

\( E[I|T, V] \) is a prior on the transformation.

\[ P[I, V] = \frac{1}{Z} e^{-E[I, V]} \]

Write as:

\[ P[Y \mid I, V] P[I, V] \]

\[ = P[Y, I \mid V] \]

Maximize to estimate \( I, V \).

All naturally:

\[ P[Y, I \mid X] = \sum_{V} P[Y, I \mid V, X] \]

Make this more complex by adding attribute variables for the feature embedded points:

\[ \langle (\phi_i, x_i), \langle (y_a, t_a) \rangle \]

Similarity measure on attributes:

\[ S(\phi_i, y_a) \quad S(\cdot, \cdot) \text{ similarity} \]
(3)

Oclusion: Dummy node $a = 0$.

$V_{io} = 1$, means node $x_i$ is occluded

pay penalty for occlusion

E.g. add $\lambda \sum V_{io}$ to the energy.

Formally, the model can be expressed as a mixture of distribution.

Another variant, no explicit transformation $T$. Instead treat the $\{x_i\}$ as random variables, with a prior $p(\{x_i\})$ usually Markov.

Another variant, explicit transformation $T$ plus prior on the $\{x_i\}$.

Many variants in the 1990s, Ranganujan, Tuille...
Constellation Models Perona et al.

Features: X - location
Parts A - appearance
S - scale

Appearance will correspond to intensity patches.

Parts P (3-7)

h - assignment variables (like V)
can allow for occlusion.

Preprocessing of the image extracts features
(to be described later). N features extracted
- match to P parts.

Two Tasks:
(i) Unsupervised learning of the models from training data.
(ii) Detection and recognition of the constellation model from an image.
(5) \[ \text{Appearance model} \]

\[
\begin{align*}
\text{mean & covariance} & \\
\Theta_p^\text{app} & = \begin{bmatrix} \ell_{SP} & \ell_{VP} \end{bmatrix}^T \quad \Theta_{bg}^\text{app} & = \begin{bmatrix} \ell_{bg} & \ell_{bg} \end{bmatrix}^T
\end{align*}
\]

\[
P(A | X, S, h, \theta) = \prod_{p=1}^{dp} \frac{G(A(h_p) | S_p, V_p)}{G(A(h_p) | \ell_{bg}, \ell_{bg})}
\]

log-likelihood ratio is used to discriminate between parts and non-parts.

\[ dp = 1, \text{ if part } p \text{ is non-occluded} \]
\[ dp = 0, \text{ otherwise.} \]

\text{Shape model} \[
P(X | S, h, \theta) = G(X | \ell, \Sigma)\chi_f
\]

\[ P(X | S, h, \theta_{bg}) \]

\text{uniform distribution on background.}
(6) Scale

\[
\frac{\text{Scale}}{\text{the scale of each point, relative to a reference frame is a Gaussian density}} \quad \theta_{\text{scale}} = \langle t_p, v_p \rangle
\]

\[
P(l \leq h, \theta) = \frac{\int_{l}^{\theta} \mathcal{G}(l, v_p) \, dv_p}{\int_{l}^{\theta} \mathcal{G}(l, v_p) \, dv_p + \text{uniform distribution on background.}}
\]

Occlusion &

\[
P(h_l \theta) = \frac{P_{\text{polar}}(h_l \theta) \cdot \text{p}(l | \theta)}{P(h_l \theta | \text{polar}) \cdot \text{p}(l | \theta) \cdot \text{p}(h_l | \theta)}
\]

Combined Model:

\[
P(A_1 \times S, h_l \theta) \cdot P(S \mid h_l \theta) \cdot P(h_l \theta)
\]

\[
\frac{P(A_1 \times S, h_l \theta)}{P(S \mid h_l \theta | \text{polar}) \cdot P(h_l \theta | \text{polar})} \cdot \text{p}(h_l \theta)
\]

Need to sum out the matching variables

h.e.
Feature Detection

Use Kadir & Brady to detect salient image regions. (scale invariant)

Take histogram PCIs in curvile radius $s$.

Calculate entropy $H(s)$ as functions.

Local maxima of $H(s)$ are candidates.

Saliency is measured by $H(s)$ (normalized $H(s)$ for scale). This gives a local measure for the scale of the template.

Feature Representation

Rescale detect regions to $11 \times 11$ patches.

Use PCA to reduce dimension to 20. Then use a Gaussian.
Learning:

The learning task is to estimate the model parameters

\[ \Theta = \langle \mu, \Sigma, c, v, M, \text{plate}, t, U \rangle \].

Strategy: EM algorithm

Input - dataset of images including the target object.

E-Step

\[ \rightarrow \text{use current estimate of } \Theta \text{ to perform best match of DI to image, and gather statistics.} \]

M-Step

\[ \rightarrow \text{use statistics to update value of } \Theta. \]

Requires \( \approx \text{estimating, or summing out the } h \text{ variables, can be very demanding computationally (e.g., pruning, etc.).} \)
(9) Extensive & Limitations

1. The approach relies on patch feed-
   - not invariant to spatial warps
   - unsuitable for deformable objects (e.g. horses)

2. The model does not exploit all
   the knowledge about the object.
   - e.g. humans can still recognize the
     object once the parts have been
     removed.

3. Some extension in the Google Image
   paper. But still limited. Constraining
   the types of features.