A Graphical Models based Image Segmentation Method

Rui Huang, Vladimir Pavlovic, and Dimitris N. Metaxas
Division of Computer and Information Sciences, Rutgers University
{ruihuang, vladimir, dnm}@cs.rutgers.edu

Comparison of MRFs and DMs:

	Pros	Cons
MRFs (Region-based methods)	robust to noise	rough edges; holes; hard to constrain the object shape and topology
DMs (Edge-based methods)	easy to incorporate the shape prior and the object topology	sensitive to noise; over-smoothed boundaries; good initialization needed

Previous work: Loosely coupled method

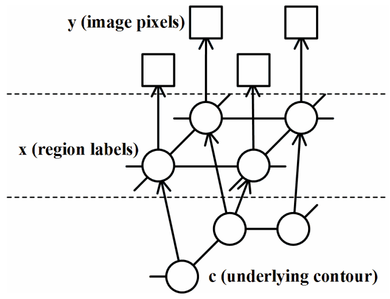
- MRFs are used to initially estimate the boundary of objects in noisy images.
- Balloons are then fitted to the estimated boundary. The result of the fitting is in turn used to update the MRF parameters.
- Final segmentation is achieved by iteratively integrating these processes.
- Take advantage of both MRFs and deformable models.
- Loose coupling may cause failure of deformable models if the initial estimation of the boundary by MRF is not closed, and it may also yield over-smoothed boundaries.

Our method: Tightly coupled method

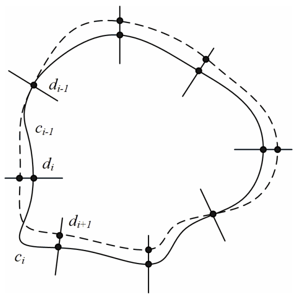
- Tight coupling of MRFs and deformable models using graphical models.
- Belief propagation (BP) in a banded area of the Bayesian multinets.

The new integrated model:

- A graphical model constructed by adding a new layer representing the underlying object contour to the traditional MRF model.
- The contour representation is based on segments, that is, each node c_i in our model is a segment of the actual contour. The edges between the **x** layer and the **c** layer are determined based on the distance between the image pixels and the contour segments: each pixel label node x_i is connected to its nearest contour segment.



Graphical representation of the integrated model



Contour representation

The segmentation problem:

$$(\mathbf{c},\mathbf{x},\theta)_{MAP} = \arg\max_{\mathbf{c},\mathbf{x},\theta} P(\mathbf{c},\mathbf{x},\theta \mid \mathbf{y}),$$

where $P(\mathbf{c},\mathbf{x},\theta \mid \mathbf{y}) \propto P(\mathbf{y} \mid \mathbf{x},\theta)P(\mathbf{x} \mid \mathbf{c},\theta)P(\mathbf{c} \mid \theta)P(\theta)$

$$P(\mathbf{y} \mid \mathbf{x}) = \prod_i P(y_i \mid x_i) = \prod_i p_i(x_i) = \prod_i \frac{1}{\sqrt{2\pi\sigma_{x_i}^2}} \exp\left(-\frac{(y_i-\mu_{x_i})^2}{2\sigma_{x_i}^2}\right)$$

$$P(\mathbf{x} \mid \mathbf{c}) = \prod_{(i,j)} P(x_i,x_j) \prod_i P(x_i \mid c_{i'}) = \prod_{(i,j)} p_{ij}(x_i,x_j) \prod_i p_{ii'}(x_i \mid c_{i'})$$

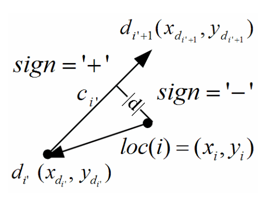
$$= \frac{1}{Z_x} \prod_{(i,j)} \exp\left(\frac{\delta(x_i - x_j)}{\sigma^2}\right) \prod_i p_{ii'}(x_i \mid c_{i'}),$$

where $i' = \arg\min_j dist(i, c_j)$

$$p_{ii'}(x_i = inside \mid c_{i'}) = \frac{1}{1+\exp(-dist(i,c_{i'}))}$$

$$p_{ii'}(x_i = outside \mid c_{i'}) = 1 - p_{ii'}(x_i = inside \mid c_{i'})$$

$$dist(i,c_{i'}) = (d_{i'} - loc(i)) \times (d_{i'+1} - d_{i'}) / |d_{i'+1} - d_{i'}|$$



