Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

ELSEVIER

Computer Vision and Image Understanding 92 (2003) 26–55

Computer Vision and Image Understanding

www.elsevier.com/locate/cviu

# A unified framework for alignment and correspondence

Bin Luo[a,b,*] and E.R. Hancock[a]

[a] *Department of Computer Science, University of York, York YO1 5DD, UK*
[b] *The Key Lab of IC&SP, Anhui University, China*

Received 27 July 2001; accepted 27 June 2003

**Abstract**

This paper casts the problem of 2D point-set alignment and correspondence matching into a unified framework. Our aim in providing this unification is to constrain the recovery of pose parameters using relational constraints provided by the structural arrangement of the points. This structural information is provided by a neighbourhood graph for the points. We characterise the problem using distinct probability distributions for alignment errors and correspondence errors. The utility measure underpinning the work is the cross-entropy between probability distributions for alignment and assignment errors. This statistical framework interleaves the processes of finding point correspondences and estimating the alignment parameters. In the case of correspondence matching, the probability distribution models departures from edge consistency in the matching of the neighbourhood graphs. We investigate two different models for the alignment error process. In the first of these, we study Procrustes alignment. Here we show how the parameters of the similarity transform and the correspondence matches can be located using dual singular value decompositions. The second alignment process uses a point-distribution model. We show how this augmented point-distribution model can be matched to unlabelled point-sets which are subject to both additional clutter and point drop-out. Experimental results using both synthetic and real images are given.


## 1. Introduction

Point pattern matching is a problem of pivotal importance in computer vision that continues to attract considerable interest. The problem may be abstracted as

* Corresponding author. Fax: +44-1904-432-767.
*E-mail addresses:* luo@cs.york.ac.uk (B. Luo), erh@cs.york.ac.uk (E.R. Hancock).

1077-3142/$ - see front matter © 2003 Elsevier Inc. All rights reserved.
doi:10.1016/S1077-3142(03)00097-3

either alignment or correspondence. Alignment involves explicitly transforming the point positions under a predefined geometry so as to maximise a measure of correlation. Examples here include Procrustes normalisation [13,1], affine template matching [27], deformable point models [5] and various methods for matching 2D point-sets to 3D models [3,7,11]. Correspondence, on the other hand, involves recovering a consistent arrangement of point assignment labels. The correspondence problem can be solved using a variety of point assignment [20,22] and graph matching [2,9,14,28] algorithms.

The problem of point pattern matching has attracted sustained interest in both the vision and statistics communities for several decades. For instance, Kendall [13] has generalised the process to projective manifolds using the concept of Procrustes distance. Ullman [23] was one of the first to recognise the importance of exploiting rigidity constraints in the correspondence matching of point-sets. Recently, several authors have drawn inspiration from Ullman's ideas in developing general purpose correspondence matching algorithms using the Gaussian weighted proximity matrix. For instance Scott and Longuet-Higgins [20] locate correspondences by finding a singular value decomposition of the inter-image proximity matrix. Shapiro and Brady [21], on the other hand, match by comparing the modal eigenstructure of the intra-image proximity matrix. In fact these two ideas provide some of the basic groundwork on which the deformable shape models of Cootes et al. [5] and Sclaroff and Pentland [18] are built. This work on the co-ordinate proximity matrix is closely akin to that of Umeyama [24] who shows how point-sets abstracted in a structural manner using weighted adjacency graphs can be matched using an eigen-decomposition method. These ideas have been extended to accommodate parametrised transformations [25] which can be applied to the matching of articulated objects [26]. More recently, there have been several attempts at modelling the structural deformation of point-sets. For instance, Amit and Kong [2] have used a graph-based representation (graphical templates) to model deforming two-dimensional shapes in medical images. Lades et al. [14] have used a dynamic mesh to model intensity-based appearance in images. Belongie et al. [19] have used so-called shape contexts based on the chord length distribution associated with boundary points to recover point correspondences and then estimate the transformation between two shapes.

In a recent paper Cross and Hancock [6] developed a unified statistical framework for alignment and correspondence. The motivation for the work was that the dichotomy normally drawn between the two processes overlooks considerable scope for synergistic interchange of information. In other words, there must always be bounds on alignment before correspondence analysis can be attempted, and vice versa. The approach adopted in developing this new point-pattern matching method was to embed constraints on the spatial arrangement of correspondences within an EM algorithm for alignment parameter recovery. This process has many features reminiscent of Jordan and Jacob's hierarchical mixture of experts algorithm [10]. The observation underpinning this paper is that although the method proved effective it fails to put the alignment and correspondence processes on a symmetric footing. The relational constraints were simply used to gate the contributions to the log-likelihood function for the alignment errors.

To overcome this shortcoming, in this paper we provide a new framework for the maximum likelihood matching of 2D point-sets which allows a symmetric linkage between alignment and correspondence. Specifically, we aim to realise interleaved iterative steps which communicate via an integrated utility measure. The utility measure is the cross-entropy between the probability distributions for alignment and correspondence.

We consider two matching applications. The first of these involves Procrustes alignment. By casting the cross-entropy in terms of matrices, we realise optimisation via dual singular value decompositions. The first of these transforms the point-set positions so as to locate an alignment that maximises the weighted correlation between the point-set co-ordinates. The second singular value decomposition updates the set of correspondence probabilities that maximise the weighted correlation between the edge-sets of the adjacency graphs for the point-sets. These processes are interleaved and iterated to convergence.

In the second application we apply the new matching framework to the non-rigid alignment of point-sets which deform according to a point-distribution model. This is a class of deformable shape model [4,12,18] recently developed by Cootes et al. [5]. The idea underpinning point-distribution models is to learn the modes of variation of point-patterns by computing the eigenmodes of the co-variance matrix for a set of training examples. The eigenvectors of the co-variance matrix define directions in which the points can move with respect to the mean-pattern. Once trained in this way, a point-distribution model can be fitted to data by estimating the proportions of each eigenmode that minimise the distance between the data and the aligned model. When carefully trained, the method can be used to model quite complex point deformations. Moreover, the modal deformations defined by the eigenvectors can be reconciled with plausible natural modes of shape variation. While much effort has been expended in improving the training of PDMs (point-distribution model), there are a number of shortcomings that have limited their effective matching to noisy point-data. First, in order to estimate the modal displacements necessary to align a point-distribution model with data, the data points must be labelled. In other words, the correspondences between points in the model and the data must be known a priori. Second, the point-sets must be of the same size. As a result, point-distribution models cannot be aligned reliably to point-sets which are subject to contamination or dropout. Finally, information concerning the relational arrangement of the points is overlooked. Our new matching framework provides a natural way of overcoming these shortcomings.

Our aim in this paper is to present a statistical framework for aligning point-distribution models. We depart from the conventional treatment in three ways. First, we deal with unlabelled point-sets. That is to say we commence without knowledge of the correspondences between the model and the data to be matched. Our second contribution is to deal with the case in which the model and the data contain different numbers of points. This might be due to poor feature location or contamination by noise. Finally, we aim to exploit constraints on the relational arrangement of the points to improve the fitting process.

## 2. Point-Sets

Our goal is to recover the co-ordinate transformation that best aligns a set of image feature points $\mathbf{w}$ with their counterparts in a model $\mathbf{z}$. In order to do this, we represent each point in the image dataset by a co-ordinate position vector $\vec{w}_i = (x_i, y_i)^{\mathrm{T}}$ where $i$ is the point index. In the interests of brevity we will denote the entire set of image points by $\mathbf{w} = \{\vec{w}_i \ \forall i \in \mathcal{D}\}$ where $\mathcal{D}$ is the point index-set. The corresponding fiducial points constituting the model are similarly represented by $\mathbf{z} = \{\vec{z}_j \ \forall j \in \mathcal{M}\}$ where $\mathcal{M}$ denotes the index-set for the model feature-points $\vec{z}_j$.

Later on we will show how the two point-sets can be aligned using singular value decomposition. In order to establish the required matrix representation of the alignment process, we construct two co-ordinate matrices from the point position vectors. The data-points are represented by the matrix $D = (\vec{w}_1 \ \vec{w}_2 \ \cdots \ \vec{w}_{|D|})$ whose columns are the co-ordinate position vectors. The corresponding point-position matrix for the model is $M = (\vec{z}_1 \ \vec{z}_2 \ \cdots \ \vec{z}_{|M|})$. The model-points are brought into alignment with the data-points. Later we consider how this may be done using Procrustes analysis and a point-distribution model. There are also alternatives not considered in this paper, including affine and perspective transformations. The alignment transformation is specified by a vector of parameters. In the case of Procrustes analysis, these parameters are the translation, scaling, and rotation of the points. In the case of the point-distribution model, the parameters are the proportions of the eigenvectors of the variance–covariance matrix of the training shapes. At iteration $n$ of the algorithm the parameter-vector is denoted by $r^{(n)}$. The result of applying this transformation to the set of model-points is an updated matrix of co-ordinates $M^{(n)} = (\vec{z}_1^{(n)} \ \vec{z}_2^{(n)} \ \cdots \ \vec{z}_{|M|}^{(n)})$.

One of our goals in this paper is to exploit structural constraints to improve the recovery of alignment parameters from sets of feature points. To this end we represent point adjacency using a neighbourhood graph. There are many alternatives including the N-nearest neighbour graph, the Delaunay graph, the Gabriel graph, and the relative neighbourhood graph. Because of its well documented robustness to noise and change of viewpoint, we adopt the Delaunay triangulation as our basic representation of image structure [8]. We establish Delaunay triangulations on the data and the model, by seeding Voronoi tessellations from the feature-points.

The process of Delaunay triangulation generates relational graphs from the two sets of point-features. More formally, the point-sets are the nodes of a data graph $G_{\mathrm{D}} = \{\mathcal{D}, E_{\mathrm{D}}\}$ and a model graph $G_{\mathrm{M}} = \{\mathcal{M}, E_{\mathrm{M}}\}$. Here $E_{\mathrm{D}} \subseteq \mathcal{D} \times \mathcal{D}$ and $E_{\mathrm{M}} \subseteq \mathcal{M} \times \mathcal{M}$ are the edge-sets of the data and model graphs. Later on we will cast our optimisation process into a matrix representation. Here we use the notation

$$E_{\mathrm{D}}(i, i') = \begin{cases} 1 & \text{if } i = i' \text{ or if } (i, i') \in E_{\mathrm{D}}, \\ 0 & \text{otherwise} \end{cases}$$

to represent the adjacency matrix for the data graph and

$$E_{\mathrm{M}}(j, j') = \begin{cases} 1 & \text{if } j = j' \text{ or if } (j, j') \in E_{\mathrm{M}}, \\ 0 & \text{otherwise} \end{cases}$$

be the corresponding matrix for the model graph. We represent the state of correspondence between the two graphs using the function $f : \mathcal{D} \rightarrow \mathcal{M}$ from the nodes of the model graph onto the nodes of the data graph. To facilitate the matrix development, we represent the correspondences by a matrix of assignment variables $S^{(n)}$ whose elements convey the following meaning:

$$S_{i,j}^{(n)} = \begin{cases} 1 & \text{if } f^{(n)}(i) = j, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $n$ is the iteration index.

