FRANKENSTEIN3D: HUMAN BODY RECONSTRUCTION FROM LIMITED NUMBER OF POINTS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

OĞUZHAN TAŞTAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

DECEMBER 2019

Approval of the thesis:

FRANKENSTEIN3D: HUMAN BODY RECONSTRUCTION FROM LIMITED NUMBER OF POINTS

submitted by **OĞUZHAN TAŞTAN** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil KALIPÇILAR
Dean, Graduate School of **Natural and Applied Sciences** ______________

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** ______________

Assoc. Prof. Dr. Yusuf Sahillioğlu
Supervisor, **Computer Engineering, METU** ______________

Examining Committee Members:

Prof. Dr. Uğur Güdükbay
Computer Engineering, Bilkent University ______________

Assoc. Prof. Dr. Yusuf Sahillioğlu
Computer Engineering, METU ______________

Assoc. Prof. Dr. Ahmet Oğuz Akyüz
Computer Engineering, METU ______________

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Oğuzhan Taştan

Signature :

ABSTRACT

FRANKENSTEIN3D: HUMAN BODY RECONSTRUCTION FROM LIMITED NUMBER OF POINTS

Taştan, Oğuzhan
M.S., Department of Computer Engineering
Supervisor: Assoc. Prof. Dr. Yusuf Sahillioğlu

December 2019, 66 pages

We propose a novel approach for reconstructing high-resolution 3D human body models from extremely small number of 3D points which represent body parts. We leverage a database of high-resolution 3D models of 100 humans varying from each other by physical attributes such as age, weight, size etc. We, first, divide the bodies in database into seven semantic regions. Then, for each input region consisting of maximum 40 points, we search the database for the best matching body part. For the matching criterion, we use the distance between novel point-base features of input points and body parts in the database. We further combine the matched parts from different bodies into one body which result in a high-resolution human body, with the help of Laplacian deformation. To evaluate our results, we pick points from each part of the ground-truth human body models, then reconstruct them using our method and compare the resulting bodies with corresponding ground-truths. Also, our results are compared with ARAP-based results. In addition, we run our algorithm with noisy data to test robustness of our method and run it with input points whose body parts are manually edited, which produces plausible human bodies that do not even exist in our database. Our experiments verify qualitatively and quantitatively that the proposed approach reconstructs human bodies with different physical attributes from small number of points successfully and prove that our method is robust to noisy data.

Keywords: 3D Body Reconstruction, Modelling, Database-Assisted Modelling

ÖZ

FRANKENSTEIN3D: LİMİTLİ SAYIDA NOKTADAN İNSAN VÜCUDU OLUŞTURMA

Taştan, Oğuzhan
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Yusuf Sahillioğlu

Aralık 2019 , 66 sayfa

Bu tezde, yüksek çözünürlüklü 3 boyutlu insan vücudunu, insan vücudunun parçalarını temsil eden son derece az sayıda 3B noktadan yeniden oluşturmak için yeni bir yaklaşım önerdik. Yaş, ağırlık, boyut vb. gibi fiziksel özelliklere göre birbirinden farklı 100 insandan oluşan yüksek çözünürlüklü 3 boyutlu modellerin bulunduğu veri setinden yararlanıyoruz. İlk önce veri tabanındaki modelleri yedi semantik bölgeye (yani baş, sol kol, sağ kol, göğüs, göbek, sol bacak, sağ bacak) bölüyoruz. Sonra, maksimum kırk noktadan oluşan her girdi bölümü için, veritabanında en iyi eşleşen şekli arıyoruz. Eşleştirme kriteri olarak, girdi noktalarının özgün nokta bazlı özellikleri ile veritabanındaki vücut parçaları arasındaki mesafeyi kullanıyoruz. Laplacian deformasyonun yardımı ile, farklı vücutlardan eşleşen parçaları, yüksek çözünürlüklü bir insan vücudu ile sonuçlanan tek bir vücutta birleştiriyoruz. Sonuçlarımızı değerlendirmek için, referans bedenlerin her bir kısmından noktalar aldık, daha sonra yöntemimizi kullanarak onları yeniden yapılandırdık ve sonuçtaki bedeni karşılık gelen referans beden ile karşılaştırdık. Ayrıca, sonuçlarımızı ARAP-tabanlı sonuçlar ile karşılaştırdık. Buna ek olarak, yöntemimizin dayanıklılığını test etmek için, yöntemimizi gürültü eklenmiş girdi noktaları ile denedik ve ayrıca yöntemimizi, veritabanımızda bulunmayan makul insan vücutları üreten, parçaları elle düzenlenmiş girdi noktaları ile çalıştırdık. Deneylerimiz, önerilen yaklaşımın, az sayıdaki noktadan farklı fiziksel özelliklere sahip insan bedenlerini nicel ve nitel olarak başarılı bir şekilde oluşturabildiğini ve gürültülü girdilere karşı dayanıklı olduğunu göstermektedir.

