V. Sparse model matching - global methods

V.1 Introduction

- Sparse models matched directly to data, no initialization required.
- Data models - point processes calculated using binary local features.
- Local features - photometric and geometric invariant - estimated from training examples in a robust way.
- Each landmark in the object has characteristic local topography. See figure V.1 (left). Level curves around left ventricle in an axial MRI image. Characterize topography with photometric invariant local feature.
- Impossible to find feature which *only* flags correct location. See level curves in a neighborhood of a sulcus figure V.1 (right).
- Find feature with *low density* of hits away from the correct location on the object.
- One local feature insufficient to detect object: many false positives, minimal instantiation information - use a collection of points associated with a set of points Z on the reference grid R.

Figure V.1: Left: Contour plot of neighborhood of frontal horn. Right: Contour plot of neighborhood of a sulcus

- $Z = \{z_1, \ldots, z_n\}$ relatively small - a small number of landmark points on the object images registered to the reference grid.
- Instantiation parameters given directly in terms of constraints on the relative locations of the n points - **admissible configurations** of the n landmarks.

  The set of constraints is called $\Theta \subset L^n$.

  For $\theta \in \Theta$, θ_i is matched to landmark $z_i \in Z$.

  $\Theta \subset L^n$.

- Develop efficient matching procedure for detecting admissible configurations.
- The local operators should be invariant within the range of instantiations Θ.
- See Amit & Kong (1996), Amit (1997), Amit et al. (1998), Amit & Geman (1998).
- For a very similar approach in spirit, see Burl et al. (1995),Cootes & Taylor (1996).

  They use linear filters for local features which is problematic from the point of view of photometric and scale invariance, estimation, as well as computation.

  Their matching procedures are more intensive computationally.

V.2 The model

V.2.1 Data model

- Binary local features $X_i, i = 1 \ldots, n$ defined to detect the object landmarks associated with $Z = \{z_1, \ldots, z_n\}$.

  Data transform: $FI(x) = [X_1, \ldots, X_n]$.

- Detection of the object involves identifying admissible configurations $\theta \in \Theta \subset L^n$ of instances of the n features, i.e. $X_i(\theta_i) = 1, i = 1, \ldots, n$.

- Invariance of the local features: $P(X_i(\theta_i) = 1|\theta) = p_o$ does not depend on θ.

  This implies certain limitations on Θ.

- For generic images on the image lattice not containing the object assume that $P(X_i(x) = 1) = p_b << 1$, for any $x \in L$.

- Given no object in the image $X_i(x), i = 1, \ldots, n, x \in L$ are assumed independent binary variables.

  Given an object at θ the $X_i(x), i = 1, \ldots, n, x \in L$ are assumed independent binary variables.

- Joint likelihood

$$P(X_i(x), x \in L, i = 1, \ldots, n|\theta) = \prod_{i=1}^{n} \Big[p_o^{X_i(\theta_i)} (1 - p_o)^{(1-X_i(\theta_i))} \prod_{x \neq \theta_i} p_b^{X_i(x)} (1 - p_b)^{(1-X_i(x))} \Big]. \quad \text{(V.1)}$$

- p_b small. Given no instance of object - X_i homogenous Poisson with intensity p_b. Multiply and divide each interior product by the missing term for the point θ_i.

$$
\begin{aligned}
P(X_i(x), x \in L, i = 1, \ldots, n|\theta) = \\
\left(\frac{p_o}{p_b}\right)^{n_y} \quad \left(\frac{(1-p_o)}{(1-p_b)}\right)^{n-n_y} \prod_{i=1}^{n} \frac{e^{-\lambda_b}\lambda_b^{n_i}}{n_i!} \\
\propto \left(\frac{p_o(1-p_b)}{p_b(1-p_o)}\right)^{n_y}
\end{aligned}
\tag{V.2}
$$

n_y - number of i's for which $X_i(\theta_i) = 1$, and $\lambda_b = p_b|L|$.

n_i - total number of hits of X_i in image.

V.2.2 The prior - hard constraints

- **Translation invariance:**

  $\theta = (\theta_1, \ldots, \theta_n) \in \Theta \leftrightarrow \theta + x \equiv (\theta_1 + x, \ldots, \theta_n + x) \in \Theta$

  for any $x \in L$.