## 3. Dual step matching algorithm

We characterise the matching problem in terms of separate probability distributions for alignment and correspondence. In the case of alignment, the distribution models the registration errors between the data-point positions and their counterparts in the model. The correspondence process on the other hand captures the consistency of the pattern of matching assignments to the graph representing the point-sets. In Section 3.1 we will detail a method for modelling the distribution of correspondence errors associated with the matrix of assignment variables $S^{(n)}$, while in Sections 4 and 5 we, respectively, model the distribution of alignment error associated with Procrustes normalisation and the fitting of a point-distribution model. For now, suppose that $p_{i,j}^{(n)}$ is the probability distribution associated with the alignment errors for nodes $i$ of the data graph and node $j$ of the model graph. From this probability distribution we can compute the a posteriori probability $P_{i,j}^{(n)}$ that node $i$ from the data graph is in alignment with node $j$ from the model graph given the current alignment parameters at iteration $n$. Similarly, $q_{i,j}^{(n)}$ is the probability distribution associated with the matrix of correspondence indicators $S^{(n)}$ when node $i$ is placed in correspondence with node $j$. Again, $Q_{i,j}^{(n)}$ is the a posteriori correspondence probability computed from this probability distribution given the matrix of correspondences indicators $S^{(n)}$ at iteration $n$. With these ingredients the utility measure which we aim to maximise in the dual alignment and correspondence steps is

$$\mathcal{E} = \sum_{i \in \mathcal{D}} \sum_{j \in \mathcal{M}} \left[ Q_{i,j}^{(n)} \ln p_{i,j}^{(n+1)} + P_{i,j}^{(n)} \ln q_{i,j}^{(n+1)} \right]. \tag{2}$$

In other words, the two processes interact via a symmetric expected log-likelihood function. The correspondence probabilities weight contributions to the expected log-likelihood function for the alignment errors, and vice versa. In our previous work, we showed how the first term arises through the gating of the log-likelihood function of the EM algorithm [6].

The alignment parameters and correspondence indicators are recovered via the dual maximisation equations

$$r^{(n+1)} = \arg\max_{M} \sum_{i \in \mathcal{D}} \sum_{j \in \mathcal{M}} Q_{i,j}^{(n)} \ln p_{i,j}^{(n+1)} \tag{3}$$

and

$$S^{(n+1)} = \arg\max_{S} \sum_{i\in\mathcal{D}} \sum_{j\in\mathcal{M}} P_{i,j}^{(n)} \ln q_{i,j}^{(n+1)}. \tag{4}$$

At the point it is worth pausing to consider the relationship between our unified matching framework and some alternatives. For the conventional EM algorithm the alignment process aims to find the transformation parameters which satisfy the condition

$$\vec{r}^{(n+1)} = \arg\max_{M} \sum_{i\in\mathcal{D}} \sum_{j\in\mathcal{M}} P_{i,j}^{(n)} \ln p_{i,j}^{(n+1)}. \tag{5}$$

By contrast the update equations for the Cross and Hancock [6] dual-step EM algorithm are

$$\vec{r}^{(n+1)} = \arg\max_{M} \sum_{i\in\mathcal{D}} \sum_{j\in\mathcal{M}} P_{i,j}^{(n)} Q_{i,j}^{(n)} \ln p_{i,j}^{(n+1)} \tag{6}$$

and

$$S^{(n+1)} = \arg\max_{S} \sum_{i\in\mathcal{D}} \sum_{j\in\mathcal{M}} q_{i,j}^{(n+1)} P_{i,j}^{(n)}. \tag{7}$$

Hence the a posteriori correspondence probabilities are used to weight contributions to the expected log-likelihood function for the alignment process. The correspondences are recovered so as to maximise the a posteriori probability. Our new framework therefore places correspondence and alignment on a more symmetric footing.

### 3.1. Correspondences

To model the distribution of correspondence errors, we draw on the framework developed in our recent paper [17]. In this paper we develop a model for the probability distribution for the observed set of correspondences between the nodes of the data and the model graphs. Our model draws on the recent work of Wilson and Hancock [28] and assumes that the observed data-graph nodes are derived from the model-graph nodes through a Bernoulli distribution. The parameter of this distribution is the probability of correspondence error $P_e$. The idea behind this model is that the data-graph node $i$ can emit a symbol $j$ drawn from the set of model-graph nodes. The probability that this symbol is the correct correspondence is $1 - P_e$ while the probability that it is in error is $P_e$. To gauge the correctness of the emitted symbol, we check whether the nodes $i$ and $i'$ of the data-graph are matched to a valid edge $(j, j') \in E_m$ of the model-graph. To test for edge-consistency, we make use of the quantity $E_{\mathrm{D}}(i, i')E_{\mathrm{M}}(j, j')s_{i'j'}$. This is unity if the label-assignment $f(i') = j'$ can be made to node $i'$, in such a way that the data-graph edge $(i, i') \in E_{\mathrm{D}}$ is matched to an edge $(j, j') \in E_{\mathrm{M}}$ of the model-graph. When this condition is not met then the quantity is zero.

To recover the correspondences, we pursue the matrix decomposition method recently proposed by Luo and Hancock [16]. To make this paper more self contained, we briefly review the method in this section. What should be mentioned is that to make this paper consistent, we have changed the notations which were used in the above paper.

By accepting Wilson and Hancock's[28] Bernoulli distribution model of correspondences between two point-sets and introduce the following quantity

$$E_{\mathrm{D}}(i',i)E_{\mathrm{M}}(j',j)s_{i,j} = \begin{cases} 1 & \text{if}(i',i) \in E_{\mathrm{D}} \text{ and } (j',j) \in E_{\mathrm{M}}, \text{ and } s_{i,j} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

the probability distribution for the assignment variables is

$$q_{i,j}^{(n)} = K_a \exp\left[ -\mu \sum_{i' \in \mathcal{D}} \sum_{j' \in \mathcal{M}} E_{\mathrm{D}}(i,i')E_{\mathrm{M}}(j,j')s_{i'j'} \right], \tag{9}$$

where $K_a = P_e^{|\mathcal{M}|\times|\mathcal{D}|}$ and $\mu = \ln\frac{1-P_e}{P_e}$ are constants.

Our aim is to recover correspondences that maximise the quantity

$$\mathcal{E}_c = \sum_{i \in \mathcal{D}} \sum_{j \in \mathcal{M}} P_{i,j}^{(n)} \ln q_{i,j}^{(n+1)}. \tag{10}$$

By substituting Eq. (9) into the above equation, the correspondence step reduces to one of maximising the quantity

$$\mathcal{F}_c = \sum_{i \in \mathcal{D}} \sum_{i' \in \mathcal{D}} \sum_{j \in \mathcal{M}} \sum_{j' \in \mathcal{M}} P_{i,j}^{(n)} E_{\mathrm{M}}(j,j')s_{j',i'}^{(n+1)} E_{\mathrm{D}}(i',i)^{\mathrm{T}}, \tag{11}$$

where $E_{\mathrm{D}}(i,i')$ and $E_{\mathrm{M}}(j,j')$ are the elements of the adjacency matrices for the data and model graphs. In more compact notation, the updated matrix of correspondence indicators $S^{(n+1)}$ satisfies the condition

$$S^{(n+1)} = \arg\max_{S} \mathrm{Tr}[E_{\mathrm{D}}^{\mathrm{T}} P^{(n)} E_{\mathrm{M}} S^{(n)T}] \tag{12}$$

where $P^{(n)}$ is a matrix whose elements are the alignment probability $P_{i,j}^{(n)}$. In other words, the utility measure gauges the degree of correlation between the edge-sets of the two graphs under the permutation structure induced by the alignment and correspondence probabilities. Using the method outlined in [16], we recover the matrix of assignment variables that maximises $\mathcal{F}_c$ by performing the singular value decomposition $E_{\mathrm{D}}P^{(n)}E_{\mathrm{M}} = V\Delta U^{\mathrm{T}}$, where $\Delta$ is a diagonal matrix and $U$ and $V$ are orthogonal matrices. The matrices $U$ and $V$ are used to compute a matrix $R^{(n+1)} = VEU^{\mathrm{T}}$ which maximises $Tr(E_{\mathrm{D}}^{\mathrm{T}} P^{(n)} E_{\mathrm{M}} S^{\mathrm{T}})$. The updated set of correspondence indicators is

$$s_{i,j}^{(n+1)} = \begin{cases} 1 & \text{if } R_{i,j} = \arg\max_{i'j'} R_{i',j'}, \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

The a posteriori probabilities are updated in an expectation step. As outlined in [16], the updated probabilities are

$$Q_{i,j}^{(n+1)} = \frac{q_{i,j}^{(n)} \pi_j^{(n)}}{\sum_{j' \in \mathcal{M}} q_{i,j'}^{(n)} \pi_{j'}^{(n)}}, \tag{14}$$

where

$$\pi_j^{(n)} = \frac{1}{|\mathcal{D}|} \sum_{i \in \mathcal{D}} Q_{i,j}^{(n)}. \tag{15}$$

## 4. Rigid point-set alignment

In this paper, we tackle two kinds of point pattern matching problems. When the deformation is rigid, we use the Procrustes alignment technique to match the point sets. If there exists non-rigid deformation, the Procrustes alignment will fail to match the point sets. We will derive a new alignment method to match point sets with non-rigid deformations by appealing the point-distribution models. For the sake of completeness, we summarise the Procrustes alignment method next. Complete details of the method can be found in a recent paper [17] where we develop an EM algorithm for Procrustes alignment. It should be stressed that the method reported here differs from that presented in this recent paper. Here, we are interested in augmenting the alignment process with structural constraints provided by a neighbourhood graph, and use the cross entropy between the probability distributions for correspondence and alignment, rather than using the expected log-likelihood function as a utility measure. In the next subsection, we extend this discussion to incorporate non-rigid alignment by using a point-distribution model.

Alignment is achieved via maximisation of the quantity

$$\mathcal{E}_a = \sum_{i \in \mathcal{D}} \sum_{j \in \mathcal{M}} Q_{i,j}^{(n)} \ln p_{i,j}^{(n+1)}. \tag{16}$$

By assuming the point sets alignment error to be distributed according to a Gaussian, we have

$$p_{i,j}^{(n)} = \frac{1}{2\pi\sqrt{|\Sigma|}} \exp\left[ -\frac{1}{2} \left( \vec{w}_i - \vec{z}_j^{(n)} \right)^{\mathrm{T}} \Sigma^{-1} \left( \vec{w}_i - \vec{z}_j^{(n)} \right) \right], \tag{17}$$

where $\Sigma$ is the covariance matrix for the alignment error. With this distribution at hand, our next step is to minimise the quantity

$$\mathcal{F}_a = \sum_{i \in \mathcal{D}} \sum_{j \in \mathcal{M}} Q_{i,j}^{(n)} \left( \vec{w}_i - \vec{z}_j^{(n+1)} \right)^{\mathrm{T}} \left( \vec{w}_i - \vec{z}_j^{(n+1)} \right), \tag{18}$$

where $Q_{i,j}^{(n)}$ is the probability of point $i$ in correspondence with point $j$.

