

A Probabilistic Approach to Robust Shape Matching and Part Decomposition

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

January 19, 2006

Abstract

We present a probabilistic approach to shape matching which is invariant to rotation, translation and scaling. Shapes are represented by unlabeled point sets, so discontinuous boundaries and non-boundary points do not pose a problem. Occlusions, significant dissimilarities between shapes and image clutter are explained by a ‘background model’ and hence, their impact on the overall match is limited. By simultaneously learning a part decomposition of both shapes, we are able to successfully match shapes that differ as a result of independent part transformations – a form of variation common amongst real objects of the same class. The effectiveness of the matching algorithm is demonstrated using the benchmark MPEG-7 data set and real images.

1 Introduction

In many object recognition problems, the examples are most easily disambiguated on the basis of *shape* - as opposed to properties such as color or texture. Content-based image retrieval (CBIR) is a prime example of an application that can benefit from shape information. However, defining a shape descriptor and associated matching algorithm that is both generic and discriminative is a difficult challenge. Despite recent developments in this area, the available techniques typically struggle to match perceptually similar shapes in at least one of the following cases:

Occlusion: one or both shapes are partially occluded.

Nonlinearity: a good match is not possible using linear transformations.

Localized dissimilarity: *e.g.* a ‘feature’ of one shape is not present or is significantly different on the other shape.

Discontinuity: the observed part of a shape is only available as an unordered point set.

Consideration of all possible *occlusion* states is often prohibitively expensive. *Non-linear* differences in shape can be dealt with by introducing nonlinear transformations into the matching process (e.g. [2, 13]). Unfortunately, shape similarity assessments based on any particular class of transformations will inevitably fail to agree with human judgement in some cases. The challenge is to identify the commonly encountered transformations that we expect shape to be invariant to (or partially invariant to) and include these in our shape matching algorithm. *Localized dissimilarities* are often responsible for low global similarity scores between perceptually similar shapes. It is therefore important to limit the impact of any region of the shapes on the overall similarity score. Most matching algorithms assume the shape boundary is *continuous* – even those which use point representations generally rely on the cyclic ordering of the points. Ghosh and Petkov [6] note that continuous contours cannot be extracted from many real images and argue for matching techniques which do not require the continuity assumption. Shape Contexts [2] and Distance Multisets [7] are two such methods. The latter seems to perform better [7, 6], but it is not scale invariant.

Probabilistic methods which find a *soft* correspondence between points [9, 11, 4] can potentially overcome problems associated with occlusion, localized dissimilarity and discontinuity. However, such methods have not been adopted by the shape retrieval community. This may be due to their reliance on global linear transformations (with the exception of [4]) which are too restrictive in many matching problems. This paper makes two important contributions to the field of shape retrieval. Firstly, we introduce a probabilistic approach to shape matching which overcomes many of the difficulties associated with deterministic algorithms. This performs well on the popular MPEG-7 “bullseye” retrieval test and is shown to correctly match objects in real images despite partial occlusion and image clutter. Secondly, by generalizing the initial model, we are able to learn part decompositions and soft correspondences simultaneously. This allows us to handle cases where *different* parts of a shape undergo *different* linear transformations.

2 Weighted Procrustes Matching

We start by giving a brief introduction to *weighted Procrustes matching* which is fundamental to our approach. Let us assume that we have two shapes, each represented by N 2D points, $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^T$, $\mathbf{Y} = (\mathbf{y}_1, \dots, \mathbf{y}_N)^T \in \mathbb{R}^{N \times 2}$, and that the point-to-point correspondence is known to be $\mathbf{x}_j \leftrightarrow \mathbf{y}_j$. Intuitively, the shape of an object should not change under translation, rotation or scaling. Therefore, a reasonable measure of dissimilarity between \mathbf{X} and \mathbf{Y} is the minimum of the weighted sum of squared distances over corresponding point pairs:

$$d^2(\mathbf{X}, \mathbf{Y}) \equiv \min_{s, \Gamma, \mathbf{c}} \sum_{j=1}^N \alpha_j^2 (\mathbf{y}_j - s\Gamma\mathbf{x}_j - \mathbf{c})^2, \quad (1)$$

where s is a scale parameter, $\mathbf{c} \in \mathbb{R}^2$ is a translation vector and $\Gamma(\theta)$ is a 2D rotation matrix. The weights, α_j , allow us to express the relative importance of closely matching the j -th pair of points. In this paper, we are interested in the transformation