- For $\theta \in \Theta$ let $x_c(\theta) = \frac{1}{n}\sum_{i=1}^{n}\theta_i$. $x_c(\theta)$ - **virtual center**.

  $\Theta^0 = \{(\theta - x_c(\theta)) : \theta \in \Theta\} = \Theta/L$.

  $\Theta = L + \Theta^0 = \{x + \theta; x \in L, \theta \in \Theta^0\}$.

  Examples of Θ^0

  1. Independent variations of the landmarks relative to the center.

     $$\Theta^0 = \{(\theta_1, \ldots \theta_n) : x_i \in B_i, i = 1, \ldots, n\},$$

     with B_i subsets of R.

  2. Linear and local nonlinear transformations:

     $$\Theta^0 = \{(\theta_1, \ldots, \theta_n) : \theta_i = Az_i + C_i, i = 1, \ldots, n,\ A \in \mathcal{A}\}, \tag{V.3}$$

     with $\mathcal{A}$ subset of the invertible 2d linear transformations containing the identity map, and $0 \in C_i \subset R$.

- Also possible to put a prior $P(\theta)$ on Θ^0.

V.2.3 The posterior

- Using equation V.2,

  Posterior:

$$P(\theta|X_i(x), x \in L, i = 1, \ldots) \propto \left(\frac{p_o(1-p_b)}{p_b(1-p_o)} \right)^{n_y} P(\theta) \tag{V.4}$$

- Possible to write similar form for the posterior in general case where $P(X_i(\theta_i) = 1|\theta)$ depends on i.

- Optimizing posterior - finding admissible configurations or sub-configurations.

- Two approaches for optimizing the posterior: section V.4 and section VI.

  1. At most one instance of the object is present. $p_o = 1$ - only $P(\theta)$ needs to be optimized over Θ.

     The points $\theta_i, i = 1, \ldots, n$ are pointed out on a list of training images by the user. No explicit use is made of the reference grid.

     A local feature is chosen for each $i = 1, \ldots, n$ and set so that p_o is very close to 1.

     Optimization computationally feasible when $P(\theta)$ is assumed to have a decomposable form (see section V.4), and local features are low density in background.

  2. Several instances of the object can be present. They may be partially occluded.

     Assume $p_o \sim .5$. Find local maxima of full posterior: values of θ for which n_y is above some threshold τ.

     The computation is done in stages using a variation on the Hough transform.

Only 3 points are marked by the user on training data. These 3 points are matched to three fixed points on the reference grid. The training data then registered to the reference grid via the associated affine map and stable local operators are found, through systematic training.

V.3 The local features

Two approaches to defining local features: stable on object and low density in background.

V.3.1 Comparison counts - masks

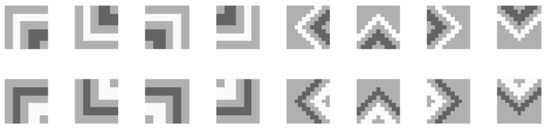

Figure V.2: 16 masks used for creating the graph models. White means intensity greater than center value, dark means intensity less than middle value, grey means ignored.

- Choose $m \times n$ array A of 1-s, -1-s and 0-s.
- Calculate *sign* of difference between the intensity at any pixel x, and the intensity at each of the pixels in the $m \times n$ neighborhood, into an array $A(x)$ of 1-s and -1-s.
- If the percentage of matched 1-s and -1-s between A and $A(x)$ is above a prescribed threshold the pixel x is considered a candidate for the corresponding landmark - The 0-s region in A represents a region where the results of the $A(x)$ are ignored.

- Figure V.2 has 16 masks used in creating models.

  White - intensity greater than center value.

  Black - intensity less than center value.

  Grey - ignored.

- Loosely speaking these 16 masks identify ‘turns’ of level curves of the image pointing in eight different orientations, (see figure V.2)

- Many possible variation on this theme.

- These local features are robust to rather significant variations in pixel intensities, smooth deformations and a certain range of scales and small rotations.

- Figure V.3, bottom panel, shows 11×11 neighborhoods of the tip of the frontal horn of the lateral ventricle in four images. All four were activated by the first operator in figure V.3, middle panel. An indication that p_o is close to 1.