We would like to recover the maximum likelihood alignment parameters by applying Procrustes normalisation to the two point-sets. This involves performing singular value decomposition of a point-correspondence matrix. In order to develop the necessary formalism, we rewrite the weighted squared error criterion using a matrix

representation. Suppose that $Q^{(n)}$ is the $|\mathcal{D}| \times |\mathcal{M}|$ correspondence probability matrix whose elements are the a posteriori correspondence probabilities $Q_{i,j}^{(n)}$. With this notation the quantity $\mathcal{F}_a$ can be expressed in the following matrix form:

$$\mathcal{F}_a = \mathrm{Tr}[D^{\mathrm{T}}D] - 2\mathrm{Tr}[DQ^{(n)}M^{(n+1)\mathrm{T}}] + \mathrm{Tr}[M^{(n+1)\mathrm{T}}M^{(n+1)}]. \tag{19}$$

Since the first and third terms of this expression do not depend on the alignment of the point-sets we can turn our attention to maximising the quantity

$$\hat{\mathcal{F}}_a = \mathrm{Tr}[DQ^{(n)}M^{(n+1)\mathrm{T}}], \tag{20}$$

where $M^{(n+1)}$ is the revised matrix of aligned model point positions.

This quantity is in fact a weighted measure of correlation between the point-sets under the current alignment estimate. Here it is the structurally based correspondence probability matrix $Q^{(n)}$ that weights the correlation. In our previously reported work on using the EM algorithm to perform Procrustes alignment, we focused on maximising the quantity $DP^{(n)}M^{(n+1)\mathrm{T}}$. Hence, it is now the a posteriori correspondence probabilities that fulfill the weighting function rather the alignment probabilities.

The quantity $\hat{\mathcal{F}}_a$ can be maximised by performing a singular value decomposition. The procedure is as follows. The matrix $DQ^{(n)}M^{(n+1)\mathrm{T}}$ is factorised into a product of three new matrices $U$, $V$, and $\Delta$, where $\Delta$ is a diagonal matrix whose elements are either zero or positive, and $U$ and $V$ are orthogonal matrices. The factorisation is as follows:

$$DQ^{(n)}M^{(n+1)\mathrm{T}} = U\Delta V^{\mathrm{T}}. \tag{21}$$

The matrices $U$ and $V$ define a rotation matrix $\Theta$ which aligns the principal component directions of the point-sets $M$ and $D$. The rotation matrix is equal to

$$\Theta = VU^{\mathrm{T}}. \tag{22}$$

With the rotation matrix at hand we can find the Procrustes alignment which maximises the correlation of the two point sets. The procedure is to first bring the centroids of the two point-sets into correspondence. Next, the data points are scaled so that they have the same variance as those of the model. Finally, the scaled and translated point-sets are rotated so that their correlation is maximised.

To be more formal the centroids of the two point-sets are

$$\langle \vec{w} \rangle = \frac{\sum_{i \in \mathcal{D}} \vec{w}_i}{|\mathcal{D}|} \tag{23}$$

and

$$\langle \vec{z}^{(n)} \rangle = \frac{\sum_{j \in \mathcal{M}} \vec{z}_j^{(n)}}{|\mathcal{M}|}. \tag{24}$$

The corresponding covariance matrices are

$$\Sigma_{\mathrm{D}} = \frac{\sum_{i \in \mathcal{D}} \left( \vec{w}_i - \langle \vec{w} \rangle \right) \left( \vec{w}_i - \langle \vec{w} \rangle \right)^{\mathrm{T}}}{|\mathcal{D}|} \tag{25}$$

and

$$\Sigma_{\mathrm{M}}^{(n)} = \frac{\sum_{j\in\mathcal{M}} \left(\vec{z}_j^{(n)} - \langle\vec{z}^{(n)}\rangle\right)\left(\vec{z}_j^{(n)} - \langle\vec{z}^{(n)}\rangle\right)^{\mathrm{T}}}{|\mathcal{M}|}. \tag{26}$$

With these ingredients the update equation for re-aligning the data-points is

$$\vec{z}_j^{(n+1)} = \langle\vec{w}\rangle + \sqrt{\frac{\mathrm{Tr}\Sigma_{\mathrm{D}}}{\mathrm{Tr}\Sigma_{\mathrm{M}}}} V U^{\mathrm{T}}\left(\vec{z}_j^{(n)} - \langle\vec{z}^{(n)}\rangle\right). \tag{27}$$

With the re-alligned point positions at hand, the a posteriori alignment probabilities can be updated using the Bayes rule. The revised values are

$$P_{i,j}^{(n+1)} = \frac{p_{i,j}^{(n)}\pi_j^{(n)}}{\sum_{j'\in V_{\mathrm{M}}} p_{i,j'}^{(n)}\pi_{j'}^{(n)}}, \tag{28}$$

where

$$\pi_j^{(n)} = \frac{1}{|\mathcal{D}|}\sum_{i\in\mathcal{D}} P_{i,j}^{(n)}. \tag{29}$$

To conclude the section, we describe the algorithm for the unified correspondence and rigid point-set alignment. The steps are as follows:

(1) Set the parameters. In this paper, we set $\sigma = 30$, $N = 10$.
(2) Calculate the initial alignment probabilities and correspondence probabilities by using the normalised Gaussian weighted inter-point distances measure

$$P_{i,j}^{(0)} = \frac{\exp(-\epsilon_{i,j}^2/2\sigma^2)}{\sum_{i\in\mathcal{D}}\sum_{j\in\mathcal{M}}\exp(-r_{i,j}^2/2\sigma^2)},$$

where $\epsilon_{i,j}^2 = (\vec{w}_i - \vec{z}_j)^{\mathrm{T}}(\vec{w}_i - \vec{z}_j)$ is the Euclidean distance between point $i$ and point $j$.
(3) Recover the optimal correspondences using Eq. (12) by SVD decomposition of the matrix $E_{\mathrm{D}}PE_{\mathrm{M}}$. Set the correspondence indicators using Eq. (13).
(4) Update the a posteriori correspondence probabilities using Eqs. (14) and (15).
(5) Align the two point-sets using Eqs. (23)–(27).
(6) Update the a posteriori alignment probabilities using Eqs. (28) and (29).
(7) Repeat steps (3)–(6) until the maximum iteration $N$ is reached, or the average point-to-point Euclidean distance between the two point-sets is less than a predefined threshold.

## 5. Non-rigid point-set alignment

In this section we consider how our unified framework for alignment and correspondence can be applied to non-rigid deformations. We focus on point-distribution models, which is based on the principal component analysis(PCA) of the datasets.

### 5.1. Point-distribution models

The point-distribution model of Cootes and Taylor [5] commences from a set of training patterns. Each training pattern is a configuration of labelled point co-ordinates or landmarks. The patterns of landmark points are collected as the object in question undergoes representative changes in shape. To be more formal, each pattern of landmark points consists of $M$-labelled points whose co-ordinates are represented by the set of position co-ordinates $\{\vec{x}_1,\vec{x}_2,\ldots,\vec{x}_{\mathrm{M}}\}=\{(x_1,y_1),\ldots,(x_{\mathrm{M}},y_{\mathrm{M}})\}$. Suppose that there are $N$ patterns of landmark points. The $t$th training pattern is represented using the long-vector of landmark co-ordinates $\vec{X}_t=(x_1,y_1,x_2,y_2,\ldots,x_{\mathrm{M}},y_{\mathrm{M}})^{\mathrm{T}}$, where the subscripts of the co-ordinates are the landmark labels. For each training pattern the labelled landmarks are identically ordered. The mean landmark pattern is represented by the average long-vector of co-ordinates

$$\hat{X}=\frac{1}{N}\sum_{t=1}^{N}\vec{X}_t.$$

The covariance matrix for the landmark positions is

$$T=\frac{1}{N}\sum_{t=1}^{N}(\vec{X}_t-\hat{X})(\vec{X}_t-\hat{X})^{\mathrm{T}}. \tag{30}$$

The eigenmodes of the landmark covariance matrix are used to construct the point-distribution model. First, the eigenvalues $\lambda$ of the landmark covariance matrix are found by solving the eigenvalue equation

$$|T-\lambda I|=0,$$

where $I$ is the $2M\times 2M$ identity matrix. The eigenvector $\vec{\phi}^{\lambda_i}$ corresponding to the eigenvalue $\lambda_i$ is found by solving the eigenvector equation $T\vec{\phi}^{\lambda_i}=\lambda_i\vec{\phi}^{\lambda_i}$. According to Cootes and Taylor, the landmark points are allowed to undergo displacements relative to the mean-shape in directions defined by the eigenvectors of the covariance matrix $T$. To compute the set of possible displacement directions, the $K$ most significant eigenvectors are ordered according to the magnitudes of their corresponding eigenvalues to form the matrix of column-vectors $\Phi=(\phi^{\lambda_1}|\phi^{\lambda_2}|\cdots|\phi^{\lambda_K})$, where $\lambda_1,\lambda_2,\ldots,\lambda_K$ is the order of the magnitudes of the eigenvectors. The landmark points are allowed to move in a direction which is a linear combination of the eigenvectors. The updated landmark positions are given by

$$\vec{X}^{(n)}=\hat{X}+\Phi\vec{r},$$

where $\vec{r}$ is a vector of modal coefficients. This vector represents the free-parameters of the global shape-model.

### 5.2. Landmark displacements

The matrix formulation of the point-distribution model allows the global shape-deformation to be computed. However, in order to develop our EM-like matching

method we will be interested in individual point displacements. We will focus our attention on the displacement vector for the landmark point indexed $i$ produced by the eigenmode indexed $\lambda$. The two components of displacement are the elements of long-vector $\vec{\phi}^{\lambda}$ indexed $2i-1$ and $2i$. We denote the displacement vector by $\vec{u}_i^{\lambda} = (\phi_{2i-1}^{\lambda}, \phi_{2i}^{\lambda})^{\mathrm{T}}$. For each landmark point the set of displacement vectors associated with the individual eigenmodes are concatenated to form a $2 \times K$ displacement matrix. For the $j$th landmark, the displacement matrix is

$$\Delta_j = \left(\vec{u}_j^{\lambda_1} | \vec{u}_j^{\lambda_2} | \cdots | \vec{u}_j^{\lambda_K}\right).$$

The point-distribution model allows the landmark points to be displaced by a vector amount which is equal to a linear superposition of the displacement-vectors associated with the individual eigenmodes. To this end let $\vec{r} = (r_1, r_2, \ldots, r_K)^{\mathrm{T}}$ represent a vector of modal superposition coefficients for the different eigenmodes. With the modal superposition coefficients at hand, the position of the landmark $j$ is displaced by an amount $\Delta_j \vec{r}$ from its mean position

$$\widehat{z}_j = \frac{1}{N} \sum_{t=1}^{N} \vec{z}_j^{t}.$$

The aim in this section is to develop an iterative method for aligning the point-distribution model. At iteration $n$ of the algorithm we denote the aligned position of the landmark point $j$ by the vector

$$\vec{z}_j^{(n)} = \widehat{z}_j + \Delta_j \vec{r}^{(n)}. \tag{31}$$

We wish to align the point-distribution model represented in this way to a set of observed data-points. The data-points to be fitted are represented by an unlabelled set of $\mathcal{D}$ point position vectors $\mathbf{w} = \{\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_{|\mathcal{D}|}\}$. The size of this point set may be different to the number of landmark points $|\mathcal{M}|$ used in the training of the point-distribution model. The free parameters that must be adjusted to align the landmark points with $\mathbf{w}$ are the modal coefficients $\vec{r}$.

