A Hierarchical Model for 3D Human Feature Point Extraction

X. Pan, L. Chen, A. Agathos

Abstract—**Feature point extraction is a fundamental problem in human-centered automation. Addressing this research, one challenging problem is how to consistently extract feature points under variant poses. Previous methods use multidimensional scaling to resolve the problem. However, multidimensional scaling will reduce the accuracy of extracted feature points since the transformed human shapes lose important joint features. In this paper, we present a hierarchical model that directly extracts feature points on the original 3D human data by taking full advantage of human knowledge. The proposed algorithm first detects some feature points defined by extremities. For each point, its semantic label is also recognized using human knowledge. Then, these feature points with their semantic labels are used for detecting other feature points. On the other hand, we consider how to improve the accuracy of those feature points by the property of human joint. We use graph cut to capture the position of human joints since the position of human joint is usually concave. In experiments, we verify the proposed algorithms using human data with different poses. The experimental results further prove the algorithm's robustness.**

Index Terms—3D human data, Feature point extraction, Hierarchical model, Semantic recognition, Geodesic distance, Graph cut

I. Introduction

The 3D human data has lots of applications in different fields, including anthropometrical, clothing design, prosthetic design and virtual human animation[1]. For automatic design of human-centered modeling, a key problem is how to extract feature points for establishing correspondence. Manually annotating feature points cannot be practical for automatic design. Therefore, how to automatically extract human feature points has attracted considerable research interests.

However, this research topic is challenging because of the following two reasons. Firstly, the general detecting algorithms cannot accurately locate human feature points like crotch, neck. Since these algorithms assume that feature points must be defined as local extremities. Apparently, not all feature points of human shape have such a characteristic. A second problem is that human data represents different poses, which causes the shape similarity based point correspondence be unstable. Therefore, those methods transferring the feature point from template to the input 3D data cannot obtain expected precision due to the shape difference between template and the input shape.

Our work is motivated by Leong's method[2]. Leong et al try to give a mathematical definition for finding accurate positions of human's feature points. Therefore, it is fully automatic, and no user's interaction is required. However, their method is not stable under pose variation and noise distortion since the definition of human's feature points are defined by profile and range information. Differing with Leong's method, our algorithm directly provides definitions on the 3D human data. We highlight some distinctive features of our algorithm:

1) Accurate definition using extremities and spatial relation of human's feature points. For most points, such as hands, feet, crotch, armpits, we find them using extremity. Therefore, the algorithm is very robust and accurate even under noise distortion. For example, the crotch point must be located on the path from left foot to right foot, and it has the minimum distance to head point.

2) Refined feature points using Graph cut. For those points without definition of extremities, we use graph cut to fit the human's joint for refined feature points. Since the graph cut is capable of finding smooth boundary instead of a point, it is very stable even under noise distortion. Whereas the accuracy of most other methods will decrease due to pose variation. Moreover the pose variation can improve our algorithm's accuracy due to the more obvious cut boundary, like bent arms.

To further prove our algorithm's robustness, we tested human models with different poses and noises. The experimental results showed the method can fit the manual landmarks very well.

The rest of the paper is organized as follows: Section 2 provides a brief review of the related work; Section 3 gives an overview of the proposed algorithm; Section 4 discusses the algorithm in detail; Section 5 performs experimental analysis on the proposed algorithm; Finally, Section 6 gives a conclusion and some recommends for future work.

Manuscript received ***** **, 2012. This work was supported in part by the China National Funding under Grant 60703001.

X. Pan is with the Zhejiang University of Technology, Hangzhou, Zhejiang, 310023 China (86-571-85290024; fax: 86-571-85290668; e-mail: panx@ zjut.edu.cn).

L. Chen is a graduate at Carnegie Mellon University(e-mail: chenli19871016@gmail.com).

A. Agathos is with The Institute of Informatics & Telecommunications of Demokritos(e-mail: alexander.agathos@gmail.com).

II. Related work

There are two research topics related to our work. One is the general feature point extraction methods for any 3D model. The other is specifically used for 3D human data. This section gives a brief overview of the two methods.

A. *General feature point extraction methods*

Feature point extraction is a very interesting research topic both in image and 3D model field due to its wide applications, such as registration, point-to-point correspondences and partial matching. There are lots of algorithms proposed for detecting feature points of images. Recently, many of them have been extended to detect 3D feature points. For example, one of the most famous methods is SIFT(Scale-invariant Feature Transform)[3], which has been extended to 3D model(meshSIFT). meshSIFT has been proven to be very effective in describing 3D local shape features. Similarly, Harris operator has been widely used for image feature point detection[4], and Sipiran et al extended their work and use Harris operator for 3D feature point extraction[5]. Another theory DoG(Difference of Gaussian) has been used for 3D feature point extraction as well. DoG seeks the extremity of the Laplacian of a scale-space representation of any scalar function defined on a discrete manifold[6]. Sun et al used heat kernel in defining shape descriptor and extracting feature points[7]. Mian et al not only used the covariance matrix for selecting stable points, but also provide a quality measurement for ranking the key points[8]. SHREC also held the contest to compare the algorithms' robustness under different transformations[9].