Anahtar Kelimeler: 3B Vücut Oluşturma, Modelleme, Veritabanı Yardımlı Modelleme

To my parents, my wife and my beloved ones.

ACKNOWLEDGMENTS

I would first like to thank my thesis advisor Assoc. Prof. Dr. Yusuf Sahillioğlu for the continuous and invaluable support for my M.Sc. study and research throughout my graduate education. In addition to his technical support, he encourages me all the time to complete this study. I have taken the advantage of his immense knowledge, motivation and enthusiasm.

My sincere thanks also goes to my great colleagues Cemal Aker, Oğulcan Eryüksel, Mehmet Ali Deligöz for their psychological support before and during my M.Sc. studies.

Last but not least, for their encouragement and support, I would like to thank my parent, parents-in-law and my lovely wife, Ece Nur Taştan, who always stands back of me and provides full support with all her soul and heart.

This work was supported by TUBITAK under the project EEEAG-215E255.

TABLE OF CONTENTS

CHAPTERS

APPENDICES

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

LIST OF ABBREVIATIONS

2D	2 Dimensional
3D	3 Dimensional
RGB-D	Red Green Blue-Depth
CNN	Convolutional Neural Network
DHF	Distance Histogram Feature
AHF	Angle Histogram Feature
ARAP	As-rigid-as-possible
ICP	Iterative Closest Point
DSLR	Digital Single-Lens Reflex

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

Modelling human body digitally is one of the most attractive field in computer graphics and these models have wide-ranging applications. In clothing, size and shape of the body are important to improve cloth comfort and fit [1]. In medicine, the human model data can be used for arranging the doses of medication [2] and monitoring the posture of human body [3]. There exist some researches which aims to create dynamic digital models which mimic physiological and behavioral realism [4], [5].

This thesis considers the problem of generating high-detail 3D human body from extremely small number of points. Generating human body is already a difficult task because human body can be in enormous number of different sizes and shapes. Moreover, our task is even more challenging due to the lack of surface features such as meshes and normal vectors in the input. Another cause of being challenged is the absence of the correspondence between sparse points and high-resolution models.

In real life, the sparse points can be obtained from 2D images [6] or array of sensors [7]. [6] proposes a neural network which learns a non-linear mapping from an image to a set 3D points which represent a human face. [8] proposes a method which learns salient points on the fingers and pose of hand. Their method can be extended to learn salient points on human body which can be input to our method. [7] uses wearable sensors to capture sparse 3D point sets from the human parts. [9] makes use of a small set of inertial sensors attached to the body. They deforms a base human body based on the measurements from the sensors to capture the motion of human. Although they place six sensors to the wrists, lower legs, back and head, increasing the number of sensor can produce 3D points on human body which can be input to our method.

1.2 Contributions and Novelties

The main contributions of the thesis are as follows:

- **Novel Features For 3D Points :** In this thesis, we propose a set of features which represents the geometry of a set of 3D points. Our experiments show that the proposed features can effectively distinguish between 3D point sets in different geometry and also recognize 3D point sets in similar geometry.
- **Novel Method for Reconstructing Human Body :** A novel method is proposed to reconstruct human body from small number of 3D points by following a data-driven approach which exploits a database of 3D human body models.

1.3 The Outline of the Thesis

In Chapter 2, the related literature is reviewed by focusing on the studies which aim to reconstruct human bodies and on specifically the ones which take sparse inputs. In Chapter 3, the database used, our proposed feature sets and the method for reconstructing 3D human body are explained. In Chapter 4, the experiments conducted to show how successful the proposed method reconstructs 3D human bodies are demonstrated. In Chapter 5, the drawback and limitations of our method are explained. In Chapter 6, conclusion and future work are discussed. Finally, all models used in this study are presented in Appendix A.

