Pre-organizing Shape Instances for Landmark-Based Shape Correspondence

**Brent C. Munsell · Andrew Temlyakov · Martin Styner ·
Song Wang**

Abstract The major challenge in constructing a statistical shape model for a structure is shape correspondence, which identifies a set of corresponded landmarks across a population of shape instances to accurately estimate the underlying shape variation. Both global or pairwise shape-correspondence methods have been developed to automatically identify the corresponded landmarks. For global methods, landmarks are found by optimizing a comprehensive objective function that considers the entire population of shape instances. While global methods can produce very accurate shape correspondence, they tend to be very inefficient when the population size is large. For pairwise methods, all shape instances are corresponded to a given template independently. Therefore, pairwise methods are usually very efficient. However, if the population exhibits a large amount of shape variation, pairwise methods may produce very poor shape correspondence. In this paper, we develop a new method that attempts to address the limitations of global and pairwise methods. In particular, we first construct a shape tree to globally organize the population of shape instances by identifying similar shape instance pairs. We then perform pairwise shape correspondence between such similar shape instances with high accuracy. Finally, we combine these pairwise correspondences to achieve a unified correspondence for the entire population of shape instances. We evaluate the proposed method by comparing its performance to five available shape correspondence methods, and show that the proposed method achieves the accuracy of a global method with the efficiency of a pairwise method.

Keywords Shape correspondence · Statistical shape modeling · Statistical shape analysis · Shape tree

B.C. Munsell (✉) · A. Temlyakov · S. Wang
Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA
e-mail: munsell@cec.sc.edu

A. Temlyakov
e-mail: temlyaka@cec.sc.edu

S. Wang
e-mail: songwang@cec.sc.edu

M. Styner
Departments of Psychiatry and Computer Science, University of North Carolina, Chapel Hill, NC 27599, USA
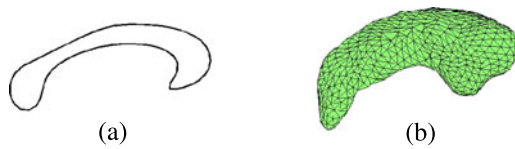e-mail: martin_styner@ieee.org

1 Introduction

Shape plays a central role in computer vision: Many structures in the real world exhibit specific shapes and, with appropriate models, such shape information can be quantized and used to facilitate matching, recognition, registration, segmentation, categorization, classification and many other computer-vision tasks (Frangi et al. 2001; Rueckert et al. 2001; Veltkamp and Hagedoorn 1999; Matthews and Schneider 2004; Marsland et al. 2008; Heimann and Meinzer 2009; Amit and Geman 1997). In particular, being capable of modeling both the base shape and possible shape variation of a structure, statistical shape models have attracted special interest in recent years, with successful applications in many computer vision and medical imaging tasks. For example, in Duta and Sonka (1998), Leventon et al. (2000), van Ginneken et al. (2002), Lekadir et al. (2007), Heimann and Meinzer (2009), statistical shape models are applied to guide the segmentation of anatomic structures that cannot be accurately segmented based solely on image intensities. In Bookstein (1989, 1997), Shenton et al. (2002), medical researchers have successfully used statistical shape models to accurately locate subtle shape

Fig. 1 Example shape instances in the form of: (**a**) 2D shape contour of the corpus callosum, and (**b**) 3D shape surface of the hippocampus

Fig. 2 Three example 2D shape contours of the human hand. Each contour has 24 landmarks, where *one white* landmark corresponds the tip of the thumb and the *other white* landmark corresponds the tip of the index finger

differences in the corpus callosum and hippocampus between schizophrenia patients and normal controls. Statistical shape models have also been successfully applied to facial identification and recognition (Milborrow and Nicolls 2008; Seshadri and Savvides 2009; Gonzalez-Jimenez and Alba-Castro 2007; Lanitis et al. 1995).

Statistical shape models (Dryden and Mardia 1998) are usually constructed from a population of shape instances of the structure of interest in the form of smooth 2D shape contours or smooth 3D shape surfaces as shown in Fig. 1. To construct a statistical shape model, the first step is to identify a set of landmarks across the population of shape instances, where a *landmark* is a point of correspondence used to examine and measure shape change (Bookstein 1989, 1991). For example, using the three 2D contours of the human hand in Fig. 2, *one white* landmark corresponds the tip of the thumb and the *other white* landmark corresponds the tip of the index finger. This landmark identification step is usually referred to as *(landmark-based) shape correspondence*, and it is critical that the identified landmarks be of the same number, well corresponded, and sufficiently dense to accurately reflect the geometry of the structure (Richardson and Wang 2005). With shape correspondence, each shape instance can be represented by a fixed-dimensional vector consisting of all its landmarks and a statistical shape model can then be constructed by using classical statistical analysis on the entire population of shape instances. In the computer vision and medical imaging communities, the most widely used and recognized statistical shape model is the point distribution model (PDM) (Cootes et al. 1995), which uses a multivariate Gaussian distribution to model the base shape and the possible shape variation.

Manually labeling corresponded landmarks across a large population of shape instances is very laborious and time consuming, especially for 3D shape surfaces. Furthermore,

manual correspondence can be very subjective and inaccurate: i.e., different experts may produce different manual correspondences. In recent years, many automatic shape-correspondence methods have been developed that attempt to address these limitations. Previous shape correspondence methods are usually developed by optimizing a specific physical or mathematical model. In Davies et al. (2002), the minimum description length (MDL) method considers the landmarks across the population of shape instances to be well corresponded when the bit length of the constructed PDM is minimum. MDL is widely recognized as a state-of-the-art method for shape correspondence in statistical shape modeling. In Bookstein (1989), Powell (1995), Wang et al. (2004), 2D thin-plate splines are used to model the non-rigid deformation between a pair of shape instances, and shape correspondence is achieved by identifying landmarks that minimize the thin-spline bending energy. In Meier and Fisher (2002), spherical harmonics are used to model the shape difference between a pair of shape instances, and shape correspondence is achieved when the spherical harmonic coefficients minimize the differences in curvature, distance, and normal direction. In Xie and Heng (2005), the medial-axis is used to identify matched contour segments between a pair of shape instances and the landmarks within these identified segments are considered to be well corresponded when an optimal one-to-one point matching cost is found.

In computer vision, there are numerous works on shape matching that may also identify corresponded landmarks between two shape instances. However, shape correspondence and shape matching have completely different goals and challenges that require different solutions. Specifically, shape correspondence is applied to shape instances of the same deformable structure where the goal is to construct a statistical shape model to accurately describe the underlying shape variation that may be very subtle. In general, shape matching is developed for recognition with the goal of identifying a set of matched shape features that can distinguish different structures (Belongie et al. 2002; Schmidt et al. 2007; Basri et al. 1995). Therefore, the landmarks identified in shape matching are usually not sufficiently dense and accurate for constructing a statistical shape model. In addition, most shape matching methods are developed to match a pair of shape instances and are not concerned with identifying landmark correspondence over the entire population. Finally, partial shape matching, i.e., matching shape instances with possible occlusions, is very important in many computer vision applications (DiIanni et al. 1996; Ozcan and Mohan 1997; Bruckstein et al. 1992; Saber et al. 2005; Veltkamp and Tanase 2005; Latecki et al. 2007; Chen et al. 2008). However, in shape correspondence the goal is to construct an accurate statistical shape model for a structure and, as in previous work on shape correspondence,
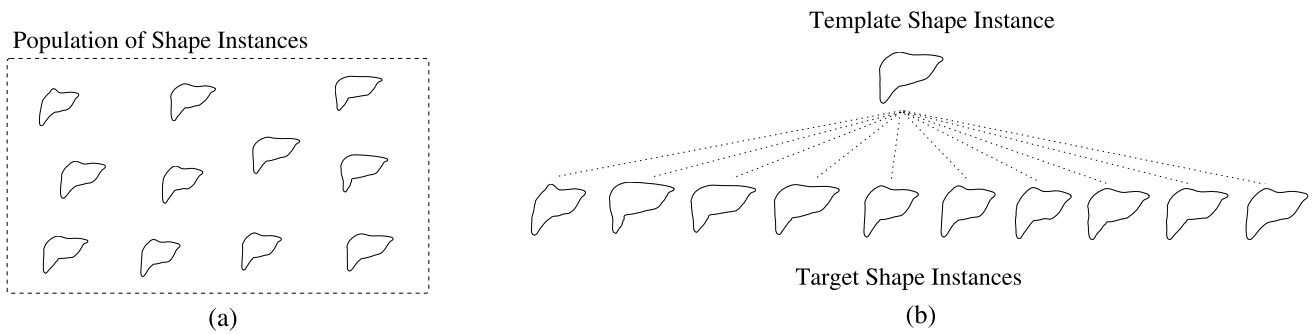
Population of Shape Instances

Template Shape Instance

Target Shape Instances

(a)

(b)

Fig. 3 (**a**) For global methods, shape correspondence is achieved when an objective function that considers the entire population of shape instances is optimized. (**b**) For pairwise methods, one template (or reference) shape instance is selected from the population and the remaining shape instances are corresponded to the template independently

we usually do not consider shape instances with occlusions (Davies et al. 2002).

In general, the available shape correspondence methods can be grouped into one of two categories: global methods and pairwise methods. For global methods a comprehensive objective function that considers the entire population of shape instances is optimized, as illustrated in Fig. 3(a). Genetic algorithms (Davies et al. 2002, 2008) and gradient descent algorithms (Heimann et al. 2005) have been used to optimize the objective function. These optimization methods may require a large number of iterations to converge and are usually vulnerable to be trapped in a local optimum. While global methods may produce a more accurate shape correspondence, they tend to be inefficient when the population size is large. For pairwise methods, as illustrated in Fig. 3(b), one shape instance in the population is designated as the template whose landmarks are fixed and, for each remaining target shape instance in the population, the pairwise method identifies a set of landmarks that are well corresponded to those on the template. Because pairwise methods only consider two shape instances at any time, they tend to be more efficient and scale favorably to the size of the population. However, since a single template shape instance is chosen from the population, pairwise methods tend to be less accurate and can produce poor shape correspondence when the population has a large amount of shape variation (Munsell et al. 2008).

Algorithm efficiency is important for shape correspondence and statistical shape modeling. As shown by the experiments in Sect. 3, even for 2D shape instances, an inefficient global method may require weeks of compute time when the population size is large. Such a method may not be suitable for time-critical applications such as those found in clinical, near-realtime tracking, or biometric settings, where the new, incoming shape instances require frequent shape correspondence to update the statistical shape model. Furthermore, while this paper is focused on 2D shape correspondence, the proposed method can be extended to 3D

shape correspondence and modeling. In the 3D case, manual correspondence is not an option and inefficient global methods may take weeks to correspond only a small population of shape instances (Dalal et al. 2010).

In this paper we attempt to address these limitations by developing a shape correspondence method that has the accuracy of a global method with the efficiency of a pairwise method. This is accomplished by first pre-organizing the input population of shape instances into a *shape tree*, where each node represents a shape instance, and two nodes connected by an edge represent two similar shape instances. Using this shape tree, we achieve shape correspondence for the entire population by corresponding and combining the pairwise shape correspondence between neighboring shape instances. In particular, the shape tree is constructed by evaluating a rough shape similarity between every pair of shape instances in the population. This important step allows us to incorporate the global information underlying the population. From this perspective, our method can be considered a global method. However, the final refined shape correspondence is performed only between neighboring shape instances that are very similar. This step ensures that the proposed method is both efficient and accurate.