parameters which minimize eq.(1). For 2D shapes, these can be calculated using a simple closed form expression as follows.¹ Represent each 2D point as a complex number: $\mathbf{x}_j = (x_j^{(1)}, x_j^{(2)}) \rightarrow x_j^{(1)} + ix_j^{(2)} \equiv v_j \in \mathbb{C}$, then $\mathbf{X} \rightarrow \mathbf{v}$ and $\mathbf{Y} \rightarrow \mathbf{z} \in \mathbb{C}^N$. Shift \mathbf{v} and \mathbf{z} such that $\sum_j \alpha_j^2 v_j = \sum_j \alpha_j^2 z_j = 0$ and then form the weighted shapes \mathbf{v}_α and \mathbf{z}_α , where $(\mathbf{v}_\alpha)_j \equiv \alpha_j v_j$. The minimizing parameters are given by

$$\hat{s} = |\mathbf{z}_\alpha^* \mathbf{v}_\alpha| / (\mathbf{v}_\alpha^* \mathbf{v}_\alpha), \quad \hat{\theta} = -\arg(\mathbf{z}_\alpha^* \mathbf{v}_\alpha), \quad \hat{\mathbf{c}} = \mathbf{0}, \quad (2)$$

where \mathbf{v}^* denotes the complex transpose of \mathbf{v} and, in this section only, $\arg(\cdot)$ denotes the complex argument.

3 Basic Probabilistic Model

We now develop a probabilistic approach to shape matching which utilizes the ideas introduced in the previous section. The first step is to describe a model for matching unlabeled point sets.

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 define and compute the optimal match between \mathbf{X} and \mathbf{Y} . We assume that the \mathbf{y}_j are observations from a Gaussian mixture model (GMM) [3], *i.e.* $p(\mathbf{y}) = \sum_k p(\mathbf{y}|k)p(k)$ where each $p(\mathbf{y}|k)$ is a bivariate Gaussian distribution. The center of the Gaussian $p(\mathbf{y}|k)$ depends on \mathbf{x}_k , but unlike a standard GMM, the movement of the centers is controlled by a single set of transformation parameters s, Γ and \mathbf{c} :

$$\mathbf{y}|k \sim \mathcal{N}(s\Gamma\mathbf{x}_k + \mathbf{c}, \sigma^2\mathbf{I}). \quad (3)$$

Note that the model can ‘delete’ some of the \mathbf{x}_k by taking the appropriate $p(k)$ to be zero. However, there is no such mechanism for deleting points of \mathbf{Y} since the model will try to explain all of the data. To rectify this, we introduce a ‘background’ component k_b to explain outliers and assume that $\mathbf{y}|k_b \sim \mathcal{N}(\mathbf{0}, \sigma_b^2\mathbf{I})$, where σ_b^2 is very large.

The model described so far is a variation of that used by Kent *et. al.* to match the active sites of proteins [9]. In shape matching tasks, we have found that this approach struggles with occlusion and local dissimilarities – the mechanism for handling outliers is ineffective. We modify this mechanism by introducing a variable q which takes the value q_b (background) or q_f (foreground). The joint distribution of this modified model can be written as $p(\mathbf{y}, k, q) = p(\mathbf{y}|k)p(k|q)p(q)$, where the $p(\mathbf{y}|k)$ remain unchanged (*c.f.* eq.(3) and $\mathbf{y}|k_b$ above). The mixture component k_b ‘belongs’ solely to the background (*i.e.* $p(k_b|q_b) = 1$), whereas components k_1, \dots, k_M belong solely to the foreground (*i.e.* $p(k|q_b) = 0$ when $k \neq k_b$). In this model, the old $p(k)$ are replaced by the $p(k|q_f)$. To compute these, we identify each k with the point \mathbf{x}_k that it indexes and take

$$p(k|q_f) \equiv p(\mathbf{x}_k|q_f) \equiv \frac{g(\mathbf{x}_k; \boldsymbol{\mu}, \Sigma)}{\sum_{k=1}^M g(\mathbf{x}_k; \boldsymbol{\mu}, \Sigma)}, \quad (4)$$

¹Eq.(1) is minimized using the approach described in [5] for standard (unweighted) Procrustes matching; an alternative derivation involving singular value decomposition is presented in [11].

where $g(\cdot; \boldsymbol{\mu}, \Sigma)$ is the density function of a bivariate Gaussian distribution. Note that $p(k|q_f)$ is a distribution over $\{1, \dots, M\}$, whereas $p(\mathbf{x}_k|q_f)$ is a unimodal (yet still discrete) distribution over $\{\mathbf{x}_1, \dots, \mathbf{x}_M\} \subset \mathbb{R}^2$. We refer to the two distributions interchangeably.

Maximum likelihood estimates for the model parameters are found using the expectation maximization (EM) algorithm [3] as follows. The value of σ^2 in eq.(3) (which reflects the predicted variation between ‘similar’ shapes) and σ_b^2 remain fixed throughout. Let $w_{k,j,q} \equiv p(k, q|\mathbf{y}_j)$ represent the probability that mixture component k and foreground/background = q were responsible for generating the data point \mathbf{y}_j . The EM updates are given by:

E-step: Update the $w_{k,j,q}$ using the current estimates of the parameters $s, \Gamma, \mathbf{c}, \boldsymbol{\mu}, \Sigma$ and $p(q)$ (c.f. eqs.(3) & (4)).

$$w_{k,j,q}^{new} = \frac{p(\mathbf{y}_j|k)p(k|q)p(q)}{\sum_{k,q} p(\mathbf{y}_j|k)p(k|q)p(q)}. \quad (5)$$

M-step: Update the parameters using the $w_{k,j,q}$

$$p(q) = \frac{1}{N} \sum_{j,k} w_{k,j,q}, \quad (6)$$

$$\boldsymbol{\mu} = \frac{\sum_{j,k} w_{k,j,q_f} \mathbf{x}_k}{\sum_{j,k} w_{k,j,q_f}}, \quad (7)$$

$$\Sigma = \frac{\sum_{j,k} w_{k,j,q_f} (\mathbf{x}_k - \boldsymbol{\mu})(\mathbf{x}_k - \boldsymbol{\mu})^T}{\sum_{j,k} w_{k,j,q_f}}, \quad (8)$$

$$(s, \Gamma, \mathbf{c}) = \arg \min_{s, \Gamma, \mathbf{c}} \sum_{j,k} w_{k,j,q_f} (\mathbf{y}_j - s\Gamma\mathbf{x}_j - \mathbf{c})^2. \quad (9)$$

Eq.(9) is a weighted Procrustes problem that can be solved analytically as described in Sec. 2. The initial value of $p(q_b)$ is set at 0.001, random initialization is used for the $p(k|q_f)$ and the same values of σ^2 and σ_b^2 are used in all examples.² In all figures, we remove the \mathbf{y}_j that are explained primarily by q_b (i.e. with $\arg \max_q p(q|\mathbf{y}_j) = q_b$). Each of the remaining \mathbf{y}_j is then assigned to an \mathbf{x}_k using $\arg \max_k p(\mathbf{x}_k|\mathbf{y}_j)$, and those \mathbf{x}_k with no \mathbf{y}_j assigned to them are also removed. Fig. 1 shows how the model performs on matching problems involving occlusion, discontinuity and localized dissimilarity. The bold points are the \mathbf{x}_k that are collectively transformed to match the faint \mathbf{y}_j ; the final matches are scaled up for clarity. See that many of the \mathbf{y}_j have been deleted in example (a), whereas it is the \mathbf{x}_k that have been deleted in (b) and (d). In all cases, the sampling frequency of corresponding sections is different, so there is no perfect point-to-point match. When such a perfect match does exist, the model will find it.

Why does the modified model handle outliers more effectively than the simpler model? In the simpler model, the mixture component k_b competes with the other components directly. This reflects an assumption that each non-outlier was generated from

²The initial values $p(k|q_f) = 1/M$ for $k = 1, \dots, M$ were used for the tests in Sec. 4.1 – this effectively means $\Sigma = \gamma\mathbf{I}$ where γ is very large.

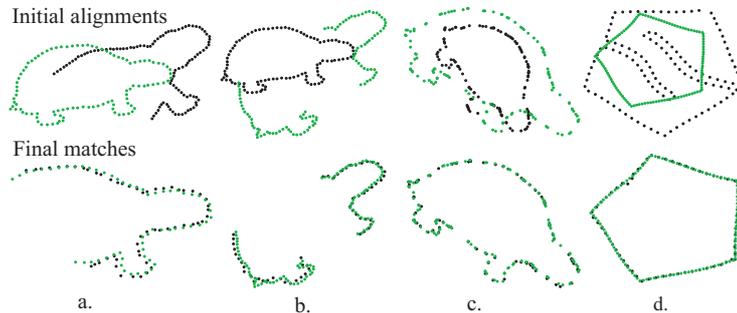


Figure 1: Examples - basic model.

a single mixture component – an assumption that is justified when matching the active sites of proteins [9] since the ultimate aim is to recover a 1-1 mapping. In the modified model, q_b competes against the mixture model rather than the individual components. This is consistent with the assumption that a non-outlier can be ‘generated’ by more than one component – an assumption that is justified here since the ultimate aim is the soft assignment itself, *i.e.* there is generally no 1-1 correspondence associated with the correct match (*c.f.* Figs. 1-7). A further advantage of having q_b compete against a fixed combination of the foreground components is that $p(q_b)$ need not be large for q_b to explain outliers.

The assumed form of $p(\mathbf{x}_k|q_f)$ (eq.(4)) might seem unjustified, but experiments have shown that parameterizing $p(\mathbf{x}_k|q_f)$ in this way is preferable to learning an unconstrained distribution. In practice, $p(\mathbf{x}_k|q_f)$ does not become sharply peaked, ensuring that the learnt transformation parameters depend on a large number of the \mathbf{x}_k . Also, the EM algorithm converges quickly when the assumption is used, presumably due to the reduction in the number of parameters that are learnt. Fig. 1b demonstrates that disjoint clusters of \mathbf{x}_k can remain undeleted despite $p(\mathbf{x}_k|q_f)$ being unimodal.

4 Basic Model – Results and Examples

4.1 Shape Retrieval and Classification

The ‘‘Bullseye Test’’ on the MPEG-7 shape data set³ has been used extensively to assess the performance of shape matching algorithms. The data set is composed of 1400 binary images with a single shape in each image. There are 70 different classes and 20 observations in each class. Some of the shapes are shown in Figure 2. In the bullseye test, a shape is presented as a query and the top 40 matches are retrieved (from the entire data set – the test shape is not removed). The task is repeated for each shape and the number of correct matches (out of a maximum possible 20) are noted. A perfect performance results in $1400 \times 20 = 28000$ matches. Results are given as a percentage of this maximum score.

³<http://www.cis.temple.edu/~latecki/research.html#shape>

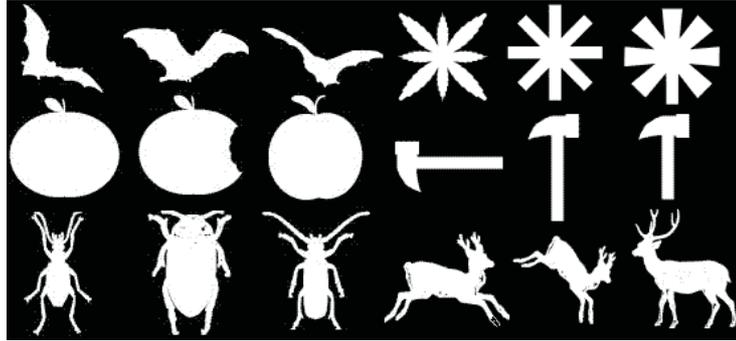


Figure 2: Example shapes from the MPEG-7 data set.

To apply probabilistic matching to this problem, we must first address the issue of initialization. The EM algorithm often finds the correct match despite a seemingly poor initial alignment (*c.f.* Figs. 1 and 3), but inevitably, sub-optimal matches (local minima) are chosen on some occasions. It is therefore important to find as good an initial alignment as possible. The following procedure is a fast, simple heuristic which satisfies this objective. Shapes (represented by the pixels/points they contain) are centered, aligned along their first principal component, and then scaled to have the same interior area. The alignment between two of these normalized shapes is found using a 25 point boundary representation of each shape. The first shape is compared to 32 poses of the second shape (16 rotation states and reflection). The best pose is selected using the sum of squared distances over each point of the first shape and the closest point to it on the second. The transformations associated with the chosen alignment (including the initial normalizing transformations) are then applied to 100 point representations of the original shapes to which the probabilistic matching procedure is applied. Note that the sole objective of this initialization algorithm is to find *sufficiently good matches as often as possible*, so that the local minimum found by the EM algorithm corresponds to the correct match. Various fast alignment techniques (*e.g.* [12]) will find *very good matches quite often*, but they will also choose nonsensical matches for shapes that probabilistic matching can handle given a suitable initial alignment (*c.f.* Fig. 1d).

The dissimilarity score between two shapes is based on the hard assignment of undeleted \mathbf{y}_j to undeleted \mathbf{x}_k (Sec. 3). Specifically, the average squared distance between matched point-pairs (post transformation) is multiplied by a penalty term λ^a , where a is the number of deleted \mathbf{x}_k and $\lambda > 1$ is a constant (set at 1.1 in our experiments):

$$d_{\mathbf{X}}(\mathbf{Y}) \equiv \left(\frac{1}{L} \sum_{l=1}^L |\mathbf{x}_{l_x} - \mathbf{y}_{l_y}|^2 \right) \lambda^a. \quad (10)$$

The pair (l_x, l_y) indexes the matched pairs of \mathbf{X} and \mathbf{Y} and L is the total number of matched pairs.⁴ The EM algorithm converges after 5 iterations on average.

⁴ L need not be $100-a$ since multiple \mathbf{y}_j can be assigned to one \mathbf{x}_k .

Table 1: Bullseye scores for best performing algorithms.

Score (%)	78.18	78.38	78.80	79.36	84.33
Reference	[13]	[7]	[8]	[14]	[1]

Table 1 shows, to the best of our knowledge, the best known results for the bullseye test *using a single shape descriptor and basic nearest neighbor retrieval*. Our probabilistic approach achieved a score of 79.44% using the dissimilarity measure defined in eq. 10 and nearest neighbor retrieval. This score is second only to the method proposed by Attalla and Siy [1] which is strictly dependent on continuous boundary information. Whilst we did utilize boundary information to some extent here – to ensure that the 100 points were equally spaced along the boundary,⁵ the key observation is that probabilistic matching remains applicable and effective when no such information is available (*c.f.* Figs. 1c, 3 and 7).

By removing the query shape from the database and retrieving just one shape, we can compute the leave-one-out classification accuracy of a 1-nearest neighbor classifier. Our algorithm misclassified only 35 shapes out of 1400. Again, only the technique described in [1] has achieved a higher level of accuracy ([12]: 60 errors, [14]: 43 errors, [1]: 33 errors).

4.2 Matching Shapes in Real Images

The MPEG-7 data set consists of binary images, each containing a single object whose continuous boundary is easily extracted. Since real images rarely fall into this category, it is important that a shape matching algorithm can operate effectively on partial, noisy shape information. Fig. 3 demonstrates that our model can be used to successfully match shapes in real images despite occlusion, clutter and sub-optimal performance of the edge detector.⁶ The correct match in Fig. 3a involves significant deletion from both point sets (the corkscrew’s blade in the first image and the pencil in the second). Fig. 3b demonstrates a ‘query by example’ problem where a sketch of the required object is found in a cluttered image.

4.3 Part-based variation

In Fig. 4 (first two rows) we consider shapes whose parts have undergone different transformations. The basic model has no way of recognizing part structure and at best will identify localized dissimilarities (Fig. 4c). This limitation is important since part-based variation is commonly observed between perceptually similar shapes (*c.f.* Figs. 5 and 7). As we shall see in the next section, by introducing q and the ‘discrete Gaussian assumption’ for $p(\mathbf{x}_k|q_f)$, we have done all the hard work required for the construction of a part-based model.

⁵Note that area-based scale normalization can always be used when complete boundaries are available.

⁶The output of edge detectors is often noisy, particularly in CBIR where it is unrealistic to choose different parameters for each image.

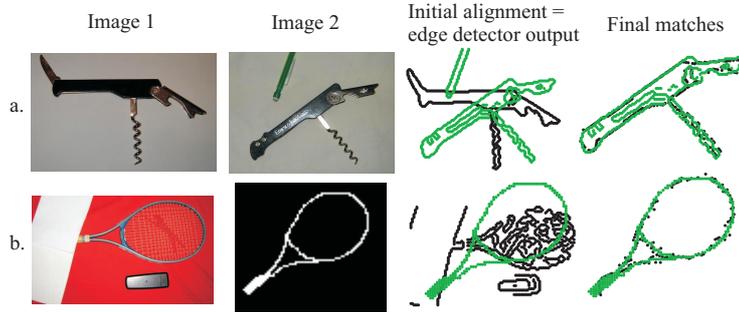


Figure 3: Matching shapes in real images.

5 Part-based Model

The model introduced in Sec. 3 can be seen as a special instance of a part-based model: it has one background part (q_b) and one foreground part (q_f). We now consider models with one background part, q_b , and Q foreground parts, q_f^1, \dots, q_f^Q , where

$$\mathbf{y}|k, q_f^l \sim \mathcal{N}(s_l \Gamma_l \mathbf{x}_k + \mathbf{c}_l, \sigma^2 \mathbf{I}). \quad (11)$$

Thus, the part ‘label’ q_f^l 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. This is in complete contrast to methods which seek an independent part decomposition of each shape (e.g. [10]). The simplest way to proceed is to write $p(\mathbf{y}_j|k, q) = p(\mathbf{y}_j|k)p(k, q)$ and learn every entry of the $p(k, q)$ matrix. Our experiments have indicated that this approach learns parts that are widely dispersed around the shape, *i.e.* the decomposition that maximizes the expected likelihood does not agree with the intuitive decomposition. Here, as before, we write $p(k, q) = p(k|q)p(q)$ and use the Gaussian assumption introduced in Sec. 3 (eq.(4)) which will encourage parts to have spatial coherence:

$$p(k|q_f^l) \equiv p(\mathbf{x}_k|q_f^l) \equiv \frac{g(\mathbf{x}_k; \boldsymbol{\mu}_l, \Sigma_l)}{\sum_{k=1}^M g(\mathbf{x}_k; \boldsymbol{\mu}_l, \Sigma_l)}. \quad (12)$$

This assumption means that the \mathbf{x}_k themselves are generated by a GMM with Q components. However, note that this GMM is embedded in the larger model. Using EM to maximize the expected likelihood will balance this GMM’s desire for coherent parts against the need for the parts and transformations to explain the real data (the \mathbf{y}_j).

There are now l lots of parameters $s_l, \Gamma_l, \mathbf{c}_l, \boldsymbol{\mu}_l, \Sigma_l$, and $p(q_f^l)$ to update. The update equations themselves (eqs. 5-9) remain the same except that $p(\mathbf{y}_j|k)$ in eq.(5) becomes $p(\mathbf{y}_j|k, q)$. This highlights the fact that each \mathbf{x}_k ($k \neq k_b$) now belongs to every q_f^l to some extent – a soft assignment of mixture components to parts. The initial $p(q)$ are chosen at random, except for $p(q_b)$ which is initialized to 0.001. We assume

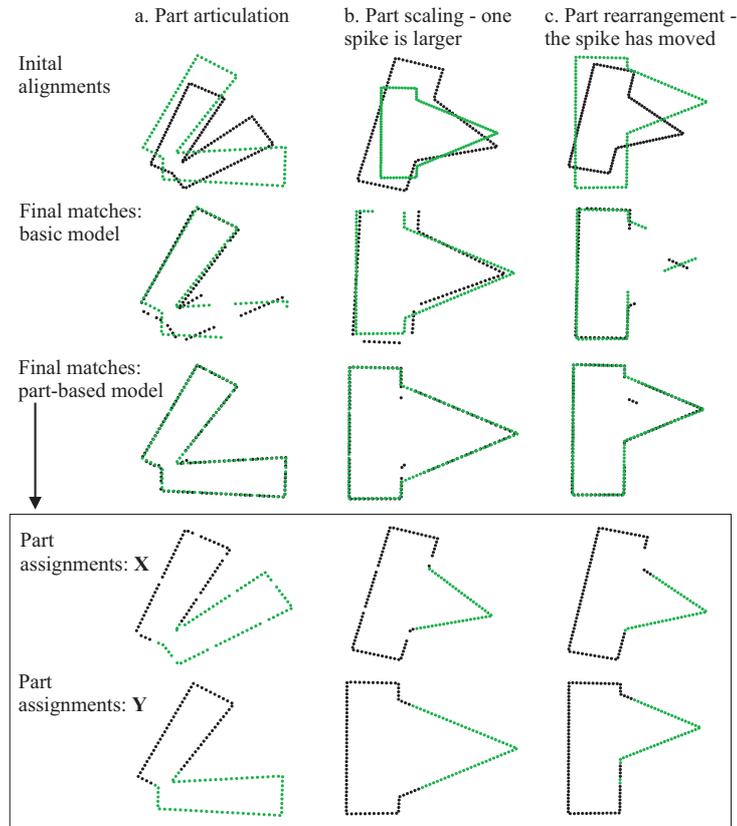


Figure 4: Part-based variation.

that the optimal number of parts, Q , is known; techniques for learning Q are discussed in Sec. 7.

To visualize the matches found using this model, we first delete unmatched points using the procedure described in Sec. 3. This leaves a subset of the y_j , each element of which has been assigned to an undeleted x_k . The part membership of an x_k (and those y_j assigned to it) is given by $\arg \max_q p(x_k|q)$. Finally, each x_k is transformed using the transformation parameters of the part to which it belongs. Fig. 4 (third row) demonstrates that the new model can correctly match shapes whose parts have undergone independent transformations. The learnt decompositions (bottom two rows of Fig. 4) show that the matches were achieved by learning the natural part structure of the objects. We now consider more challenging problems, the results of which give valuable insight into the model's behavior.

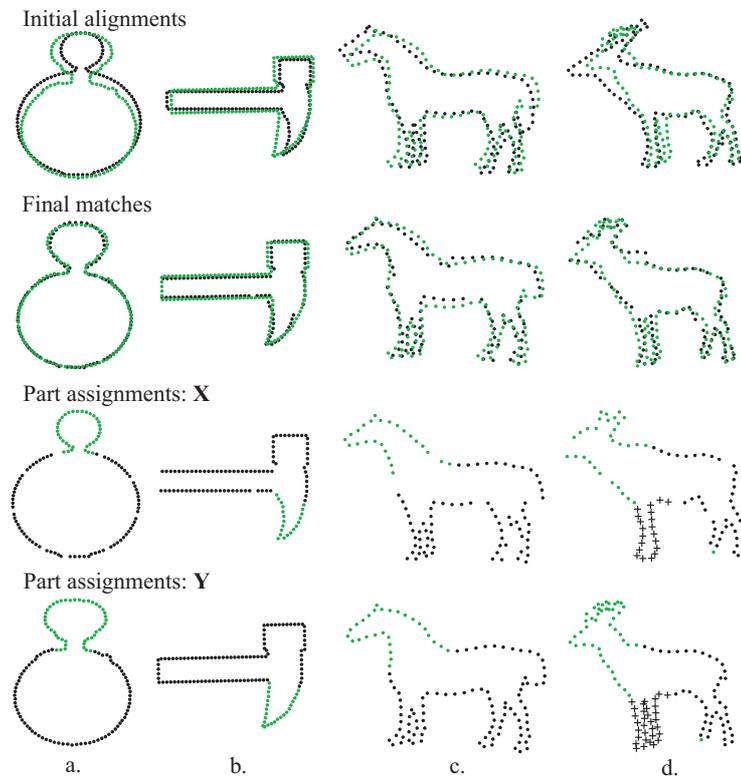


Figure 5: Part-based matching of some MPEG-7 shapes.

6 Part-Based Model – Examples

Fig. 5 shows the results achieved on pairs of shapes from the MPEG-7 data set. All shapes are represented by the same number of points (120), enabling us to use standard Procrustes matching to find the initial alignments [12]. Observe that the learnt parts are either ‘natural parts’ (Fig. 5a), composites of natural parts (Figs. 5c, 5d), or ‘minor natural parts’ that arise through recursive decomposition of the shape (Fig. 5b). Thus, despite the model not always learning the most obvious decomposition, the learnt parts do correspond to perceptually meaningful elements of the shapes. This suggests that shape similarity measures based on our approach may mimic human judgement more accurately than standard nonlinear methods. The challenges associated with defining such measures are discussed in Sec. 7.

Different aspects of the model’s behavior can be explained using specific examples. The tendency to learn composites of natural parts is particularly clear for the horse shapes. Here, the legs and bodies of the horses can be well matched by a single transformation, so the similarity between these regions can be established without learning the composite parts. Indeed, the very fact that these regions can be well matched means

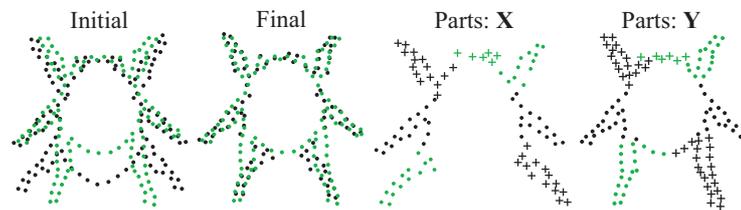


Figure 6: A difficult problem.

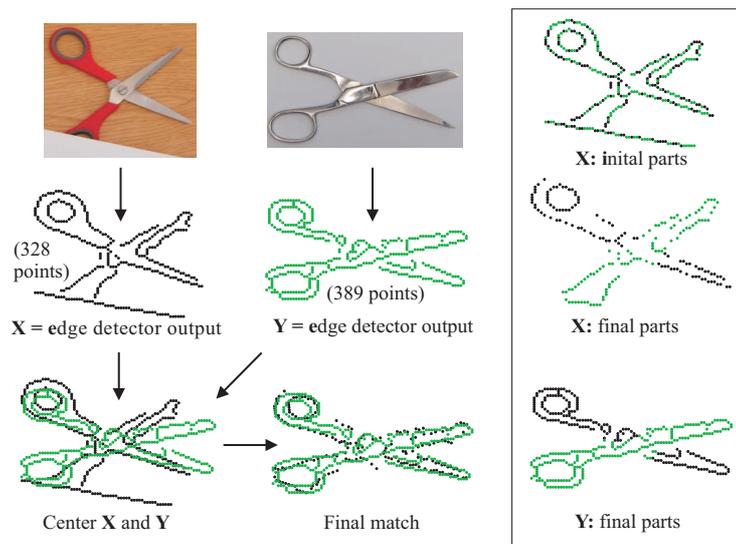


Figure 7: Matching shapes in real images.

