Bending-Invariant Correspondence Matching on 3D Human Bodies for Feature Point Extraction

Samuel S.-M. Li, Charlie C.L. Wang*, *Member, IEEE*, and Kin-Chuen Hui

***Abstract*— In this paper, we present an automatic approach to match correspondences on 3D human bodies in various postures so that feature points can be automatically extracted. The feature points are very important to the establishment of volumetric parameterization around human bodies for the human-centered customization of soft-products [1]. For a given template human model with a set of predefined feature points, we first down-sample the input model into a set of sample points. Then, the corresponding points of these samples on the human model are identified by minimizing the distortion with the help of a series of transformations regardless of their differences in postures, scales or positions. The basic idea of our algorithm is to transform the template human body to the shape of the input model iteratively. To generate a bending invariant mapping, the initial correspondence/transformation is computed in a *Multi-Dimensional Scaling* (MDS) embedding domain of 3D human models, where the Euclidean distance between two samples on a 3D model in the MDS domain corresponds to the geodesic distance between them in $\Re^3$. As the posture change (i.e., the body bending) of a human model can be considered as approximately isometric in the intrinsic 3D shape, the initial correspondences established in the MDS domain can greatly enhance the robustness of our approach in body bending. Once the correspondences between the surface samples on the template model and the input model are determined after iterative transformations, we have essentially found the corresponding feature points on the input model. Lastly, the locations of the feature points on the input model are refined by a curvature map based local matching step..**

***Note to Practitioners*—The research work presented in this paper is to support the geometric solution for the design automation of human-centered customization of freefrom products (e.g., apparel products, shoes and glasses). The design automation of such products with complex geometric shapes is based on how to establish the relationship between the product and the human body and how to maintain such a relationship. This is quite different from the design automation function provided by current commercial 3D/2D *Computer-Aided Design* (CAD) systems. A prior work presented in [1] gives a solution to such problems. However, in that approach, the correspondences of the feature points on human bodies are interactively specified by users. The algorithm proposed in this paper aims at providing an automatic method to extract the feature points on the input human bodies, which serves as the preprocessing step of [1] in the pipeline of design automation for human-centered freeform products. On the other aspect, most of the existing cloth CAD systems focus on the simulation of virtual wear and do not provide the design automation function for the personal customization of apparel products.**

***Index Terms*— correspondence matching, bending invariant, multi-dimensional scaling, sign-flip, feature points.**

Manuscript received August 28, 2010; revised December 20, 2010; accepted April 8, 2011.

All the authors are with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong. Corresponding Author: Charlie C.L. Wang; E-mail: cwang@mae.cuhk.edu.hk; Tel: (852) 2609 8052; Fax: (852) 2603 6002.

I. Introduction

DIFFERENT from the design automation functions provided by current commercial *Computer-Aided Design* (CAD) systems that are developed for products with regular shapes and are usually driven by dimensional parameters, the design automation of human-centered soft-products relies on establishing the volumetric parameterization of the spaces around human bodies (see Fig.1), where one of the challenging steps is how to extract the feature points on the 3D models of human bodies to serve as the anchors to constrain the volumetric parameterization. In our prior research [1], the feature points (at least part of them) are specified by users or semi-automatically selected by rule-based systems (e.g., [2] and [3]). An automatic method is proposed here to extract them for the downstream processing in [1].

A. Problem definition

Given a template 3D human model T represented as a polygonal mesh surface $M_T \in \Re^3$ with a set of predefined feature points, G_T, we are going to find the corresponding feature points, G_H, on the surface M_H of an input human model H. Without loss of generality, M_H is also represented by a polygonal mesh in $\Re^3$ and both M_T and M_H have surface normals facing outwards.

The automatic extraction is challenging for two reasons. First, the feature points on human body are not always located at the shape extremities, therefore the local shape matching based methods cannot robustly give satisfactory results. Second, the robustness of local shape matching is more problematic when the postures of human bodies are varied (i.e., the 3D bodies are bended).

In this paper, we propose a global deformation based fitting method to automatically find the correspondences between M_T and M_H, and thus the locations of feature points on H. Specifically, we are going to find a mapping Υ to minimize the distortion function E as

$$E(\Upsilon) = \int \|M_H - \Upsilon(M_T)\|^2 ds \tag{1}$$

with $\|\cdots\|$ being the L^2-norm in $\Re^3$. In other words, by the optimal mapping function

$$\Upsilon = \arg\min E(\Upsilon), \tag{2}$$

we can determine the feature points by

$$G_H = \{\mathbf{g} \mid \mathbf{g} = \Upsilon(\mathbf{q}), \forall \mathbf{q} \in G_T\}. \tag{3}$$

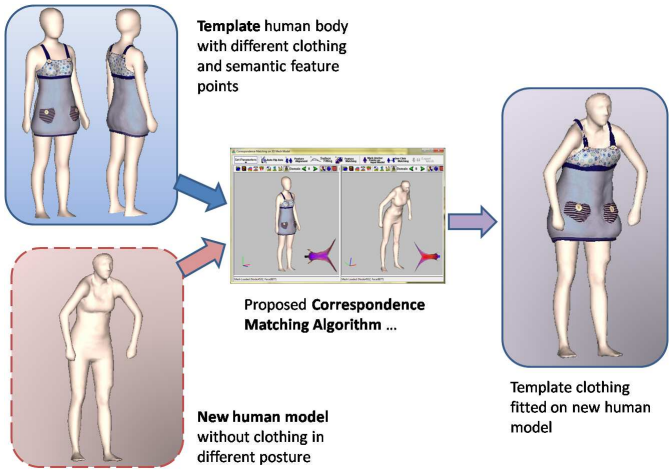


Fig. 1. Illustration of garment design automation, where the template model has predefined features and the clothes are designed around the template body. By the correspondences between human bodies, the garments are refit to the new human model.

To have a refined matching, some important points in G_H should have local shape distributions similar to their corresponding points in G_T. This serves as constraints for the minimization problem defined in Eq.(2).

B. Main features

The main features of our method are outlined as follows.

- An MDS-based point matching algorithm is investigated to align the initial correspondences between the template human model and the given 3D human model. A sign-flip correction technique is developed to enhance the robustness of MDS embedding. The details of sign-flip problem can be found in section IV-B. Without this sign correction technique, the MDS-based method cannot be applied to find correct matching on those nearly symmetric models like human bodies.
- Starting from the initial correspondences, a global alignment technique is exploited to iteratively find a mapping function (via the point correspondences) that optimizes surface proximities and is constrained by feature points (see sections IV-D and IV-E).

These main features of our method lead to a robust feature extraction technique for 3D human bodies in various postures. In fact, the method proposed in this paper can also be applied to other classes of models which are approximately isometric. Although whole human bodies are employed as examples in this paper, there is no difference if we apply it to parts of human bodies (e.g., feet, hands, and faces).

The rest of the paper is organized as follows. After reviewing the related work in section II, the overview of our algorithm is given in Section III and the detailed methodology of our algorithm is presented in Section IV. The experimental results are shown and studied in Section V. Lastly, our paper ends with the conclusion section.

II. Literature Review

Point matching algorithms in literature can be classified into two major categories: *local feature matching* and *global iterative alignment* techniques.

A. Local feature matching techniques

Feature based matching has been a common approach in shape matching [4]–[7]. It can be found in 2D applications such as photo panorama, text recognition and animation morphing. The features are always represented by grouping regional information in point, known as descriptor. Two well-known descriptors for image are shape contexts and spin images [8], both utilizing a histogram obtained by binning the space around a point according to the Euclidean metric and collecting point counts. These methods have subsequently been generalized in a straight-forward manner to handle 3D point sets. However, neither space contexts nor spin images are invariant to shape bending. Some extra work has to be done to deal with the non-rigid object matching problem in 3D scenario.