B. *Human domain based feature point extraction methods*

These mentioned methods have a potential application in extracting feature points of 3D human. However, it is not enough since human point is a high level semantic definition. Therefore, many researches improve the above methods by considering human knowledge. Leong et al try to give a mathematical definition by combining human profile and range information[2]. It works well only if the 3D model has a very standard pose since the profile and range information are highly dependent on the 3D human pose. Therefore, researches are now considering how to extract feature points directly on the 3D data to avoid occlusion problem. Template matching is the most typical method of these researches. The method manually annotates feature points on the human template. Then the general registration method is used to build a correspondence between the template and the input model. In this way, the position of feature points can be transferred from the template to input 3D model. Allen et al. use a set of landmarks and a template model to deform a template model to human shapes in similar postures[10]. Anguelov et al. extend Allen et al.'s approach to work for varying postures[11]. Azouz et al. use statistical learning to find reliable correspondences between input model and the template[12]. However, the defined shape features are sensitive to pose variation. Therefore, their method can work well only if the input model has similar pose with template.

To fill this limitation, the most common solution is to use MDS(Multidimensional Scaling) to preprocess the input model. MDS normalize the input model by approximating Euclid distance to geodesic distance over the surface. As a result, MDS can build a pose-invariant representation[13]. MDS has been well applied in detecting human's feature point. For example, Samuel et al iteratively transformed the template to the input model using MDS embedding domain of 3D human models[14]. The similar idea also occurs in Wuhrer et al's work. They used Markov network to learn the spatial relation among the feature points. Then the feature points can be found by maximizing a joint probability over all possible configurations[15]. MDS method can resolve pose variation very well. However, it will cause decreased accuracy since it misses some important features, such as high curvature property of human joint.

Our method is similar to Leong et al's method. Both methods try to give a mathematical definition of human feature points. Compared to Leong et al's method, our method has two advantages. Firstly, it is independent of any pose variation. Secondly, it refines the feature point position using graph cut.

III. Overview the proposed algorithm

In this paper, we consider thirteen feature points of the human body, including hands, elbows, armpits, feet, head, and crotch points. These feature points are usually sufficient to allow a satisfactory correspondences for different human data[14]. The main idea of our algorithm is to detect feature points using hierarchical structure instead of simultaneously detecting all feature points. Those feature points having properties of accurate extremity are extracted firstly. Based on the constraint of these accurate feature points, other feature points can be recursively detected using spatial relation and human knowledge. Figure 1 visualizes the flowchart of the proposed algorithm.

As shown in Figure 1, the whole algorithm mainly consists of four steps. Firstly, it detects exterior feature points using global extremities (Figure 1(a)). Secondly, it recognizes semantic label of each exterior feature point using human knowledge (Figure 1(b)). Here different colors represent different semantic labels. This step performs a very important role in our algorithm since we need to use semantic labels to define spatial relation of feature points. Thirdly, Based on semantic labels, we recursively find another feature points (Figure 1(c)). Finally, it uses graph cut to refine positions of some feature points and judge the left and right arm(Figure 1(d)). Notice the refined position of neck point is much better than that of before refinement.

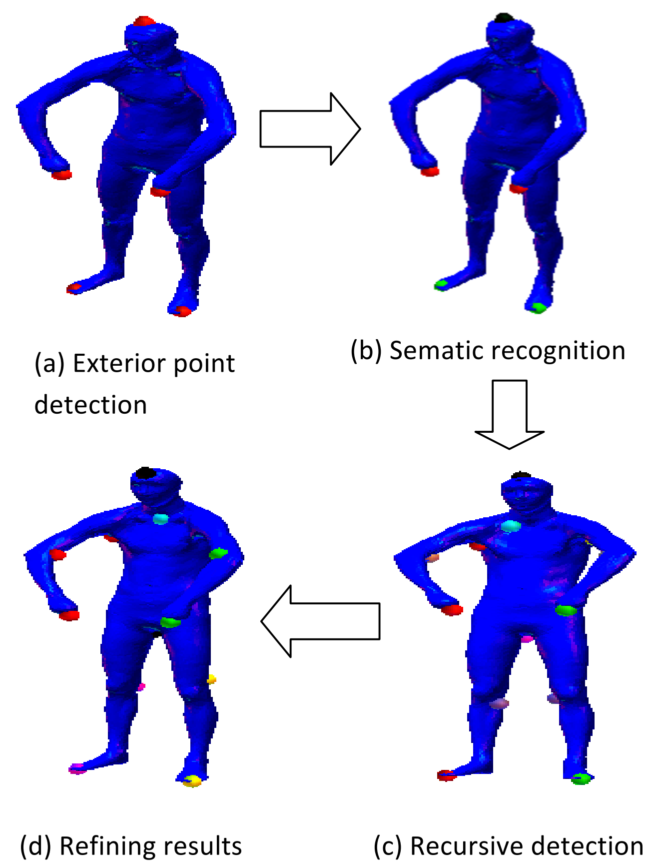

Figure 1 The flowchart of our algorithm

IV. ALGORITHM DETAILS

Based on the above flowchart, this section provides details for each step. For a 3D shape S represented by meshes, it consists of N connected faces $f_i, i = 1,2,3,\cdots i,\cdots N$, denoted by a set TS. For each face, its center is denoted by c_i. For any two faces f_i and f_j, their geodesic distance and geodesic path are denoted by $geod(f_i, f_j)$ and $path(f_i, f_j)$ respectively. The goal of our algorithm is to automatically find some faces representing feature points of 3D human data.