that there is insufficient variation to learn the composite parts. With the hammers, there is only significant variation on one side of the head and consequently this minor part is recognized. There are various examples of the algorithm deleting significant sections of the shape to achieve a good match, most notably: the horse's tail, the deer's antlers and one end of the beetle's body (Fig. 6). In Fig. 6, the model has paired together legs which are spatially separated. Here, the mechanism for encouraging spatial coherence is ineffective and the model has learnt part composites that are counter-intuitive. If seven parts are used, the intuitive decomposition (a body and six legs) is not learnt, suggesting that there is insufficient variation of the real parts. It is interesting to consider how an algorithm including part number selection should behave in this case and how such behavior could be achieved (*c.f.* Sec. 7).

Fig. 7 demonstrates that the model can be used to successfully match shapes in real images despite part-articulation, occlusion and sub-optimal performance of the edge detector.

7 Summary and Discussion

We have presented a probabilistic approach to shape matching which is invariant to rotation, translation and scaling. The algorithm operates on unlabeled point sets of arbitrary size and uses a background model to handle occlusion, significant dissimilarities between shapes and image clutter. By simultaneously learning a part decomposition of both shapes, we are able to successfully match shapes that differ as a result of independent part transformations.

The examples in this paper indicate that our method overcomes the problems associated with state-of-the-art techniques. However, further work is required to produce a complete matching algorithm with an associated similarity score. Choosing the number of parts is the most important issue still to be addressed. We are assessing the performance of penalized log-likelihood approaches⁷ on this model selection problem. Currently, there is no ‘cost’ associated with the part transformations. Thus, the degree of part rotation, scaling and translation does not alter the likelihood of \mathbf{X} generating \mathbf{Y} . It may be 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. Initialization is another important issue. The final match is sensitive to the initial values of $p(k|q)$ and $p(q)$. Selecting these values based on a standard Q -component GMM for the \mathbf{x}_k works well when this naive clustering approximates the part structure. The EM algorithm often finds the correct match despite a seemingly poor initial alignment (*c.f.* Figs. 1 and 4), but inevitably, sub-optimal matches (local minima) are chosen on some occasions. Basic matching algorithms (*c.f.* Fig. 5) can provide good initial alignments. Since the EM algorithm converges quickly when a good match exists (generally <20 iterations), we need not restrict ourselves to a single initial alignment for each pair of shapes.

References

- [1] E. Attalla and P. Siy. Robust shape similarity retrieval based on contour segmentation polygonal multiresolution and elastic matching. *Patt. Recog.*, 38:2229–2241, 2005.
- [2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24:509–522, 2002.
- [3] C. Bishop. *Neural Networks for Pattern Recognition*. Oxford Univ. Press, 1995.
- [4] H. Chui and A. Rangarajan. A new algorithm for non-rigid point matching. In *CVPR*, 2000.
- [5] I. Dryden and K. Mardia. *Statistical Shape Analysis*. Wiley, 1998.
- [6] A. Ghosh and N. Petkov. Robustness of shape descriptors to incomplete contour representations. *PAMI*, 27(11):1793–1804, 2005.
- [7] C. Grigorescu and N. Petkov. Distance sets for shape filters and shape recognition. *IEEE Trans. Image. Proc.*, 12(10):1274–1286, 2003.
- [8] A. Jalba, M. Wilkinson, and J. Roerdink. Shape representation and recognition through morphological curvature scale spaces. *IEEE Trans. on Image Processing*, In Press.
- [9] J. Kent, K. Mardia, and C. Taylor. Matching problems for unlabelled configurations. In *LASR*, 2004.

⁷The objective function is augmented with a term based on the number of parts.

- [10] D. Kim, I. Yun, and S. Lee. A new shape decomposition scheme for graph-based representation. *Pattern Recognition*, 38:673–689, 2005.
- [11] B. Luo and E. Hancock. Iterative Procrustes alignment with the EM algorithm. *Im. and Vis. Comp.*, 20:377–396, 2002.
- [12] G. McNeill and S. Vijayakumar. 2D shape classification and retrieval. In *IJCAI*, 2005.
- [13] T. Sebastian, P. Klein, and B. Kimia. On aligning curves. *PAMI*, 25(1):116–125, 2003.
- [14] B. Super. Learning chance probability functions for shape retrieval or classification. In *IEEE W'shop on Learning in Comp. Vis. and Pat. Recog.*, 2004.