CHAPTER 2

RELATED WORK

In this chapter, the studies which aim to reconstruct human bodies are reviewed. Since there is no directly related work when the input is considered, we focused on the studies which aim to reconstruct human bodies and faces. Due to the the growing demand for virtual human models, there is a huge body of literature on reconstructing human bodies.

The paper which is most similar to ours in the literature handles a dense point cloud as input to the method [10]. As we do, they leverage a database of high-resolution 3D meshes. Their method semantically divides the meshes into several parts in order to compare the input with the meshes in the database. Differently from ours, their method reconstructs a face rather than a full human body. They first align input points to database by detecting predefined facial landmarks and using them for rigid pose alignment via Procrustes analysis [11] and then dense alignment to a generic mesh [12]. After alignment process, each face part is matched to the database using distance function which is a weighted average of pseudo-landmarks and histograms of azimuth and elevation components of the surface normals [13, 14]. In contrast to our extremely sparse input points, they uses dense point clouds;therefore, they are able to make use of surface-related features such as normals and pseudo-landmarks.

[15] processes several depth frames;which are dense point clouds, by detecting and cropping the face in each depth frame. Then, these extracted 3D faces are aligned with each other by using Coherent Point Drift algorithm [16] in order to build a cumulated face. Then, the lowess non-parametric regression method is used to approximate the face surface from the cumulated face model and remove outliers from the data. The resulting 3D face is still noisy, so as to get rid of this noise, first they model the face

as a 2D manifold embedded in the 3D space. Then,they treat the problem as manifold reconstruction from noisy data and solved this problem by applying a method which extends the one described in [17] which is based on a combination of dimensionality reduction and local weighted regression.

In order to reconstruct a full body, [18] built a full-body scanner and a separate face scanner, consisting of 40 and 8 DSLR cameras, respectively. This setup creates a huge body scan data with 4M points. After they scan a full body, they try to align and fit to a template model by performing the following steps:In initialization step, they optimize scaling, rotation and translation by using manually selected landmarks using [19] and optimize the joint angles using inverse kinematics based on linear blend skinning [20]. After good initialization, fine-scale non-rigid registration is applied by minimizing an energy which a weighted average of landmark term, fitting term and regularization term. The landmark term is to minimize the squared distance between the manually selected landmarks, the fitting term penalizes the squared distance between corresponding points and the regularization term penalizes the geometric distortion from the undeformed model ;which is the result of the initialization phase, to the deformed state, measured by the squared deviation of the per-edge Laplacians.

[21] trains a convolutional neural networks with RGB-D face data in order to reconstruct faces. Their network has coarse-to-fine architecture in which a medium scale CNN to regress a medium scale face model followed by a fine scale CNN to recover the surface details. Both CNN models are trained in a completely self-supervised manner in that the face shape and details are automatically learned from large scale unlabelled RGB-D data.

[22] exploits a database of 3D heads of 200 adults which are in a cylindrical representation produced by a commercial laser scanner. First, they put all the head models in the database in full point-to-point correspondence. Then, they create a morphable model from the head models. Finally, using the morphable model, they create a 3D head from 2D input image. To do so, a set of rendering parameters and coefficients of the 3D model are optimized until an image as close as possible to the input image are produced. Their algorithm generates a 3D face from the current parameters, and renders an image, and updates the parameters according to the residual difference.

They started with the average head and with rendering parameters roughly estimated by the user.

As we do, [23] follows a data-driven method which reconstructs articulated 3D motion. Their method extracts the all shape information from a set of range scans which is formed by 200K points and 50K triangles. They put the range scans in full correspondences bu using markers obtained by Correlated Correspondence [24] which computes the consistent embedding of each instance mesh into the template mesh, which minimizes deformation, and matches similar-looking surface regions. Then, their model tries to align the template with each mesh in the data set consisting of different poses of a human. The deformations are modeled for each triangle of the template such that the deformations are applied in local coordinate of each triangle by translating a point of the triangle to the origin. A specific transformation matrix is applied to each triangle and then a rotation is applied which is the rotation of the triangle's articulated part. They try to minimize least square error by optimizing transformation and rotation matrices.

