

## Part-Based Probabilistic Point Matching

Graham McNeill, Sethu Vijayakumar

Institute of Perception, Action and Behavior

School of Informatics, University of Edinburgh, Edinburgh, UK. EH9 3JZ

Email: g.j.mcneill-2@sms.ed.ac.uk, sethu.vijayakumar@ed.ac.uk

### Abstract

We present a probabilistic technique for matching part-based shapes. Shapes are represented by unlabeled point sets, so discontinuous boundaries and non-boundary points do not pose a problem. Occlusions and significant dissimilarities between shapes are explained by a 'background model' and hence, their impact on the overall match is limited. Using a part-based model, we can successfully match shapes which differ as a result of independent part transformations – a form of variation common amongst real objects of the same class. A greedy algorithm that learns the parts sequentially can be used to estimate the number of parts and the initial parameters for the main algorithm.

### 1 Introduction

Shape-based object recognition is a key problem in computer vision and content-based image retrieval (CBIR). Over the last decade, numerous shape matching algorithms have been proposed leading to a dramatic improvement in performance on benchmark shape retrieval tests. However, there are two important limitations common to many algorithms: Firstly, they operate on continuous shape boundaries (*i.e.* the ordering of the boundary points matters) and have no mechanism for handling occlusion, absent features, non-boundary points and clutter. Secondly, they struggle to handle shapes that display significant part-based variation. The first problem becomes important if working with real images where often one can only extract incomplete, noisy shape information. The second is important because many common objects (natural and man made) have a part structure – single part objects can be seen as a special case.

Techniques that match unordered point sets [1, 4] are appealing since they do not require ordered boundary information and can work with non-boundary points. The soft correspondence method described in [2], and the probabilistic methods in [7] and [5] can handle outliers, occlusions and clutter, but have not been extended to handle part-based shapes explicitly. We address this issue by first describing

a simple point matching model (Sec. 2), and then extending it so that the most likely part structure and match are found simultaneously (Sec. 3). A sequential approach for estimating the number of parts and initial parameter values is described in Sec. 5.

### 2 Probabilistic Point Matching (PPM)

We start by describing the probabilistic point matching (PPM) algorithm used in [5]. Each shape is represented by an arbitrary number of points; these need not belong to the shape boundary and the ordering of the points is irrelevant. Given two such shapes,  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M)^T \in \mathbb{R}^{M \times 2}$  and  $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N)^T \in \mathbb{R}^{N \times 2}$  (generally  $M \neq N$ ), our task is to compute the correspondence and match between  $\mathbf{X}$  and  $\mathbf{Y}$ . We assume that the  $\mathbf{y}_n$  are observations from a mixture model:

$$p(\mathbf{y}_n) = p(\mathbf{y}_n|v_1)p(v_1) + p(\mathbf{y}_n|v_0)p(v_0) \quad (1)$$

where

$$p(\mathbf{y}_n|v_1) = \sum_{m=1}^M p(\mathbf{y}_n|m, v_1)p(m), \quad (2)$$

$$\mathbf{y}_n|v_0 \sim \text{Uniform}. \quad (3)$$

The mixture component  $v_0$  represents the 'background model' which ensures that all data points are explained to some extent, and hence, robustifies the model against outliers. The 'foreground' component  $v_1$  is the interesting part. The distribution  $p(\mathbf{y}_n|v_1)$  is itself a mixture model (eq.(2)), where the  $p(\mathbf{y}_n|m, v_1)$  are assumed to be Gaussian. The center of each Gaussian depends on  $\mathbf{x}_m$ , but unlike standard Gaussian mixture models (GMMs), the movement of the centers is controlled by a single set of transformation parameters  $s, \Gamma$  and  $\mathbf{c}$ :

$$\mathbf{y}_n|m, v_1 \sim \mathcal{N}(s\Gamma\mathbf{x}_m + \mathbf{c}, \sigma^2\mathbf{I}), \quad (4)$$

where  $s$  is a scale parameter,  $\mathbf{c} \in \mathbb{R}^2$  is a translation vector and  $\Gamma$  is a 2D rotation matrix. For now, we assume that  $m$  follows a discrete uniform distribution over  $\{1, \dots, M\}$ .

Maximum likelihood estimates (MLEs) of the unknown model parameters ( $(s, \Gamma, \mathbf{c})$  and  $p(v)$ ) are found using the expectation maximization (EM) algorithm as follows

**E-step:** Compute the *responsibilities* using the current parameter values and eq.(4):

$$p(m, v_1 | \mathbf{y}_n) = \frac{p(\mathbf{y}_n | m, v_1) p(m) p(v_1)}{p(\mathbf{y}_n)} \quad (5)$$

**M-step:** Update the parameters using the responsibilities:

$$p(v_1) = \frac{1}{N} \sum_n p(v_1 | \mathbf{y}_n), \quad p(v_0) = 1 - p(v_1) \quad (6)$$

$$(s, \Gamma, \mathbf{c}) = \arg \min_{s, \Gamma, \mathbf{c}} \sum_{m, n} p(m, v_1 | \mathbf{y}_n) \|\mathbf{y}_n - s\Gamma \mathbf{x}_m - \mathbf{c}\|^2 \quad (7)$$

where  $p(\mathbf{y}_n) = p(\mathbf{y}_n | v_0) p(v_0) + \sum_m p(\mathbf{y}_n | m, v_1) p(m) p(v_1)$ , and  $p(v_1 | \mathbf{y}_n) = \sum_m p(m, v_1 | \mathbf{y}_n)$ . Eq.(7) is a *weighted Procrustes matching problem* between two points sets, each of size  $N \times M$  – the importance of matching the pair  $(\mathbf{x}_m, \mathbf{y}_n)$  is given by  $p(m, v_1 | \mathbf{y}_n)$ . This least squares problem can be solved analytically using a similar approach to that used for unweighted Procrustes problems (see *e.g.* [3]). The uniform distribution  $p(\mathbf{y}_n | v_0)$  (eq.(3)) is set to  $U(\mathbf{y}_n) \equiv 0.001$ , and the variance  $\sigma^2$  (eq.(2)) is fixed at 0.05 in all examples.

Fig. 1 shows how PPM performs on matching problems involving occlusion, irregular sampling and *local dissimilarity* – a ‘feature’ of one shape is not present or is significantly different on the other shape.<sup>1</sup> The bold points are the  $\mathbf{x}_m$  that are collectively transformed to match the faint  $\mathbf{y}_n$ ; the final matches are scaled up for clarity. See that many of the  $\mathbf{y}_n$  have no counterparts in (a), whereas it is the  $\mathbf{x}_m$  that have no counterparts in (c). In all cases, the sampling frequency of corresponding sections is different, so there is no perfect point-to-point match. Fig. 2 (first two rows) shows that PPM struggles to match shapes that display part-based variation; this limitation is addressed in the next section.

### 3 Part-Based Probabilistic Point Matching (PB-PPM)

The PPM algorithm can be seen as a special instance of a part-based model: it has one background part,  $v_0$ , and one foreground part,  $v_1$ . We now consider models with one background part,  $v_0$ , and  $Q$  foreground parts,  $v_1, v_2, \dots, v_Q$ , where

$$\mathbf{y} | m, v_q \sim \mathcal{N}(s_q \Gamma_q \mathbf{x}_m + \mathbf{c}_q, \sigma^2 \mathbf{I}), \quad q = 1, \dots, Q. \quad (8)$$

Thus, the part label  $q$  indexes transformation parameters where a (foreground) part is implicitly defined as those points of  $\mathbf{X}$  that undergo the same transformations. Note that parts of  $\mathbf{X}$  are only defined in the context of a given  $\mathbf{Y}$  that  $\mathbf{X}$  is being matched to.

<sup>1</sup>The shapes in Fig. 1 and many of the other figures are from the MPEG-7 data set: <http://www.cis.temple.edu/~latecki/research.html#shape>

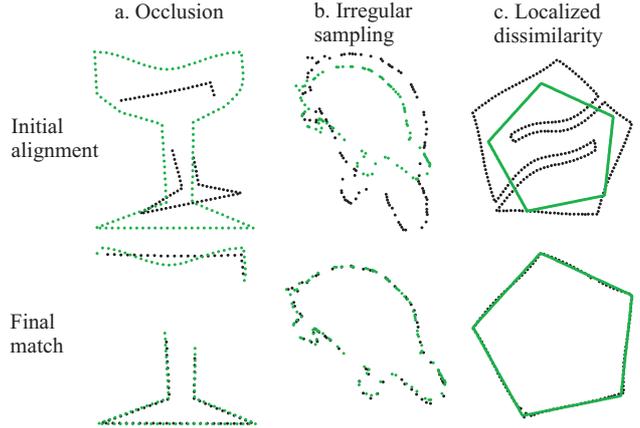


Figure 1. PPM – examples.

We could proceed by writing  $p(\mathbf{y}, m, v) = p(\mathbf{y} | m, v) p(m | v) p(v)$  and learning every entry of the  $p(m | v)$  matrix. However, this places no constraints on what a part should look like – for example, the points belonging to a single part could be spread across the shape. Intuitively, we expect that the correct correspondence and match is more likely to be found if the learnt parts have some perceptual meaning. Even when this is not the case, it seems reasonable that difficult correspondence problems may be solved by splitting shapes into spatially localized pieces. With this in mind, we look for foreground parts that are *spatially coherent* by replacing the uniform distribution of  $m$  with

$$p(m | v_q) = \frac{\exp\{-(\mathbf{x}_m - \boldsymbol{\mu}_q)^T \Sigma_q^{-1} (\mathbf{x}_m - \boldsymbol{\mu}_q) / 2\}}{\sum_m \exp\{-(\mathbf{x}_m - \boldsymbol{\mu}_q)^T \Sigma_q^{-1} (\mathbf{x}_m - \boldsymbol{\mu}_q) / 2\}}, \quad (9)$$

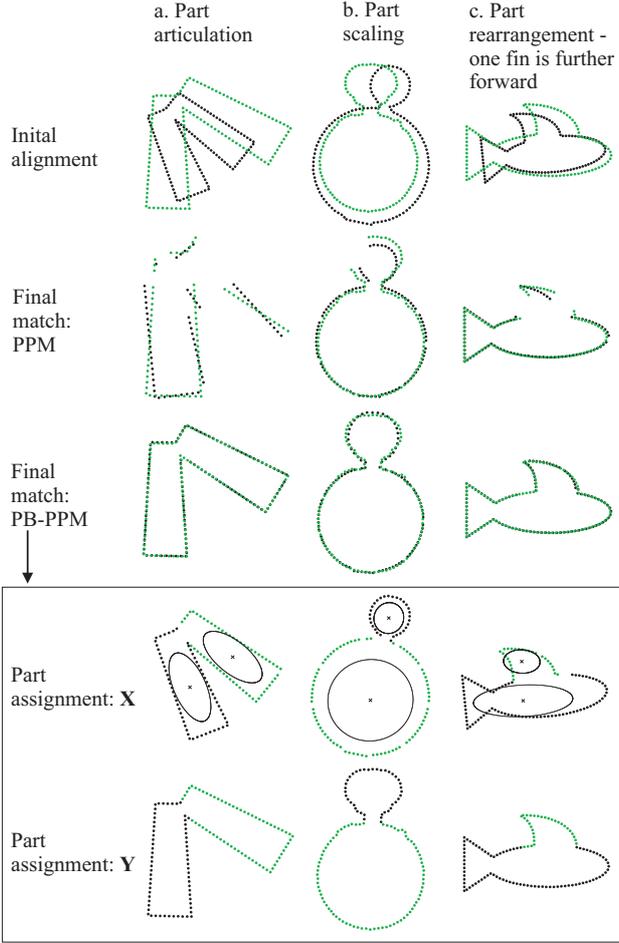
where  $\boldsymbol{\mu}_q \in \mathbb{R}^2$  is a mean vector and  $\Sigma_q \in \mathbb{R}^{2 \times 2}$  is a covariance matrix.<sup>2</sup> In words, we identify  $m \in \{1, \dots, M\}$  with the point  $\mathbf{x}_m$  that it indexes and assume that the  $\mathbf{x}_m$  follow a bivariate Gaussian distribution. Since  $m$  must take a value in  $\{1, \dots, M\}$ , the distribution is normalized using the points  $\mathbf{x}_1, \dots, \mathbf{x}_M$  only. This assumption means that the  $\mathbf{x}_k$  themselves are essentially generated by a GMM with  $Q$  components. However, this GMM is embedded in the larger model and maximizing the data likelihood will balance this GMM’s desire for coherent parts against the need for the parts and transformations to explain the actual data (the  $\mathbf{y}_n$ ).

The EM algorithm is similar to that used for PPM (eqs.(5-7)), but now we run over each of the  $Q$  foreground parts, and there are the additional parameters  $\boldsymbol{\mu}_q$  and  $\Sigma_q$  to estimate:

**E-step:** (compute responsibilities)

$$p(m, v_q | \mathbf{y}_n) = \frac{p(\mathbf{y}_n | m, v_q) p(m | v_q) p(v_q)}{p(\mathbf{y}_n)}$$

<sup>2</sup>The likelihood of a point  $\mathbf{y}_n$  given the background  $v_0$  remains uniform ( $p(\mathbf{y}_n | m, v_0) = p(\mathbf{y}_n | v_0)$ ) so  $p(m | v_0)$  is not important here.



**Figure 2. Part-based variation.**

**M-step:** (update parameters)

$$p(v_q) = \frac{1}{N} \sum_n p(v_q | \mathbf{y}_n), \quad q = 0, \dots, Q$$

$$(s_q, \Gamma_q, \mathbf{c}_q) = \arg \min_{s, \Gamma, \mathbf{c}} \sum_{m,n} p(m, v_q | \mathbf{y}_n) \|\mathbf{y}_n - s\Gamma \mathbf{x}_m - \mathbf{c}\|^2$$

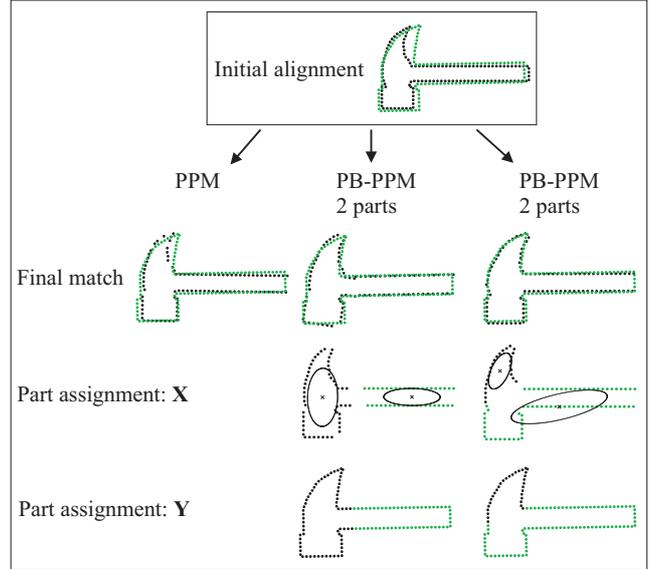
$$\boldsymbol{\mu}_q = \frac{\sum_{m,n} p(m, v_q | \mathbf{y}_n) \mathbf{x}_m}{\sum_{m,n} p(m, v_q | \mathbf{y}_n)}$$

$$\Sigma_q = \frac{\sum_{m,n} p(m, v_q | \mathbf{y}_n) (\mathbf{x}_m - \boldsymbol{\mu}_q)(\mathbf{x}_m - \boldsymbol{\mu}_q)^T}{\sum_{m,n} p(m, v_q | \mathbf{y}_n)}$$

For the examples in Sec. 4, we used the initial values  $p(v_q) = 1/(Q+1)$  (there are  $Q+1$  parts including the background component),  $\Sigma_q = 10^3 \mathbf{I}$ , and the  $\boldsymbol{\mu}_q$  were positioned at randomly chosen  $\mathbf{x}_m$ .

## 4 Examples of Part-Based Matching

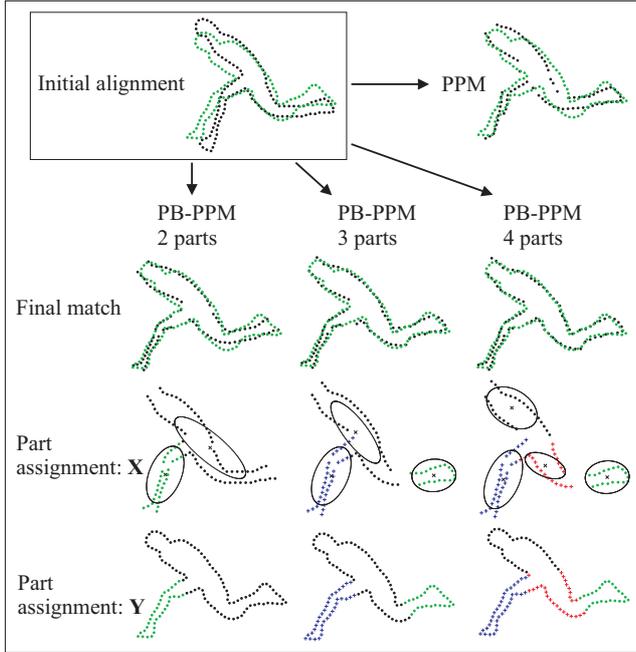
To visualize the matches found by PB-PPM, each point  $\mathbf{y}_n$  is assigned to a part  $v$  using  $\max_v p(v | \mathbf{y}_n)$ . Points assigned to  $v_0$  are removed from the figure. For each



**Figure 3. Alternative solutions.**

$\mathbf{y}_n$  assigned to some  $v_q \in \{v_1, \dots, v_Q\}$ , we find  $m_n \equiv \arg \max_m p(m, v_q | \mathbf{y}_n)$  and assign  $\mathbf{x}_{m_n}$  to  $v_q$ . Those  $\mathbf{x}_m$  not assigned to any parts are removed from the figure. The means and the ellipses of constant probability density associated with the distributions  $\mathcal{N}(\boldsymbol{\mu}_q, \Sigma_q)$  are also plotted.

Fig. 2 shows that PB-PPM correctly matches the part-based shapes considered earlier. Fig. 3 highlights the dependence of PB-PPM on the initial parameter values. In this case, different solutions are found due to the random initialization of the  $\boldsymbol{\mu}_q$  (see Sec. 5). One of the suggested decompositions is intuitive (roughly the head and the handle), whereas the alternative decomposition is perhaps the correct choice since the final match is better and it is clear from the initial alignment that the significant variation is concentrated about the point of the hammer. Fig. 4 shows the matches obtained for different values of  $Q$  when matching two human body shapes. Though our objective is to find the correct correspondence, it is reassuring that the coarse-to-fine decomposition associated with an increasing number of parts is closely related to the ‘natural’ part decomposition (NPD). Most shapes/objects have a NPD and the significant variation in shape is often restricted to changes in the individual parts or composites of them. Examples of this are the possible movements of a single articulated object or the variation across a class of part-based objects (*e.g.* the head sizes of different tennis rackets). Thus, we would expect PB-PPM to frequently find natural parts or composites. Of course, PB-PPM can only recognize parts that have been transformed and consequently many natural parts are invisible to it (*e.g.* it will not identify the upper and lower arm if the arm does not bend in either image). In some cases, seemingly minor natural parts are identified due to



**Figure 4. Varying the number of parts.**

their contribution to the overall difference in shape (e.g. the point of the hammer in Fig. 3).

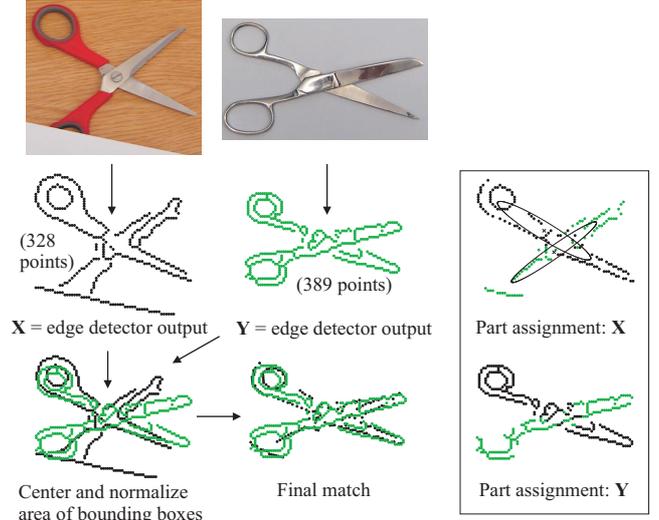
Shapes extracted from real images are often of much poorer quality than those in Figs. 1-4. It is therefore important that a shape matching algorithm can operate effectively on partial, noisy shape information. Fig. 5 demonstrates that PB-PPM can be used to successfully match shapes in real images despite part-articulation, occlusion and sub-optimal performance of the edge detector.<sup>3</sup>

The above examples demonstrate that PB-PPM can match part-based shapes. However, we have found many examples where it fails to identify the correct part structure and correspondence. In some cases this is unavoidable – shape matching is an inherently under-constrained problem and allowing multiple parts makes it all the more so. However, local minima may be responsible for many of the failures since we are attempting to minimize a complicated objective function (recall that all the parts are learnt simultaneously) using random initialization. In the next section, we describe a sequential approach where a simpler problem is solved at each stage. This can be used to select the number of parts and initialize the parameters of PB-PPM.

## 5 Sequential Algorithm for Initialization

When part variation is present, one would expect PB-PPM with  $Q=1$  to find the most significant part and allow the background to explain the remaining parts (c.f. Fig. 2,

<sup>3</sup>The output of edge detectors is often noisy, particularly in CBIR where it is unrealistic to choose different parameters for each image.



**Figure 5. Matching shapes in real images.**

second row). This suggests a sequential approach whereby a single part is learnt and removed from further consideration at each stage. Each new part/component should focus on data points that are currently explained by the background. This is achieved using the idea described in [6]. Specifically, assume that the first part has been learnt using PB-PPM with  $Q=1$ . We now learn the second part using the modified objective function

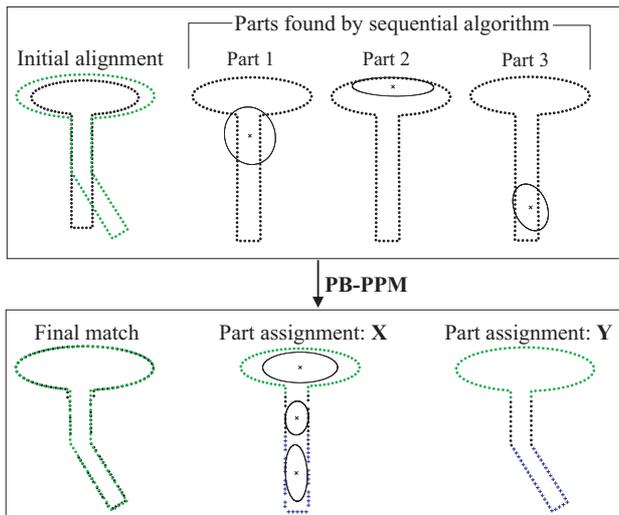
$$\mathcal{J} \equiv \sum_n z_n^1 \log\{p(\mathbf{y}_n|v_2)p(v_2) + U(\mathbf{y}_n)(1 - p(v_1) - p(v_2))\}, \quad (10)$$

where  $p(v_1)$  is known and<sup>4</sup>

$$z_n^1 \equiv \frac{(1 - p(v_1))U(\mathbf{y}_n)}{p(\mathbf{y}_n|v_1)p(v_1) + (1 - p(v_1))U(\mathbf{y}_n)}. \quad (11)$$

Note that  $\mathcal{J}$  is essentially a weighted log-likelihood, where each data point  $\mathbf{y}_n$  is weighted by  $z_n^1$  – the responsibility of the uniform component for that  $\mathbf{y}_n$  under the current model. This has the desired effect of forcing the new component ( $v_2$ ) to explain the data currently explained by the uniform component. The data points generated by the learnt component  $v_1$  are effectively removed (they have small weights) and accordingly  $v_1$  does not play a part in eq.(10). However, since  $p(v_1)$  of the available probability mass has already been assigned to  $v_1$ , the mixing coefficient of the uniform component in eq.(10) is  $1-p(v_1)-p(v_2)$ , rather than  $1-p(v_2)$ . The parameters associated with  $v_2$  are learnt by applying EM to the log-likelihood function in eq.(10). Additional components are learnt in the same way except for minor adjustments to eqs.(10) and (11) to incorporate the previously learnt components. The algorithm

<sup>4</sup>The superscript of  $z$  indicates the number of components that have already been learnt.



**Figure 6. Sequential algorithm – example.**

terminates when the background is responsible for very few data points or very few  $x_m$  (evaluated using  $p(v_b|m)$ , where  $m|v_b \sim Uniform$ ), or when the mixing coefficient for the most recently learnt part is very small. The number of learnt parts provides the  $Q$  for PB-PPM, and the other parameters of PPB-PM are initialized using the values found by this sequential approach.

As discussed in [6], the sequential algorithm is expected to have fewer problems with local minima since the objective function will be smoother (a single component competes against a uniform component at each stage) and the search space smaller (fewer parameters are learnt at each stage). This is supported by our own observations that PPM and PB-PPM with  $Q=1$  often recover from poor initial alignments. Preliminary experiments suggest that the sequential algorithm is capable of solving the model selection problem (choosing the number of parts) and providing good initial values for PB-PPM. This is demonstrated on a simple 3-part problem in Fig. 6. The correct number of parts are found and the suggested parameters are roughly correct before being improved upon by PB-PPM. We are currently investigating a number of modifications that may improve the performance and reliability of the sequential algorithm. These include refining the previously learnt components during learning [6] and experimenting with different initialization strategies.

## 6 Summary and Discussion

We have presented a probabilistic technique for matching part-based shapes. Shapes are represented by unlabeled point sets, so discontinuous boundaries and non-boundary points do not pose a problem. Occlusions and significant dissimilarities between shapes are explained by a ‘background model’ and hence, their impact on the overall match

is limited. A greedy algorithm that learns the parts sequentially can be used to estimate the number of parts and initialize the main algorithm.

The model is limited by the simple transformations used. In particular, non-isotropic scaling of parts and shear are commonly observed transformations that cannot be handled. This limitation could be addressed by modifying PB-PPM to learn affine transformations. Ideally, we could go further and allow parts to deform nonlinearly. The techniques described in [2] and [7] allow nonlinear transformations but are not equipped to deal with the discontinuity associated with part-based variation. Conversely, PB-PPM struggles to match shapes that have undergone a significant nonlinear transformation. Any attempt to combine parts with nonlinear transformations is likely to suffer from problems with local minima. These could perhaps be overcome by incorporating stronger prior knowledge about what parts are expected to look like or using training data to learn about valid modes of variation.

We are currently investigating how PB-PPM can be used to quantitatively assess shape similarity. The similarity score should depend on the number of parts required to match the shapes and the extent of the transformations as well as the match itself. There is currently no ‘cost’ associated with the part transformations. Thus, the degree of part rotation, scaling and translation does not alter the likelihood of  $X$  generating  $Y$ . It seems appropriate to put prior distributions over the transformation parameters that encourage small transformations. These should reflect the fact that we are not concerned about transformations common to all parts, but rather the difference between transformations applied to different parts.

## References

- [1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24:509–522, 2002.
- [2] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Comp. Vis. and Image Understanding*, 89:114–141, 2003.
- [3] I. L. Dryden and K. V. Mardia. *Statistical Shape Analysis*. Wiley, 1998.
- [4] C. Grigorescu and N. Petkov. Distance sets for shape filters and shape recognition. *IEEE Trans. Image Proc.*, 12(10):1274–1286, 2003.
- [5] G. McNeill and S. Vijayakumar. A probabilistic approach to robust shape matching. In *ICIP, 2006* – to appear.
- [6] M. Titsias. *Unsupervised Learning of Multiple Objects in Images*. PhD thesis, University of Edinburgh, 2005.
- [7] Z. Tu and A. Yuille. Shape matching and recognition using generative models and informative features. In *ECCV, 2004*.