### 5.3. Alignment method

To develop a useful alignment algorithm we require a model for the measurement process. Here we assume that the observed position vectors, i.e., $\vec{w}_i$ are derived from the model points through a Gaussian error process. According to our Gaussian model of the alignment errors

$$p(\vec{w}_i | \vec{z}_j, \vec{r}^{(n)}) = \frac{1}{2\pi\sqrt{|\Sigma|}} \cdot \exp\left[ -\frac{1}{2} \left(\vec{w}_i - \vec{z}_j^{(n)}\right)^{\mathrm{T}} \Sigma^{-1} \left(\vec{w}_i - \vec{z}_j^{(n)}\right)\right], \tag{32}$$

where $\Sigma$ is the variance–covariance matrix for the point measurement errors. Here we assume that the position errors are isotropic, in other words the errors in the $x$ and $y$ directions are identical and uncorrelated. As a result we write $\Sigma = \sigma^2 I_2$ where $I_2$ is the $2 \times 2$ identity matrix and $\sigma^2$ is the isotropic noise variance for the point

positions. With this model, the alignment step is concerned with minimising the weighted square error measure

$$\mathcal{E}_c = \sum_{i=1}^{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{M}|} Q_{i,j}^{(n)} \left( \vec{w}_i - \vec{z}_j^{(n+1)} \right)^{\mathrm{T}} \left( \vec{w}_i - \vec{z}_j^{(n+1)} \right). \tag{33}$$

With the point-distribution model displacement process detailed earlier, the weighted squared-error becomes

$$\mathcal{E}_c = \sum_{i=1}^{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{M}|} Q_{ij}^{(n)} \left( \vec{w}_i - \hat{z}_j - \Delta_j \vec{r}^{(n+1)} \right)^{\mathrm{T}} \left( \vec{w}_i - \hat{z}_j - \Delta_j \vec{r}^{(n+1)} \right). \tag{34}$$

Our aim is to recover the vector of modal coefficients which minimize this weighted squared error. To do this we solve the system of saddle-point equations which results by setting

$$\frac{\partial \mathcal{E}_c}{\partial \vec{r}^{(n+1)}} = 0. \tag{35}$$

After applying the rules of matrix differentiation and simplifying the resulting saddle-point equations, the solution vector is

$$\vec{r}^{(n+1)} = \left( \sum_{j=1}^{|\mathcal{M}|} \Delta_j^{\mathrm{T}} \Delta_j \right)^{-1} \left\{ \sum_{i=1}^{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{M}|} Q_{ij}^{(n)} \vec{w}_i^{\mathrm{T}} \Delta_j - \sum_{j=1}^{|\mathcal{M}|} \hat{z}_j^{\mathrm{T}} \Delta_j \right\}. \tag{36}$$

Further simplification results if we note that the landmark covariance matrix $T$ is symmetric. As a result its individual eigenvectors are orthogonal to one another, i.e., $\vec{\phi}^{\alpha \mathrm{T}} \vec{\phi}^{\beta} = 0$ if $\alpha \neq \beta$. As a result the modal coefficient for the eigenmode indexed $\lambda_k$ is

$$\vec{r}_k^{(n+1)} = \sum_{i=1}^{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{M}|} Q_{i,j}^{(n)} \sum_{k=1}^{K} \vec{w}_i^{\mathrm{T}} \vec{u}_i^{\lambda_k} - \sum_{j=1}^{|\mathcal{M}|} \sum_{k=1}^{K} \hat{z}_j^{\mathrm{T}} \vec{u}_j^{\lambda_k}. \tag{37}$$

Finally, we update the a posteriori alignment probabilities. This is done by substituting the revised parameter vector into the conditional measurement distribution. Using the Bayes rule, we can re-write the a posteriori alignment probabilities using the measurement density function

$$P_{i,j}^{(n+1)} = \frac{\pi_j^{(n)} p\left( \vec{w}_i | \vec{z}_j, \vec{r}^{(n)} \right)}{\sum_{j'=1}^{|\mathcal{M}|} \pi_{j'}^{(n)} p\left( \vec{w}_i | \vec{z}_{j'}, \vec{r}^{(n)} \right)}, \tag{38}$$

where

$$\pi_j^{(n)} = \frac{1}{|D|} \sum_{i \in \mathcal{D}} P_{i,j}^{(n)}. \tag{39}$$

Upon substituting the Gaussian distribution appearing in Eq. (32), the revised alignment probabilities are related to the updated point positions in the following manner:

$$P_{i,j}^{(n+1)} = \frac{\pi_j^{(n)} \exp\left(-\frac{1}{2\sigma^2}\left(\vec{w}_i - \vec{z}_j^{(n)}\right)^{\mathrm{T}}\left(\vec{w}_i - \vec{z}_j^{(n)}\right)\right)}{\sum_{j'=1}^{|\mathcal{M}|} \pi_{j'}^{(n)} \exp\left(-\frac{1}{2\sigma^2}\left(\vec{w}_i - \vec{z}_{j'}^{(n+1)}\right)^{\mathrm{T}}\left(\vec{w}_i - \vec{z}_{j'}^{(n+1)}\right)\right)}. \tag{40}$$

As in the previous section, we present a summary of the algorithm for correspondence and non-rigid point-set alignment. To be compact, we have omitted steps (1)–(4) which are identical to the rigid case discussed in the previous section. Only steps (5)–(7) are different.

(1)–(4) As for the rigid point-set alignment method described in the previous section.
(5) Calculate the covariance matrix of landmark positions using Eq. (30), and compute the eigenvalues and the eigenvectors of the covariance matrix.
(6) From the eigenvectors compute the optimal alignment parameters as described in Section 5.2, using Eqs. (36) or (37).
(7) Compute the optimal alignment using Eq. (31).

## 6. Experiments

In this section, we provide some experimental evaluation of the new unified approach to correspondence and alignment. We separately experiment with rigid and non-rigid point-set alignment. In the case of rigid point-sets we use the iterative Procrustes alignment method outlined in Section 4. When dealing with non-rigid point-set alignment we use the point-distribution model outlined in Section 5.

### 6.1. Procrustes alignment

We commence by considering the Procrustes alignment process which we detailed in Section 4. For the experimental evaluation of the method we use a sequence of images of a model house taken from the CMU/VASC database. The images used in our study are shown in Fig. 1 and correspond to different camera viewing directions. Here we are concerned with matching the Delaunay triangulations of corner-features. We use the corner detector recently reported by Luo et al. [15] to extract point features. The detected corner features and their Delaunay triangulations are overlaid on the images.

We match the first house image in the sequence to the remaining nine images in the sequence. Fig. 2 shows the first six alignment results using the unified matching method described in this paper. From left to right, and from top to bottom, the panels in Fig. 2, respectively, show the alignment of the first image to the remaining images in the house sequence. We display the results of the alignment process by overlaying the Delaunay graph of the feature points extracted from the first image in the sequence on the remaining grey-scale images in the sequence. Table 1 summarises the alignment errors obtained by using the proposed unified matching method (referred to as UM). From these results it is clear that the method fails after the sixth image in the sequence.

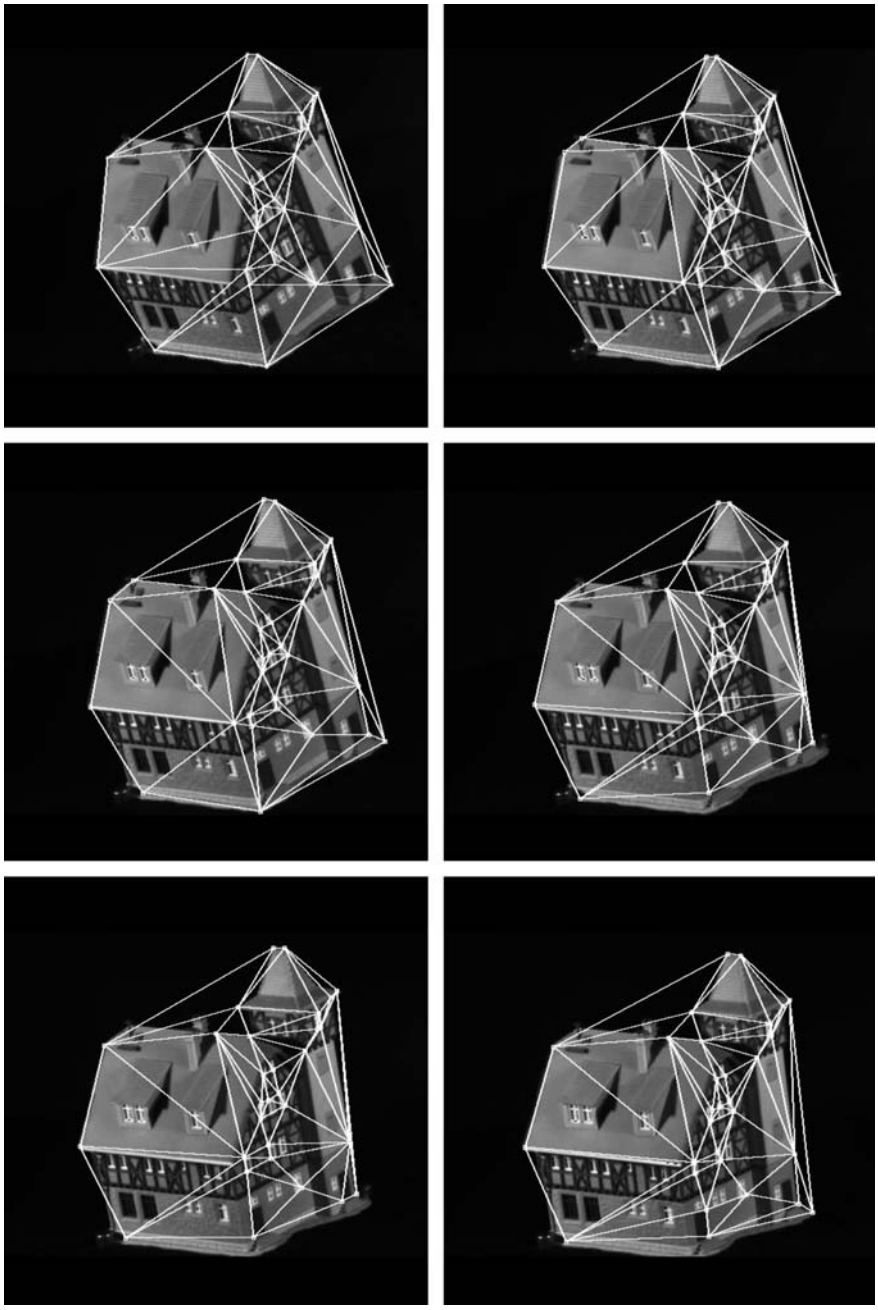

Fig. 1. Delaunay graphs overlaid on the toy house images.

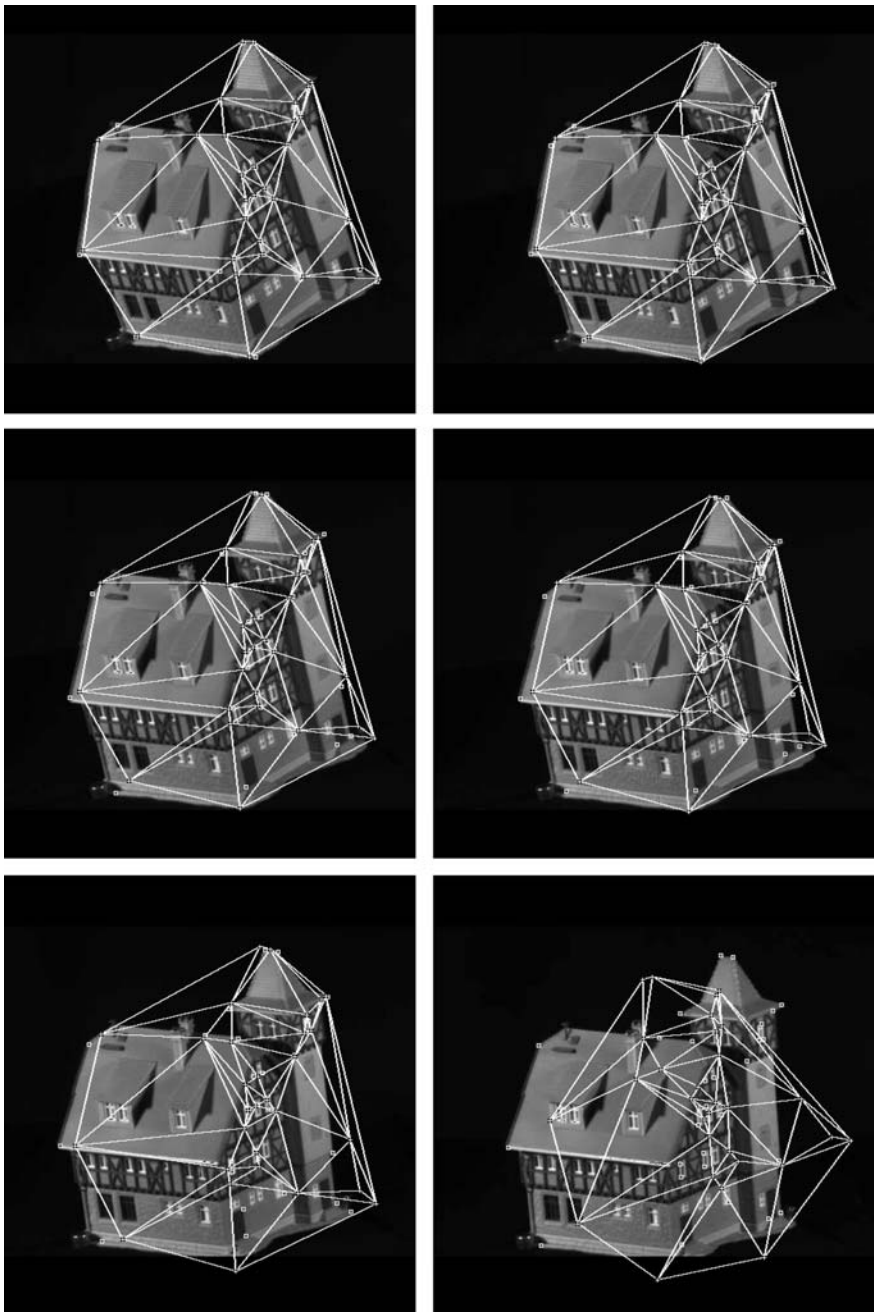

Fig. 2. House image alignment results (UM).

Table 1
Alignment errors of the unified matching (UM) method

| House index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| UM | — | 1.07 | 1.90 | 4.75 | 5.17 | 6.87 | 23.4 | 24.8 | 27.1 | 36.3 |

We have compared the alignment results obtained with the new unified matching method and those obtained by using the EM algorithm to perform iterative Procrustes analysis[17] (referred to as IP). Here we investigate the effect of removing the structural constraints (i.e., the graph) from the matching process. The EM algorithm does not use structural information to constrain the alignment process and aims to recover the set of alignment parameters that maximise the quantity $DP^{(n)}M^{(n+1)\mathrm{T}}$ rather than $DQ^{(n)}M^{(n+1)\mathrm{T}}$. In other words the weighting of the correlation between the point position matrices is controlled by the matrix of a posteriori alignment probabilities $P^{(n)}$ rather than the matrix of correspondence probabilities $Q^{(n)}$.

In Fig. 3 we show the alignment error as a function of the index number of the matched image in the sequence. The two curves show the results of applying the EM algorithm (IP) and the unified method (UM). From this data, it is clear that unified method outperforms the EM Procrustes method and the graph matching method with respect to improved alignment error.

Next we turn our attention to the effect of the unified method on correspondence. In Figs. 4–8 we show the resulting correspondences with segments connecting the

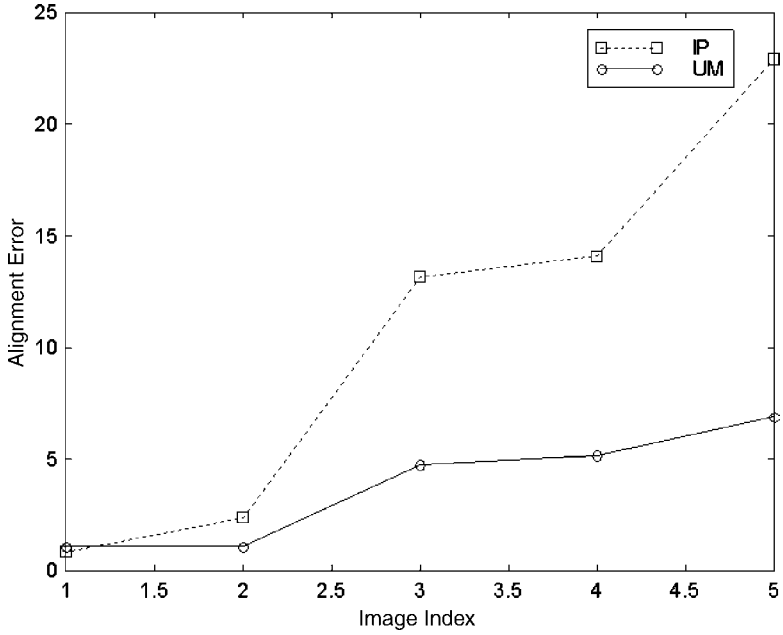


Fig. 3. Comparison of the IP and UM methods—alignment error.

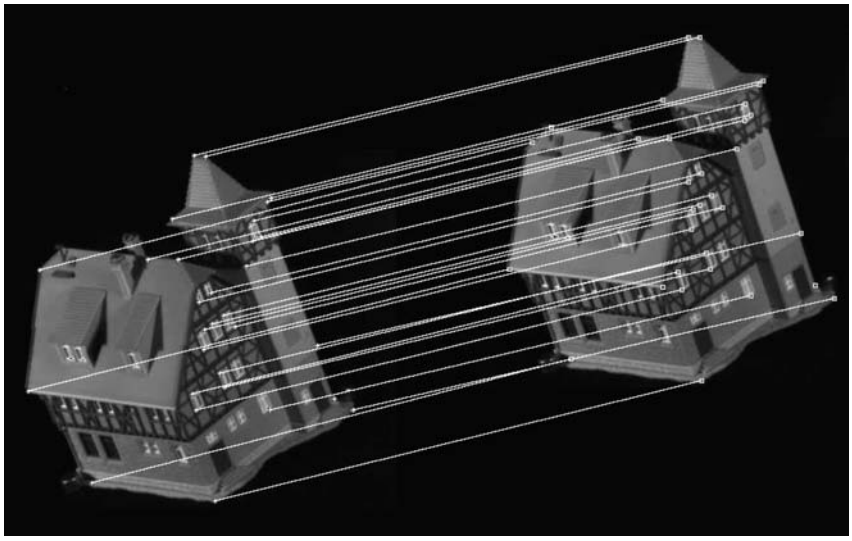

Fig. 4. Correspondences between the first and the second images.

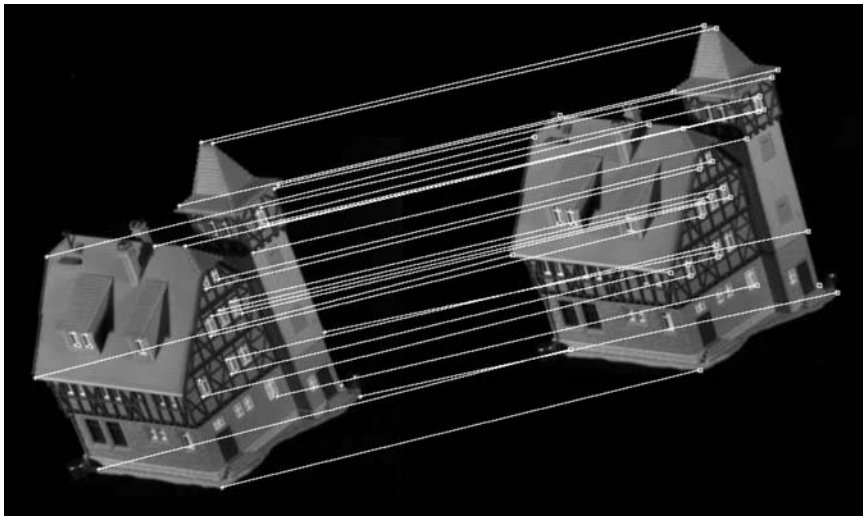

Fig. 5. Correspondences between the first and the third images.

points in the first image with their correspondences in subsequent images in the house sequence. Table 2 summarises the results. We can see that from the sixth image there is a significant number of false correspondences.

To conclude this subsection, we compare the unified method with the iterative Procrustes method and a purely structural graph matching method [16]. The graph

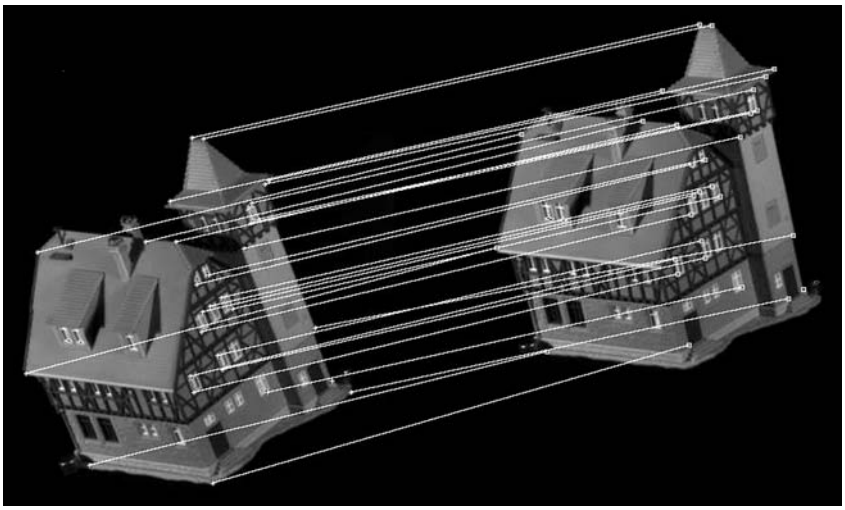

Fig. 6. Correspondences between the first and the fourth images.

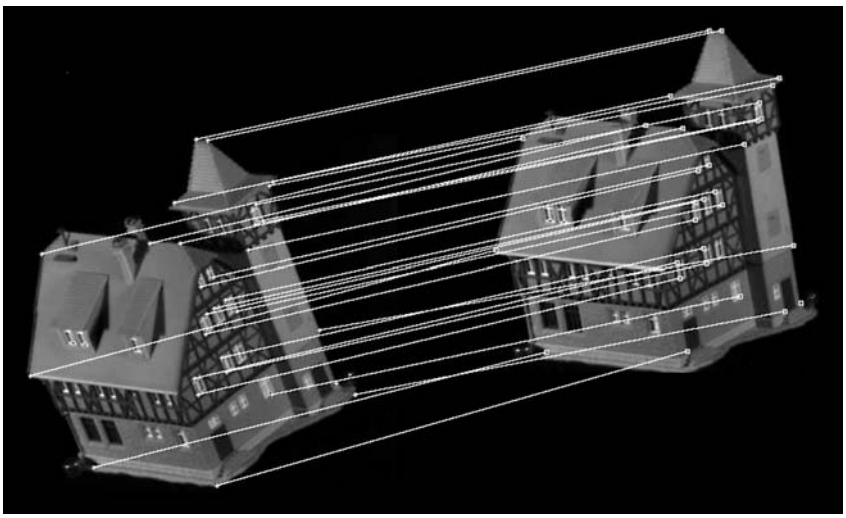

Fig. 7. Correspondences between the first and the fifth images.

matching method aims to iteratively find the matrix of correspondences $S^{(n+1)}$ which maximise the quantity $E_{\mathrm{D}}^{\mathrm{T}} Q^{(n)} E_{\mathrm{M}} S^{(n+1)\mathrm{T}}$. We denote the results obtained with this method by GM.

We compare three measures of performance. These are the fractions of correct correspondences, false correspondences, and missed correspondences. From the performance curves in Figs. 9–11, we conclude that the graph matching method (GM) outperforms the iterative EM Procrustes alignment method (IP) although it has

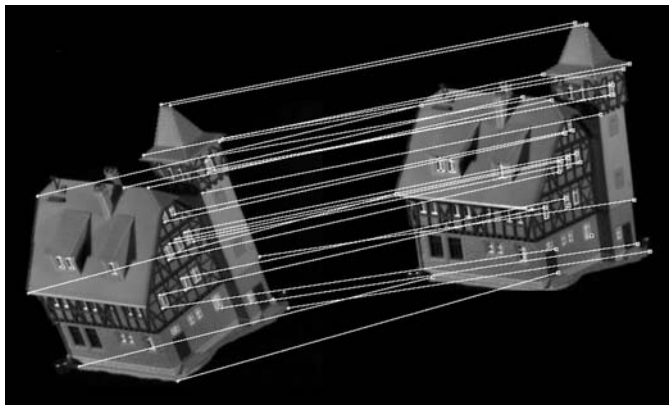

Fig. 8. Correspondences between the first and the sixth images.

Table 2
Correspondence allocation results of the UM method

| House index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| Correct | — | 29 | 28 | 29 | 29 | 26 | 7 | 4 | 4 | 5 |
| False | — | 0 | 0 | 0 | 0 | 2 | 17 | 26 | 20 | 16 |
| Missed | — | 1 | 2 | 1 | 1 | 2 | 6 | 0 | 6 | 9 |

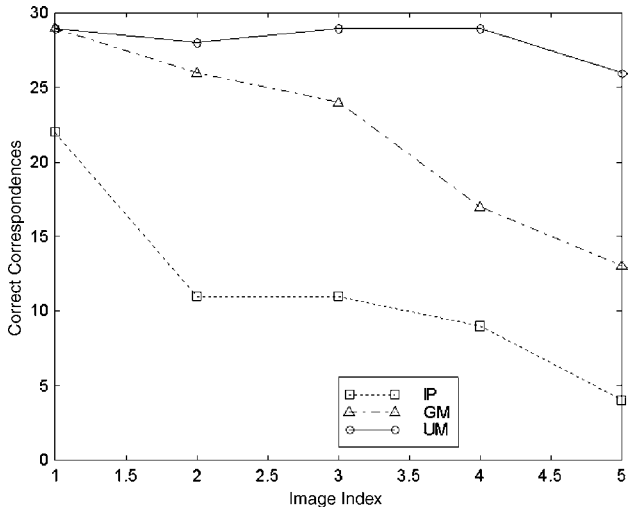


Fig. 9. Comparison of the IP, GM, and UM methods—correct correspondences.

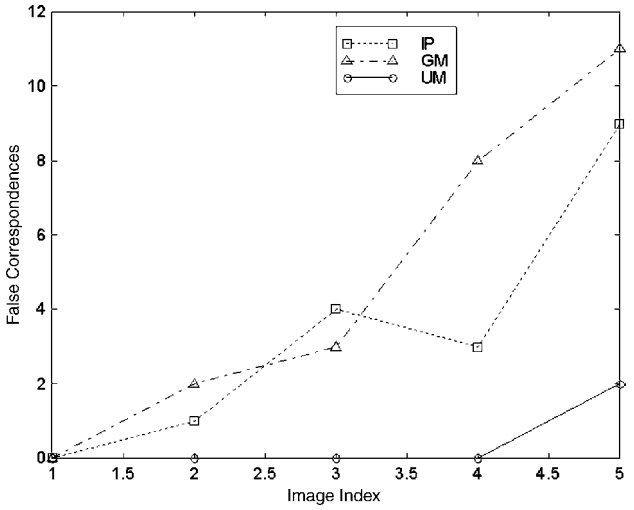


Fig. 10. Comparison of the IP, GM, and UM methods—false correspondences.

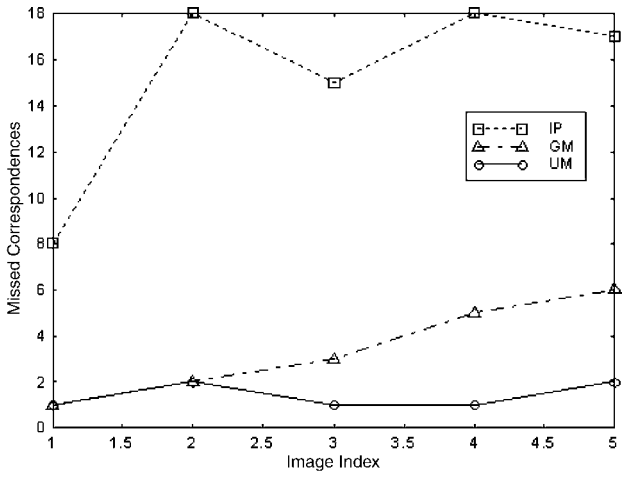


Fig. 11. Comparison of the IP, GM, and UM methods—missed correspondences.

relatively more false correspondences. However, the unified matching method (UM) is the best with respect to all three performance measures.

### 6.2. Point distribution model fitting

We have experimented with our new point-distribution model alignment method on an X-ray angiogram image sequence of a beating heart. Here the feature points are hand-labelled locations of maximum curvature on the outline of the heart.

There are 16 feature points in each image. In total we have used 19 frames to train the point-distribution model. The mean shape is shown in Fig. 12a superimposed on one of the images from the sequence; the different frames used for training are shown in Fig. 12b. In Fig. 13 we show the modal displacements corresponding to the first six eigenmodes.

An example of the alignment of the PDM to an image which was not part of the training set is shown in Fig. 14. The sequence shows the PDM iterating from the mean shape to the final alignment. The different panels in the figure show different iterations of the algorithm. The process converges in 10 iterations. In this example there is no initial correspondence. The final set of correspondences obtained are shown in Fig. 15. These are all correct.

In Figs. 16 and 17 we illustrate the structural aspect of the matching process. Fig. 16 shows the Delaunay graphs for the model point-set image and another image from the sequence. The second image contains clutter which is introduced by non-perfect segmentation. In Fig. 17 we show the result obtained when we align the points in the model with those in the second image. Here we use the positions of the points from model to initialise the fitting of the point-distribution model. The crosses represent the initial positions of the model landmark-points. The lines represent the displacements of these points resulting from the alignment of the PDM. Notice that none of the feature-points is displaced in error so that it is brought into correspondence with the noise point.

In Figs. 18 and 19 we illustrate the effect of removing the adjacency graph from the point representation and the correspondence step from the matching process. To meet this goal we simply fit the point-distribution model so as to minimise the quantity

$$\mathcal{E}_{\mathrm{EM}} = \sum_{i=1}^{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{M}|} \left[ P_{i,j}^{(n)} \ln p_{i,j}^{(n+1)} \right]. \tag{41}$$

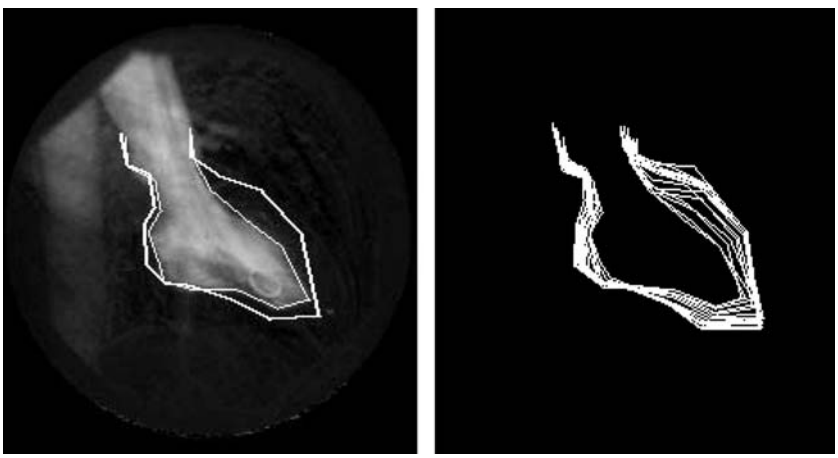

Fig. 12. Overlapped mean shape and training images.