Related to our work is the shape correspondence method developed in Hill and Taylor (1994), where a binary tree is used for finding an initial correspondence for the entire population. Different from our method, only the leaf nodes in the binary tree represent shape instances in the population, and each non-leaf node represents the average shape of its two children. In this binary tree, the neighboring non-leaf shape instances may not be similar, especially for those near the root. Therefore, this initial shape correspondence may not be very accurate and a separate method that does not include the binary tree is used to improve the correspondence accuracy (Hill and Taylor 1994). Also related to our work is the registration and motion analysis method developed in Cristinacce et al. (2008), where a set of prototype facial images are organized into a shortest-path tree. Different from
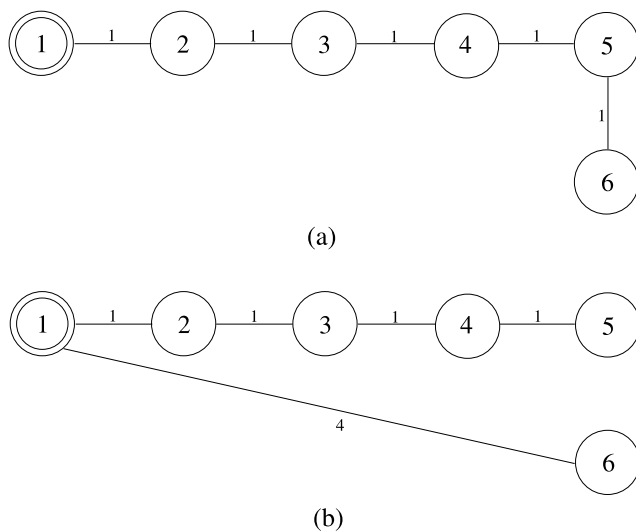
Fig. 4 An example that illustrates the difference between (**a**) the shape tree in our method and (**b**) the shortest-path tree in Cristinacce et al. (2008). Lower edge weight indicates larger pairwise similarity. In the shortest-path tree, node 6 is directly connected to the root 1 because the edge weight between them is less than the total edge weight along the path $6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1$

the shape tree in our method, this tree is constructed by combining the shortest path from each node to a selected root. As illustrated in Fig. 4, shape correspondence using the shortest path tree may still have a shape instance (e.g., node 6) directly connected to the root (e.g., node 1) even if there is large difference between them. The direct correspondence between such different shape instances may lead to large errors.

The remainder of this paper is organized as follows. In Sect. 2 we describe in detail the proposed method. In Sect. 3 we evaluate the performance of the proposed method against five available shape-correspondence methods, and in Sect. 4 we provide a brief conclusion.

2 The Proposed Method

In this section a new shape correspondence method that attempts to address the limitations of global and pairwise methods is developed. Throughout the remainder of this paper we will focus on continuous 2D shape contours, however the concepts presented in this paper can be extended to 3D shape surfaces.

In general, the proposed method first pre-organizes a population of shape instances into a shape tree $T = (V, E)$ where each node in V is a shape instance in the population, and each edge in E connects two nodes that represent two shape instances that are very similar. A root node that represents the starting template shape instance is selected. Starting from the selected root node, the proposed method recursively walks the shape tree where the current node is the

template shape instance and its child nodes are target shape instances. Along each edge in the downward path from the root to the leaf nodes, the landmarks of the target shape instance are corresponded to the fixed landmarks of the template instance using a pairwise method. When each node in the shape tree has been visited, the landmarks of each shape instance in the shape tree are considered to be well corresponded and the method terminates.

For example, given a population of eleven shape instances in the form of 2D contours that resemble the human liver, the proposed method first pre-organizes this population into the shape tree shown in Fig. 5, where v_1 is the root node, and $\{v_5, v_8, v_9, v_7, v_3, v_{10}, v_{11}\}$ are leaf nodes. Starting from the root node, the proposed method designates the shape instance at node v_1 as the template and its child nodes $\{v_2, v_3, v_4\}$ as the targets. We sample a set of landmarks on the template shape instance as the reference and then identify landmarks on each target, say v_2, so that they are well corresponded to the landmarks on the template using a pairwise method. When completed, the method recursively walks the shape tree where the shape instance at node v_2, together with its landmarks, becomes template and its child nodes $\{v_5, v_6, v_7\}$ become the targets. Likewise, we correspond each target, say v_5, to the template v_2 by identifying corresponded landmarks on v_5. Because v_5 is a leaf node the recursive process terminates. After the pairwise correspondence between each pair of neighboring nodes, we achieve shape correspondence for the entire population where the landmarks in v_1 are the common reference.

This brings us to several key issues which need to be addressed: (a) How to construct the shape tree, (b) how to determine the root node of the shape tree, (c) the algorithm used for pairwise shape correspondence, and (d) how to determine and handle error conditions during shape correspondence.

2.1 Constructing the Shape Tree

As mentioned above, we construct a shape tree such that neighboring nodes represent similar shape instances. In this paper, we use the minimum spanning tree (MST) algorithm for this purpose. Given a population of shape instances $\tilde{S} = \{\tilde{S}_i, \ i = 1, \ldots, n\}$ in the form of continuous 2D contours, we must estimate the similarity between each pair of shape instances. To achieve this, we first find a *rough* correspondence across the entire population. Specifically, for open shape contours, we can select one endpoint as the first landmark and equally sample each shape contour for m roughly corresponded landmarks $\tilde{U}_i = \{\tilde{\mathbf{u}}_{ik}, \ k = 1, \ldots, m\}$, where $\tilde{\mathbf{u}}_{ik} = (\tilde{x}_{ik}, \tilde{y}_{ik})$ is the kth landmark on $\tilde{S}_i$. For closed shape contours, we need to search for the first corresponded landmark across the population. Specifically, we use a simple exhaustive search detailed in Appendix. More efficient

Fig. 5 An example shape tree that pre-organizes a population of shape instances. Each node is a shape instance in the form of a 2D shape contour that resembles the human liver
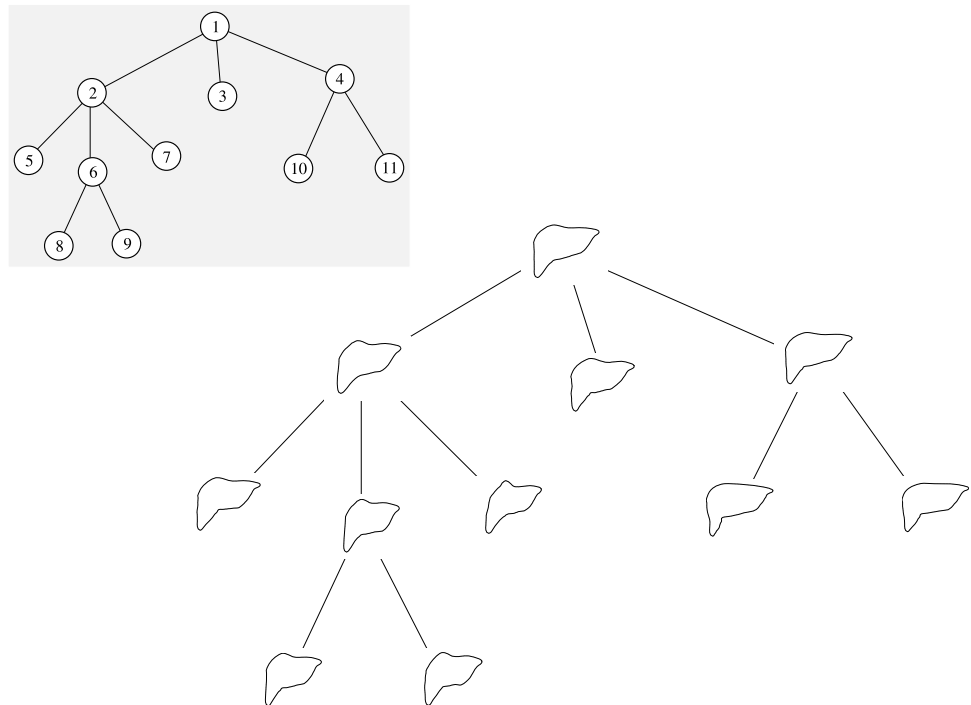
algorithms with sub-cubic runtime can also be used for this purpose (Schmidt et al. 2007).

We then estimate a set $U = \{U_i, \ i = 1, \ldots, n\}$ of normalized shape instances using the roughly corresponded landmarks identified above by removing scaling, rotation, and translation transformations using Procrustes analysis (Gower and Dijksterhuis 2004). A set of normalized continuous shape contours $S = \{S_i, \ i = 1, \ldots, n\}$ is found by interpolating the m landmarks in $U_i, i = 1, \ldots, n$ using the Catmull-Rom cubic spline (Catmull and Rom 1974). Using these normalized shape instances, we construct a fully connected, undirected graph with n nodes, where node v_i represents shape instance U_i, and the edge (v_i, v_j) that connects nodes v_i and $v_j, i \neq j$, is weighted by

$$w(v_i, v_j) = \Delta(S_i, S_j) \cdot \Omega(U_i, U_j) \tag{1}$$

for $i, j = 1, \ldots, n$. This edge weight reflects the shape difference between S_i and S_j. In this equation, $\Delta(S_i, S_j)$ measures the area coincident between two normalized continuous shape contours S_i and S_j, and $\Omega(U_i, U_j)$ measures the nonrigid shape difference between two shape instances using their roughly corresponded landmarks U_i and U_j.

In particular, the first term

$$\Delta(S_i, S_j) = 1 - \frac{|R(S_i) \cap R(S_j)|}{|R(S_i) \cup R(S_j)|} \tag{2}$$

measures the shape difference using the Jaccard coefficient, where $R(S)$ is the region enclosed by the shape contour S

and $|R|$ computes the area of the enclosed region. For open shape contours, the first and last landmark are tied together to enclose the region. This term takes a value in [0, 1] where zero indicates S_i is exactly the same as S_j.

However, the Jaccard coefficient cannot well describe local shape variations between two shape contours. To overcome this potential limitation the second term

$$\Omega(U_i, U_j) = \frac{\beta(U_i, U_j) + \beta(U_j, U_i)}{2}$$

is included that measures the amount of bending energy required to deform shape instance U_i to U_j (and shape instance U_j to U_i) using the 2D thin-plate spline model (Bookstein 1989). In Sect. 3, we conducted experiments to show that both terms in (1) are important for shape-tree construction and accurate shape correspondence.
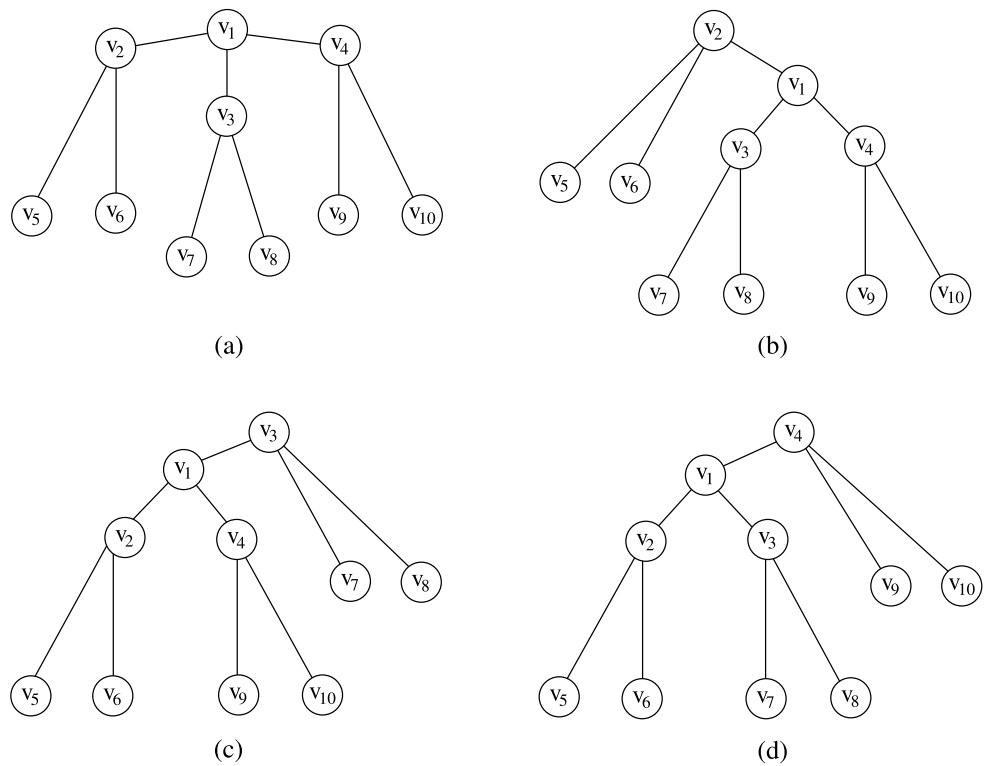
Given the roughly corresponded landmarks in a template U_i and target U_j, the bending energy can be calculated as

$$\beta(U_i, U_j) = \mathbf{x}^T \mathbf{L} \mathbf{x} + \mathbf{y}^T \mathbf{L} \mathbf{y}, \tag{3}$$

where $\mathbf{x} = (x_1, \ldots, x_m)^T$ and $\mathbf{y} = (y_1, \ldots, y_m)^T$ are vectors that contain the x and y coordinates of the landmarks in the target U_j. The bending matrix L is the $m \times m$ upper left sub-matrix of

$$\begin{bmatrix} \mathbf{K} & \mathbf{D} \\ \mathbf{D}^T & 0 \end{bmatrix}^{-1},$$

(a)

(b)

(c)

(d)

where $D = (\mathbf{1}, \hat{\mathbf{x}}, \hat{\mathbf{y}})$ is a $m \times 3$ matrix whose column vectors $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_m)^T$ and $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_m)^T$ are the x and y coordinates of the landmarks in the template U_i. K is an $m \times m$ matrix with element $K(a,b) = \|\mathbf{u}_{ia} - \mathbf{u}_{ib}\|^2 \log \|\mathbf{u}_{ia} - \mathbf{u}_{ib}\|$ for $a, b = 1, \ldots, m$ and $\mathbf{u}_{ia}$ is the coordinate of the ath landmark in U_i. Lastly, because $\beta(U_i, U_j)$ and $\beta(U_j, U_i)$ are typically not the same, the proposed method averages them to measure the nonrigid shape difference. If the averaged value is small then the two shape instances are very similar.