The *Curvature Map* introduced by Gatzke et al. in [9] is a kind of feature descriptor which gathers local differential geometry information at a point. A curvature map is first defined around a point $\mathbf{v}$, and then accumulates curvature information from a region around $\mathbf{v}$ and takes one of two forms: a one-dimensional (1-D) map, which only considers the distance from v, and a two-dimensional (2-D) map that uses both distance and orientation information.

In most shape matching applications, geometric feature is a very important portion to be preserved during matching processes (e.g., reverse engineering of mechanical parts [5]). One significant drawback to incorporating curvature map in 3D object matching is its disability to handle bended objects. Although geodesic binning is invariant to bending, the histograms computed are based on curvature distributions, which are not invariant to bending.

B. Global iterative alignment techniques

In global iterative alignment matching approaches, there are two unknown variables that have to be determined: the *correspondence* and the *transformation*. While it is impossible to solve either variable without information regarding the other, it is possible to optimize these unknowns by determining them iteratively. Once the correspondence is given, the transformation can be guessed with reasonable knowledge. On the other hand, the correspondence can be searched if the transformation is known. Hence, it leads to a solution of the correspondence problem by alternating the estimations of correspondence and transformation (e.g., [10]–[12]).

The ICP algorithm is the simplest one among these methods. It utilizes the nearest-neighbor relationship to assign a binary correspondence at each step. This estimation of the correspondence is then used to refine the transformation, and vice versa. It is a very simple and fast algorithm which is guaranteed to converge to a local minimum. Chui et al. enhanced this algorithm in [13] by making two significant improvements: *Soft-assign* idea and *Robust Point Matching – Thin Plate Spline* (RPM-TPS) algorithm.

The basic idea of the soft-assign [12] is to relax the binary correspondence variables to be a continuous valued matrix M in the interval $[0, 1]$, while enforcing the row and column constraints. The continuous nature of the correspondence matrix M basically allows fuzzy, partial matches between the point sets. Hence the correspondences are able to improve gradually and continuously during the optimization without jumping around in the space of binary permutation matrices. The row and column constraints are enforced via iterative row and column normalization of the corresponding matrix M. Chui et al. also proposed RPM-TPS as the parameterization of non-rigid spatial mapping transformation. Their work is based on the RPM algorithm that involves a dual update process embedded within an annealing scheme. Obviously, the iterative alignment approach does not require any complicated algorithm or computation. Nevertheless, the initial guess mapping of correspondence must be good enough in order to solve the bending invariant matching problem.

On the contrary, a hybrid approach of this technique on the MDS signature would be a much reliable and advanced approach for deformable shape matching. Recently, Lipman and Funkhouser [14] proposed a surface correspondence matching method by repeatedly computing Möbius transformations, which requires the input models to be two-manifold – this is more restricted than our approach that is based on spatial transformation.

C. Other Approaches

Apart from the feature based and iterative alignment approaches, other previous work of matching approaches for shape matching, such as skeletal based matching [15] and image based matching, has also been studied. Some famous approaches, such as the shock graph [16], reeb graph [17], conformal geometry [18], and canonical homology basis [19], achieved the shape matching goal in certain fields of applications. Nevertheless, the structural information of these approaches does not provide detailed matching ability on mesh surfaces. The recent development of semantic features and relevant applications can be found in [20]–[23].

III. Algorithm Overview

The proposed bending-invariant matching algorithm integrates *Global Surface Alignment* and *Feature Based Matching* techniques, which are found as the two major techniques in shape matching studies. The integration of these two techniques inherits their advantages. The matching algorithm has three steps: 1) *posture alignment*, 2) *surface fitting* and 3) *feature matching refinement*.

Firstly, the *posture alignment* step transforms the template model to the input model non-rigidly according to the control point mapping defined by their similar isometric signatures – the *multi-dimensional scaling* (MDS) embedding. The MDS embedding of a given model is defined in a k-dimensional domain according to the relative distribution of surface points on the model. The robustness of finding good initial correspondences according to the MDS embedding is guaranteed by the observation that 1) the shapes of a human body in different postures are nearly isometric to each other and 2) the isometric shapes have the same MDS embedding.

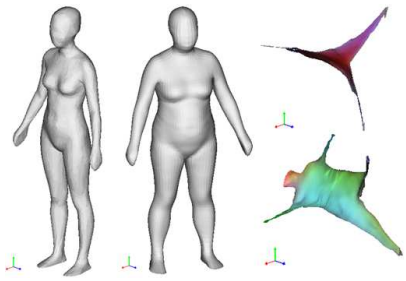


Fig. 2. Examples of the template model and the input model before versus after the MDS transformation. Note that, the Euclidean coordinates in $\Re^3$ and the RGB colors of the transformed models (in the right) represent the normalized coordinates in the 6-dimensional MDS domain.

Secondly, the *surface fitting* step refines the surface of the transformed template by optimizing the fitness and the smoothness iteratively. Two main processes, *surface fitness optimization* and *surface smoothing*, are repeatedly applied until changes on the surface converge to a limited amount. The surface fitting procedure employs a bi-directional mapping concept and an orientation-aware movement, which greatly improve the fitting quality of the template model.

Finally, the *feature matching refinement* step further refines the correspondences by adopting the feature descriptor constraints on particular surface regions. At this stage, the descriptor is encoded on a surface point with curvature distribution information on the surface around it. The concept is similar to the Curvature Maps presented by Gatzke et al. in [9] but in a constrained manner. Hence, the pre-defined feature points on the template model can be mapped to the input model according to the feature-aligned models.

IV. Methodology

This section presents the details of our method. First of all, two important techniques to enhance the robustness of our matching algorithm, MDS transformation and sign-flip correction, are introduced. After that, the three steps of our algorithm, 1) pose alignment, 2) surface fitting and 3) feature matching refinement, are detailed.

A. MDS transformation

To robustly establish the initial correspondences between the template human model T and the input model H, their MDS embeddings T_{MDS} and H_{MDS} are computed via the classical MDS transformation, which involves a computationally expensive step – eigenvector analysis. In order to simplify and speed up the computation, their surface models M_T and M_H are sampled into m points, $\tilde{M}_T = \{\mathbf{t}_1, \ldots, \mathbf{t}_m\}$ and $\tilde{M}_H = \{\mathbf{h}_1, \ldots, \mathbf{h}_m\}$, by the *Farthest Point Sampling* (FPS) method in [24]. According to this simplified shape representation, when the samples of $\tilde{M}_T$ are mapped to new positions, e.g., $\tilde{M}^*_T = \{\mathbf{t}^*_1, \ldots, \mathbf{t}^*_m\}$, the newly mapped (or warped) shape of M_T can be determined by a *Radial Basis Function* (RBF) based warping function (ref. [1], [25]).

For these sample points, we calculate their geodesic distance map by the fast marching algorithm on triangulated domains

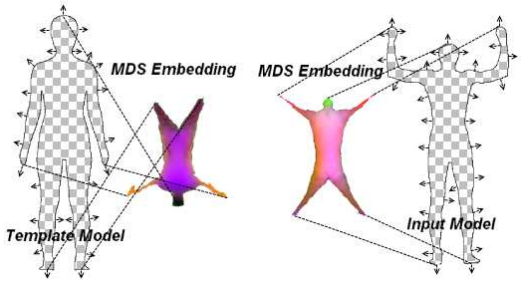


Fig. 3. The MDS embeddings (colored) of two human bodies (checkerboard) are significantly different due to the sign-flip, where the closest point search will give wrong correspondences. The correct point correspondences are specified by the dashed lines between the models in the spatial and MDS domains.