A. Exterior point detection

The whole algorithm begins with finding five exterior points due to their extremities and stability. There are many methods that can be used for finding exterior points. Here, we use geodesic distance based method proposed by Kata et al[16]. The algorithm is particularly good for 3D human shape since this kind of 3D shapes composed of a main part (e.g, the torso of human) and several exterior parts (e.g., the head, arm, and feet).

The algorithm mainly consists of two steps. Firstly, it extracts a main face $f_{main} \in TS$ to represent the torso of human. The main face can be detected using the sum of geodesic distance. For all faces of the 3D shape, the main face has a minimum sum. Therefore, The definition of main face is given by the following equation:

$$f_{main} = \{f_i \mid \min \sum_{j \neq i} geod(f_i, f_j),\ f_j \in TS\} \qquad (1)$$

Secondly, the algorithm recursively detect feature points by maximizing their minimum distance to the previously detected exterior points. The recursive process exits until all five exterior points are detected. The whole process can be concluded in the following table.

Table 1Finding exterior points by recursive process

Step 1:Initialize set $ExtPt = \{f_{main}\}$

Step 2: For any face $f_i \in TS$, compute its distance dis_i to the set $ExtPt$ by the following equation:

$$dis_i = \max_{f_i \in TS} geod(\{f_i \mid \min geod(f_i, f_k),\ f_k \in ExtPt\}$$

Step 3: Update set

$$ExtPt = ExtPt \cup \{f_i \mid dis_i = \max(dis_j),\ f_j \in TS\}\}$$

Step 4: If $|ExtPt| = 6$, exit, otherwise Go to step2

B. Semantic recognition

With the above process, we can obtain five exterior feature points. However, the semantic label of each feature point is unknown. In other words, we do not know which point is denoting the head and which is denoting the foot. If we can get the semantic label of each point, we can employ human domain knowledge to detect other feature points. For example, we can get the crotch point using the path between feet points. Therefore, it is necessary to recognize semantic labels of these feature points by taking advantage of humans' domain knowledge. To simplify the next description, we use a set containing five exterior points.

$$ExtPt = \{kp_1, kp_2, kp_3, kp_4, kp_5\}$$

here kp_i denotes a detected feature point. Therefore, the problem is how to assign a semantic label to each point, and formulate a semantic-aware label set:

$$ExtPt = \{kp_{head}, kp_{hand\_1}, kp_{hand\_2}, kp_{foot\_1}, kp_{foot\_2}\}$$

As for human knowledge, it is natural to use the percentage of human size for semantic recognition. The reason is that the percentage of human size remains very similar regardless of any kind of human. This characteristic has been proved in anthropometric[17]. On the other hand, the human size can be easily approximated using geodesic distance, which is robust under pose variation. In Appendix A, we make an analysis about the human size. It shows why human size can be used for semantic recognition. Based on human size, we find the sum from a certain feature point to another feature points are very distinctive. The head point has the smallest value, and the second one is hand. More details can be found in Appendix A. Therefore, for each feature point kp_i, we calculate its sum $SDis_i$ to other feature points using the following equation:

$$SDis_i = \sum_{j=1\ to\ 5} geod(kp_i, kp_j),\ \ j \neq i \qquad (2)$$

After getting the sum of each feature point, the algorithm first recognizes the head point. Secondly, it classifies hand and feet.

Finally, the relation between hand and feet are determined. The recognizing rules are defined as the follows:

1) Head point

Based on the percentage of human size, the value $SDis_i$ of head point will be the minimum one. In this way, the head point can be judged by the following equation:

$$kp_{head} = \{kp_i \mid \min(SDis_i), i = 1\ to\ 5\} \quad (3)$$

2) Hand and foot point

After recognizing head point, we need to distinguish hand and foot point. Notice the value $SDis_i$ of foot will be larger than that of hand. So we sort the value$SDis_i$ in ascend order. The first two points denote hands and the other two will denote feet.

3) Relation between foot and hand

The Rule 2 only classifies the foot and hand point, but it cannot judge the side relation between foot and hand. That is to say, which foot is on the same side of one specified hand? Obviously, the foot and hand in the same side has the smaller geodesic distance than that not in the same side. So we select one feature point $kp_{hand_1}$and compute its geodesic distance to points $kp_{foot_1}$and kp_{foot_2} , denoted by $geod_1$ and $geod_2$. Then the relation between foot and hand can be determined by the following equation:

$$\begin{cases} kp_{hand\_1}, kp_{foot\_1}, & if \quad geod_2 > geod_1 \\ kp_{hand\_1}, kp_{foot\_2}, & Otherwise \end{cases} \quad (4)$$

C. *Recursive detection*

After semantic recognition, the detection of other feature points can be easily done through a recursive process. The whole process can be further classified into two stages. In the first stage, the crotch and armpit points are detected by extremity analysis. Then, other feature points can be detected by considering results from the first stage.

In the first stage, the detected feature points will be accurate due to their extremities. Taken for an example, we describe the idea that how to detect crotch point. If one path over the surface is defined between two foot's points, the crotch point is definitely lying on this path. In addition, the crotch point must have the minimum distance to the head point. With these two conditions, we can accurately detect the position of crotch point using geodesic distance. Figure 3 further verifies the above property. Similarly, we can detect armpit using geodesic path between hand and foot point.

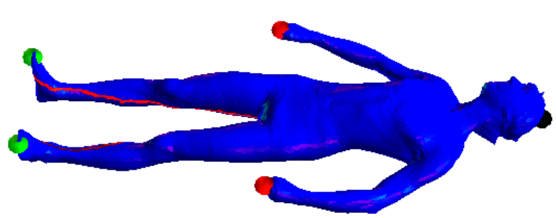

Figure 3 The crotch point has the minimum distance to the head point(The red line denotes the geodesic path between two foot points)

Based on the above analysis, the following rules are used for detecting crotch and armpit point.

1) Crotch kp_{crotch} : The crotch feature point can be determined by the following equation:

$$kp_{crotch} = \{f_i \mid \min geod(f_i, f_{head}), \\ f_i \in path(f_{foot\_1}, f_{foot\_2})\} \quad (5)$$

2) Armpit kp_{armpit_1}: The left armpit point can be judged by the following equation:

$$f_{armpit\_1} = \{f_i \mid \min geod(f_i, f_{head}), \\ f_i \in path(f_{foot\_1}, f_{hand\_1})\} \quad (6)$$

The same for right armpit point kp_{armpit_2}.

The previously detected feature points are rather accurate because they are purely based on the extreme analysis. However, the remained five feature points do not have such a property. Fortunately, the detected feature points have split the human model into different parts. Therefore, the distance's error has not a huge effect on the feature point position. Therefore, we can further get another five points using human percentage.

1) Neck point kp_{neck}: Base on the human percentage, the splitting percentage defined by feature point f_{neck} is nearly 9:17. Therefore, we can get an approximate position denoting neck by the following equation:

$$kp_{neck} = \{f_i \mid \min\left(\frac{geod(f_i, f_{crotch})}{geod(f_i, f_{head})} - 9/17\right), \\ f_i \in path(f_{crotch}, f_{head})\} \quad (7)$$

2) Elbow point kp_{elbow_1}: Based on the percentage of human size, the splitting percentage defined by feature point f_{elbow_1} is nearly 3:2. We can get an approximate position denoting neck by the following equation:

$$kp_{elbow\_1} = \{f_i \mid \min\left(\frac{geod(f_i, f_{hand\_1})}{geod(f_i, f_{armpit\_1})} - 3/2\right), \\ f_i \in path(f_{armpit\_1}, f_{hand\_1})\} \quad (8)$$

3) Knee point kp_{knee_1}: Based on the percentage of human size, the split percentage by feature face f_{knee_1} is nearly 5:7. We can get an approximate feature point denoting knee point by the following equation:

$$kp_{knee\_1} = \{f_i \mid \min\left(\frac{geod(f_i, f_{crotch})}{geod(f_i, f_{foot\_1})} - 5/7\right), \\ f \in path(f_{crotch}, f_{foot\_1})\}$$

(9)

D. *Graph cut based refinement*

In the above recursive detection process, the position of some feature points will not be accurate enough due to the percentage of human's size. Though different humans have similar percentage, they always have some minor variation. In addition, the computation of geodesic distance will induce some errors for detected positions. Therefore, those feature points defined by distance's percentage should be refined by analyzing local shape character. In this subsection, we discuss how to use graph cut to refine positions.

Observing those feature points, like elbow, knee and neck points, we find that they often appear highly concave. Therefore, a good position of feature point can be found using this characteristic. Though geometric feature curvature is good in judging whether the position is concave or not, directly using curvature is not feasible since it is sensitive to noise. Fortunately, concave region near feature points can formulate a close boundary. Therefore, the problem is reduced to finding a cutting boundary, which can be extracted using graph cut. After finding the boundary, each refined feature point can be considered as a point of the boundary that is the closest to another accurate feature point.

There are five points that need to be refined, and they are two elbow points, two knee points and neck point. Notice there is no huge difference in extracting their refined boundaries. Therefore, we only discuss how to get cut boundary for neck point.

1) Performing a coarse binary segmentation using geodesic distance. Get two patches PA and PB. The faces belonging to PA satisfy the following condition:

$$PA = \{f_i | geod(kp_{head}, f_i) < geod(kp_{head}, kp_{neck})\}$$

And other faces are classified into the other patch PB.