$$P(\mathbf{c}) = \prod_{i'} [P(c_{i'-1}, c_{i'})P(c_{i'})] = \frac{1}{Z_c} \prod_{i'} [p_{i'-1,i'}(c_{i'-1}, c_{i'}) p_{i'}(c_{i'})],$$

where $p_{i'-1,i'}(c_{i'-1}, c_{i'}) = \exp\left(-\omega_1 |d_{i'-1} - d_{i'+1}|^2 / 4h^2 - \omega_2 |d_{i'-1} + d_{i'+1} - 2d_{i'}|^2 / h^4\right)$

$$p_{i'}(c_{i'}) = [p_{i'}^1, \ldots, p_{i'}^k]$$

Parameter estimation

$$\mu_l = \frac{\sum_{i \in bandB} belief(x_i = l) y_i}{\sum_{i \in bandB} belief(x_i = l)}; \sigma_l^2 = \frac{\sum_{i \in bandB} belief(x_i = l)(y_i - \mu_l)^2}{\sum_{i \in bandB} belief(x_i = l)}$$

Algorithm description

Initialize contour **c**
while (error > maxError) {
1. Calculate a band area B around **c**, Perform remaining steps inside B;
2. Build links between the **x** layer and the **c** layer;
3. Calculate the discretized states at each contour node along its normal;
4. Estimate the MAP solution $(\mathbf{c}, \mathbf{x})_{\mathrm{MAP}}$ using BP with some schedule S;
5. Update model parameters and contour position θ_{MAP}, d_{MAP};
}

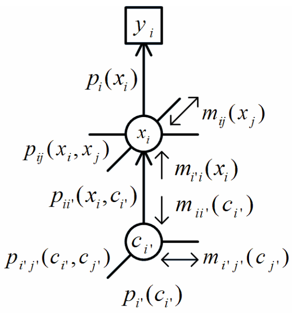
Belief Propagation

Some of the message passing rules

$$m_{ij}(x_j) = \max_{x_i}\left[p_i(x_i) \prod_{k \in \aleph(i) \setminus j} m_{ki}(x_i) m_{i'i}(x_i) p_{ij}(x_i, x_j)\right]$$

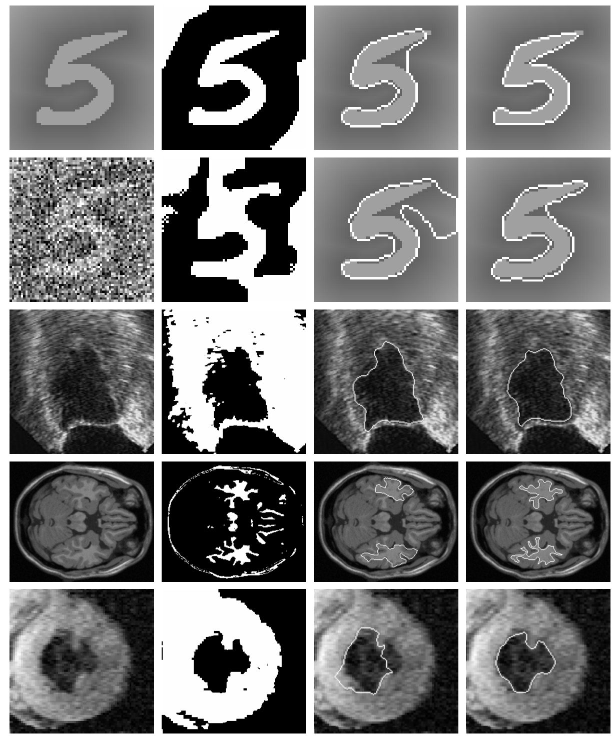
$$m_{i'i}(x_i) = \max_{c_{i'}}\left[p_{i'}(c_{i'}) \prod_{k' \in \aleph(i')} m_{k'i'}(c_{i'}) \prod_{k \in \aleph'(i') \setminus i} m_{ki'}(c_{i'}) p_{ii'}(x_i, c_{i'})\right]$$

$$b_i(x_i) = Z_i p_i(x_i) \prod_{k \in \aleph(i)} m_{ki}(x_i) m_{i'i}(x_i)$$



Experimental results:

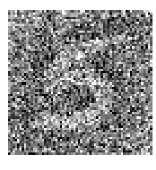
Original image | MRF | DM | DMRF



Experiments with different message-passing schedules:

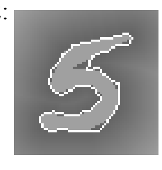
S1: update each message once at each iteration
S2: update messages in one layer for several times (or until convergence) then pass messages to the other layer, and so on.

Test:



S1:

S2:



Loglikelihood of the whole model vs. iterations: