Note Title 11/14/2007

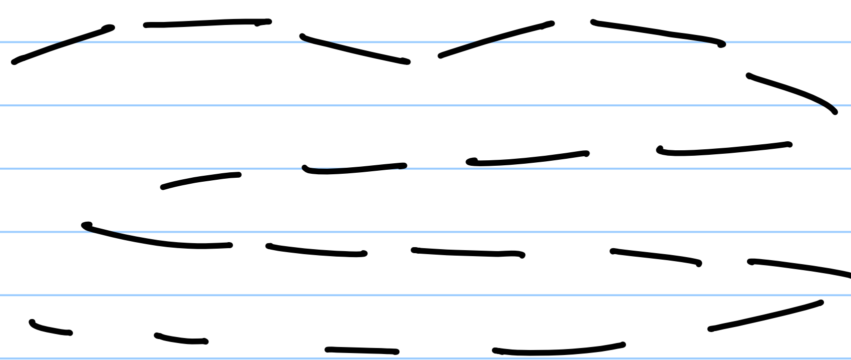
(1) Deformable Templates

Detection in cluttered background.

Dynamic Programming. (Belief Propagation)

→ Coughlan et al.

Discrete Formulation → Felzenszwalb. & Huttenlocher.



Set of **edglets**

positions $\{(y_a, \theta_a) : a = 1 \text{ to } M\}$

Prior Probability $z_a = (y_a, \theta_a)$

Distribution Prior. $\prod_{a,b} \psi_{ab}(z_a, z_b)$

(2) Likelihood term

Image $\rightarrow$ detect edges

$$\{(x_i, \gamma_i) : i = 1 \text{ to } N\}$$

$$\xi_i = (x_i, \gamma_i)$$

likelihood

$$P(\{(x_i,\gamma_i)\} \mid (y_a, \theta_a)\}, \{V_{ai}\})$$

$$= \prod_{i,a} \{\Psi(x_i, \gamma_i, y_a, \theta_a)\}^{V_{ai}}$$

$\{V_{ai}\}$ correspondence variables.
(as in previous lecture)

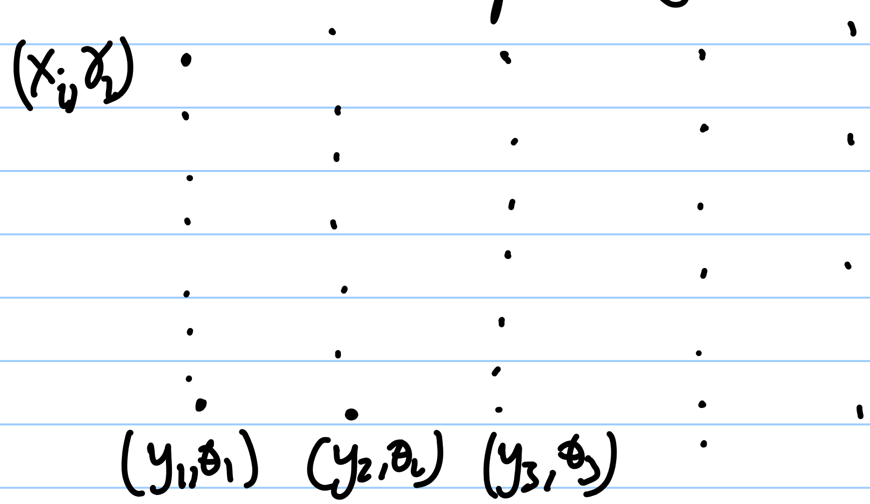
Again. use a dummy node $i = 0$
so that $V_{a0} = 0$ means
that (y_a, θ_a) is unmatched.

Simplify use the potential Ψ
to ensure that points y_a must be
matched directly to some x_i i.e. $y_a = x_i$ for some i
penalty on difference between θ_a & γ_i

(3) This spatial constraint ensures that each y_a can take $N+1$ possible values — the N image edglets, or the junk node.

If the prior is tree-like, then we can detect the object by using dynamic programming.

$O(N^2 M)$ complexity.



Coughlan's paper was rejected twice by CVPR. Reviewers thought that DP required initialization.

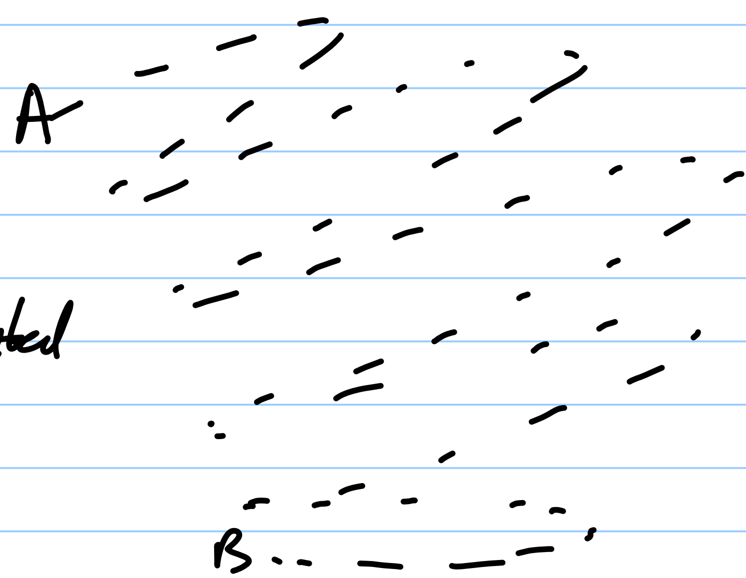
(4) In practice $O(N^2M)$ is still too large.

Use pruning heuristics

→ use neighbourhood relationships.

(<u>e.g.</u> if y_a is matched to x_i then y_{a+1} can only match a point close to x_i)

Applied to detecting hands in <u>images</u>:

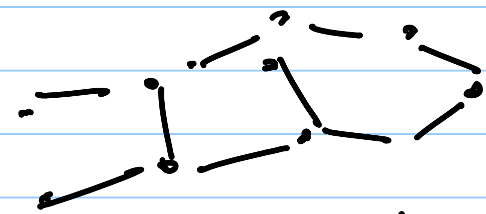


This was finally accepted by CVPR.

<u>Algorithm</u> → slightly unintuitive, seach starts at one end on the hand (A) and proceeds to the other end (B)

(5) What if the deformable template model has closed loops.

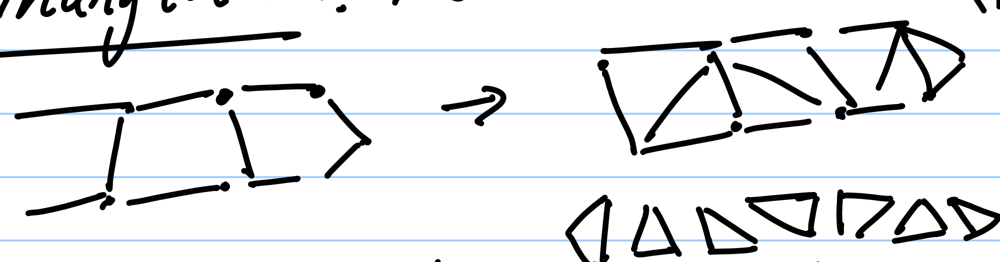
→ e.g.



(1) Belief Propagation can be applied.
This has been proposed Coughlan, Huttenlocher & Felzenszwalb.

Performance is reasonable, but heuristics still needed to make this effective

(2) Triangulation. → Junction Trees. (Amit Felzenszwalb)

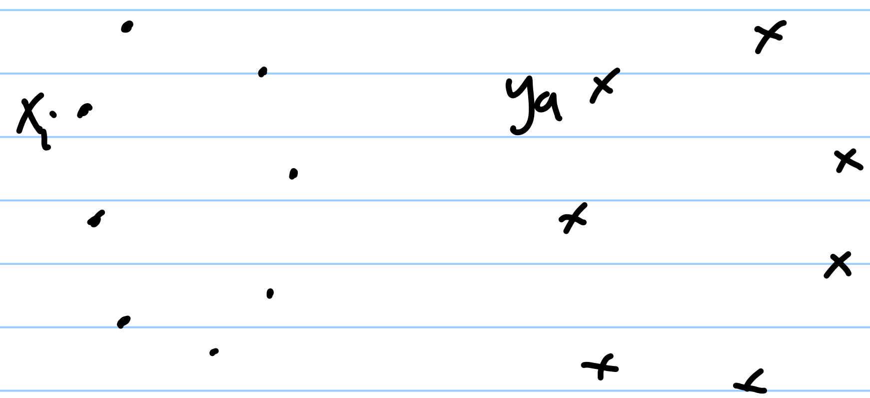


A tree in triangles.
Dynamic Programming applied to triangle variables

(6) Shape Matching

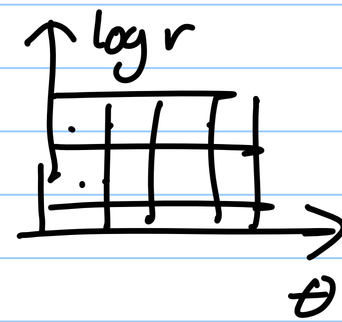
shape context.

individual points are ambiguous
→ but local configurations are not.



For each point
→ compute the **shape context**

~~the~~ $\psi_i(x)$ → histogram



Define a measure of similarity $M(\psi_i(\underline{x}), \psi_a(\underline{y}))$

Matching criteria $\sum_{i,a} V_{ia} M(\psi_i(\underline{x}), \psi_a(\underline{y}))$

(7) $$E[V] = \sum_{i,a} V_{ai} M(\Psi_i(\underline{x}), \Psi_a(\underline{y}))$$

<u>Constraint:</u> $\sum_a V_{ai} = 1, \quad \forall i = 1 \text{ to } M$

<u>1-1</u> $\sum_i V_{ai} = 1, \quad \forall a = 1 \text{ to } M.$

This is the linear assignment problem.

<u>Many</u> algorithm can solve it efficiently

Hungarian
Auction
Sinkhorn . . . .

<u>This</u> approach can be extended to allow for geometric transformation.

<u>Similary</u> to EM. algorithm

$$E[V, A] = \sum_{ia} V_{ai} M(\Psi_i(\underline{x}), \Psi_a(\underline{\underline{A}}\, \underline{y}))$$

alternate. Initialize: $\underline{\underline{A}} = \underline{\underline{I}}$ identity.

<u>Minimize</u> w.r.t. $\{V_{ai}\}$ & $\underline{\underline{A}}$ alternately.

(8) <u>Attempt</u> to combine shape Context with generative models (Tu et al).

<u>Define</u> a generative model (last lecture).

<u>Problem</u> with this generative model.

→ scale, rotation are unknown.

→ Algorithm will get stuck in local minima.

<u>Define</u> a discriminative model using shape context and other features.

These features can be used to estimate properties that are invariant of the rotation

local orientation → use relative orientation. (scale not posed)

(9) Recall that the EM algorithm for the generative model involves estimating the correspondences $\{V_{ia}\}$ and the deformation parameters.

Tu et al use the E-step from the discriminative model and the M-step from the generative model.