Given this fully connected graph, the minimum spanning tree $T = (V, E)$ is found using either Prim's or Kruskal's algorithm (Cormen et al. 1990), where T is a connected, acyclic, undirected graph defined by the set of vertices V with size $|V| = n$, and edges E with size $|E| = n - 1$, whose total edge weight is minimum.

2.2 Selecting the Root Node of the Shape Tree

The root node defines the common reference for the shape correspondence of the entire population. It is critical to select a shape instance that is a good representative of the entire population as the root node. Particularly, the proposed method propagates the landmarks on the root node to other nodes by a sequence of pairwise shape correspondences. The correspondence error accumulates during propagation and, to reduce this error, we select a root node such that pairs of shape instances that have a larger shape difference are located closer to the leaves in the shape tree. Conceptually, we

attempt to push outlier shape instances to the bottom of the tree so they become (or are located very near) leaf nodes in the shape tree.

Specifically, given the (unrooted) shape tree $T = (V, E)$ found in Sect. 2.1, we calculate the cost of selecting node v_i as the root using

$$C(v_i) = \sum_{k=1}^{|V|} \sum_{j=1}^{|\tilde{V}_k|} h(v_k) w(v_k, \tilde{v}_{kj}), \qquad (4)$$

where $v_i \in V$, $\tilde{V}_k = \{\tilde{v}_{kj}, \ j = 1, \ldots, |\tilde{V}_k|\}$ is the set of nodes with parent v_k, $h(v_k)$ is the height (Cormen et al. 1990) of the node in the tree that is equal to the number of edges on the longest simple downward path from node v_k to a leaf node, and $w(v_k, \tilde{v}_{kj})$ is the weight of the edge that connects nodes v_k and $\tilde{v}_{kj}$ in (1). Note that the parent nodes and node heights are determined by the selected root v_i. For each node in the shape tree this cost is calculated, and the node with the least cost is then selected as the root node.

For example, assume we have the shape trees illustrated in Fig. 6 where the length of the edge is proportional to its weight and we only consider nodes $\{v_1, v_2, v_3, v_4\}$ that have degree ≥ 2 as candidate root nodes. If v_1 is selected to be the root node of the tree as illustrated in Fig. 6(a), its cost would be

$$C(v_1) = h(v_1)(w(v_1, v_2) + w(v_1, v_3) + w(v_1, v_4))$$
$$+ h(v_2)(w(v_2, v_5) + w(v_2, v_6))$$

$$+ h(v_3)(w(v_3, v_7) + w(v_3, v_8))$$

$$+ h(v_4)(w(v_4, v_9) + w(v_4, v_{10})).$$

Using node height $h(v_i)$ as an additional weight, neighboring shape instances that are not similar will incur a heavy cost if located near the selected root node. Likewise, the cost of selecting v_2, v_3, and v_4 as the root node of the shape tree are calculated, and the trees are illustrated in Figs. 6(b), (c), and (d) respectively. The proposed cost favors root nodes that produce shape trees like those illustrated in Fig. 6(a) over those in Figs. 6(b–d).

2.3 Shape Correspondence Algorithm

With the shape tree and the selected root node, we can carry out the shape correspondence by propagating the landmarks from the root to all the other nodes by a sequence of pairwise correspondences. This algorithm is outlined in Algorithm 1. Given the shape tree T and the root node v as input, on *line 1* the shape instance P at node v is designated as the template. *Line 2* locates the child nodes of v, and the shape instance U at child node $\tilde{v}$ is designated as the target on *line 3*.

Algorithm 1 WALKSHAPETREE(T, v)

1:   $P \leftarrow getTemplateShapeInstance(v)$
2: **for all** $\tilde{v} \in children[v]$ **do**
3:     $U \leftarrow getTargetShapeInstance(\tilde{v})$
4:     $\min \phi(P, U)$
5:     WALKSHAPETREE(T, $\tilde{v}$)
6: **end for**

On *line 4*, shape correspondence between the template and the target is achieved by moving landmarks in the target to minimize the pairwise shape-correspondence error (Wang et al. 2004)

$$\phi(P, U) = \beta(P, U) + \lambda \, R(U),$$

where $\beta(P, U)$ measures the bending energy between the target U and template P using (3) and it is invariant to affine transformations, $R(U)$ is the shape representation error of the target shape instance, and $\lambda > 0$ is a balance factor that prevents all the landmarks on the target from moving too close to each other. Specifically, the shape representation error is defined by

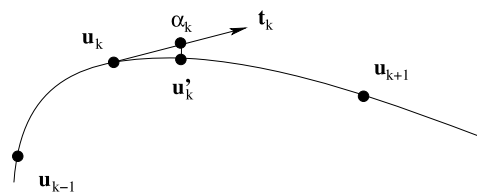$$R(U) = \sum_{k=1}^{m} (l_k \hat{l}_{k+1|m} - \hat{l}_k l_{k+1|m})^2.$$

Fig. 7 An illustration of landmark sliding where α_k is the sliding distance along the normalized tangent direction $\mathbf{t}_k$ for landmark $\mathbf{u}_k$, and $\mathbf{u}'_k$ is the location of the new landmark projected onto the shape contour

This representation error favors the landmarks on the target that have a similar spatial distribution to those on the template, i.e.,

$$\frac{l_k}{l_{k+1|m}} \approx \frac{\hat{l}_k}{\hat{l}_{k+1|m}},$$

where l_k is the curve length from landmark $\mathbf{u}_k$ to $\mathbf{u}_{(k+1|m)}$ in the target, and $\hat{l}_k$ is the curve length from landmark $\mathbf{p}_k$ to $\mathbf{p}_{(k+1|m)}$ in the template.[1]

The landmarks on the target are iteratively refined by calculating the optimal *sliding distances* as illustrated in Fig. 7, where α_k is the optimal sliding distance along the normalized tangent direction $\mathbf{t}_k$ for landmark $\mathbf{u}_k$. Because the location of $\mathbf{u}_k + \alpha_k \mathbf{t}_k$ may not be on the continuous shape contour, a projection step is used to map it back. Specifically, we project $\mathbf{u}_k + \alpha_k \mathbf{t}_k$ to a landmark $\mathbf{u}'_k$ on the shape contour so that $l(\mathbf{u}_k, \mathbf{u}'_k) = \alpha_k$, where $l(\mathbf{u}_k, \mathbf{u}'_k)$ is the curve length between $\mathbf{u}_k$ and $\mathbf{u}'_k$. At each iteration, a quadratic solver is used to find a set of optimal sliding distances $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)$ that minimize

$$(\mathbf{x} + T_x \boldsymbol{\alpha})^t \mathbf{L} (\mathbf{x} + T_x \boldsymbol{\alpha}) + (\mathbf{y} + T_y \boldsymbol{\alpha})^t \mathbf{L} (\mathbf{y} + T_y \boldsymbol{\alpha})$$

$$+ \lambda \sum_{a=1}^{m} \left\{ ((l_a + \alpha_a)\hat{l}_{a+1|m} - (l_{a+1|m} + \alpha_{a+1|m})\hat{l}_a)^2 \right\}, \quad (5)$$

where $T_x = \mathrm{diag}(t_{x_1}, \ldots, t_{x_m})$ and $T_y = \mathrm{diag}(t_{y_1}, \ldots, t_{y_m})$ are diagonal matrices that contain the x and y coordinates of the m normalized tangent vectors. Vectors $\mathbf{x}$ and $\mathbf{y}$ are the x and y coordinates of the m landmarks in U and $\mathbf{L}$ is the bending matrix defined using the landmarks in P. Additional constraints

$$l_k - \alpha_k + \alpha_{k+1} > 0, \quad k = 1, 2, \ldots, m$$

are imposed to ensure that each landmark does not move beyond its neighbors, which preserves the topology in the target. Shape correspondence between P and U is achieved when the landmarks on the target stop moving or a preset maximum number of iterations has been reached.

Since the above shape correspondence error contains a shape-representation term $R(U)$, it is not invariant under

---

[1] Modulo arithmetic $k + 1|m$ is used for closed shape contours.

Fig. 8 An illustration of our error handling strategy. (**a**) Remove edge $(v, \tilde{v})$ between two shape instances that fail to correspond. (**b**) Create a new edge $(\hat{v}_1, \tilde{v})$ to designate a new template at $\hat{v}_1$ for the target at $\tilde{v}$

(a)   (b)
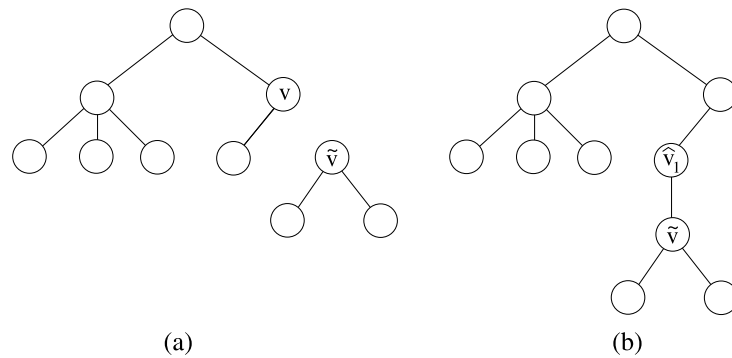
certain affine transformations, such as stretching and shrinking. Therefore, this refined pairwise correspondence algorithm may not handle well a pair of template and target with a large aspect-ratio change. However, this is not an issue to the proposed method because we only apply this refined pairwise correspondence algorithm between very similar shape pairs identified by the shape tree. The correspondence between a very different pair is obtained by combining a sequence of refined pairwise correspondences along the path between them.

Lastly, on *line 5*, the shape instance represented by node $\tilde{v}$ becomes the template and its child nodes become the targets. The proposed algorithm recursively walks the shape tree until a leaf node is reached, in a depth-first fashion. The algorithm terminates after all the nodes have been visited.

2.4 Error Handling

During shape correspondence it is possible that neighboring shape instances in the shape tree are still quite different from each other. For such pairs, the pairwise correspondence algorithm described in Sect. 2.3 may not produce an accurate pairwise correspondence. We can identify such pairs by evaluating the thin-plate bending energy between the landmarks on the template and resulting landmarks (after iterative sliding as described in Sect. 2.3) on the target. If this bending energy is significantly larger than those between other corresponded pairs, we consider the correspondence between the template and target as a failure. In this case, we take the following steps to avoid propagating this pairwise correspondence error to other shape instances.

Given the shape tree $T = (V, E)$, if the proposed method fails to correspond the target shape at node $\tilde{v}$ and the template shape at node v, we first determine the size of the subtree $\tilde{T} = (\tilde{V}, \tilde{E})$ rooted at node $\tilde{v}$, where $\tilde{T}$ is a subtree in T. If $|\tilde{V}| \ll |V|$, and the weight $w(v, \tilde{v})$ of edge $(v, \tilde{v})$ is significantly large, then it is reasonably safe to assume that the nodes in $\tilde{T}$ represent shape instances that may be outliers in the population. In such conditions, the subtree $\tilde{T}$ rooted at $\tilde{v}$ is pruned from the tree and discarded.