Training

- User points to the n landmarks in each training image.

- The best matching mask from the pool of 16 masks of figure V.2 is chosen for each point.

  Or user determines which mask to use.

- The matching threshold for each mask is taken as the lowest match observed at the correct location of the landmark over the training images.

V.3.2 Edge groupings

The edges

Define eight oriented edge features.

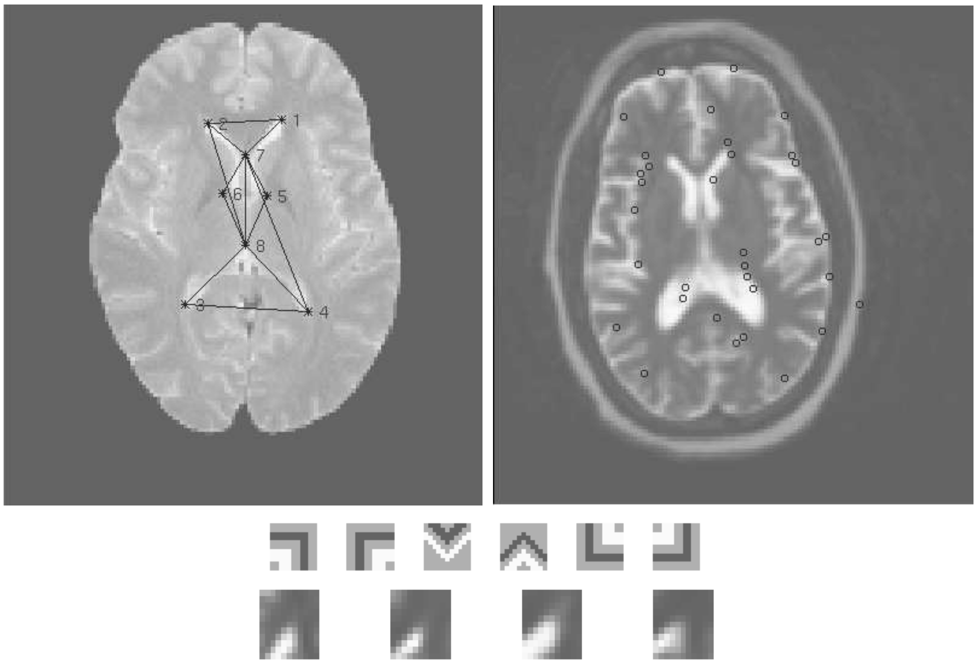

Figure V.3: Top Left: Model graph and six landmarks determined on a sample image. Top right: Candidates for the first landmark found in a data image. Middle: The six masks used for identifying candidates for the model landmarks. Bottom: Neighborhoods of tip of upper right ventricle in four images

- z, y adjacent pixels. $v = z - y$, w - 90 degree rotation of v.

  $z_1 = z + w, z_2 = z - w, z_3 = z + v$ and $y_1 = y + w, y_2 = y - w, y_3 = y - v$.

  An edge is present at z if

$$|I(z) - I(y)| > \max_{i=1,2,3} \left(\max\left(|I(z) - I(z_i)|, |I(y) - I(y_i)|\right)\right).$$

- Orientation of the edge is v if $I(z) > I(y)$ and $-v$ otherwise.

  Eight edge types.

  Often only use four: two horizontal and two vertical.

- An edge is some type of local maximum of the modulus of the gradient.

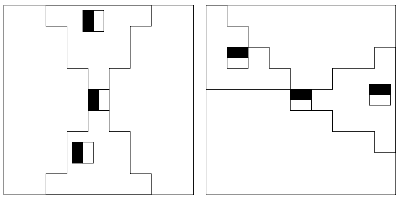

Figure V.4: Two examples of local edge groupings with $N_{edges} = 2$ edges in addition to the center one, each allowed to lie anywhere in a subregion of size $N_{pixels} \approx 10$.

- Invariance to significant gray scale transformations, smooth deformations and large ranges of scale and rotation are inherent in the definition of these features.