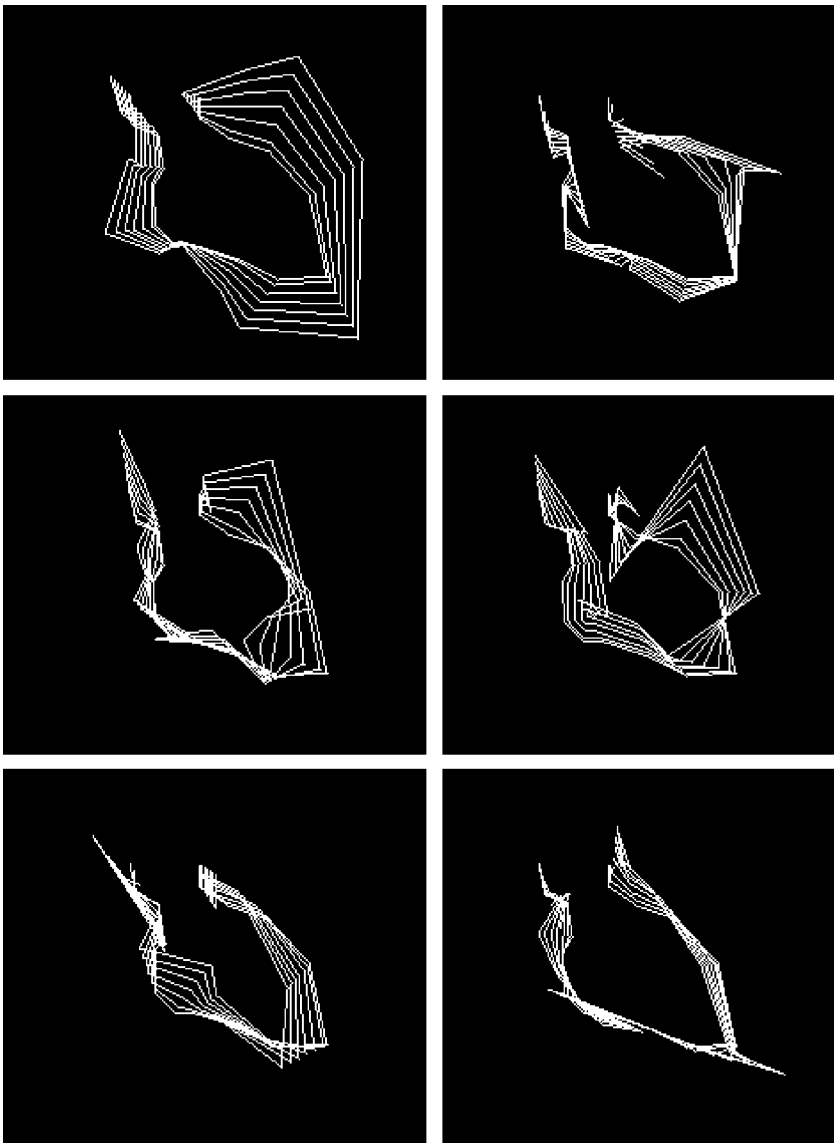

Fig. 13. Eigenmodes of the training images.

This means that we use the EM algorithm to estimate the vector of modal parameters $\vec{r}$. Fig. 18 shows the triangulated point-sets. The second point set contains a clutter point. When the EM algorithm is used to align the PDM, then this clutter point is matched in error to one of the landmark points. However, when our new unified matching framework is used to match the augmented PDM, then the clutter point finds no correspondence in the model.

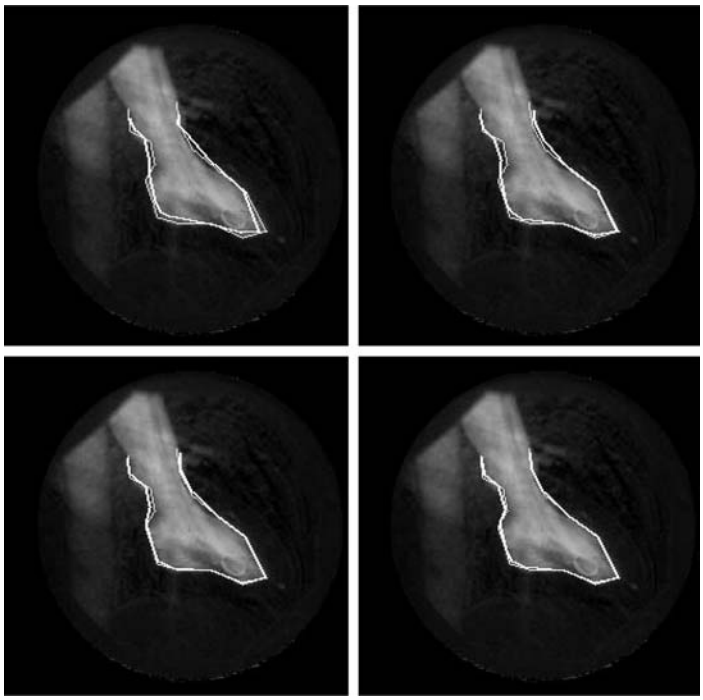

Fig. 14. Alignment results of the heart images.

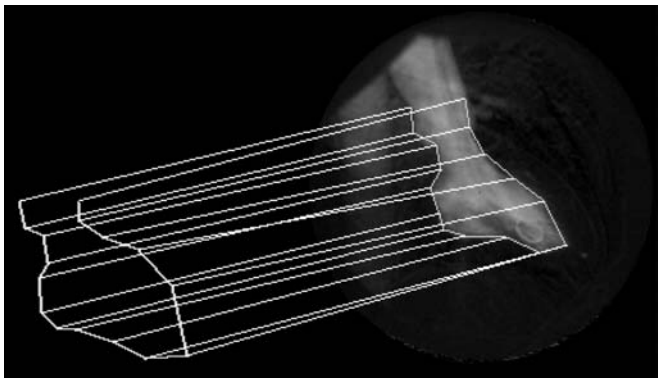

Fig. 15. Correspondences of the heart images.

Next, we turn our attention to some of the quantitative properties of the method. In Fig. 20 we show the final RMS alignment error on the number of eigenmodes (ordered according to the size of their respective eigenvalues) used in the alignment process. Here there is slow improvement once we use more than 20% of the eigenmodes in the fitting process.

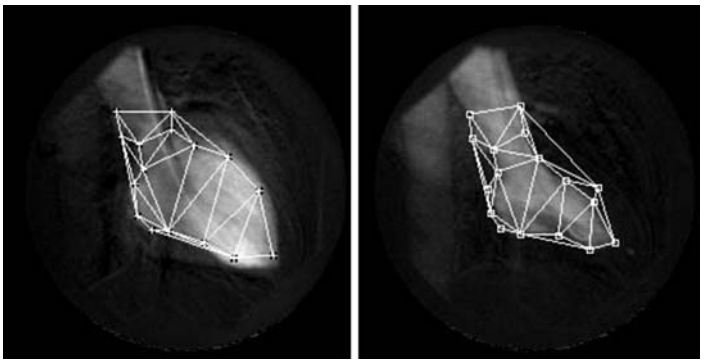

Fig. 16. Graph structure of two point sets with clutter.

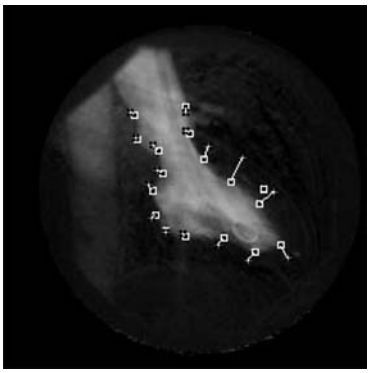

Fig. 17. Correspondence result of two point sets with clutter.

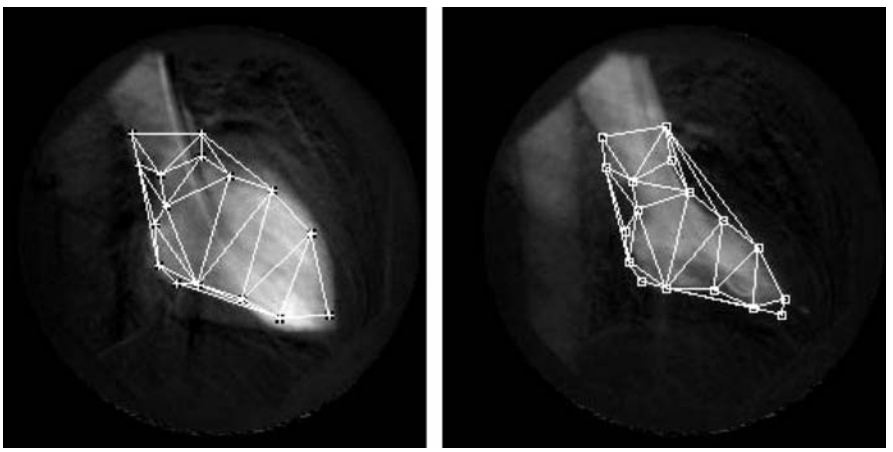

Fig. 18. Graphs for algorithm comparison.

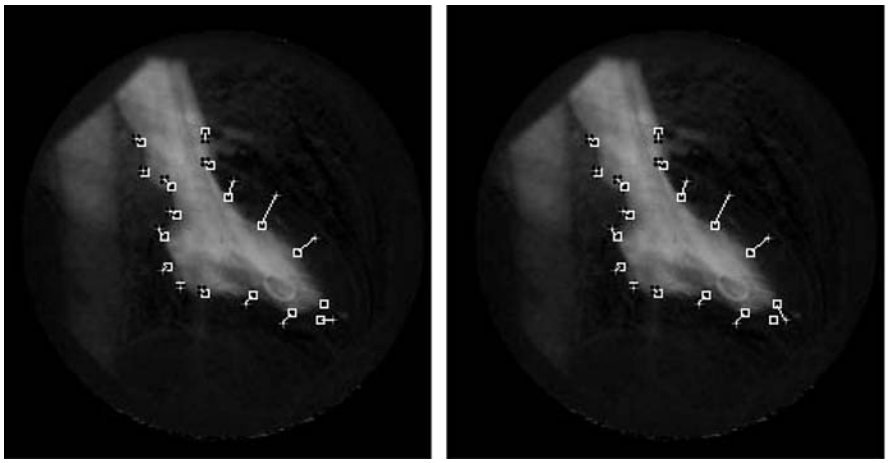

Fig. 19. Comparison of EM-PDM and GRAPH-PDM.

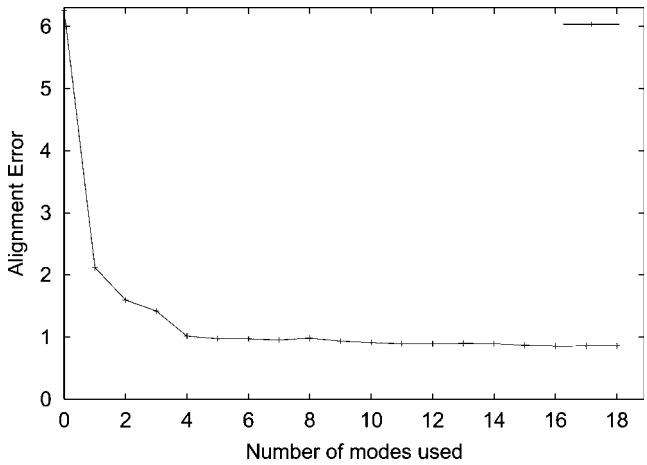


Fig. 20. The effect of number of modes used to alignment error.