However, if the number of nodes in $|\tilde{V}|$ is very large we locate a new template in the shape tree for $\tilde{v}$. This is achieved by calculating the set of weights $w(v_k, \tilde{v})$ for $k = 1, \ldots, |V|$ such that $v_k \neq v$ and $v_k \notin \tilde{V}$, where $w(\tilde{v}, v_k)$ is the weight of edge $(\tilde{v}, v_k)$ in (1). The weights are then sorted in ascending order resulting a set of nodes $\{\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_z\}$ that represent candidate templates for $\tilde{v}$, where $\hat{v}_1$ represents the node whose edge weight $w(\hat{v}_1, \tilde{v})$ is minimal. As illustrated in Fig. 8, we remove the edge $(v, \tilde{v})$ and create a new edge $(\hat{v}_1, \tilde{v})$ in the shape tree.

The proposed method then attempts to correspond the target to the template at $\hat{v}_1$. If fails again, the next candidate at $\hat{v}_2$ will become the template. This process is repeated until the target is successfully corresponded, or when all candidates in $\{\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_z\}$ have failed. If all candidates have failed, the subtree $\tilde{T}$ rooted at node $\tilde{v}$ is pruned from the tree and discarded.

3 Experiments

In this section we conduct experiments to evaluate the accuracy and efficiency of the proposed shape correspondence method by comparing with four available global methods and one pairwise method. The four global methods are: Thodberg's implementation (Thodberg 2003) of the MDL method (T-MDL),[2] Ericsson and Karlsson's implementations (Karlsson and Ericsson 2006) of MDL (E-MDL), MDL with curvature distance minimization (E-MDL+ CUR), and a reparameterisation method by minimizing Euclidean distance (EUC). For these four global methods, the maximum number of iterations is set to 20 and we use their recommended settings. Additionally, for T-MDL the endpoints were not allowed to move for open shape contours. The pairwise shape correspondence method is the landmark sliding, insertion, and deletion method (SDI) developed by

---

[2]Note that both T-MDL and E-MDL are based on a simplified version of the description length cost function and are considered to be approximate versions of MDL (Davies et al. 2002).

Richardson and Wang (2005). In these experiments, we primarily use the shape correspondence benchmark developed in Munsell et al. (2008) for performance evaluation. We also conducted experiments to show the importance of root selection for the shape tree and error handling.

The reported performance values were found using a C++ implementation of the proposed method. Specifically, the quadratic programming solver used to minimize the shape correspondence error was implemented using the OOQP C++ library developed by Gertz and Wright (2001), where the balance factor λ and the maximum number of iterations used by the quadratic programming solver were set to 1 and 20 respectively. The enclosed regions required to compute the Jaccard coefficient between two shape contours was computed using the C++ OpenCV image processing library, and the minimum spanning tree was implemented using the C++ BOOST graph library and Kruskal's algorithm. Furthermore, the reported CPU times were acquired on Linux workstations running Intel Xeon 3.4 GHz processors with 4 GB of RAM.

3.1 Shape Correspondence Benchmark

Performance evaluation of shape correspondence is a very difficult problem because ground truth is typically not available and the optimal shape correspondence may not be unique (Munsell et al. 2008). Davies et al. propose three measures to describe the compactness, specificity, and generality of the constructed PDM and suggest the use of these three measures to evaluate the shape correspondence (Davies 2002; Styner et al. 2003). Without using ground truth, these three measures may not provide an objective shape-correspondence evaluation, as demonstrated by an example in Sect. 2.3 of Munsell et al. (2008). Shape correspondence and the resulting PDM can be used for different vision applications, such as shape classification and segmentation. We can use such applications to evaluate shape correspondence, however, the evaluation result using one application may not be generalized to others.

In our experiments, we primarily use the shape correspondence benchmark developed in Munsell et al. (2008) for performance evaluation. Like the compactness, specificity, and generality measures in Davies (2002), Styner et al. (2003), the benchmark evaluates the accuracy of the resulting PDM to estimate shape correspondence performance. The major difference is that this benchmark starts from a known ground truth PDM, which is randomly sampled to generate a population of shape instances. A shape correspondence method is then applied to correspond these shape instances and a derived PDM is constructed using the correspondence results. Finally shape-correspondence performance is evaluated by checking if the derived PDM accurately recovers the probabilistic shape-space defined by
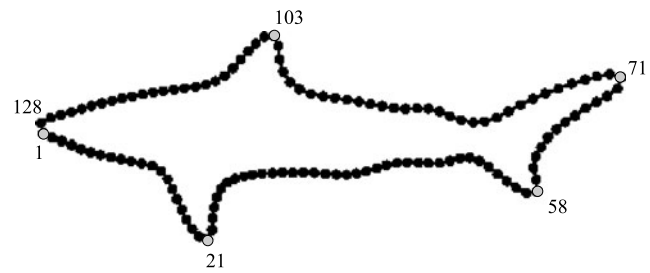

Fig. 9 Landmarks identified for constructing a ground-truth PDM that resembles the shark. Landmarks 1, 21, 58, 71, 103 were manually identified, and the remaining 123 landmarks were automatically placed at equal arc-length intervals between these manually identified landmarks
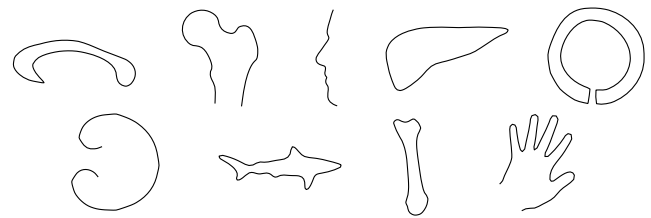

Fig. 10 *From left to right*, and *top to bottom*, the mean shapes of the ground-truth PDM that resemble the corpus callosum, human femur, human face silhouette, human liver, human heart valve, human kidney, shark, human metacarpal, and human hand

a ground-truth PDM. By using a ground-truth PDM, this benchmark is more objective than the compactness, specificity, and generality measures.

In this paper, we construct the ground truth PDM by manually identifying a set of corresponded landmarks over a small population of real shape instances. In particular, manual correspondence is established by first locating five to nine landmarks with distinguished geometric features, such as high curvature points or turning points. The remaining landmarks are then automatically placed at equal arc-length intervals between these manually identified landmarks. For example, along the shape contour of the shark shown in Fig. 9, manual correspondence identifies the snout (landmark 1) and the tips of the four fins (landmarks 21, 58, 71, 103). The remaining 123 landmarks are then sampled at equal arc-length intervals between these five landmarks: 19 are placed between landmarks 1 and 21, 36 between landmarks 21 and 58, 12 between landmarks 58 and 71, 31 between landmarks 71 and 103, and 25 between landmarks 103 and 128. Using this strategy, we construct nine ground-truth PDMs that resemble the human corpus callosum (corpus callosum for short), human femur, human face silhouette (face silhouette for short), human liver, human heart valve (heart valve for short), human kidney, shark, human metacarpal, and human hand structures, as shown in Fig. 10, for our experiments.

For each ground-truth PDM, we randomly generate a population of 1,200 synthetic shape instances for testing

shape correspondence. In our experiments, each shape correspondence method is asked to identify 64 corresponded landmarks across the entire population. To check whether the derived PDM recovers the probabilistic shape space of the ground truth PDM, we randomly generate 2,000 synthetic shape instance from each PDM and then calculate the bipartite measure Δ_b and the Wald-Wolfowitz measure Δ_w as suggested in Munsell et al. (2008). For both measures, smaller values indicate better performance. These two measures usually lead to similar results when comparing different shape correspondence methods. However, the Δ_b measure is based on the Jaccard coefficients between shape instances generated from the derived PDM and the ground-truth PDM. It reflects the detailed correspondence errors between shape instances. The Δ_w measure only counts the number of certain edges in a tree where nodes represent shape instances generated from the derived PDM and the ground-truth PDM. It does not reflect the detailed correspondence error between shape instances. Therefore, the Δ_b measure may be more accurate, but is more sensitive to noise (e.g., outliers) than the Δ_w measure. In evaluating two shape correspondence methods, if both measures indicate that one method is better than the other, we then defer to the Δ_b measure since it is more accurate. If two measures are inconsistent, we defer to the Δ_w measure since it is more robust. By using both measures, we can achieve more thorough and comprehensive performance evaluations.

3.2 Shape Correspondence Performance

As mentioned above, in the shape correspondence benchmark the Δ_b and Δ_w measures are obtained by random simulation: We compare a set of synthetic shape instances randomly generated from the ground-truth and the derived PDMs. To increase the statistical confidence of these measures, we perform 50 rounds of random simulation for each derived PDM to obtain a median value and a range for both Δ_b and Δ_w measures. The box plots in Fig. 11 show the median, first quartile, third quartile, inter-quartile range, lower extreme ($Q1 - 1.5 * IQR$), and upper extreme ($Q3 + 1.5 * IQR$) statistics of the Δ_b measure for the test shape correspondence methods over 50 rounds of random simulation. Note that 'TRUTH' indicates the Δ_b measure between the ground-truth PDM and itself, which can be treated as a lower bound for this measure. 'MST' indicates the proposed method. Out of the six test methods (MST, E-MDL, E-MDL+CUR, EU, T-MDL, SDI), we only show the four with better performance, i.e., with smaller Δ_b values. In each box plot, the selected methods are sorted increasingly in terms of the median Δ_b value from left to right. Likewise, the box plots illustrated in Fig. 12 show performance in terms of the Δ_w measure.

From the box plots in Fig. 11, we can see that the proposed method achieves the best Δ_b performance for the ground-truth PDMs that resemble the human femur, human hand, face silhouette, human metacarpal, and heart valve. The proposed method achieves a comparable Δ_b performance to that of E-MDL for the ground-truth PDM that resembles the shark. The proposed method achieves a comparable Δ_b performance to that of both E-MDL and E-MDL+CUR for the ground-truth PDM that resembles the corpus callosum and human liver. However, the inter-quartile range value for E-MDL+CUR is greater than the proposed method and MDL for the ground-truth PDM that resembles human liver. E-MDL demonstrates the best Δ_b performance for the ground-truth PDM that resembles the human kidney. From the box plots in Fig. 12 we can see that the proposed method achieves the best Δ_w performance for the ground-truth PDMs that resemble the human femur, shark, human hand, corpus callosum, human liver, human metacarpal, and heart valve. The proposed method achieves a comparable Δ_w performance to that of E-MDL for the ground-truth PDM that resembles the face silhouette, and E-MDL demonstrates the best Δ_w performance for the ground-truth PDM that resembles the human kidney. The performance of the EUC, T-MDL and SDI methods consistently rank in the bottom half of the six test methods. These results show that the performance of the proposed method is comparable to, or even better than, the state-of-the-art global shape-correspondence methods, such as E-MDL.

The CPU time (in seconds) taken by the six test shape correspondence methods to correspond 1,200 shape contours, randomly generated from each of the nine ground-truth PDMs, is provided in Table 1. For the proposed method, the reported CPU time includes both the initial correspondence, shape-tree construction, and the correspondence algorithm outlined in Sect. 2.3. We can see that the proposed method is at least 95 times faster than E-MDL for the ground-truth PDMs that resemble the human femur, shark, corpus callosum, human liver, face silhouette, and heart valve, and at least 90 times faster than E-MDL for the ground-truth PDMs that resemble the human hand and human kidney. The proposed method is at least 240 times faster than E-MDL+CUR for the ground-truth PDMs that resemble the human femur, shark, corpus callosum, human liver, human kidney, face silhouette, and heart valve, and at least 230 times faster than E-MDL+CUR for the ground-truth PDMs that resemble the human hand and human metacarpal.