Local features - flexible groupings of edges

- Local feature - "central edge" and N_{edges} other edge types, constrained to lie in specific subregions $W \in \mathcal{W}$ in neighborhood of center.
- Local feature inherits the location of the central edge.
- Sizes of the subregions W are all the same and denoted by N_{pixels}. See figure V.4.
- Disjunction - allowing the N_{edges} edges to float in their respective subregions - provides geometric invariance.

Training

- Get edge features in each training image.
- User identifies three chosen locations p_1, p_2, p_3 on the reference grid R. User marks three landmarks on each training image, to be mapped to p_1, p_2, p_3.

  Note: registration using more points is possible but there is no clear choice of interpolating map.

- The two triples determine affine map for registering edge locations of each training image to reference grid. Linear variability is factored out.
- Choose $\alpha < 1$.

Algorithm:

Detect edges in all training images. Register edge maps into reference grid.

1. Set feature counter $c = 0$. Loop over disjoint $b \times b$ (say $b = 3$ or $b = 5$), boxes on the reference grid. For each box B:
   (a) For each possible combination (e, e', W), where $e, e' = 1, \ldots, 8$, $W \in \mathcal{W}$, count number of training images for which e, the central edge, is at some $x \in B$ and $e' \in x + W$.
   (e_0, e_1, W_1) - combination with highest count.
   $\mathcal{L}_1$ - training images with an instance of (e_0, e_1, W_1) in B.
   For $d \in \mathcal{L}_1$, $x_{d,t} \in B, t = 1, \ldots, n_{d,1}$ locations of e_0 for which combination was found. Set $j = 2$.
   (b) For each possible combination e, W, $e = 1, \ldots, 8$, $W \in \mathcal{W}$,
   Count number of data images $d \in \mathcal{L}_{j-1}$ for which there is an edge $e \in x_{d,t} + W$, for some $t = 1, \ldots, n_{d,j-1}$.
   (e_j, W_j) combination with highest count.
   $\mathcal{L}_j \subset \mathcal{L}_{j-1}$ - training images which have an instance of the combination (e_j, W_j).
   For $d \in \mathcal{L}_j$, $x_{d,t}, t = 1, \ldots, n_{d,j}$ instances of e_0 for which (e_j, W_j) was found.
   (c) $j \leftarrow j + 1$. If $j < N_{edges}$ goto (b).
2. If $|\mathcal{L}_{N_{edges}}|/|\mathcal{L}| > \alpha$,

$$X_c = (e_0, e_1, W_1, \ldots, e_{N_{edges}}, W_{N_{edges}}),$$

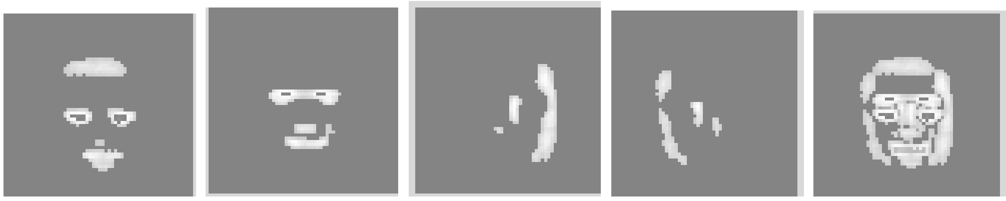

Figure V.5: The concentrations of the four edge types on registered faces.

at z_c - center of B.

All images in $\mathcal{L}_{N_{edges}}$ have an instance of e_0 at some $x \in B$ and an instance of e_k in $x + W_k$, for each $k = 1, \ldots, N_{edges}$.

$c \leftarrow c + 1$.

3. Move to the next box and goto 1.
4. The local features chosen are not necessarily at the points used for registering the images to the reference grid.

- If more than n local features found choose a random sample of size n.

Examples

- Figure V.5 - high frequency locations of the registered edges for faces. For each of four edge types a pixel is on if more than .7 of the faces had an edge of that type in the 3x3 neighborhood of that pixel.
- All detection examples shown in section VI use $N_{edges} = 3$, 16 overlapping wedges of 45 degrees as in figure V.4. N_{pixels} is then about 10, $\alpha = .5$ and $n \sim 20$.
- Figure V.6 - graphical representation of 6 of 20 local features at their model locations. The three blue dots are p_1, p_2, p_3 used for registering the training images. The user marked the two eyes and mouth on each training image.