[25] also uses a database of pre-existing rigs and extracting usable body parts from it in order to construct rigged bodies from given unrigged target meshes. To do so, they fit together pieces from several different source meshes in the database. Differently from ours, their input is a dense mesh.

Point-base registration methods are related to our work as they try to reconstruct a new shape from the existing one by deforming it. As oppose to our method, point-base registration methods handles inputs which are as dense as the base model instead of sparser inputs. [26] first embeds the shape into isometric representation. Then, they optimize this embedding as a variant of the classical intrinsic distortion [27] objective by using Markov Random Field optimization. [28] tries to align and label two point clouds in 3D in a simultaneous way until they are as similar as possible and called this procedure as Procrustes matching. Procrustes matching is defined as to find an orthogonal transformation which will be applied to input point set and a permutation which will be applied to the target point set in order to minimize the distances between resulting point sets. They propose a novel semi-definite programming relaxation solution to the problem.

CHAPTER 3

METHOD

In order to build a human body from limited number of 3D points, we make use of a database of human bodies. As a preprocess, we compute a set of features for each body part of each body in the database from the manually selected feature points and save them. Given a set of 3D points for each body part, we compute a set of features and find a best-matching part from the parts in the database by comparing the calculated features of the input points and the saved features of the body parts in the database. Then, the selected body parts from different bodies are combined together to build a human body. That is why we call our method as Frankenstein. The overview of our method shown in Figure 3.1.

3.1 The Database

In this study, the database provided by [23] has been used in order to reconstruct human body parts from small number of 3D input points. The human bodies in the database are in full correspondence with each other. Since the database is very large, only 100 of human body models have been used. Five sample from our database are shown in Figure 3.3 and the all models are shown in Appendix A.

3.2 Preprocess of the Database

We have semantically divided each body into seven segments shown in Figure 3.4. The segments are represented by at most 40 ordered 3D points which are manually selected via a base model in the database, shown in Figure 3.5; and transferred into the