To put the CPU times in perspective, for the human hand, E-MDL took approximately five days and E-MDL+CUR took approximately 13 days to identify 64 corresponded landmarks across a population of 1,200 shape instances whereas the proposed method only took approximately one hour and eighteen minutes. This is a significant performance
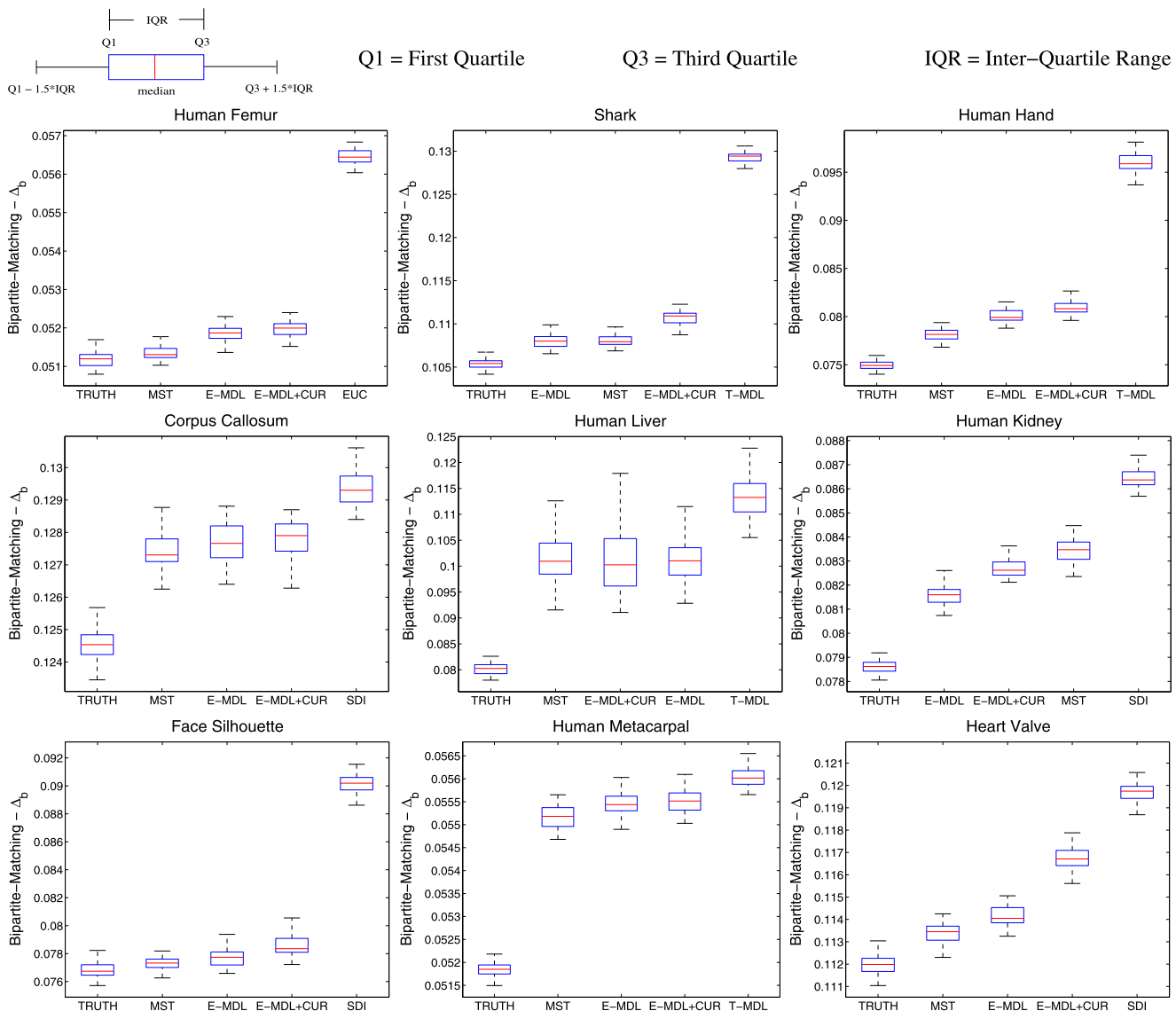
Fig. 11 Shape correspondence evaluation in terms of the Δ_b measure

tradeoff for a very small amount of gain in Δ_b and Δ_w performance that may be achieved by either E-MDL or E-MDL+CUR for the ground-truth PDMs that resemble the shark, human liver, face silhouette, and human kidney. Even though SDI (a pairwise method) is approximately 2 times faster than the proposed method for all the nine ground-truth PDMs, its Δ_b and Δ_w performance is consistently worse than the proposed method because it does not consider the global information underlying the entire population.

A detailed analysis of the algorithm complexity can be difficult because it depends on the selection of the basic operations. From a theoretic perspective, the shape tree construction involves the initial correspondence and shape similarity estimation between every pair of shape instances, which takes $O(n^2)$ time, and the refined correspondence only involves the pairwise correspondence between neigh-

boring shape instances in the shape tree, which takes $O(n)$ time. However, the initial correspondence for shape tree construction consists of very simple operations without moving landmarks (see Appendix) and the refined correspondence runs many iterations of the quadratic solver (see Sect. 2.3), which itself is a high-complexity algorithm. In our experiments, the refined correspondence on shape tree actually takes most of the CPU time reported in the first column of Table 1.

3.3 Effect of Shape Similarity Selection

In constructing the shape tree, we measure the rough shape similarity using (1), which consists of two terms: The Jaccard-coefficient-based $\Delta(S_i, S_j)$ and the bending-energy-based $\Omega(U_i, U_j)$. We conducted experiments to
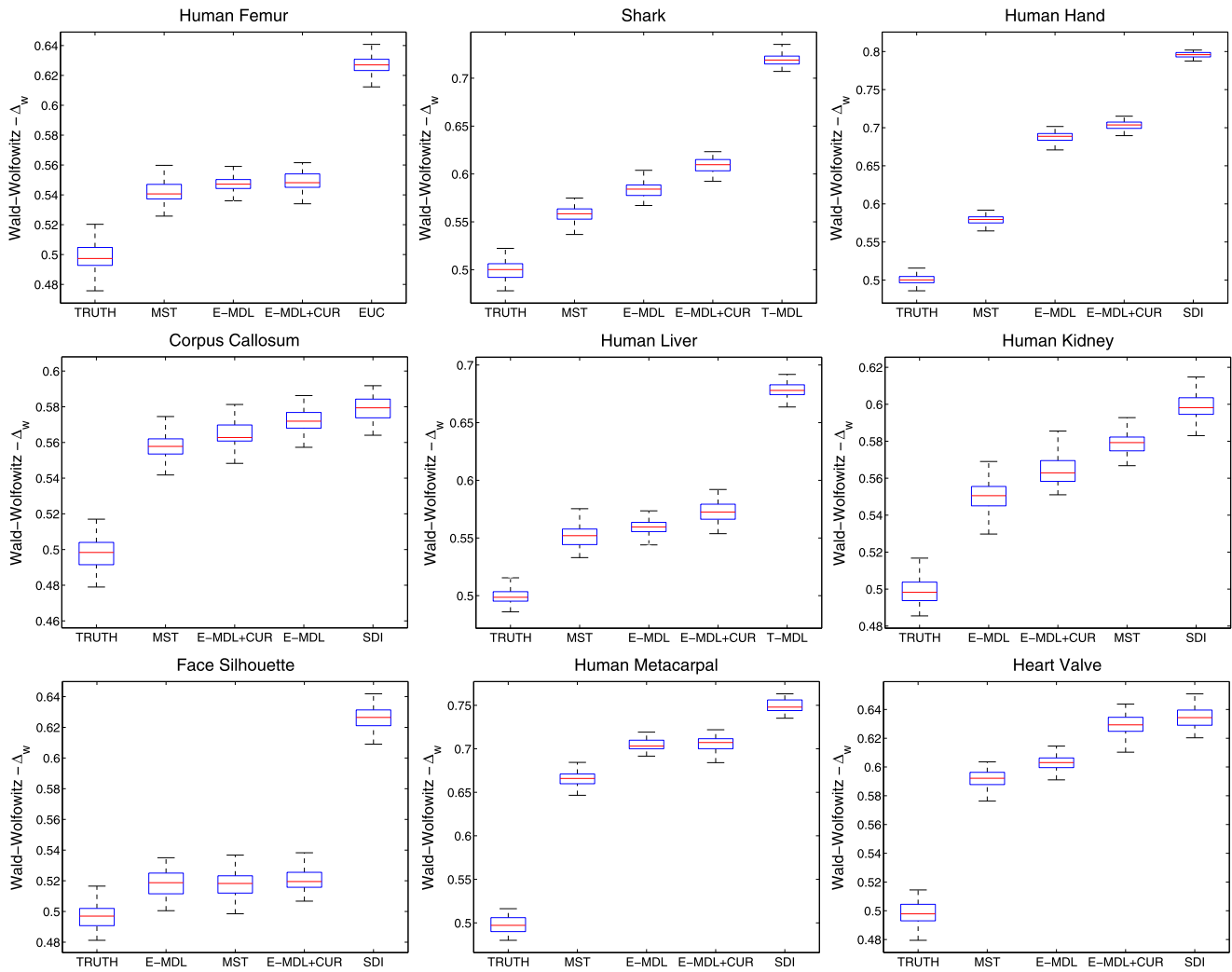
Fig. 12 Shape correspondence evaluation in terms of the Δ_w measure

Table 1 CPU time (in seconds) taken by the six test shape correspondence methods

	MST	E-MDL	E-MDL+CUR	EUC	T-MDL	SDI
Human femur	4704	457118	1160655	82250	160604	1384
Shark	4803	458735	1156086	93347	164785	1341
Human hand	4858	447094	1136918	82468	155979	1377
Corpus callosum	4773	456835	1195473	80036	226988	1336
Human liver	4745	459113	1167596	81626	136824	1362
Human kidney	4686	441907	1167605	84861	167905	1350
Face silhouette	4743	456099	1167632	92663	158910	1367
Human metacarpal	4788	433256	1143910	84968	140179	1342
Heart valve	4765	487861	1154080	79208	140991	1344

show that both terms are important for shape-tree construction and shape correspondence. The box plots in Figs. 13 and 14 show the Δ_b and Δ_w performance of the proposed method when using different shape similarity measures for the shape tree construction. In particular, 'MST-JAC' indicates the use of only $\Delta(S_i, S_j)$ for measuring shape similarity, 'MST-BE' indicates the use of only $\Omega(U_i, U_j)$ for measuring shape similarity, and 'MST' indicates the use of both terms as defined in (1) for measuring shape similarity. From Figs. 13 and 14, we can clearly see that the perfor-
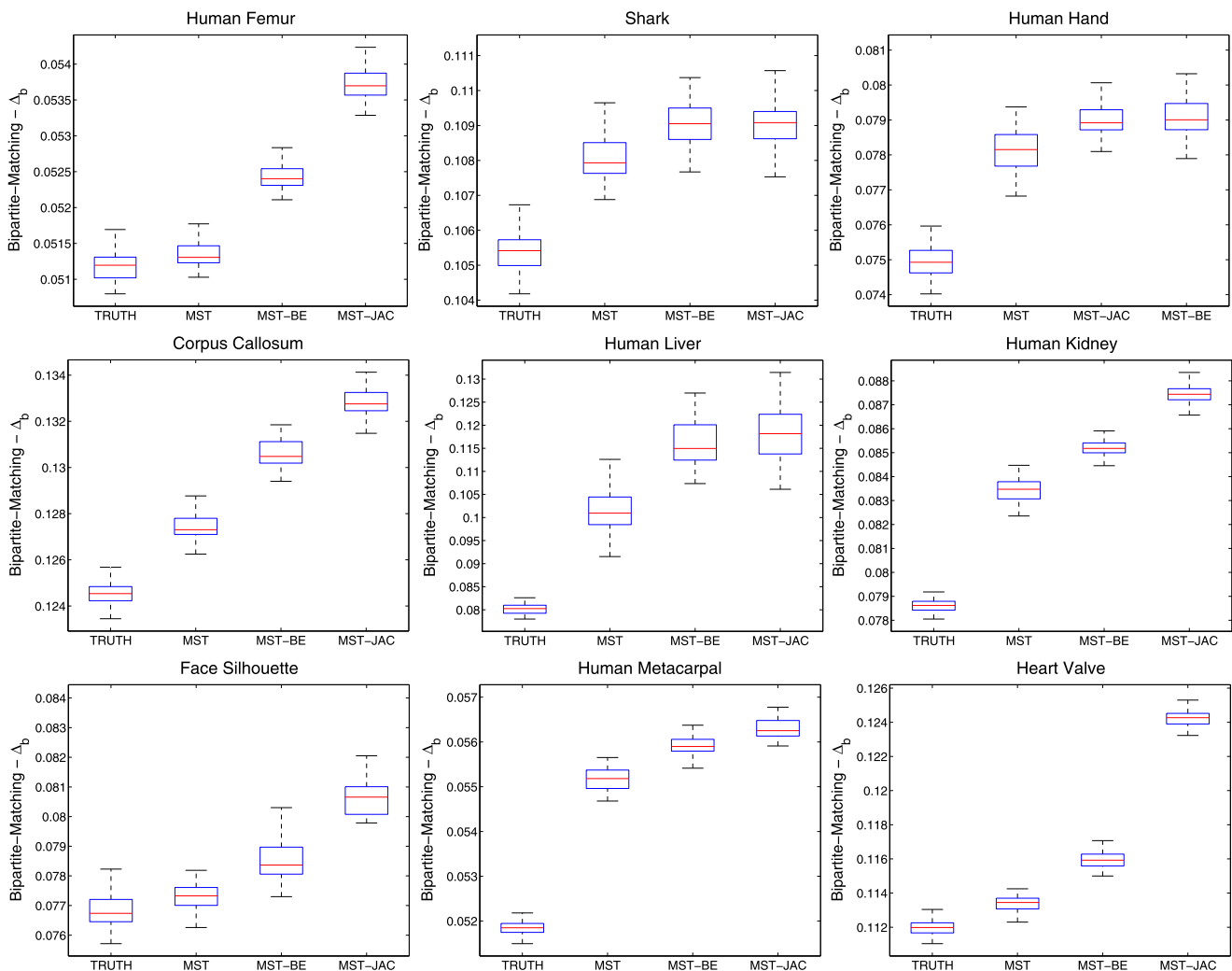
Fig. 13 Shape correspondence performance in terms of the Δ_b measure when using different shape similarity measures for constructing the shape tree