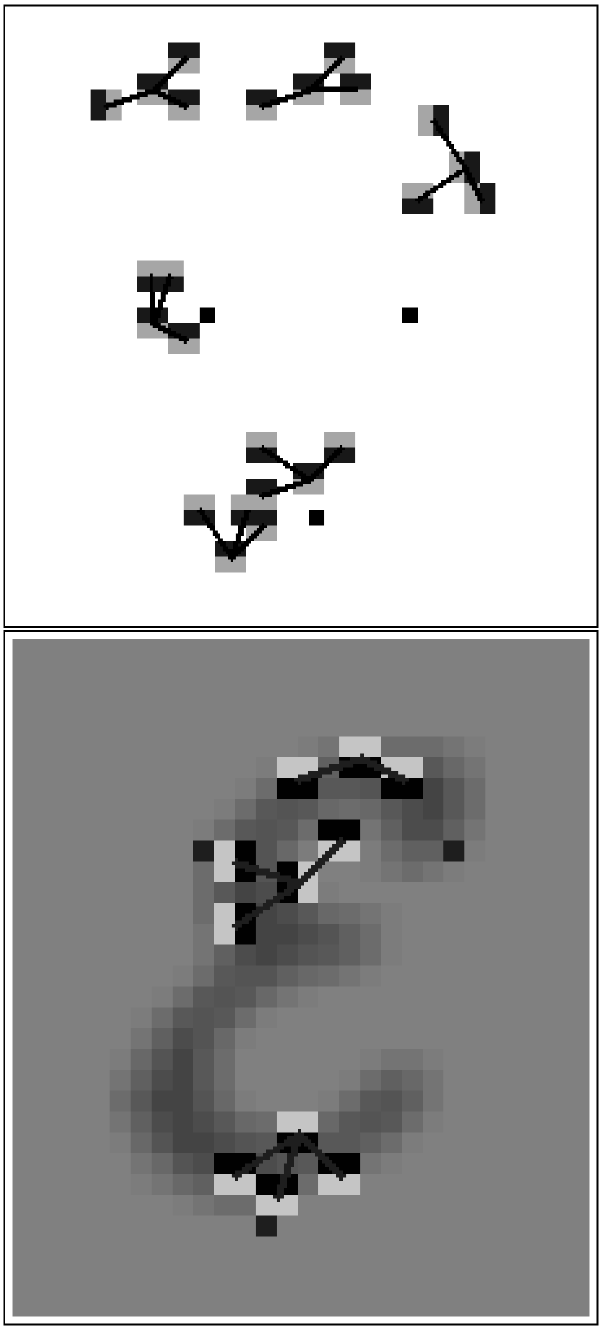

Figure V.6: The graphs representing six of the local features identified for faces, top, and three of those for deformed $\mathcal{E}$'s, bottom. Each local feature has a center edge and three other edges around it. The two adjacent rectangles describe the orientation and polarity of each edge.

Each local feature is represented with the four edges that define it. The two adjacent rectangles describe the orientation and polarity of each edge. Recall that each of the three edges can have varying locations relative to the center.

Local features detected for randomly perturbed $\mathcal{E}$'s shown below.

The upper left hand feature for example has three horizontal edges arranged more or less horizontally and a vertical edge on their left. This captures the curve of the hairline on the left part of the face.

The feature directly below that has two horizontal edges with dark part on the bottom above two horizontal edges with dark part on the top. This captures the area around the eye, which is typically darker than its surrounding.

- Features do not only capture contours, but other topographical structures as well.

V.4 Instance detection with decomposable constraints

- Our knowledge of the ways the object can appear constrains the possible configurations of the instances of the landmarks.

- Assume p_o is is very close to 1 - false negatives at local level are rare

- At most one instance of object expected in image.

- Define Θ and the prior simultaneously as follows:

  Assume Ψ is a function on L^n of the form

$$\Psi(\theta) = \sum_{C \in \mathcal{C}} \psi_C(\theta_i : i \in C), \tag{V.5}$$

  where $\mathcal{C}$ is some collection of triples (not necessarily all) from the set $\{1, \ldots, n\}$, and $\theta \in L^n$.

The collection of triples can be represented through a graph. The nodes represent $\theta_1, \ldots, \theta_n$ and an edge exists between any two points belonging the same triple. (See figure V.3).