(ref. [26]) and store the result as a $m \times m$ symmetric and zero diagonal matrix D

$$D = \{d_{ij}\} = \{\xi^2(i,j)\} \tag{4}$$

where $\xi(i,j)$ evaluates the geodesic distance between the sample points $\mathbf{t}_i$ and $\mathbf{t}_j$ on M_T (or between $\mathbf{h}_i$ and $\mathbf{h}_j$ on M_H). The *Multi-Dimensional Scaling* (MDS) process is an important step to align the given model pair into the same orientation, scale and posture. Here, we use the Gaussian affinity matrix A as the input of the MDS transformation which is similar to the approach [27].

$$A_{ij} = 1 - e^{-\frac{d_{ij}}{2\delta}} \tag{5}$$

The Gaussian kernel width δ is chosen as the maximum value among the elements of D.

The positions of m sample points in the k-dimensional MDS embedding domain can then be computed by building and decomposing the inner product matrix B

$$B = -\frac{1}{2}JAJ \tag{6}$$

where $J = I - \frac{1}{m}LL^T$ and $L_{1\times m} = [1, 1, \ldots, 1]^T$. Firstly, k most dominated eigenvalues of B, $\lambda_1 > \lambda_2 > \ldots > \lambda_k \geq 0$, and their corresponding eigenvectors are calculated by the power method. The k eigenvectors with m components for each are listed in the matrix $V_{m\times k}$. Lastly, the resultant coordinates of the sampling points in the MDS domain can be determined by

$$X_{m\times k} = V_{m\times k}\Lambda^{\frac{1}{2}}_{k\times k} \tag{7}$$

with $\Lambda_{k\times k} = diag(\lambda_1, \lambda_2, \ldots, \lambda_k)$. Each row of $X_{m\times k}$ represents a point coordinate in the k-dimensional MDS domain.

The value of k directly affects the robustness of initial shape matching. By our experimental tests, $k = 6$ can give satisfactory results in all cases while still keeping an acceptable computational speed. Therefore, in all the figures shown in this paper, the first three components of a point in the MDS domain are displayed as the Euclidean coordinates in $\Re^3$ and the next three components are displayed by the RGB colors. See Fig.2 for an example.

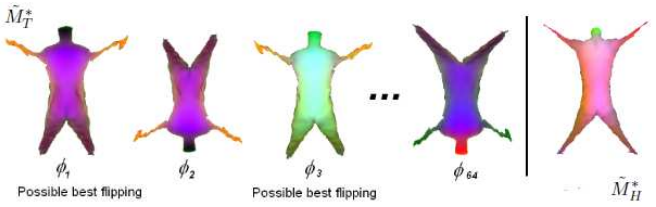


Fig. 4. Illustration of several possible combinations of axis flipping. It can be observed that two flippings, ϕ_1 and ϕ_3, have the minimal value of $C(\phi_k)$ defined in Eq.(11); however, their costs are different in $C^*(\phi_k)$ (Eq.(12)).

B. Sign-flip correction

The shapes of human bodies in the MDS domain are quite similar to each other. Ideally, the correspondences between the points $\mathbf{t}_i^* \in \tilde{M}_T^*$ and $\mathbf{h}_i^* \in \tilde{M}_H^*$ can be determined by the closest point search. However, such a mapping between $\tilde{M}_T^*$ and $\tilde{M}_H^*$ is neither bijective nor robust. Our investigation finds that the main challenge comes from the random selection of the sign of eigenvalues (therefore the direction of eigenvectors) in the MDS analysis. Thus, the shapes of $\tilde{M}_T^*$ and $\tilde{M}_H^*$ can greatly differ in terms of their axis directions – called *sign-flip* (as illustrated in Fig.3).

Under the exhaustive search framework proposed by Shapiro and Brady [28], the alignment of the MDS embedding can be achieved by finding the combination of axes swapping which minimizes a shape difference metric. For instance, there are $2^6 = 64$ sign flipping combinations for 6D MDS embeddings (see Fig.4). Therefore, 2^d different sign flipping functions can be defined for a point $\mathbf{t}^*$ in the d-dimensional MDS domain as

$$\phi_{k\ (k=1,\cdots,2^d)}(\mathbf{t}^*) = S^k_{d\times d}\mathbf{t}^* \tag{8}$$

with $S^k_{d\times d} = diag(\cdots, (-1)^{k \bmod i}, \cdots)$ $(i = 1, \ldots, d)$. Among them, the one giving the minimal cost on a shape error metric is selected as the corrected MDS embedding. The study of metrics is conducted below.

Let us assume that a function $f^*(\cdots)$ can be established[1] to map the points $\mathbf{t}_i^* \in \tilde{M}_T^*$ with sign-flip corrected to the positions of points $\mathbf{h}_i^* \in \tilde{M}_H^*$ as

$$f^*(\phi_k(\mathbf{t}_i^*)) \mapsto \mathbf{h}_i^* \quad \forall i = 1, \ldots, n. \tag{9}$$

On using the same correspondences between points but in the spatial domain, we can have a similar mapping function $f(\cdots)$

$$f(\phi_k(\mathbf{t}_i)) \mapsto \mathbf{h}_i \quad \forall i = 1, \ldots, n. \tag{10}$$

In the work of Shapiro and Brady [28], a cost function $C(\phi_k)$ is formulated to score the shape distortion by measuring the distances between sample points before versus after applying the mapping function.

$$C(\phi_k) = \sum_{i=0}^{n}\sum_{j=0}^{n} |\|\mathbf{t}_i - \mathbf{t}_j\|^2 - \|f(\phi_k(\mathbf{t}_i)) - f(\phi_k(\mathbf{t}_j))\|^2| \tag{11}$$

The major drawback of this cost function is that it only measures the distortion in shape but does not consider the

[1]Details about how to construct such a mapping function can be found in the section IV-C.

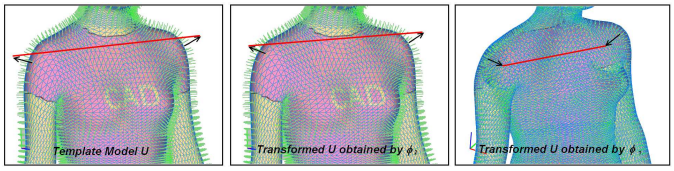


Fig. 5. Sign-flip correction with normal offset component in cost function. A significant difference can be distinguished by comparing the length of the *red* line which is constructed by the normal offsets of the vertices on the template model (left) and that on the transformed models (middle: ϕ_3 and right: ϕ_1) respectively.

swap of surface orientation. Take the human model shown in Fig.4 as an example, ϕ_1 and ϕ_3 lead to the same minimal cost by $C(\phi_k)$ in Eq.(11) – i.e., the sign-flip cannot be successfully corrected. In order to consider the surface orientation in the shape distortion metric, we modify the cost function so that it is based on the shifted positions along the normal vectors, $\mathbf{n}_{\mathbf{t}_i}$.

$$C^*(\phi_k) = \sum_{i=0}^{n}\sum_{j=0}^{n} |\|(\mathbf{t}_i + \omega\mathbf{n}_{\mathbf{t}_i}) - (\mathbf{t}_j + \omega\mathbf{n}_{\mathbf{t}_j})\|^2 - \|(f(\phi_k(\mathbf{t}_i)) + \omega\mathbf{n}_{\mathbf{t}_i}) - (f(\phi_k(\mathbf{t}_j)) + \omega\mathbf{n}_{\mathbf{t}_j})\|^2| \tag{12}$$

As shown in the example of Fig.5, the new cost function will have different values with $C^*(\phi_1)$ and $C^*(\phi_3)$, where the one with the correct surface orientation will have a smaller value. A very small number should be selected for the offset value ω; in all our tests, we use $\omega = 0.5$ centimeter. Then, an optimal sign-flip function can be found by

$$\phi^* = \arg\min C^*(\phi_k). \tag{13}$$

C. Pose alignment

After transforming the sample points from $\Re^3$ into the MDS domain, the sample points in MDS domain are then be used to correct the sign-flip and estimate the pose alignment.