mance of 'MST' is consistently better than 'MST-BE' and 'MST-JAC'. Therefore, the combination of these two terms in (1) can produce a better shape tree and leads to more accurate shape correspondence.

3.4 Performance Comparison to a Semiautomatic Method

We also conducted experiments to compare the performance of the proposed method against a semiautomatic method. This semiautomatic method starts by manually identifying a small set of corresponded landmarks with distinguished geometric features, followed by automatically identifying more landmarks at equal arc-length intervals between these manually identified landmarks. In the above benchmark, all the shape instances for testing shape correspondence are generated from a ground-truth PDM, which contains a set of manually identified landmarks as shown by landmarks 1, 21, 58, 71, and 103 in Fig. 9. Therefore, on each generated shape in-

stance, the true locations of these manually identified landmarks are actually known. Certainly this information is not passed to the above test shape correspondence methods because, in real applications, such information is unavailable. For each generated shape instance, we use the true locations to simulate the manually labeled landmarks to evaluate semiautomatic shape correspondence performance. Clearly, this simulation provides the best possible manual labeling because, in practice, manual labeling typically contains errors. Therefore, the reported performance of the semiautomatic method in this experiment is ideal, and may not be achieved in practice. Figures 15 and 16 show the Δ_b and Δ_w performance of this semiautomatic method (indicated by 'MANUAL') and the proposed method ('MST'). We can see that, for the ground-truth PDMs that resemble the human femur, human hand, face silhouette, human metacarpal, and heart valve, the proposed method achieves the same, or very similar, Δ_b and Δ_w performance to the semiautomatic
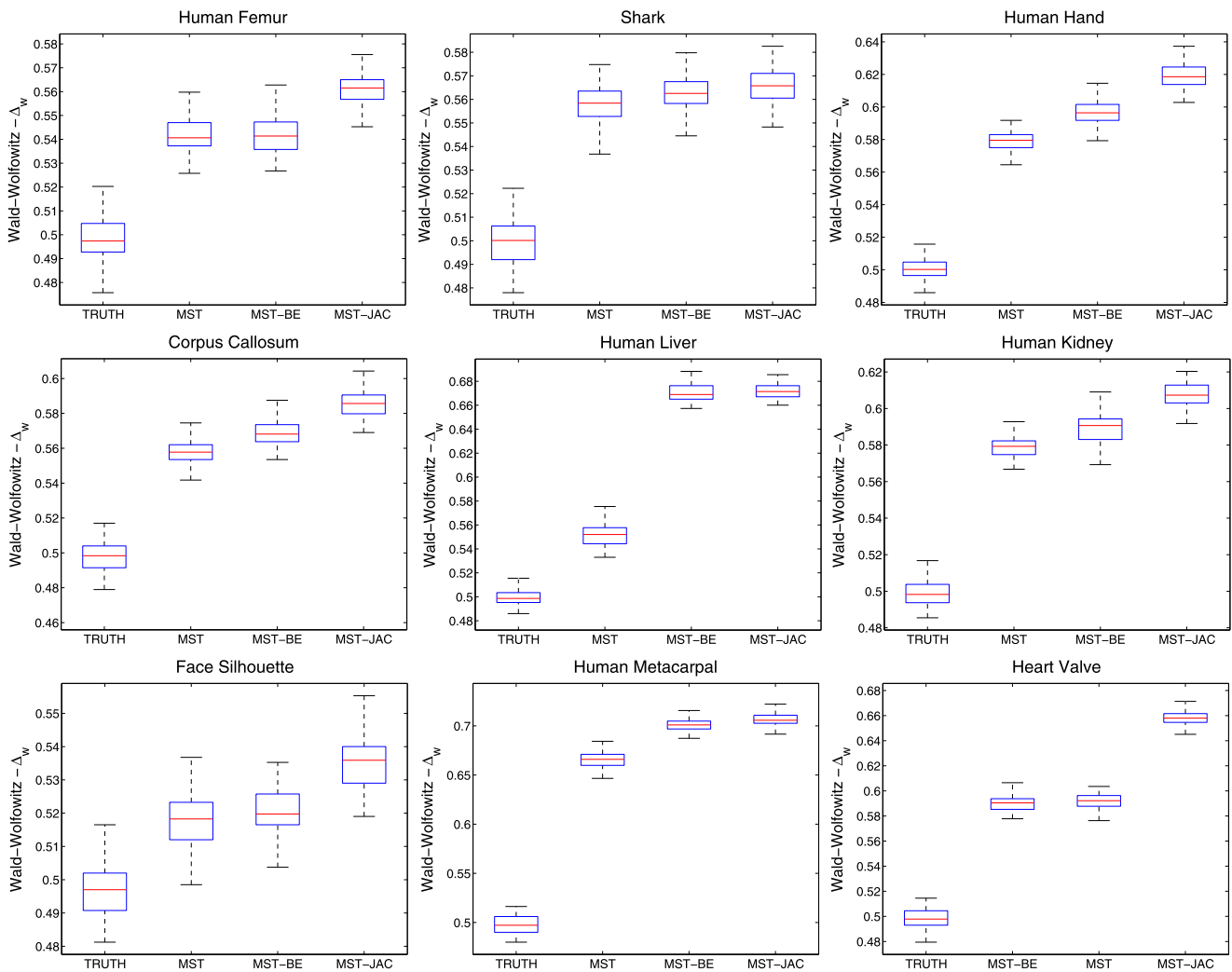
Fig. 14 Shape correspondence performance in terms of the Δ_w measure when using different shape similarity measures for constructing the shape tree

method. For the ground-truth PDMs that resemble the corpus callosum, shark, and human liver, the proposed method has comparable Δ_b and Δ_w performance to the semiautomatic method. For the ground-truth PDM that resembles the kidney, the semiautomatic method shows a better Δ_b and Δ_w performance than the proposed method.

3.5 Effect of Root-Node Selection

In this section we conduct experiments to show how the performance may be affected by choosing different root nodes for the shape tree. Specifically, by taking each shape instance (out of $n = 1,200$) as the candidate root, we calculate its root-selection cost $C(v_i)$ using (4), where v_i is the node the represents the considered shape instance. We then sort all the n shape instances in terms of their root-selection cost in increasing order. In the sorted list, we construct three categories of shape instances—Categories 1, 2 and 3—by taking the first five, median five, and last five shape instances, respectively. In total, we have 15 shape instances. We take each shape instance as the root to construct a shape tree for shape correspondence. The above-mentioned benchmark is then used for performance evaluation. Based on the median and IQR performance (the Δ_b or Δ_w measures) resulting from the five shape instances in each category, we can compare the performance by selecting a root from different categories, as shown in Tables 2 and 3. We can see that, selecting the root from Category 1 leads to consistently better performance than selecting the root from Categories 2 and 3.

3.6 Performance on Real Data and Effect of Error Handling

In this section, we collected a set of shape instances of a real object for performance evaluation. As mentioned before, it is difficult to objectively evaluate the shape-correspondence
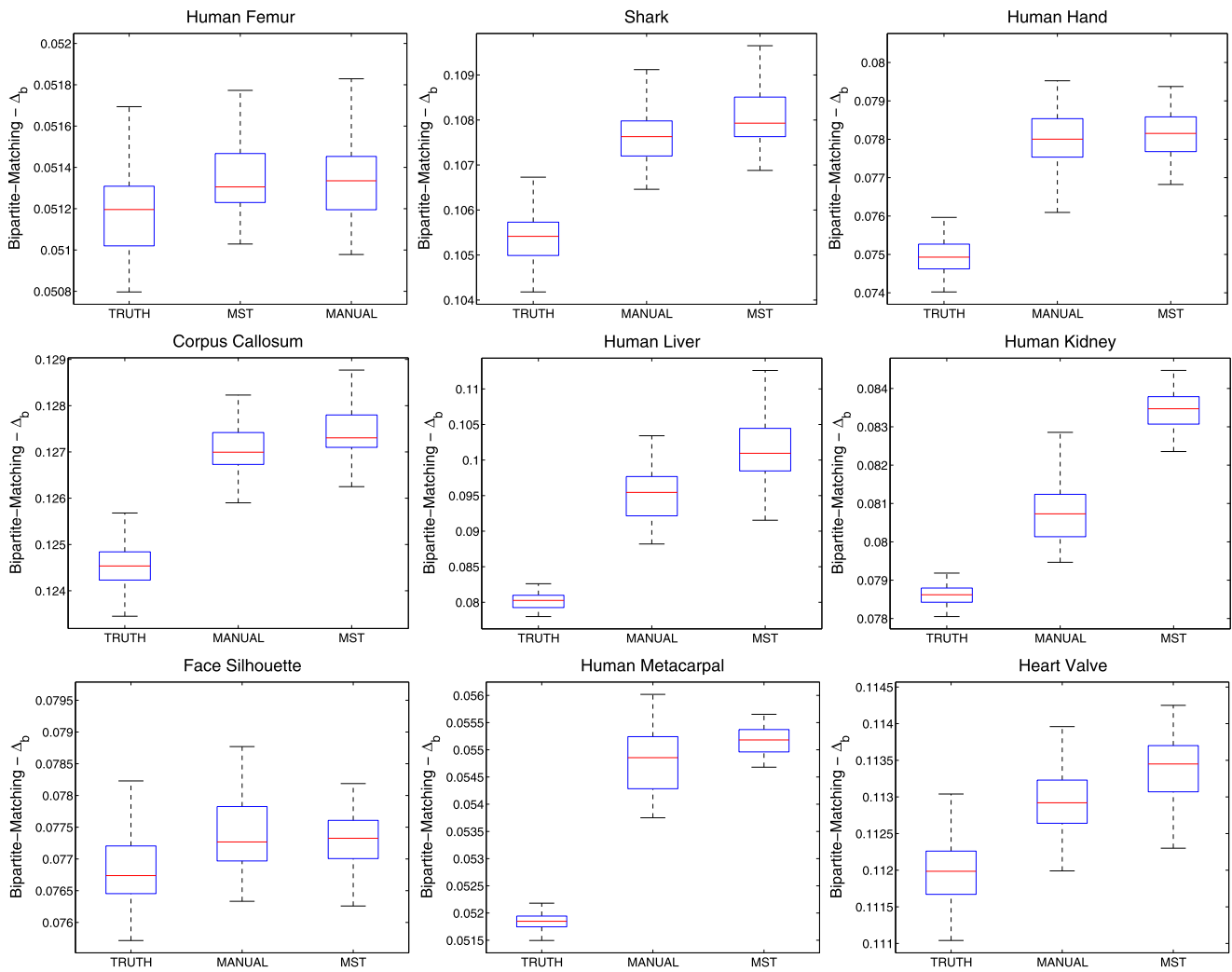
Fig. 15 Shape correspondence performance in terms of the Δ_b measure by using the semiautomatic method ('MANUAL') and the proposed method ('MST')

Table 2 Shape correspondence performance in terms of the Δ_b measure by selecting the root from different categories for the shape tree

	Category-1		Category-2		Category-3	
	Median	IQR	Median	IQR	Median	IQR
Human femur	0.05241	0.00028	0.05254	0.00028	0.05281	0.00029
Shark	0.10825	0.00095	0.10870	0.00108	0.10945	0.00140
Human hand	0.07827	0.00089	0.07854	0.00096	0.07911	0.00102
Corpus callosum	0.12868	0.00082	0.12949	0.00099	0.13094	0.00111
Human liver	0.11582	0.00608	0.11661	0.00640	0.11772	0.00632
Human kidney	0.08449	0.00049	0.08476	0.00048	0.08521	0.00061
Face silhouette	0.07853	0.00076	0.07885	0.00100	0.07920	0.00106
Human metacarpal	0.05626	0.00036	0.05643	0.00031	0.05715	0.00040
Heart valve	0.11478	0.00069	0.11541	0.00088	0.11892	0.00369

performance on these real shape instances because ground truth is not available. Therefore, the Δ_b and Δ_w measures used in the above benchmark are not applicable to real data.