In Figs. 21a and b, we investigate the effect of clutter on the matching process. In Fig. 21a we show the evolution of one of the fitted PDM parameters with iteration number. The dotted curve is the result obtained with the EM algorithm, while the solid curve is the result obtained when we use the new method reported in this paper. The main feature to notice is that both methods converge to the same parameter values, but that the convergence of the graph-based method is faster. In Fig. 21b we repeat this experiment when random noise points are added to the data. Here we should recover the same parameter values as in Fig. 21a. Both algorithms result in

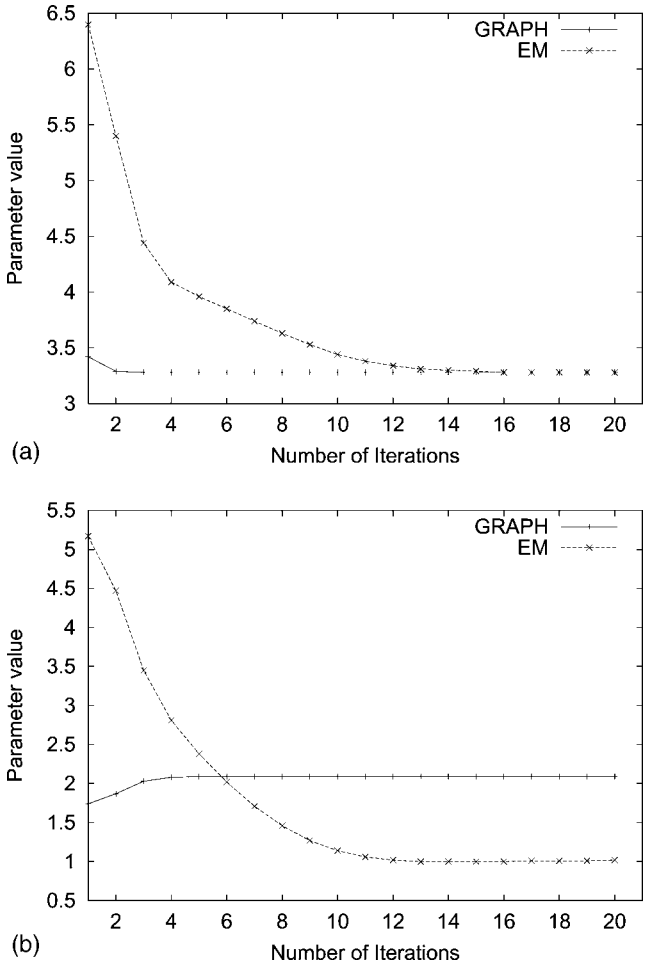


Fig. 21. Comparison of PDM parameter value: (a) without clutter; (b) with 18.75% clutter.

a significant parameter error. However, the graph-based method gives a final result which is closer to the correct answer. Moreover, its convergence is again much faster than the EM method.

Next, we investigate the effect of added clutter on the alignment error. Fig. 22 shows the alignment error as a function of iteration number for the EM algorithm and the graph-based method. In Fig. 22a the fraction of added clutter is 6.25%. Here both methods converge to a result in which the alignment error is consistent with zero. However, the graph-based method converges at a much faster rate. Fig. 22b repeats this experiment when the fraction of added clutter is 18.75%. Now both methods are subject to a substantial alignment error. However, in the case of the graph-based method this is smaller than that incurred by the EM method.

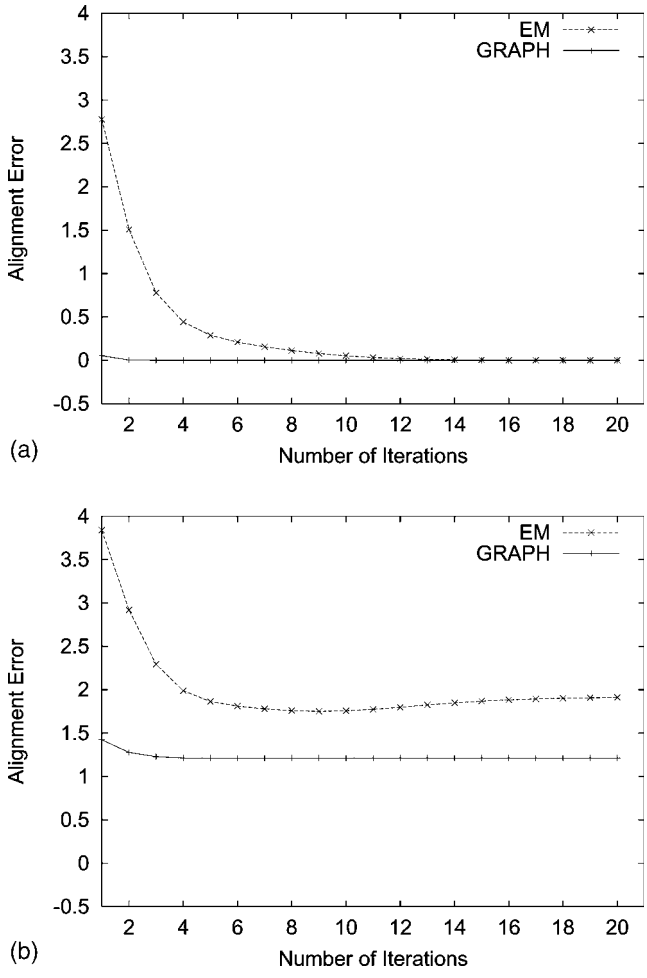


Fig. 22. Comparison of the alignment error: (a) 6.25% clutter; (b) 18.75% clutter.

## 7. Conclusions

In conclusion, we have shown how the processes of point-set alignment and correspondence analysis can be unified using a symmetric entropy. For the problem of Procrustes alignment, we have shown that by drawing on a Gaussian model of point-position errors and an exponential model of correspondence assignment errors, we are able to cast the two problems as maximisation of weighted correlation measures. In both cases the point matches can be recovered using singular value decomposition. Our new measures of point-set similarity naturally combine the ideas already developed by Scott and Longuet-Higgins and Umeyama in a single statistical utility measure. An experimental study reveals that the proposed method outperforms that

of the iterative Procrustes alignment method(IP) [17] and the matrix factorisation graph matching method(GM) [16] in terms of its ability to recover alignment parameters and correspondence.

In the case of point-distribution model alignment, we have made two contributions. First, we show how the point-distribution models can be augmented with point adjacency information. Second, we show how to fit the resulting model to noisy and unlabelled point-sets using a unified approach to correspondence and alignment. The method is shown to operate effectively when the landmark points are both unlabelled and subject to structural corruption. Although the performance is slightly degraded by increasing the clutter level, it is still reasonably rapid to converge and robust to point-set contamination.

## References

[1] H. Chui, A. Rangarajan, F.L. Bookstein, The softassign procrustes matching algorithm, in: Inf. Process. Med. Imaging, 1997, pp. 29–42.
[2] Y. Amit, A. Kong, Graphical templates for model registration, IEEE Trans. Pattern Anal. Mach. Intell. 18 (1996) 225–236.
[3] C. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1995.
[4] L.D. Cohen, I. Cohen, Finit element methods for active contour models and balloons for 2D and 3D images, IEEE Trans. Pattern Anal. Mach. Intell. 15 (11) (1993) 1131–1147.
[5] T.F. Cootes, C.J. Taylor, D.H. Cooper, J. Graham, Active shape models—Their training and application, Comput. Vision Image Understanding 61 (1995) 38–59.
[6] A.D.J. Cross, E.R. Hancock, Graph matching with a dual-step EM algorithm, IEEE Trans. Pattern Anal. Mach. Intell. 20 (11) (1998) 1236–1253.
[7] P. David, D. de Menthon, R. Duraiswami, H. Samet, Softposit: simultaneous pose and correspondence determination, in: ECCV02, 2002, page III, 698ff.
[8] O.D. Faugeras, E. Le Bras-Mehlman, J.-D. Boissonnat, Representing stereo data with the delaunay triangulation, Artif. Intell. 44 (1990) 41–87.
[9] S. Gold, A. Rangarajan, A graduated assignment algorithm for graph matching, IEEE Trans. Pattern Anal. Mach. Intell. 18 (4) (1996) 377–388.
[10] M.I. Jordan, R.A. Jacobs, Hierarchical mixture of experts and the EM algorithm, Neural Comput. 6 (1994) 181–214.
[11] F. Jurie, Solution of the simultaneous pose and correspondence problem using gaussian error model, CVIU 73 (3) (1999) 357–373.
[12] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, Internat. J. Comput. Vision 1 (4) (1988) 321–331.
[13] D.G. Kendall, Shape manifolds: Procrustean metrics and complex projective spaces, Bull. Lond. Math. Soc. 16 (1984) 81–121.
[14] M. Lades, J.C. Vorbruggen, J. Buhmann, J. Lange, C. von der Maalsburg, R.P. Wurtz, W. Konen, Distortion-invariant object-recognition in a dynamic link architecture, IEEE Trans. Comput. 42 (1993) 300–311.
[15] B. Luo, A.D.J. Cross, E.R. Hancock, Corner detection via topographic analysis of vector potential, Pattern Recognition Lett. 20 (1999) 635–650.
[16] B. Luo, E.R. Hancock, Structural graph matching using the em algorithm and singular value decomposition, IEEE PAMI 23 (10) (2001) 1120–1136.
[17] B. Luo, E.R. Hancock, Iterative procrustes alignment with the em algorithm, Image Vision Computing 20 (2002) 377–396.
[18] A.P. Pentland, S. Sclaroff, Closed-form solutions for physically based shape modeling and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 13 (7) (1991) 715–729.

[19] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape context, IEEE PAMI 24 (2002) 509–522.
[20] G.L. Scott, H.C. Longuet-Higgins, An algorithm for associating the features of 2 images, Proc. R. Soc. Lond. B 244 (1309) (1991) 21–26.
[21] L.S. Shapiro, J.M. Brady, Feature-based correspondence—an eigenvector approach, Image Vision Computing 10 (1992) 283–288.
[22] L.S. Shapiro, J.M. Brady, Rejecting outliers and estimating errors in an orthogonal-regression framework, Philos. Trans. R. Soc. A 350 (1995) 403–439.
[23] S. Ullman, The Interpretation of Visual Motion, MIT Press, Cambridge, MA, 1979.
[24] S. Umeyama, An eigendecomposion approach to weighted graph matching problems, IEEE Trans. Pattern Anal. Mach. Intell. 10 (5) (1988) 695–703.
[25] S. Umeyama, Least squares estimation of transformation parameters between two point sets, IEEE Trans. Pattern Anal. Mach. Intell. 13 (4) (1991) 376–380.
[26] S. Umeyama, Parameterised point pattern matching and its application to recognition of object families, IEEE Trans. Pattern Anal. Mach. Intell. 15 (1993) 136–144.
[27] M. Werman, D. Weinshall, Similarity and affine invariant distances between 2D point sets, IEEE Trans. Pattern Anal. Mach. Intell. 17 (8) (1995) 810–814.
[28] R.C. Wilson, E.R. Hancock, Structural matching by discrete relaxation, IEEE Trans. Pattern Anal. Mach. Intell. 19 (6) (1997) 634–648.