Given the sample points $\mathbf{t}_i^* \in \tilde{M}_T^*$, we first employ the *approximate-nearest-neighbor* (ANN) search [29] to find the closest points $\mathbf{c}^*(\mathbf{t}_i^*) \in \tilde{M}_H^*$ of $\mathbf{t}_i^*$. Using these correspondences, we can find the mapping of samples in $\Re^3$ from M_T to M_H as $\mathbf{t}_i \mapsto \mathbf{c}(\mathbf{t}_i) \in M_H$. Then, a transformation function f can be defined on n such correspondences by the RBF-based thin-plate spline transformation as

$$f(\mathbf{p}) = a_n + [a_{n+1}, a_{n+2}, a_{n+3}]\mathbf{p} + \sum_{i=1}^{n} a_i g(\|\mathbf{p} - \mathbf{c}(\mathbf{t}_i)\|), \tag{14}$$

where the coefficients $a_{n,\ldots,n+3}$ define the affine transformation of the point $\mathbf{p}$, a_is define the weights of points $\mathbf{p}$ to the control point $\mathbf{c}(\mathbf{t}_i)$, and the basis function $g(r)$ is chosen as $g(r) = r^2$. An optimal transformation function can be determined by solving the following linear equation system

$$\begin{bmatrix} G - \lambda I & P^T \\ P & 0 \end{bmatrix} [a_i] = \begin{bmatrix} Y \\ 0 \end{bmatrix} \tag{15}$$

where $G = [g_{ij}]$ with $g_{ij} = g(\|\mathbf{t}_i - \mathbf{c}(\mathbf{t}_j)\|)$, $P^T = [1, \mathbf{t}_i^x, \mathbf{t}_i^y, \mathbf{t}_i^z]$, $Y = [\mathbf{c}^x(\mathbf{t}_i), \mathbf{c}^y(\mathbf{t}_i), \mathbf{c}^z(\mathbf{t}_i)]$, and λ is the so-called regularization (smoothing) parameter [25] of the transformation function. The greater λ is, the stronger its smoothing effect is.

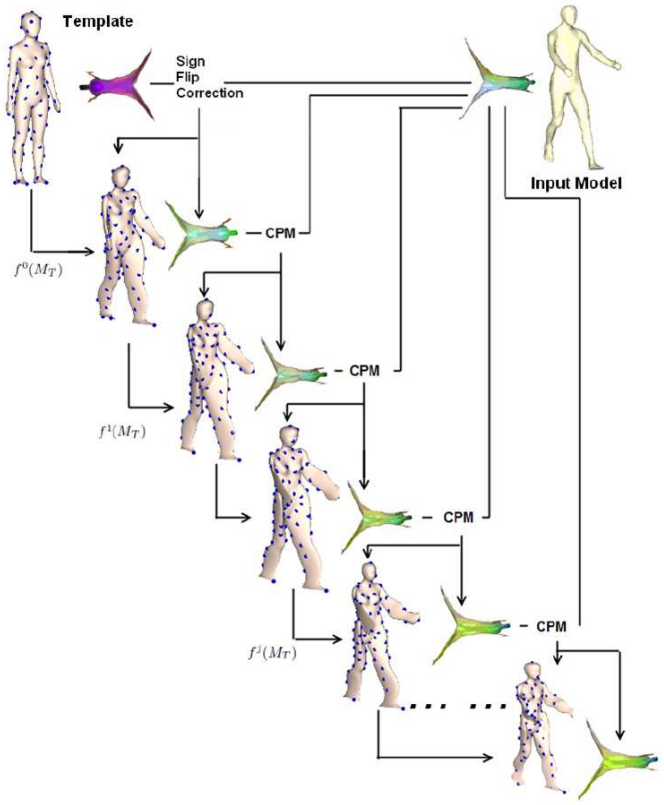


Fig. 6. Flow of the posture alignment step and the relationship between the spatial and MDS domains of the template and the input models. The correspondences between sample points in $\Re^3$ are established by the *closest point match* (CPM) conducted in the MDS-domain. Note that, the blue points on the human bodies are sample points used in this step of our algorithm.

In the pose alignment step, we first apply the sign-flip correction technique to obtain a "good" MDS-embedding for the template model. Here, a very small number of sample points are used to avoid being trapped on local optimum at the beginning of the correspondence mapping algorithm. We choose $n = 20$. Also, a large value, $\lambda = 10^8$, is adopted to obtain a correct sign-flip function ϕ^*.

Starting from a sign-flip corrected MDS-embedding $\phi^*(M_T^*)$ of the given template model, we search for the closest points $\mathbf{c}(\mathbf{t}_i^*)$ of the sample points $\mathbf{t}_i^* \in \phi^*(M_T^*)$, thus also determine the correspondences in $\Re^3$ as $\mathbf{t}_i \mapsto \mathbf{c}(\mathbf{t}_i) \in M_H$. By these correspondences, a transformation function f^0 can be determined by Eq.(15). We can compute the new sign-flip corrected MDS-embedding from $f^0(M_T)$; therefore, the new correspondences and the new transformation function f^1 can also be computed. Repeatedly applying this correspondence-transformation computing step, we can iteratively update the transformation function $f^j(M_T)$ $(j = 1, 2, 3, \cdots)$ to make it more and more aligned with the pose of M_H (see Fig.6 for an illustration). During the iteration, we decrease the value of λ by about $1/10$ gradually after each loop. According to the experiments, the changes of $f^j(M_T)$ would converge to a very small value (e.g., 10^{-5}) within ten iterations.

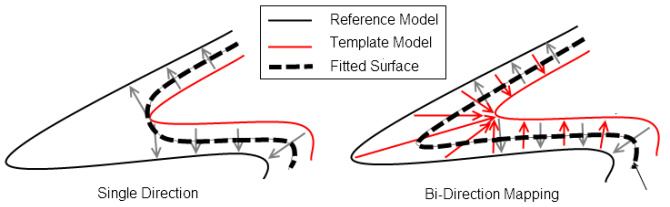


Fig. 7. Surface fitting by single-directional and bi-directional mappings of points respectively. Fitted surface is displayed in dashed line, and the mapping is displayed by arrows.

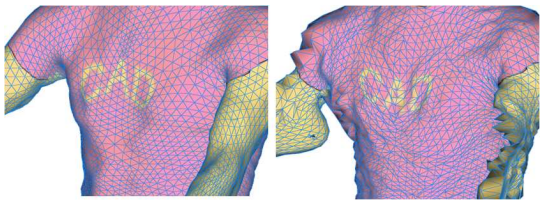


Fig. 8. Surface fitting with (left) and without (right) the smoothing term respectively.

D. Surface fitting

Given a posture-aligned template model $f^j(M_T)$ and an input model M_H, the surface fitting process in this section further increases the accuracy of correspondence mapping Υ defined in Eq.(2). However, to simplify the evaluation of the distortion function E in Eq.(1), we can actually evaluate a discrete version of it on m sample points generated by the farthest point sampling. For the examples shown in this paper, we use all the vertices of the template model (i.e., $m = 11,074$). For each sample point $\mathbf{t}_i \in M_T$, we find its closest point $\mathbf{c}(\mathbf{t}_i) \in \{\mathbf{h}_j | \forall \mathbf{h}_j \in M_H\}$. The mapping function Υ is then updated by moving the vertices $\mathbf{t}_i \in M_T$ according to $\mathbf{c}(\mathbf{t}_i)$. Meanwhile, the smoothness of surface on the template model M_T is constrained. The method was employed in [1] for a similar purpose. However, it has two defects. Firstly, wrong mappings appear on the near but opposite surfaces. Secondly, the surface smoothness is distributed in regions with dense meshes.

To overcome these defects, we add three modifications to the procedure of surface fitting.

- First of all, we do not move the vertices on the dynamic mesh (i.e., the warped template model) to their closest points on the model to fit. Instead, we move the vertices along their own normal vectors like [30]. The mapping established by the closest point match (CPM) is validated by checking two constraints: 1) normal compatibility – i.e., if the angle between $\mathbf{n}_{\mathbf{t}_i}$ and $\mathbf{n}_{\Upsilon(\mathbf{t}_i)}$ is too large (by a parameter σ), and 2) relaxation power τ that controls the maximal allowing Euclidean distance between $\mathbf{t}_i$ and its mapped position $\Upsilon(\mathbf{t}_i)$.
- Secondly, a bi-directional mapping is conducted to decide the moving direction of the vertices on the dynamic mesh. A comparison of single-directional and bi-directional mappings is given in Fig.7. It is easy to find that bi-directional mapping fits surface into narrowed (or sharped) regions like armpits and crotches more flexibly.

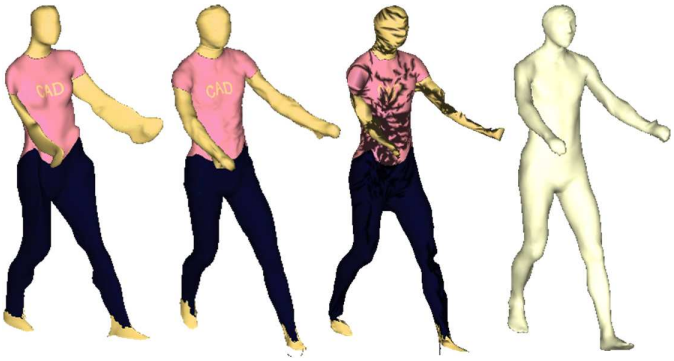


Fig. 9. An example of surface fitting. From left to right, the pose aligned template model, the surface fitting result of the template model by the method proposed in section IV-D, the surface fitting result by [1], and the input human model M_H.

- Lastly, a smoothing term is added to the fitting procedure to enhance the smoothness of the fitting result. Figure 8 shows the results with and without the smoothing term.

The pseudo-code of the surface fitting algorithm is listed in **Algorithm** SurfaceFitting. The correspondence mapping Υ is then updated by $\Upsilon : M_T \mapsto M'_T$. Figure 9 shows an example fitting result and its comparison with the method presented in [1].

Algorithm 1 SurfaceFitting

Update the template model M_T to M'_T by the optimized posture alignment function $f^j(\cdots)$;
$\tau \Leftarrow 0.5$ and $\sigma \Leftarrow 0.9$;
repeat
  $\forall \mathbf{t}_i \in M'_T$, find the closest point $\mathbf{c}(\mathbf{t}_i) \in M_H$;
  Establish a new mapping function ξ by $\mathbf{t}_i \mapsto \mathbf{c}(\mathbf{t}_i)$;
  for all $\mathbf{t}_i \in M'_T$ **do**
    $\mathbf{p}_i \Leftarrow \xi(\mathbf{t}_i)$;
    if $(\mathbf{n}_{\mathbf{t}_i} \cdot \mathbf{n}_{\mathbf{p}_i}) > \sigma$ **then**
      if $|\mathbf{t}_i + 0.5\tau\mathbf{n}_{\mathbf{t}_i} - \mathbf{p}_i| < |\mathbf{t}_i - 0.5\tau\mathbf{n}_{\mathbf{t}_i} - \mathbf{p}_i|$ **then**
        $\mathbf{t}_i \Leftarrow \mathbf{t}_i + 0.5\tau\mathbf{n}_{\mathbf{t}_i}$;
      else
        $\mathbf{t}_i \Leftarrow \mathbf{t}_i - 0.5\tau\mathbf{n}_{\mathbf{t}_i}$;
      end if
    end if
  end for
  $\sigma \Leftarrow 0.9\sigma$ and $\tau \Leftarrow 1.1\tau$;
  Smoothing M'_T by a Gaussian filter;
until $\sigma < \sigma_{\min}$ and $\tau > \tau_{\max}$
return;

E. Feature matching refinement

After determining an updated mapping Υ by the surface fitting of M_T to M_H, we can extract the feature point set G_H by Υ as Eq.(3). However, the previous fitting steps are global alignment based which do not consider the local shape distribution, and thus the locations of feature points are not accurate enough. A feature matching refinement is conducted as the last step of our algorithm to further adjust the locations of the selected feature points G_H on M_H.

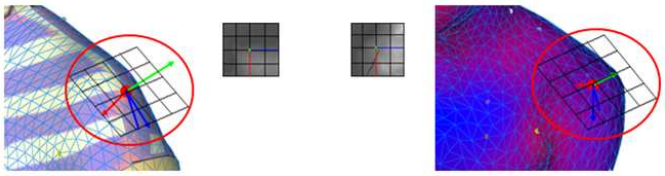


Fig. 10. Construction of the feature descriptor around a point $\mathbf{v}$ on a template model (left) and on an input human model (right). The Gaussian curvature map is generated within the red circle. Note that, for illustration purpose, only a 4×4 Gaussian curvature map is displayed. The three axes of local frames are displayed in red, green and blue arrows.

To serve the shape matching on surfaces of human bodies, a local shape descriptor is proposed. Basically, we need a shape descriptor that is invariant to the differences of scale, orientation and topology between the template model M_T and the input model M_H. Given a point $\mathbf{v}$ on a triangular mesh surface M, its feature descriptor $F_r(\mathbf{v})$ with support size r is constructed as follows.

- First, a local frame $[\hat{\mathbf{t}}_1, \hat{\mathbf{t}}_2, \hat{\mathbf{t}}_3]$ at $\mathbf{v}$ is established by letting $\hat{\mathbf{t}}_2$ be along the surface normal at $\mathbf{v}$, $\hat{\mathbf{t}}_1$ be an arbitrary unit vector on the tangent plane at $\mathbf{v}$ and $\hat{\mathbf{t}}_3 = \hat{\mathbf{t}}_1 \times \hat{\mathbf{t}}_2$.
- Second, the points around $\mathbf{v}$ within a radius r are searched and assigned to a point set V_r. The Gaussian curvatures κ_G at these points are evaluated by the method of [31], and the values of Gaussian curvature are normalized from $[\kappa_G^{\min}, \kappa_G^{\max}]$ into the range of $[-1, 1]$, where $\kappa_G^{\min}$ and $\kappa_G^{\max}$ are the minimal and maximal Gaussian curvatures among all the points in V_r respectively.
- Lastly, the normalized Gaussian curvatures at the points in V_r are projected onto the tangent plane of $\mathbf{v}$ to form a Gaussian curvature image with 10×10 pixels – this is our feature descriptor, $F_r(\mathbf{v})$.

Based on our experimental tests, selecting r as ten times of the average edge length on M_T is a good trade-off between robustness and speed. Figure 10 shows an example of the feature descriptor at a local convex region.

Once the feature descriptor scheme has been developed and the feature points on the template model have been defined, the surface of the template model M_T is refined iteratively by re-aligning the feature mapping between the template model M_T and the input model M_H once at a time.

For simplicity, the feature matching algorithm focuses on a single vertex $\mathbf{t}_a$ in G_T during each iteration. The correspondence $\mathbf{h}_a$ of $\mathbf{t}_a$ must be found on the surface of the input human model M_H so that the cost of feature descriptor

$$C_F(\mathbf{t}_a, \mathbf{h}_a) = \|F(\mathbf{t}_a) - F(\mathbf{h}_a)\| \tag{16}$$

is minimized as

$$\mathbf{h}_a = \arg\min C_F(\mathbf{t}_a, \mathbf{h}_a). \tag{17}$$

The search for an optimal $\mathbf{h}_a$ starts from $\mathbf{h}_a = \Upsilon(\mathbf{t}_a)$. A search window with a radius r is established to include all surface points (sampled) on M_H with a distance to $\mathbf{t}_a$ less than r. Then, the minimal feature descriptor cost C between $\mathbf{t}_a$ and all these surface points can be found by an exhaustive search. Note that, during the search, the local frames on the surface samples are rotated to find the best match as the axis $\hat{\mathbf{t}}_1$ of a local frame is arbitrary on the tangent plane of the surface point (see Fig.11 for the illustration). A feature matching refinement example is shown in Fig.12.

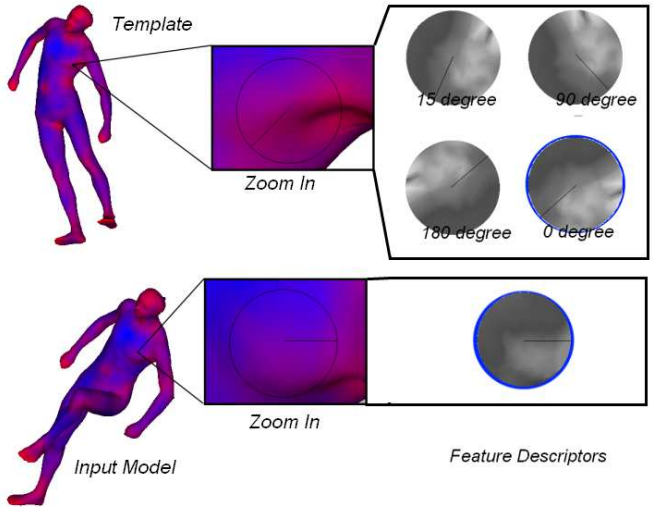


Fig. 11. Illustration of the feature descriptors on a particular region of a vertex on two human models. The top row displays the feature descriptor at different orientations by rotating the local frames around the normal vectors – maps at four example rotations are shown.

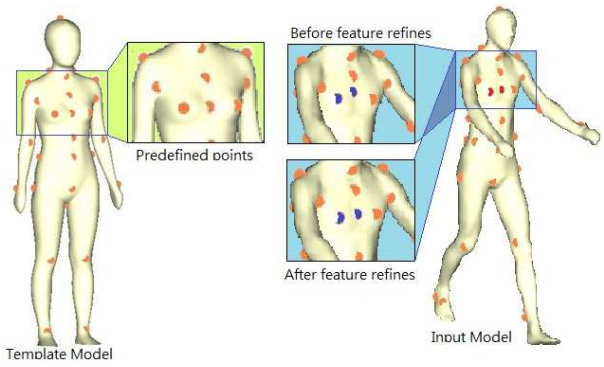


Fig. 12. An example of the feature matching refinement step. It is easy to find that, after the feature matching, the relocated feature points are positioned in places similar to where they are on the template model.

V. Results and Discussion

We have implemented the proposed algorithm in a prototype program by Visual C++ with OpenGL library for 3D visualization of models. The experimental tests are carried out on a PC with Intel Core i5 430 CPU (2.27GHz) plus 4GB main memory running 64bit MS Windows 7. Basically, the computation of all examples can be completed in less than one minute.

Figure 13 shows the results of our approach on four examples of real human bodies with different postures. All the results are generated automatically. The template model with predefined features is shown at the first column of Fig.13. The computational statistics are shown in Table 1.

In Fig.14, we compare the resultant surfaces warped from a template fat human body to a thin human body by the cor-

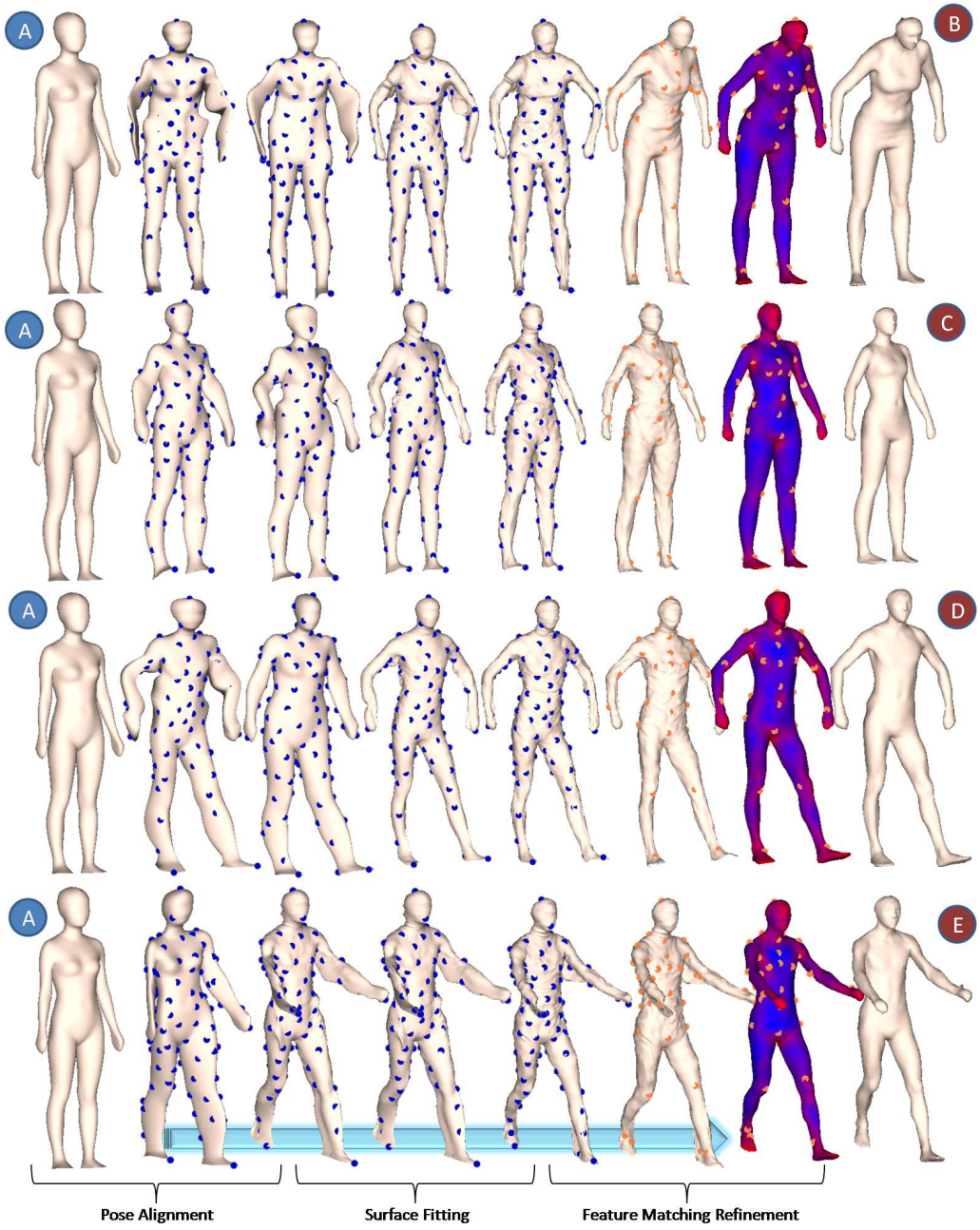


Fig. 13. Results of applying our approach to four examples (the last column of first four rows) by using the template human model shown at the first column of the figure. The progressive results in different steps of our approach are shown in different columns of the figure. The feature points are displayed in orange.

respondences established by various shape matching methods. It is not difficult to find that our method gives the best fitting results.

Another interesting study is about the number of sample points used in the pose alignment step and its effects on the final matching result. An example is given in Fig.15, where the results obtained with 250, 500 and 750 sample points are shown. It is found that the pose aligned result does not lead to a satisfactory matching result in our algorithm if too few sample points are used. However, it does not mean that the more sample points, the more accurate result can be obtained. When further increasing the number of sample points, the

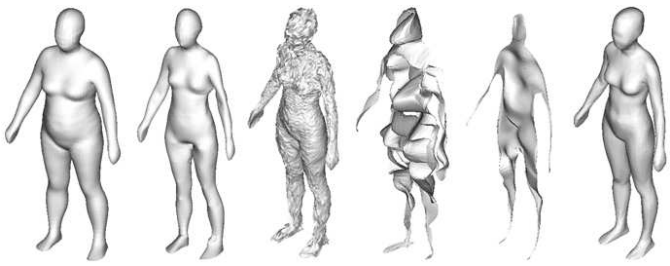


Fig. 14. Comparison of the surfaces warped from a template fat human model (leftmost) to a thin human model (rightmost) by the correspondences generated by different approaches: (2nd column) our method, (3rd column) [27] with 6D MDS, (4th column) [32] with 3D MDS, and (5th column) [13].

TABLE I
COMPUTATIONAL STATISTICS (IN SECOND)

Examples in Fig.13	$A \Rightarrow B$	$A \Rightarrow C$	$A \Rightarrow D$	$A \Rightarrow E$
Pre-computation	11.7	12.5	13.3	12.6
Posture Alignment	18.6	15.2	16.1	19.5
Surface Fitting	14.5	12.6	16.8	17.7
Feature Matching	12.1	9.6	10.9	9.9
Total Time	56.9	49.9	57.1	59.7

* The statistics are tested on a PC with Intel Core i5 430 CPU (2.27GHz) plus 4GB main memory running 64bit MS Windows 7.

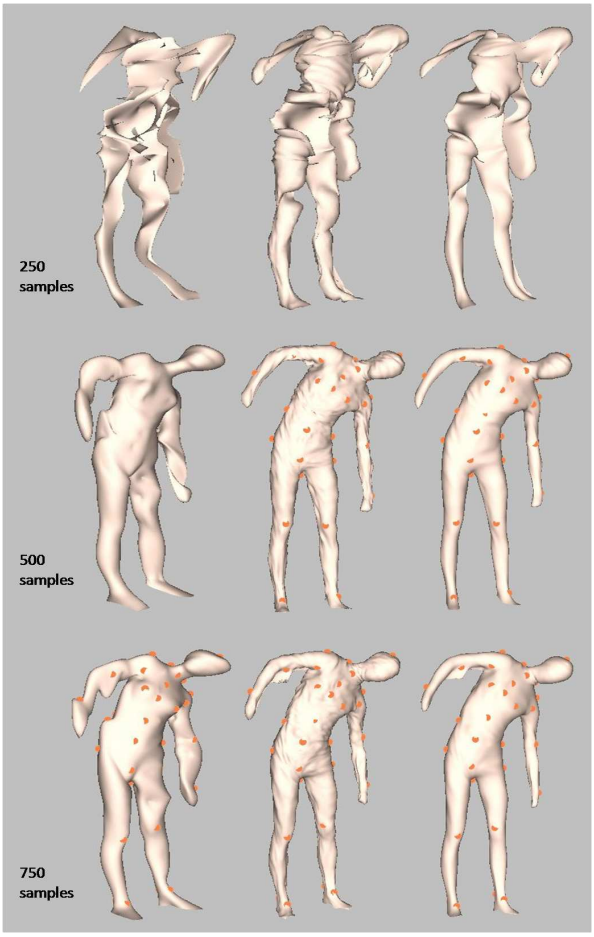


Fig. 15. Comparison of the surface matching results by using different number of samples – the template shown in Fig.13 is employed here too. From left to right, the result by pose alignment, the result after surface fitting, and the final result.

computation (i.e., surface fitting) may be stuck at some local optimum.

According to the experimental tests, satisfactory results can be obtained for those testing examples with a moderate level of deformation. However, one of the limitations of our approach is its restriction on the deformation effects between the models in local regions – specifically, isometric deformation is assumed. For instance, a particular highly stretched area, a dense point distributed region or a twisted surface may fail the validity of the algorithm. Figure 16 shows a study of the geometric errors generated on the matching results to the same human body but in different levels of bending, from which

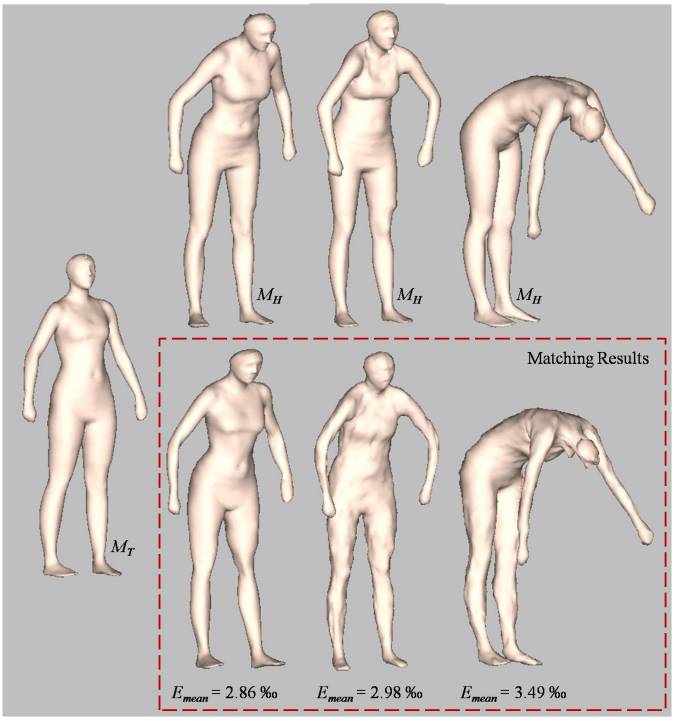


Fig. 16. The results of matching a human body to itself with different postures in different levels of bending. The error of matching results are measured by the publicly available Metro tool [33].

we can find a clear trend that the matching results become less and less accurate. Future work can be done to overcome this problem and one possible solution is to segment the mesh surface before applying MDS transformation. This can greatly reduce the stretch error accumulated in the MDS embedding and eliminate those local dense regions. However, there is a drawback of multiple sign-flip correction problems if the divided segments are symmetrically identical, for example, sign-flip correction cannot be performed on two arm segments alone. Therefore, the possibility of segmentation is still under evaluation at this moment.

Last but not the least, more future work can be done to enhance the performance of the proposed algorithm. In our current implementation, the computation time is highly dependent on the sample rates in all stages of the algorithm, the number of iterations and the RBF warping processes in each step. In the near future, we will consider using the parallel computing power, which is nowadays available on desktop PCs, will be considered to speed up our approach.

VI. CONCLUSION

The proposed algorithm in this paper presents a correspondence identification algorithm on 3D human models by referencing an isometrically similar template model. The presented approach is designed for engineering applications that require feature point identification on the surface of 3D human bodies. The experimental tests have verified the correctness and effectiveness of our approach. The research work presented in this paper can support the geometric solution for the design automation of human-centered customization of

freeform products including clothes, shoes, glasses, etc. As a preprocessing step of volumetric parameterization for design automation [1], the automatic method for extracting feature points can further shorten the time of product design and fabrication cycle.

ACKNOWLEDGMENT

This research is partially supported by the Hong Kong RGC/CERG grant CUHK/412405 and the Hong Kong ITF grant ITS/026/07. The authors would like to thank the anonymous reviewers for their helpful comments.

REFERENCES

[1] C.C.L. Wang, K.C. Hui, and K.M. Tong. Volume parameterization for design automation of customized free-form products. *IEEE Transactions on Automation Science and Engineering*, 4(1):11–21, 2007.