Instead, we evaluate the shape-correspondence performance by checking the compactness, specificity, and generality of the derived PDM (Davies 2002; Styner et al. 2003).
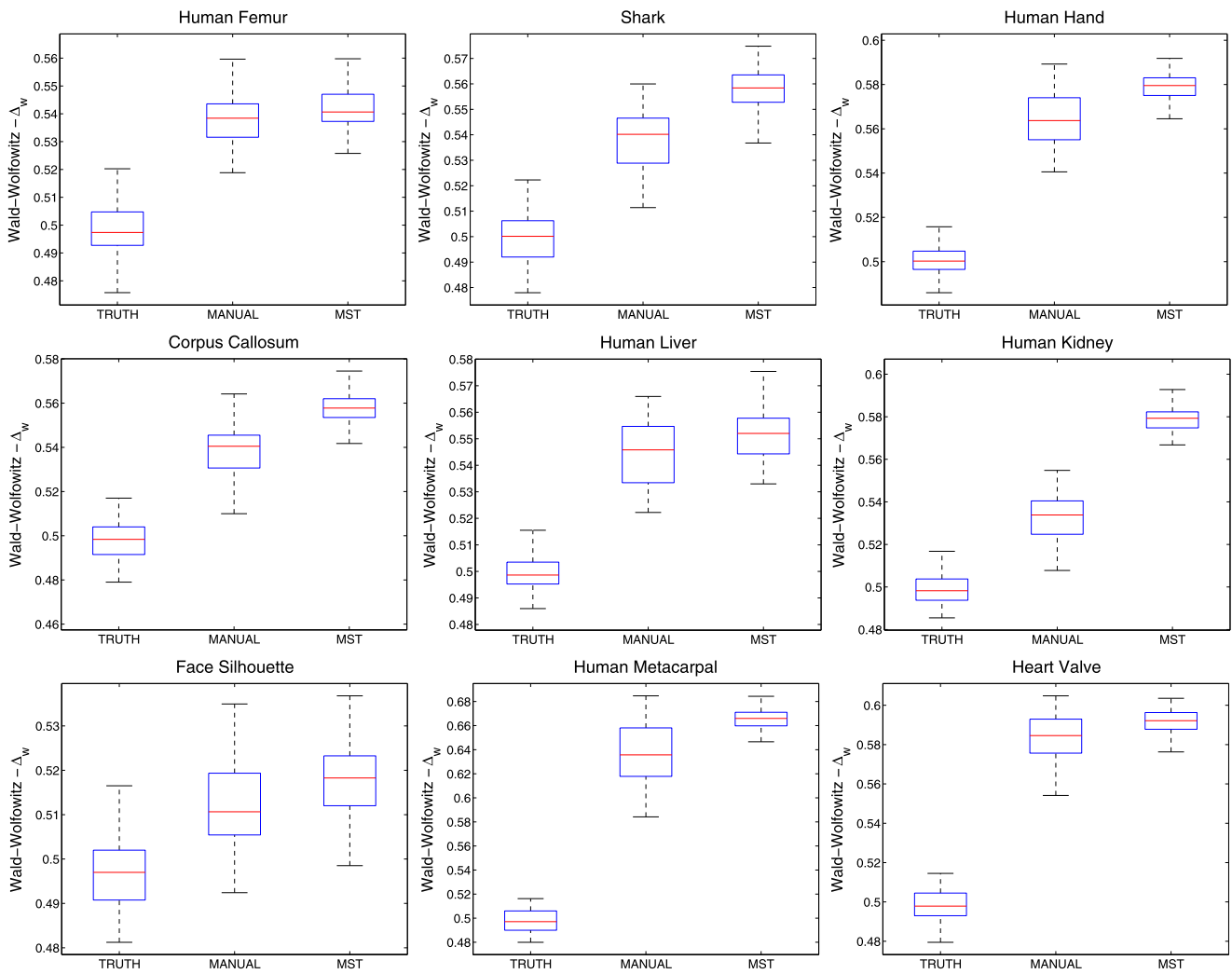
Fig. 16 Shape correspondence performance in terms of the Δ_w measure by using the semiautomatic method ('MANUAL') and the proposed method ('MST')

Table 3 Shape correspondence performance in terms of the Δ_w measure by selecting root from different categories for the shape tree

	Category-1		Category-2		Category-3	
	Median	IQR	Median	IQR	Median	IQR
Human femur	0.54227	0.01151	0.54427	0.01126	0.54877	0.01051
Shark	0.56066	0.01101	0.56428	0.01101	0.57279	0.01351
Human hand	0.58229	0.01001	0.59167	0.01176	0.59667	0.01276
Corpus callosum	0.55865	0.01151	0.56503	0.01176	0.57466	0.01201
Human liver	0.66583	0.01101	0.66983	0.01126	0.67546	0.01051
Human kidney	0.57979	0.01176	0.58242	0.01001	0.58667	0.01101
Face silhouette	0.51976	0.01026	0.52326	0.01076	0.52751	0.01126
Human metacarpal	0.67871	0.01326	0.69210	0.01276	0.70435	0.01026
Heart valve	0.59155	0.01126	0.59355	0.01051	0.60730	0.01751

Particularly, we use these real data to evaluate the effect of the error handling strategy developed in Sect. 2.4. In all the above experiments, synthetic shape instances generated

from a ground truth PDM are used for evaluation and they usually contain very few outliers that are very different from other shape instances, including other outliers, in the pop-

ulation. Therefore, although we enable the error handling in all the above experiments, it may not introduce much improvement to the shape correspondence performance. In this section, we conduct experiments to show that the error handling can improve the performance on real data. In our experiments, a correspondence between a template and target is treated as a failure if their bending energy is three times larger than the average bending energy between the corresponded neighboring pairs along the path from the template to the root in the shape tree.

In this experiment, we collected 20 images of the right hand from four different people, totalling 80 images. In 72 of them (18 from each person) individual fingers are arbitrarily moved, scrunched, or bent to cover the possible shape variation of the hand. A subset of these hand images are illustrated in Fig. 17(a). Additionally, we purposely inserted 8 images (two from each person) to test the proposed error handling strategy. In general, these 8 images may have two fingers very close to each as illustrated in Fig. 17(b).

Each of the 80 images are then converted to a binary image by simple intensity thresholding (the background is green) and from each binary image, we construct a shape contour. We use these 80 real shape contours for evaluating the performance of the proposed shape-correspondence methods and the other comparison methods. The results are shown in Fig. 18 where, for all the three measures, a smaller value indicates better performance. As in Davies (2002), in Fig. 19 we also visually demonstrate the mean shape of the derived PDMs and the shape instances after varying the principal mode by $\pm3\sigma$, where σ is the standard deviation along the principal direction of the PDM. We can see that the proposed method with error handling ('MST') achieves the best performance in terms of all the three measures.

4 Conclusion

In conclusion, this paper introduced a new method that pre-organizes a population of shape instance for landmark-based shape correspondence. Using a minimum spanning tree algorithm, the population of shape instances are pre-organized into a shape tree, where each node represents a shape instance and each edge connects two very similar shape instances. This pre-organization step allows us to incorporate global information underlying the population into shape correspondence. Shape instances are then corresponded between neighboring shape instances in the shape tree in a pairwise fashion, efficiently and accurately. The performance of the proposed shape correspondence method was evaluated against four global methods and one pairwise method. Quantitative evaluation results showed
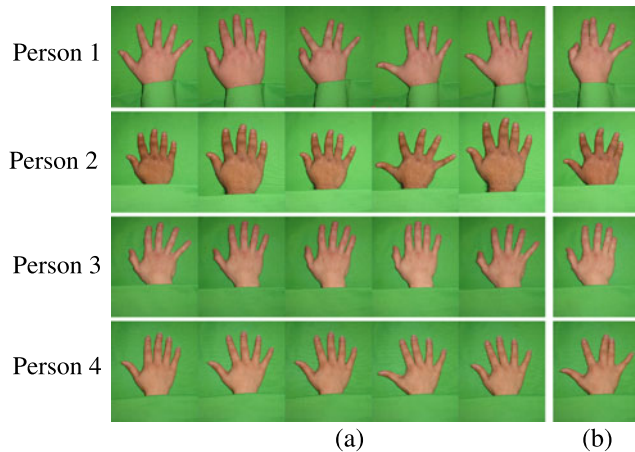

Fig. 17 Sample real images collected for performance evaluation
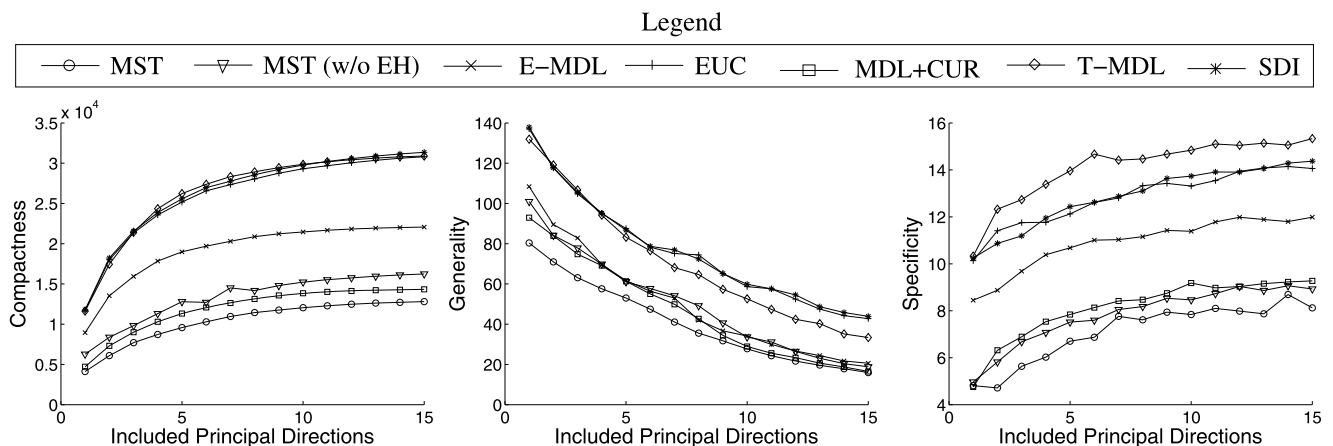