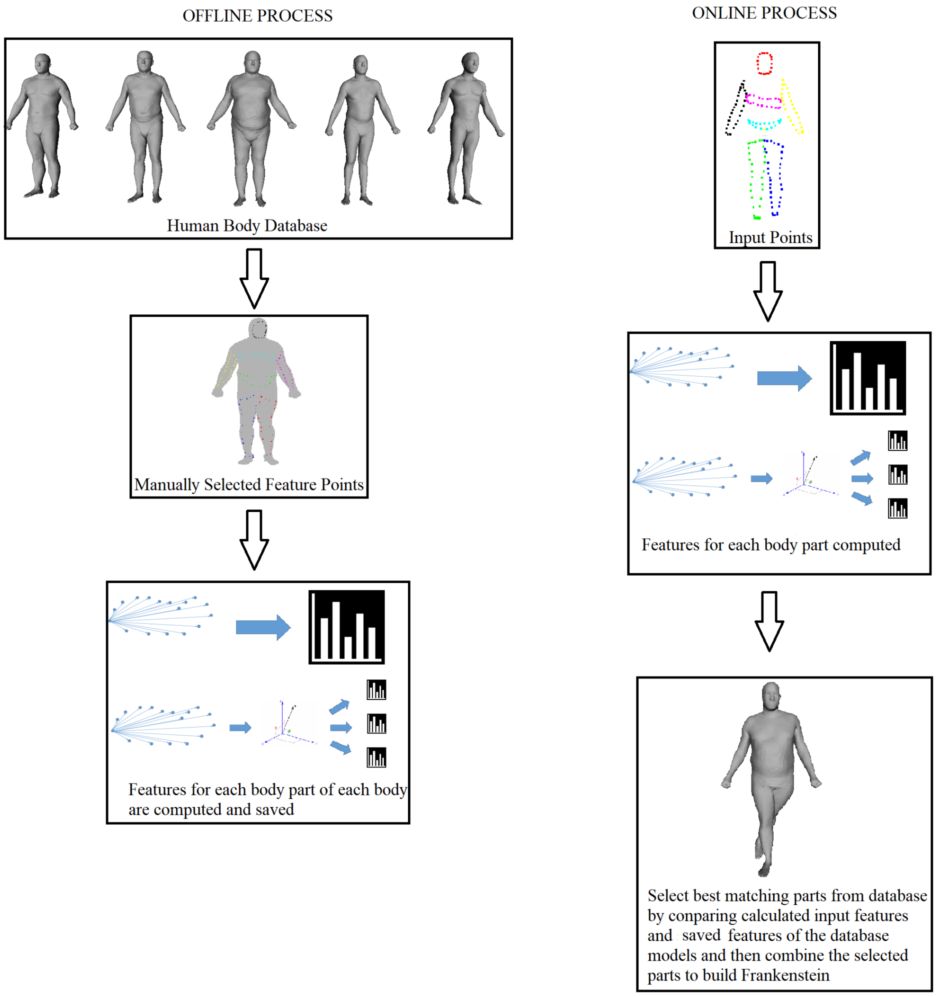


Figure 3.1: Overview of Our Method

others by using full correspondence between models. After segments are ready, we have calculated a feature vector for each segment. Our feature vector are computed from 40 3D points, so we first complete the number of segment border points to 40 with linear fashion by adding a point at the middle of the furthest consecutive points until it reaches to 40 points. Alternatively, we first fit a non-linear B-spline curve to the input points and produce 40 uniform parametric points from the fitted curve. The results of both point completion methods are shown in Fig 3.2. As we can see, the non-linear method produces smoother and more uniform points than the linear method does.

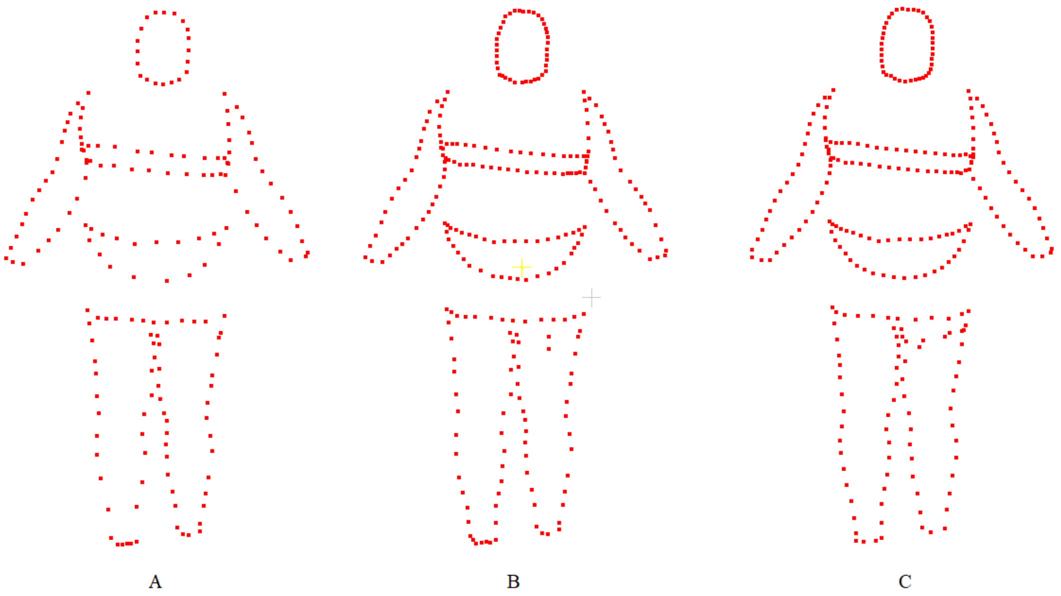


Figure 3.2: Comparison of point completion methods. A: Input points, B: Completed points by linear method, C: Completed points by non-linear method.

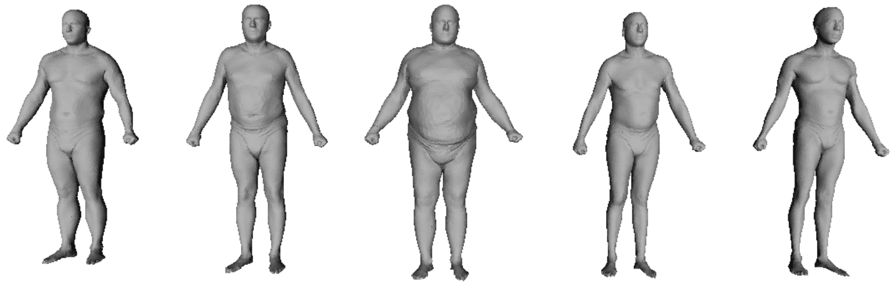


Figure 3.3: Five sample from our database

3.3 3D Features

After completing the contour points, a histogram of 3D distances between each point to the other points in the contour, called distance histogram features (DHF), is calculated. The contour points are normalized so that they have zero-mean and unit standard deviation. In addition, we calculate a vector for each point to the others, calculate the angles between that vectors and x,y,z axis. Then, the histogram of that angles, called angle histogram feature (AHF), for each axis is calculated. Both fea-