2) Constructing fuzzy region FS for searching cut boundary. The fuzzy region contains faces, whose geodesic distance to neck point is smaller than a predefined threshold th.

$$FS = \{f_i | \frac{fabs\left(geod(kp_{head}, f_i) - geod(kp_{head}, kp_{neck})\right)}{geod(kp_{head}, kp_{neck})} < th\}$$

Figure 4 gives the fuzzy region and corresponding mesh structure. Notice it includes a high concave boundary, which can be obtained by graph cut. Here we set th to be 0.3.

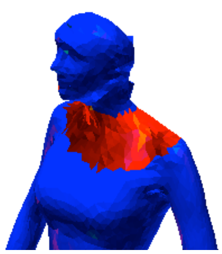

Figure 4 Fuzzy region contains a high concave boundary

3) Defining the source node S and sink node T for solving max flow problem. We have $S \in PA$ and $T \in PB$. With all defined nodes, a dual graph can be constructed for final decomposition. The node S has a connection with any face $f_i \in PA - FS$. And the node T has a connection with face $f_i \in PB - FS$.

4) Defining the edge cost for graph. The edge cost is defined by dihedral angle, which makes the cut highly concave. For two adjacent faces f_i and f_j, the capacity function can be defined by the following equation:

$$cap(i,j) = \begin{cases} \dfrac{1}{1 + \dfrac{\cos t(i,j)}{avg(\cos t)}}, & if\ (i,j) \neq T_1, T_2 \\ \infty & otherwise \end{cases} \tag{10}$$

Where the value $avg(cost)$ is the normalization factor and $cost(f_i, f_j)$ is dihedral angle, defined by the following equation:

$$\cos t(i,j) = \frac{1}{eta * (1 - \cos(f_{i,} f_j))} \tag{11}$$

where factor eta is equal to 1.0 if it is a concave angle, otherwise 0.3.

With the above defined graph and edge cost, we can find a close boundary from fuzzy region. The fine cut is performed by maximum flow algorithm[18]. The algorithm makes the cut along the concave edges and fits the human joint. Figure 5 shows some cutting boundaries. These boundaries can accurately locate the human joint.

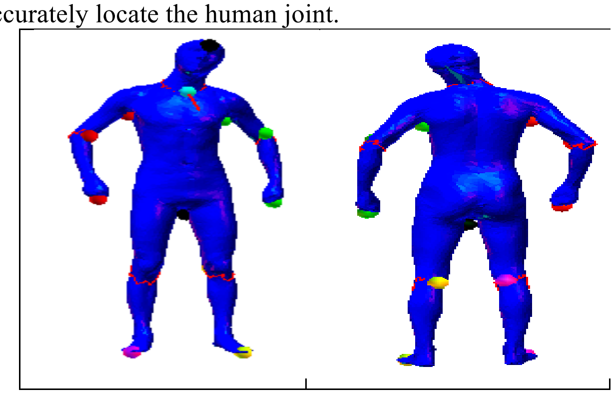

Figure 5 Cutting boundaries of 3D human model

After getting the cut boundary, we begin to discuss how to refine feature points. Here each boundary is a set of faces, denoted by $bou_i, i = 1,2,3$. The symbols bou_1, bou_2, bou_3 are the boundary between head and torso, down and upper arm, upper and lower feet(In total, there are five boundaries. While for two hands, we only define one since refinement is no difference for the other hand. The feet have same case). The position of refined points can be easily defined. They are:

1) Refined neck point: the point lying in the cutting boundary between head and torso is closest to crotch point. Therefore, the refined position can be found in the following equation.

$$kp_{neck}' = \{f_i \mid \min(f_i, f_j), f_j \in bou_1\}$$

2) Refined elbow point: the point lying in the cutting boundary between upper and lower arm is closest to armpit point.

$$kp_{elbow\_1}' = \{f_i \mid \min(f_i, f_j), f_j \in bou_2\}$$

3) Refined knee point: the point lying in the cutting boundary between head and torso is closest to crotch point.

$$kp_{knee\_1}' = \{f_i \mid \min(f_i, f_j), f_j \in bou_3\}$$

4) Refined head point: Notice the best position of head point is in the center of face set belonging to head side. Therefore, the best point position has the following characteristic: the distance to the each face in cutting boundary is almost equal. Based on this characteristic, we can calculate the minimum and maximum distance to cutting boundary. The point which has the highest rate between two distances is selected as the best position.

$$kp_{head}' = \{f_i \mid \max \frac{\min(f_i, f_j)}{\max(f_i, f_j)}, f \in bou_1\}$$

E. Judging left and right arm

Based on the refined feature points, we can further distinguish the left and right arm. Notice that the neck point must be located on the front side of human shape. The reason is that crotch is defined based on the points of the two feet, which have shorter distance to the front side than to the back side. In this way, the closest point to the crotch point must be lying on the front side as well. As a result, we can judge the left and right arm by right-hand rule.