Fig. 18 The compactness, generality, and specificity for the E-MDL, T-MDL, MDL+CUR, SDI, EUC, the proposed method (MST), and the proposed method with error handling disabled (MST w/o EH) on real data
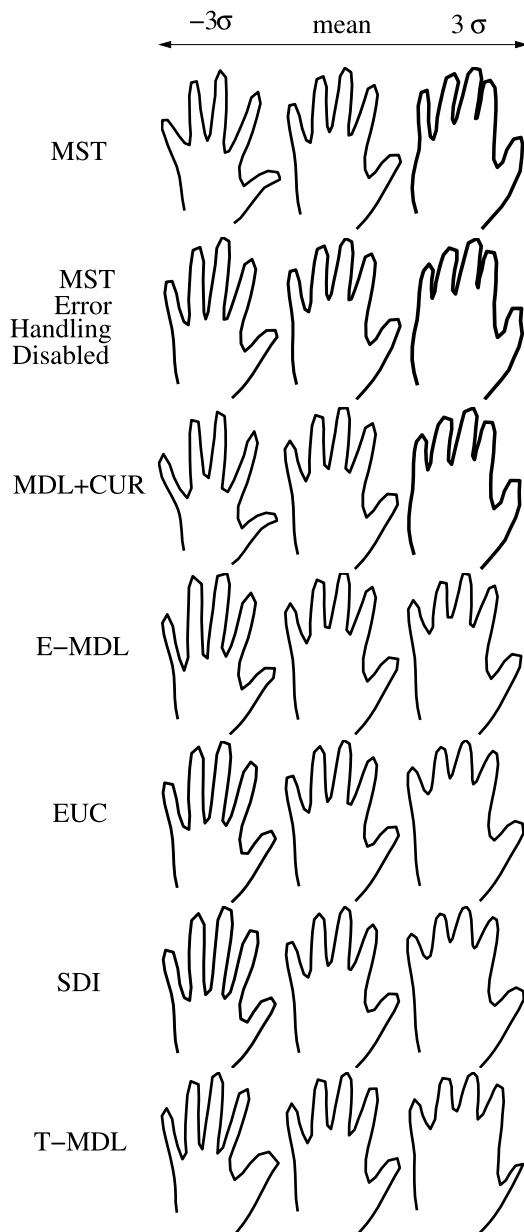
Fig. 19 A visual demonstration of the PDMs derived using different shape correspondence methods

the proposed method achieves a performance that is better than, or comparable to, the state-of-the-art global methods while using much less CPU time: At least 90 times faster than the global method that demonstrates the best performance.

Appendix: Initial Correspondence for Closed Contours

Given a population of shape instances $\tilde{S} = \{\tilde{S}_i, \ i = 1, \ldots, n\}$ in the form of closed, continuous shape contours, we construct an initial correspondence by uniformly sampling m initial landmarks on each shape instance and align these initial landmarks across different shape contours by circularly shifting the indices of these landmarks. More specifically, let $\tilde{U}_i = \{\tilde{\mathbf{u}}_{ik}, \ k = 1, \ldots, m\}$ be the sequence of m initial landmarks sampled on $\tilde{S}_i$. We can build the initial correspondence between $\tilde{U}_i$ and $\tilde{U}_j$ by circularly shifting the indices of the landmarks in $\tilde{U}_j$ and then calculate the shape similarity between them using (1). After exhaustively trying all m circular shifts of the landmarks in $\tilde{U}_j$, we select the shift that results in the best shape similarity to $\tilde{U}_i$ to build the initial correspondence. By building such an initial correspondence between each shape instance and $\tilde{U}_i$, we obtain an initial correspondence across the entire population. Note that this initial correspondence is very fast compared to the refined correspondence discussed in Sect. 2.3 because no landmark will move in the initial correspondence.

References

Amit, Y., & Geman, D. (1997). Shape quantization and recognition with randomized trees. *Neural Computation*, *9*, 1545–1588.

Basri, R., Costa, L., Geiger, D., & Jacobs, D. (1995). Determining the similarity of deformable shapes. *Vision Research*, *38*, 135–143.

Belongie, S., Malik, J., & Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(24), 509–522.

Bookstein, F. (1989). Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *11*(6), 567–585.

Bookstein, F. (1991). *Morphometric tools for landmark data*. Cambridge: Cambridge University Press.

Bookstein, F. (1997). Landmark methods for forms without landmarks: morphometrics of group differences in outline shape. *Medical Image Analysis*, *1*(3), 225–243.

Bruckstein, A., Katzir, N., Lindenbaum, M., & Porat, M. (1992). Similarity-invariant signatures for partially occluded planar shapes. *International Journal of Computer Vision*, *7*(3), 271–285.

Catmull, E., & Rom, R. (1974). A class of local interpolating splines. *Computer Aided Geometric Design*, 317–326.

Chen, L., Feris, R., & Turk, M. (2008). Efficient partial shape matching using smith-waterman algorithm. In *IEEE computer vision and pattern recognition workshops* (pp. 1–6).

Cootes, T., Taylor, C., Cooper, D., & Graham, J. (1995). Active shape models—their training and application. *Computer Vision and Image Understanding*, *61*(1), 38–59.

Cormen, T., Leiserson, C., & Rivest, R. (1990). *Introduction to algorithms*. Cambridge: MIT Press.

Cristinacce, D., Butcher, N., & Cootes, T. (2008). Facial motion analysis using clustered shortest path tree registration. In *International workshop on machine learning for vision-based motion analysis, European conference on computer vision (ECCV)*.

Dalal, P., Shi, F., Shen, D., & Wang, S. (2010). Multiple cortical surface correspondence using pairwise shape similarity. In *International conference on medical image computing and computer assisted intervention* (Vol. 1, pp. 349–358).

Davies, R. (2002). *Learning shape: optimal models for analysing natural variability*. Dissertation, University of Manchester.

Davies, R., Twining, C., Cootes, T., Waterton, J., & Taylor, C. (2002). A minimum description length approach to statistical shape modeling. *IEEE Transactions on Medical Imaging*, 21(5), 525–537.

Davies, R., Twining, C., & Taylor, C. (2008). *Statistical models of shape: optimization and evaluation*. London: Springer.

DiIanni, M., Diekrnann, R., Laling, R., Schulze, J., & Tschoeke, S. (1996). Simulated annealing and genetic algorithms for shape detection. *Control and Cybernetics*, 25(1), 159–175.

Dryden, I., & Mardia, K. (1998). *The statistical analysis of shape*. London: Wiley.

Duta, N., & Sonka, M. (1998). Segmentation and interpretation of MR brain images: an improved active shape model. *IEEE Transactions on Medical Imaging*, 17(6), 1049–1062.

Frangi, A., Rueckert, D., Schnabel, J., & Niessen, W. (2001). Automatic 3D ASM construction via atlas-based landmarking and volumetric elastic registration. In *Information processing in medical imaging conference* (pp. 77–91).

Gertz, M. E., & Wright, S. J. (2001). Object-oriented software for quadratic programming. *ACM Transactions on Mathematical Software*, 29, 58–81.

van Ginneken, B., Frangi, A., Staal, J., ter Haar Romeny, B., & Viergever, M. (2002). Active shape model segmentation with optimal features. *IEEE Transactions on Medical Imaging*, 21(8), 924–933.

Gonzalez-Jimenez, D., & Alba-Castro, J. (2007). Toward pose-invariant 2-D face recognition through point distribution models and facial symmetry. *IEEE Transactions on Information Forensics and Security*, 2(3), 413–429.

Gower, J., & Dijksterhuis, G. (2004). *Procrustes problems*. London: Oxford University Press.

Heimann, T., & Meinzer, H. P. (2009). Statistical shape models for 3d medical image segmentation: a review. *Medical Image Analysis*, 13(4), 543–563.

Heimann, T., Wolf, I., Williams, T., & Meinzer, H. P. (2005). 3d active shape models using gradient descent optimization of description length. In *Information processing in medical imaging conference* (pp. 566–577).

Hill, A., & Taylor, C. J. (1994). Automatic landmark generation for point distribution models. In *British machine vision conference* (Vol. 2, pp. 429–438).

Karlsson, J., & Ericsson, A. (2006). A ground truth correspondence measure for benchmarking. In *International conference on pattern recognition* (Vol. 3, pp. 568–573).

Lanitis, A., Taylor, C., & Cootes, T. (1995). Automatic face identification system using flexible appearance models. *Image and Vision Computing*, 13(5), 393–401.

Latecki, L. J., Megalooikonomou, V., Wang, Q., & Yu, D. (2007). An elastic partial shape matching technique. *Pattern Recognition*, 40(11), 3069–3080.

Lekadir, K., Merrifield, R., & Yang, G. (2007). Outlier detection and handling for robust 3-D active shape models search. *IEEE Transactions on Medical Imaging*, 26(2), 212–222.

Leventon, M., Grimson, E., & Faugeras, O. (2000). Statistical shape influence in geodesic active contours. In *IEEE conference on computer vision and pattern recognition* (pp. 316–323).

Marsland, S., Twining, C., & Taylor, C. (2008). A minimum description length objective function for groupwise non-rigid image registration. *Image and Vision Computing*, 26(3), 333–346.

Matthews, S. B. I., & Schneider, J. (2004). Automatic construction of active appearance models as an image coding problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10), 1380–1384.

Meier, D., & Fisher, E. (2002). Parameter space warping: shape-based correspondence between morphologically different objects. *IEEE Transactions on Medical Imaging*, 21, 31–47.

Milborrow, S., & Nicolls, F. (2008). Locating facial features with an extended active shape model. In *European conference on computer vision* (pp. 504–513).

Munsell, B. C., Dalal, P., & Wang, S. (2008). Evaluating shape correspondence for statistical shape analysis: a benchmark study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11), 2023–2039.

Munsell, B., Temlyakov, A., & Wang, S. (2009). Fast multiple shape correspondence by pre-organizing shape instances. In *IEEE conference on computer vision and pattern recognition* (pp. 840–847).

Ozcan, E., & Mohan, C. K. (1997). Partial shape matching using genetic algorithms. *Pattern Recognition Letters*, 18, 987–992.

Powell, M. (1995). A thin plate spline method for mapping curves into curves in two dimensions. In *Proc. computational techniques and applications* (pp. 43–57).

Richardson, T., & Wang, S. (2005). Nonrigid shape correspondence using landmark sliding, insertion and deletion. In *International conference on medical image computing and computer assisted intervention* (pp. II–435–442).

Rueckert, D., Frangi, A., & Schnabel, J. (2001). Automatic construction of 3D statistical deformation models using non-rigid registration. In *International conference on medical image computing and computer assisted intervention* (pp. 77–84).

Saber, E., Xu, Y., & Tekalp, A. (2005). Partial shape representation by sub-matrix matching for partial matching guided image labeling. *Pattern Recognition*, 38, 1560,573.

Schmidt, F., Farin, D., & Cremers, D. (2007). Fast matching of planar shapes in sub-cubic runtime. In *International conference on computer vision* (pp. 1–6).

Seshadri, K., & Savvides, M. (2009). Robust modified active shape model for automatic facial landmark annotation of frontal faces. In *IEEE international conference on biometrics: theory, applications and systems* (pp. 1–8).

Shenton, M., Gerig, G., McCarley, R., Szvackely, G., & Kikinis, R. (2002). Amygdala-hippocampal shape differences in schizophrenia: the application of 3d shape models to volumetric MR data. *Psychiatry Research. Neuroimaging*, 115, 15–35.

Styner, M., Rajamani, K., Nolte, L. P., Zsemlye, G., Szekely, G., Taylor, C., & Davies, R. (2003). Evaluation of 3D correspondence methods for model building. In *Information processing in medical imaging conference* (pp. 63–75).

Thodberg, H. (2003). Minimum description length shape and appearance models. In *Information processing in medical imaging conference* (pp. 51–62).

Veltkamp, R., & Hagedoorn, M. (1999). *State of the art in shape matching*. Tech. Rep. Technical Report UU-CS-1999-27, Utrecht.

Veltkamp, R., & Tanase, M. (2005). Part-based shape retrieval. In *ACM multimedia* (pp. 543–546).

Wang, S., Kubota, T., & Richardson, T. (2004). Shape correspondence through landmark sliding. In *IEEE conference on computer vision and pattern recognition* (Vol. 1, pp. 143–150).

Xie, J., & Heng, P. (2005). Shape modeling using automatic landmarking. In *International conference on medical image computing and computer assisted intervention* (pp. II–709–716).