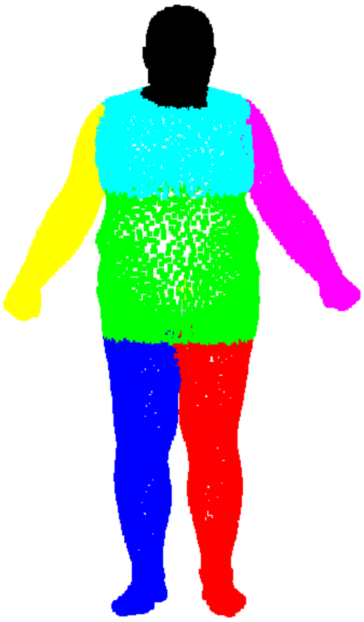


Figure 3.4: Human Body Segments

tures are shown in Figure 3.6. Finally, we combine the histograms of each contour to obtain the feature vector of a human body. Each histogram consists of 20 bins therefore the feature vector for each body part has a length of 80.

3.4 Body Part Selection

For each body part, we search for the best match in the database by comparing the feature vector of querying body part and that of each of corresponding body part in database. The body part in the database with minimum feature distance to the querying feature vector is selected as best match. In order to calculate distances between feature maps, we have used L2 distance.

3.5 Merging Selected Body Parts

Since each body part possibly is taken from different human bodies, some weird appearances come to existence when the selected parts are merged together. In order to get rid of this weird results and correct the rotation and alignment of the body parts,

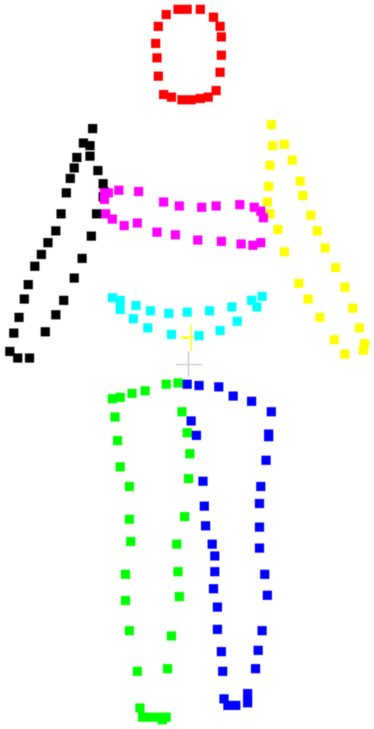


Figure 3.5: Feature points for each body part

we follow a four-step procedure which are described in the subsections. The effect of each step can be seen Figure 3.8.

3.5.1 Moving Base Points

We first select one of the models from our database and call it the base model. We will essentially deform this base model in order to get our smooth output. To this end, we first move the vertices of the base model to the corresponding positions of the parts that we selected in Section 3.4. Recall that all of the models in our database are in fixed connectivity, revealing the correspondences immediately.

As mentioned before, since the body parts belong to different bodies there exist some mismatches. In addition, since the range of rotations of the body parts in the database is limited, the selected parts do not fully match with the input points in terms of orientation. As a solution to this problems, we continue with the next steps.