Firstly, we get the cutting boundary between head and torso. Notice the neck point is in the front of boundary. Therefore, in cutting boundary, we find the other point f_{bcak} that is farthest to the point kp_{neck} using the following equation:

$$f_{back} = \{f_i \mid \max\{fabs(c_i, -c_{neck})\}, f_i \in bou_1\}$$

Then we can get the direction of body from *the point* f_{bcak} to kp_{neck}, denoted by $\mathrm{dir}_1(\overrightarrow{CE}$ in Figure 6). Similarly, we can get another direction from point kp_{crotch} to kp_{neck}, denoted by $\mathrm{dir}_2(\overrightarrow{AC}$ in Figure 6). Then for path between two hands, we can compute its direction $\mathbf{dir}_3(\overrightarrow{BD}$ in Figure 6) by right-hand rule:

$$\mathbf{dir_3} = \mathbf{dir_1} \times \mathbf{dir_2}$$

Based on the path's direction $\mathbf{dir}_3$, we can easily judge the left and right hand: the beginning point of $\mathbf{dir}_3$is the right hand, and the ending is left hand. Notice the side relation between hand and foot has been obtained in the process of semantic recognition. Therefore, we can classify the left and right foot as well.

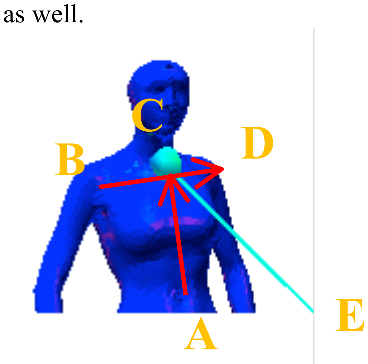

Figure 5 Judging left and right hand using right-hand rule

V. EXPERIMENTS

Experiments are carried out to test the performance of the proposed algorithm. The algorithm is implemented in a PC with Pentium 2.5Ghz CPU and 2048MB RAM. The test human models are from MPI database [1]. MPI database contains human shapes with different poses and sizes. Some shapes also contain scanning noise. Three individual experiments are conducted. Firstly, the algorithm's robustness under multi-resolution and noise is performed. Secondly, the relation between running time and data size prove the high efficiency even for big data size. Thirdly, some fitting results are shown to further verify our algorithm's accuracy.

A. Robustness analysis

Our algorithm is robust under posture variation in nature due to the property of geodesic distance. Besides this, the proposed algorithm is also robust under some geometrical transformations, such as simplification and noisy distortion. Though these geometric transformations may cause the variation of geodesic distance and local geometry, it has little effect on our algorithm due to the following reasons. Firstly, most feature points are detected using extremity definition, which is robust under local variation. Secondly, the refinement strategy is to find a cutting boundary instead of a single point. Therefore, it is still stable even with noise. Figure 6 shows some experimental results under different transformations. Results show our algorithm can robust under transformations.

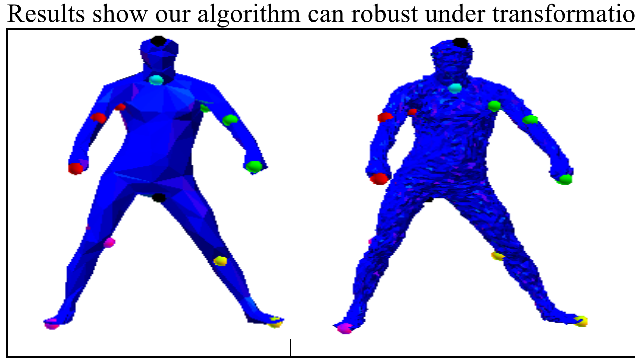

Figure 6 Robustness experiments of our algorithm

B. Computation time

As for feature point detection, the running time is one of the most important performances. Generally, the scanning data often has a large size due to the high scanning quality. Therefore, the algorithm of feature point extraction should be efficient for large size data.

The main cost of our algorithm is the computation of geodesic distance. Notice we compute the main face using the sum of geodesic distance. It has a time complexity O(N×N) , which is highly depending on the size of input 3D shapes. If the 3D shape is very dense, the computation cost will be very high. On the other hand, the center remains similar before and after simplification. Therefore, the computation efficiency can be greatly improved by simplification method. We firstly extract the main face over the simplified mesh. The step can be quickly finished if the number of faces to be simplified is under 1000. Here, we use the algorithm for simplification [20]. Secondly, the position of the main face is transferred to the original mesh. In this way, we only compute the geodesic distances from a few feature points over the original shape. Therefore, the total time complexity of feature point extraction is O(N×6). Figure 7 gives the relation between running time and different sizes. It shows our algorithm can process 7M data in less than one minute.

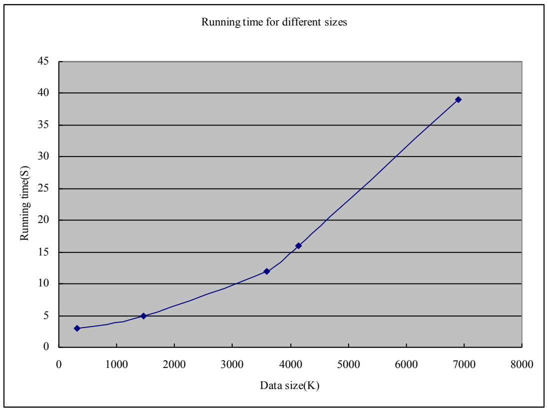

Figure 7 Running time for different size

Finally, Figure 8 gives some feature detection results of MPI database. Though these shapes have different pose, our algorithm can constantly detect feature points.

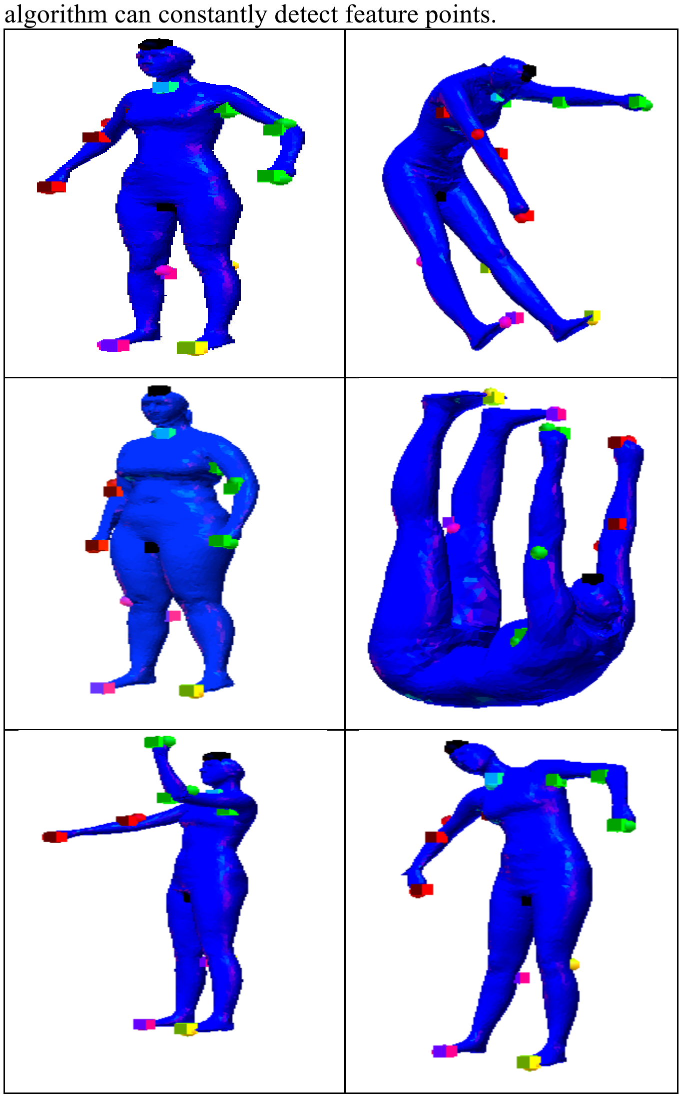

Figure 8 Some results(The detected and annotated points are denoted by sphere and cube respectively)

VI. Conclusion

This paper proposed a hierarchical model in detecting feature points of 3D human data. It aims to give accurate definitions of feature points by taking advantage of human knowledge. Our experimental results show the algorithm is robust under pose-variation, geometrical transformation.

There are still some rooms to improve algorithm accuracy. recent work about cut refinement by Kaplansky et al may be adopted to refine results [19]. On the other hand, our method produce very robust results by only taking advantage of size knowledge, while other human knowledge, like symmetry, gender, can be used for shape understanding. For example, these knowledge can be used in human animation, 3d reconstruction and so on.

Appendix A: Human Size

Figure 1 gives the standard of human size for America, Europe and Mediterranean areas. Here H denotes the human height.

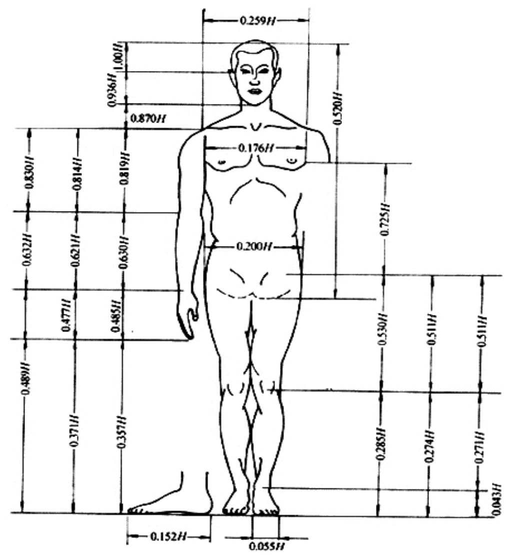

Figure 1: The standard of human size

Here, we list some human size related knowledge for our algorithm. One is to define rules of semantic recognition. The other is to find joint point.

A. *The knowledge for semantic recognition*

In section B, we use human size for recognition. Here, we show why the human size can be used for semantic recognition. Based on the human size, we can calculate the distance between any two feature points. Table 1 shows the results.