- ψ_C

  Invariant to translation and to certain ranges of scale and rotation.

  Contains hard constraints on the extent of departure of the triangle $\theta_i, i \in C$ from the triangle $z_i, i \in C$. Inadmissible departures give the value $-\infty$.

  $\Theta = \{\theta \in L^n; \psi_C(\theta_i, i \in C) > -\infty, \text{ for all } C \in \mathcal{C}\}$.

  In Θ define penalties on the extent of departure from the triangle $z_i, i \in C$.

- Set
$$P(\theta) = C \exp[\Psi(\theta)].$$

- Brute force: Evaluate Ψ for each n-tuple θ of instances of the n local features in the image - where θ_i is such that $X_i(\theta_i) = 1$ - pick the n-tuple with lowest score. Typically this is not computationally feasible.

- When $\mathcal{C}$ has a decomposable form see Amit & Kong (1996) minimization can be efficiently done through dynamic programming.

V.4.1 Decomposability of $\mathcal{C}$

- There are exactly $n-2$ triples in $\mathcal{C}$. There exists an ordering $C_1, \ldots, C_{n-2}$ and a reordering $\theta_{(i)} : (i) \in C_i$ of the index set $1, \ldots, n$ such that

  1. $\theta_{(1)}$ is only a member of C_1.

     Remove (1) from the index list and C_1 from the triples list - then (2) belongs only to C_2

     The two other points in C_1 both belong to some other triple.

2. At stage $j-1$ index $(j-1)$ belongs only to the triple C_{j-1}. Remove $(j-1)$ and C_{j-1} - then (j) belongs only to C_j,

   The two other points in C_{j-1} both belong to some other triple.

3. This 'peeling' procedure continues until all that is left are $(n-2), (n-1), (n)$ which make up the triple C_{n-2}. For more details on decomposability see Amit & Kong (1996), Rose et al. (1976).

- Example: given n indices the simplest decomposable collection would have the triples $i, i+1, i+2$ for $i = 1, \ldots, n-2$. Second order linear Markov chain.

  The graph in figure V.3 is a decomposable collection of triples. A triple of points is in the collection if all three connecting edges are present in the graph. The numbers in the image correspond to the peeling order.

V.4.2 Dynamic programming

- Assume the indices and triples are now relabeled according to the 'peeling' order. $(i) = i$.

- 1 is the vertex in C_1 which is not in any other triple (the first to be eliminated), write

$$\begin{aligned}
\Psi(\theta_1, \ldots, \theta_n) &= \sum_{t=1}^{T} \psi_{C_t}(\theta_i; i \in C_t) \\
&= \psi_{C_1}(\theta_i; i \in C_1) + \sum_{t=2}^{T} \psi_{C_t}(\theta_i; i \in C_t) \\
&\stackrel{\text{def}}{=} \psi_{C_1}(\theta_i; i \in C_1) + \Lambda(\theta_2, \ldots, \theta_n). \qquad \text{(V.6)}
\end{aligned}$$

- $C_1 = \{1, a, b\}$. For any fixed pair (θ_a, θ_b) of instances of local features X_a and X_b, let $\theta_1^*[\theta_a, \theta_b]$ be the instance of local feature X_1 which minimizes $\psi_{C_1}(\theta_1, \theta_a, \theta_b)$.

- $\Lambda(\cdot)$ does not depend on θ_1 - the optimal match $(\theta_1^*, \ldots, \theta_n^*)$ satisfies $\theta_1^* = \theta_1^*[\theta_a^*, \theta_b^*]$.

- The computation is proportional to m^3 - m is an upper bound on the number of possible matches for each landmark.

- For each pair (θ_a, θ_b) the instance $\theta_1^*[\theta_a, \theta_b]$ is stored as well as $\psi_{C_1}(\theta_1^*[\theta_a, \theta_b], \theta_a, \theta_b)$, so that the amount of storage required is proportional to m^2.