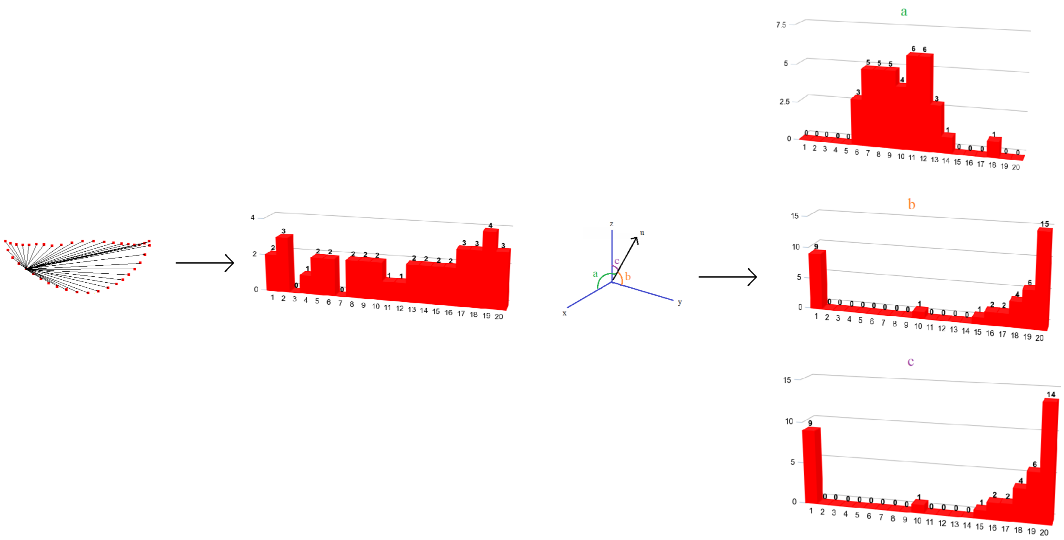


Figure 3.6: Illustration of DHF for one seed point is shown on the second column and AHF on the fourth column. The same point set of size 40 is used (first column) for both feature computations. $\mathbf{u}$ represents the vector between any two points in the point set. DHF is an histogram of $||\mathbf{u}||$ and AHF is for the angles between $\mathbf{u}$ and x-, y-, z-axis namely, a, b, c, respectively.

3.5.2 Rigid Body Part Alignment

In order to get the selected part ($\mathbf{S}$) and the input points for that part ($\mathbf{I}$) in the same orientation, we calculate and apply a transformation matrix which transform the selected part into input points. This provides individual alignment of each body parts separate from each other. To do so, we follow the steps described in [19], first both point sets are moved to origin ($\mathbf{S}^{\text{‘}}$ and $\mathbf{I}^{\text{‘}}$) and then, calculate the co-variance matrix as follows:

$$\mathbf{C} = \mathbf{S}^{\text{‘}}(\mathbf{I}^{\text{‘}})^{T} \tag{31}$$

Using this co-variance matrix, the following matrix is constructed

$$\begin{vmatrix} C_{00}+C_{11}+C_{22} & C_{12}-C_{21} & C_{20}-C_{02} & C_{01}-C_{10} \\ C_{12}-C_{21} & C_{00}-C_{11}-C_{22} & C_{01}+C_{10} & C_{20}+C_{02} \\ C_{20}-C_{02} & C_{01}+C_{10} & -C_{00}+C_{11}-C_{22} & C_{12}+C_{21} \\ C_{01}-C_{10} & C_{20}+C_{02} & C_{12}+C_{21} & -C_{00}-C_{11}+C_{22} \end{vmatrix} \tag{32}$$

As [19] suggested, in order to find the best orientation, we calculate eigen-vectors of the matrix. Using these vectors, now we can construct the rigid alignment matrix.

3.5.3 Inter-parts Alignment

To create a consistent human body, the body parts taken from different bodies should be aligned. To do so, we fix the position of the chest part, light blue colored in Figure 3.4, and add a translation to the head, arms and belly parts; black, yellow, pink and green colored, respectively, so that the borders of those parts are matched. Then, the same operation is followed for legs; blue and red colored, to match them with belly part.

In order to calculate the translation between two part, we first find neighboring edges i.e. edges which has one vertex at one part and other vertex at the other part. Then, an average displacement between the vertices of the neighboring edges is calculated. This displacement is the translation to be needed to align the borders of the body parts from different bodies.

After this operation, the average points of the borders matched; however, the job has not done yet because the parts can have different sizes; e.g. width of the borders, rough and uneven surfaces appears at the borders of parts. To avoid this, we apply a border smoothing algorithm.