[2] C.C.L. Wang, T.K.K. Chang, and M.M.F. Yuen. From laser-scanned data to feature human model: a system based on fuzzy logic concept. *Computer-Aided Design*, 35(3):241–253, 2003.

[3] C.C.L. Wang, Y. Wang, T.K.K. Chang, and M.M.F. Yuen. Virtual human modeling from photographs for garment industry. *Computer-Aided Design*, 35(6):577–589, 2003.

[4] S. Pankanti, C. Dorai, and A.K. Jain. Robust feature detection for 3d object recognition and matching. In *Proc. SPIE Conference on Geometric Methods in Computer Vision, vol.2031*, 1993.

[5] X. Jiao and M.T. Heath. Feature detection for surface meshes. In *The International Society of Grid Generation*, 2002.

[6] N. Joshi and D. Dutta. Feature simplification techniques for freeform surface models. *ASME Journal of Computing and Information Science in Engineering*, 3:177–186, 2003.

[7] N. Litke, M. Droske, M. Rumpf, and P. Schroder. An image processing approach to surface matching. In *Eurographics Symposium on Geometry Processing 2005*, 2005.

[8] D.G. Lowe. Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision*, 60(2).

[9] T. Gatzke, C. Grimm, M. Garland, and S. Zelinka. Curvature maps for local shape comparison. In *SMI '05: Proceedings of the International Conference on Shape Modeling and Applications 2005*, pages 246–255, Washington, DC, USA, 2005. IEEE Computer Society.

[10] X. Huang, N. Paragios, and D.N. Metaxas. Shape registration in implicit spaces using information theory and free form deformations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(8):1303–1318, 2006.

[11] V.S.M. Joris, S. Spanjaard, I. Horvath, and J.J.O. Jos. Fitting freeform shape patterns to scanned 3d objects. *ASME Journal of Computer and Information Science and Engineering*, 1(3):218–224, 2001.

[12] A. Rangarajan, H. Chui, and F.L. Bookstein. The softassign procrustes matching algorithm. In *IPMI '97: Proceedings of the 15th International Conference on Information Processing in Medical Imaging*, pages 29–42, London, UK, 1997. Springer-Verlag.

[13] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Comput. Vis. Image Underst.*, 89(2-3):114–141, 2003.

[14] Y. Lipman and T. Funkhouser. Möbius voting for surface correspondence. In *SIGGRAPH '09: ACM SIGGRAPH 2009 papers*, pages 1–12, New York, NY, USA, 2009. ACM.

[15] C. Aslan and S. Tari. An axis-based representation for recognition. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision*, pages 1339–1346, Washington, DC, USA, 2005. IEEE Computer Society.

[16] A. Siddiqi, A. Shokoufandeh, S.J. Dickinson, and S.W. Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 35(1):13–32, 1999.

[17] M. Hilaga, Y. Shinagawa, T. Kohmura, and T.L. Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 203–212, New York, NY, USA, 2001. ACM.

[18] S. Wang, Y. Wang, M. Jin, X. Gu, and D. Samaras. 3d surface matching and recognition using conformal geometry. In *CVPR (2)*, pages 2453–2460, 2006.

[19] C. Carner, M. Jin, X. Gu, and H. Qin. Topology-driven surface mappings with robust feature alignment. In *IEEE Visualization*, page 69, 2005.

[20] R. Bidarra and W.F. Bronsvoort. Semantic feature modelling. *Computer-Aided Design*, 32(3):201–225, 2000.

[21] P.J. Nyirenda, R. Bidarra, and W.F. Bronsvoort. A semantic blend feature definition. *Computer-Aided Design and Applications*, 4(6):795–806, 2007.

[22] E. van den Berg and W.F. Bronsvoort. Validity maintenance for freeform feature modeling. *Journal of Computing and Information Science in Engineerin*, 10(1), 2010.

[23] W.F. Bronsvoort, R. Bidarra, H.A. van der Meiden, and T. Tutenel. The increasing role of semantics in object modeling. *Computer-Aided Design and Applications*, 7(3):431–440, 2010.

[24] C. Moenning and N.A. Dodgson. Fast marching farthest point sampling. In *Proceedings of Eurographics 2003*, 2003.

[25] G. Yngve and G. Turk. Robust creation of implicit surfaces from polygonal meshes. *IEEE Transactions on Visualization and Computer Graphics*, 8(4):346–359, 2002.

[26] R. Kimmel and J.A. Sethian. Computing geodesic paths on manifolds. In *Proc. Natl. Acad. Sci. USA*, pages 8431–8435, 1998.

[27] V. Jain and H. Zhang. Robust 3d shape correspondence in the spectral domain. In *SMI '06: Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006*, page 19, Washington, DC, USA, 2006. IEEE Computer Society.

[28] L.S. Shapiro and J.. Brady. Feature-based correspondence: an eigenvector approach. *Image Vision Comput.*, 10(5):283–288, 1992.

[29] D.M. Mount and S. Arya. ANN*: A Library for Approximate Nearest Neighbor Searching*. http://www.cs.umd.edu/ mount/ANN/, 2006.

[30] Y. Wang, C.C.L. Wang, and M.M.F. Yuen. Duplicate-skins for compatible mesh modelling. In *SPM '06: Proceedings of the 2006 ACM symposium on Solid and physical modeling*, pages 207–217, New York, NY, USA, 2006. ACM.

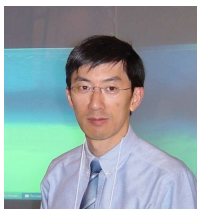
[31] P. Alliez, D. Cohen-Steiner, O. Devillers, B. Lévy, and M. Desbrun. Anisotropic polygonal remeshing. *ACM Trans. Graph.*, 22(3):485–493, 2003.

[32] A. Elad and R. Kimmel. On bending invariant signatures for surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:1285–1295, 2003.

[33] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: measuring error on simplified surfaces. *Computer Graphics Forum*, 17(2):167–174, 1998.

Samuel S.-M. Li is currently an Analyst Programmer of a local software development studio, 12sMelody.com in Hong Kong. He received his B.Eng. in Automation and Computer-Aided Engineering in 2004 and M.Phil in Automation and Computer-Aided Engineering in 2010 from the Chinese University of Hong Kong. His research interests include geometry modeling, semantic feature extraction and correspondences matching in three dimensions.

Charlie C.L. Wang is currently an Associate Professor at the Department of Mechanical and Automation Engineering, the Chinese University of Hong Kong, where he began his academic career in 2003. He gained his B.Eng. (1998) in Mechatronics Engineering from Huazhong University of Science and Technology, M.Phil. (2000) and Ph.D. (2002) in Mechanical Engineering from the Hong Kong University of Science and Technology. He is a member of IEEE and ASME, and Vice-Chair of Technical Committee on Computer-Aided Product and Process Development (CAPPD) of ASME. Dr. Wang has received a few awards including the ASME CIE Young Engineer Award (2009), the CUHK Young Researcher Award (2009), the CUHK Vice-Chancellor's Exemplary Teaching Award (2008), and the Best Paper Awards of ASME CIE Conferences (in 2008 and 2001). His current research interests include geometric modeling in computer-aided design and manufacturing, biomedical engineering and computer graphics, as well as computational physics in virtual reality.



Kin-Chuen Hui received the B.Sc. and Ph.D. degrees in mechanical engineering from the University of Hong Kong, Hong Kong, China, in 1979 and 1990, respectively. Before joining the Chinese University of Hong Kong in 1992, he was a Consultant in the CAD Services Center of the Hong Kong Productivity Council. Currently, he is a Professor of the Automation and Computer-Aided Engineering Department at the Chinese University of Hong Kong. He is on the editorial board of the Journal of Computer-Aided Design. His research interests include computer graphics, geometric and solid modeling, virtual reality, and their applications in design and manufacturing.