Table 1: the approximate distance between two feature points

Approximate distance between two feature points	Computation Equation	Value
The distance from head to hand	(0.819-0.357)+(1-0.819)	0.64H
The distance from head to foot point	1.0+0.152	1.152H
The distance between two hands	2*(0.819-0.357)+0.176	1.1H
The distance between two feet	(1-0.520)*2	0.96H
The distance from	(0.819-0.357)+0.819+0.1	1.452H

hand point to foot point	52	

Based on the values in table 1, we can further get the sum of distance from one feature point to another four feature points. As shown in Equation 1. Obviously, the sum from the head to other points will be smallest, and the foot has the largest one.

$$\begin{cases} \text{Head}: 0.64*2+1.152*2=3.584\text{H} \\ \text{Hand}: 1.1+0.64+1.452*2=4.644\text{H} \\ \text{Foot}: 1.452*2+0.96+1.152\text{H}=5.016\text{H} \end{cases} \quad (1)$$

B. Finding joint point

In section C, we consider detecting joint points by human size. For neck point, its distance to head and crotch is 0.18H and 0.34H respectively. Therefore, we can get the percentage 9:17. Similarly, the percentage for finding elbow and knee point are 2:3 and 5:7.

Acknowledgment

This research work was supported by China Natural Science Foundation (Grant No: 60703001). Thanks are also given to the anonymous reviewers for their kind help.

References

1. **Hasler, N., C. Stoll, and M. Sunkel, *A statistical model of human pose and body shape.* Computer Graphics Forum, 2009. 33(3): p. 262-269.**
2. **Leong, I.F., J.J. Fang, and M.J. Tsai, *Automatic body feature extraction from a marker-less scanned human body.* Computer-Aided Design, 2007. 39: p. 568–582.**
3. **Lowe, D.G., *Distinctive image features from scale-invariant keypoints.* International Journal of Computer vision, 2004. 60(2): p. 91-110.**
4. **SCHMID, C., R. MOHR, and C. BAUCKHAGE, *Evaluation of interest point detectors.* International Journal of Computer Vision, 2000. 37(2): p. 151-172.**
5. **Sipiran, I. and B. Bustos. *A Robust 3D Interest Points Detector Based on Harris Operator*. in *Eurographics Workshop on 3D Object Retrieval* 2010.**
6. **ZAHARESCU, A., E. BOYER, and K. VARANASI. *Surface feature detection and description with applications to mesh matching*. in *IEEE Conference on Computer Vision and Pattern Recognition*. 2009.**
7. **SUN, J., M. OVSJANIKOV, and L. GUIBAS. *A concise and provably informative multi-scale signature based on heat diffusion*. in *Eurographics Symposium on Geometry Processing*. 2009.**
8. **Mian, A., M. Bennamoun, and R. Owens, *On the Repeatability and Quality of Keypoints for Local Feature-based 3D Object Retrieval from Cluttered Scenes.* International Journal of Computer Vision 2009. 89(2): p. 348-361.**
9. **Bronsteiny, A.M., M.M. Bronsteiny, and B. Bustos6. *SHREC 2010: robust feature detection and description benchmark*. in *Eurographics Workshop on 3D Object Retrieval*. 2010.**
10. **Allen, B., B. Curless, and Z. Popovic, *The space of human body shapes: reconstruction and parameterization from range scans. .* ACM Trans on Graphic, 2003. 22(3): p. 587–594.**
11. **Anguelov, D., P. Srinivasan, and D. Koller, *Scape: shape completion and animation of people.* ACM Trans. Graphic, 2005. 24(3): p. 408–416.**
12. **Azouz, Z.B., C. Shu, and A. Mantel, *Automatic locating of anthropometric landmarks on 3d human models.* 3D Data Processing, Visualization and Transmission, 2006.**
13. **Cox, T. and M. Cox., *Multidimensional Scaling.* Chapman & Hall CRC, 2001.**
14. **Samuel S. M. Li, Charlie C.L. Wang, K. C. Hui, Bending-invariant correspondence matching on 3D human bodies for feature point extraction, IEEE Transactions on Automation Science and Engineering, 2011.**
15. **Wuhrer, S., C. Shu, and P. Xi, *Landmark-Free Posture Invariant Human Shape Correspondence.* The Visual Computer, 2011.**
16. **Katz, S. and A. Tal, *Hierarchical Mesh Decomposition using Fuzzy Clustering and Cuts.* ACM Transactions on Graphics, 2003. 22(3): p. 954-961.**
17. **Simmons, K.P. and C.L. Istook, *Body measurement techniques: A comparison of three-dimensional body scanning and physical anthropometric methods for apparel application.* Journal of Fashion Marketing and Management, 2003. 7(3): p. 306-332.**
18. **Goldberg, A. and R. Tarjan, *A new approach to the maximum flow problem.* Journal of the ACM 1988. 35(4): p. 921-940.**
19. **Kaplansky, L. and A. Tal, *Mesh Segmentation Refinement.* Computer Graphics Forum, 2009. 28(7): p. 1995-2003.**