- C_{u_1} (not necessarily the next one in the ordering) contains vertices a and b. Let $C_{u_1} = \{a, b, c\}$. Define

$$\begin{aligned}
\hat{\psi}_{C_{u_1}}(\theta_i; i \in C_{u_1}) &= \hat{\psi}_{C_{u_1}}(\theta_a, \theta_b, \theta_c) \\
&= \psi_{C_{u_1}}(\theta_a, \theta_b, \theta_c) + \psi_{C_1}(\theta_1^*[\theta_a, \theta_b], \theta_a, \theta_b), \\
\hat{\psi}_{C_t}(\theta_i; i \in C_t) &= \psi_{C_t}(\theta_i; i \in C_t) \ \text{ for } t = 2, \ldots T, t \neq u_1,
\end{aligned}$$

  and

$$\Phi(\theta_2, \ldots, \theta_n) = \sum_{t=2}^{T} \hat{\psi}_{C_t}(\theta_i; i \in C_t).$$

- $\psi_{C_1}(\theta_1^*[\theta_a, \theta_b], \theta_a, \theta_b)$ is a function of θ_a and θ_b only, and hence $\hat{\psi}_{C_{u_1}}(\cdot)$ is a function of $(\theta_i, i \in C_{u_1})$.

- $\Phi(\theta_2^*, \ldots, \theta_n^*) = \Psi(\theta_1^*, \ldots, \theta_n^*)$ and $(\theta_2^*, \ldots, \theta_n^*)$ is the minimizer of $\Phi(\cdot)$.

- Original problem reduced by one index.

- *Peeling* the vertices one at a time in increasing order the problem can be solved when only the last 3 vertices $n-2, n-1, n$, are left.

- When $\theta_{n-2}^*, \theta_{n-1}^*, \theta_n^*$ are found determine θ_{n-3}^*, ..., θ_1^* in reverse order:

  At step j set θ_j^* as $\theta_j^*[\theta_k^*, \theta_l^*]$ (which is already stored) where k, l are the other two elements of C_j. The indices k, l are necessarily greater than j so that θ_k^* and θ_l^* have already been determined.

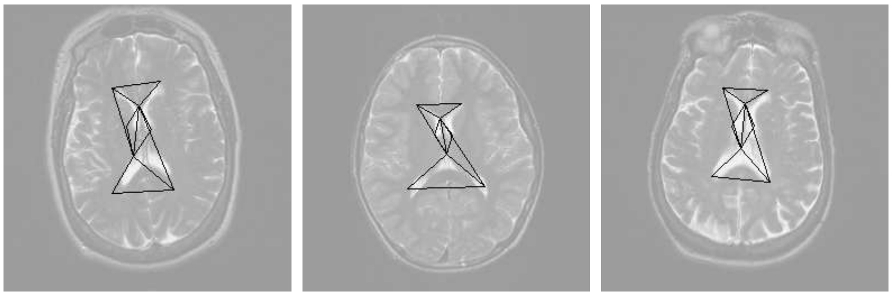

Figure V.7: The matches of the model in figure V.3 to three axial MRI images

V.4.3 Triangle functions ϕ

- Soft constraints deviations of candidate triangle $\theta_{i_1}, \theta_{i_2}, \theta_{i_3}$ from model triangle $z_{i_1}, z_{i_2}, z_{i_3}$.

  1. Sum of the squares of the angle differences of between the two triangles.
  2. Or using complex coordinates:

$$\left|\frac{z_{i_3} - z_{i_1}}{z_{i_2} - z_{i_1}} - \frac{\theta_{i_3} - \theta_{i_1}}{\theta_{i_2} - \theta_{i_1}}\right|.$$

- Penalties are *fully* translation and scale and rotation invariant. Local operators themselves are not.

- Orientation and scale assumed known in some range - add hard constraints to cost functions, limiting rotation and scaling.

- For 3 examples of the match of the model in figure V.3 see figure V.3.

V.5 Discussion

- This model is very similar to the 1-d global model. Differences:

  Local features much lower density. Contain much more information.

  Computation more efficient.

Geometric relations defined through general decomposable graph, not just second order Markov chain. Global relations can be controlled, not just local ones.

- Assumption that $p_o = 1$ makes algorithm sensitive.

- Photometric invariance dealt with directly through definition of local features through intensity comparisons.

- Geometric invariance dealt with explicitely through the functions ϕ_C. These could be different for different triangles expressing various hard and soft constraints specific to that triple of points.