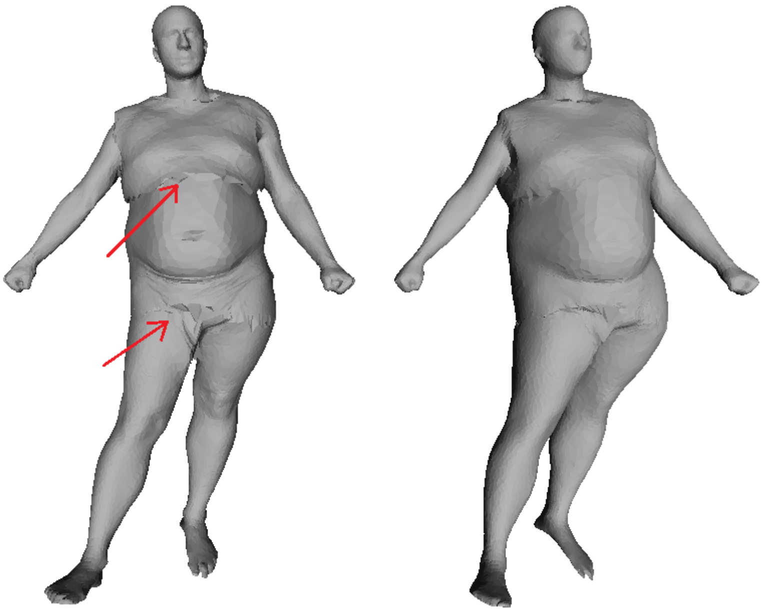


Figure 3.7: (Left) Output of the Inter-parts Alignment step (Section 3.5.3) after Laplacian smoothing does not yield a perfectly smooth mesh (problems pointed by arrows). (Right) Smoother and more natural result is obtained when the output of the Inter-parts Alignment goes directly to our Border Smoothing step (Section 3.5.4). This is our final resulting model.

3.5.4 Border Smoothing

Since the selected parts come from different human bodies, there exist "surgical scars" between neighboring body parts. In order to get rid of these abnormalities, we first simply try Laplacian smoothing [29]. However, the defects around junction points are large enough such that they cannot be cleared by simple smoothing algorithm shown in Figure 3.7. The small defects between chest and belly are cleared by the algorithm, however, the remaining larger defects still exist.

In order to remove the defects more robustly, we solved the Laplacian equation which moves the points of the base body to the selected part points smoothly. This equation is the deformation making use of differential coordinates by Laplacian of mesh,which encodes each vertex relative to its neighbors, defined in [30]. For construction of Laplacian of mesh, the cotangent weights [31] are used. Since a differential coordinate is a linear combination of a vertex and its neighbors, the process of constructing differential coordinates for all vertices is as follows:

$$\mathbf{Lv} = \delta \tag{33}$$

where $\mathbf{L}$ is *n*-by-*n* Laplacian matrix, $\mathbf{v}$ is *n*-by-3 matrix in which each row contains a vertex and δ is *n*-by-3 matrix storing differential coordinates.

The regularization term of the deformation energy is the least-squares difference between differential coordinates of the rest-pose of the base model mesh ($\mathbf{v_0}$) and the deformed pose of the base model mesh ($\mathbf{v}$):

$$E_{\text{reg}} = \| \mathbf{Lv_0} - \mathbf{Lv} \|^2 \tag{34}$$

On the other hand, the match term is the sum of the squares of the distance between the corresponding points from the deformed pose mesh and the fixed handles $\mathbf{h_k}$ designated by the selected parts from Section 3.4:

$$E_{\text{match}} = \sum_{k=1}^{N} \| \mathbf{v_k} - \mathbf{h_k} \|^2 \tag{35}$$

where N is the number of handle points, i.e., 40 from each of the 7 parts, hence 280 for our experiments.

In our case, weighted regularization term and matching term are used to construct total deformation energy (Eq. 36), then a set of vertices is searched for in order to minimize the total deformation energy.

$$E_{\text{tot}} = \alpha E_{\text{reg}} + E_{\text{match}} \tag{36}$$

To minimize E_{tot}, we first take the derivative of it with respect to $\mathbf{v}$ and then equate it to zero. The resulting sparse linear system to be solved instantly for $\mathbf{v}$ is as follows:

$$(\alpha \mathbf{L}^T\mathbf{L} + \mathbf{I})\mathbf{v} = \alpha \mathbf{L}^T\mathbf{Lv_0} + \mathbf{h} \tag{37}$$

where α controls the weight of the regularization energy.

The results of the improvements made in each step can be shown in Figure 4.3. The largest improvement has been made by rigid body part alignment as each part which is independent from each other is put together to build a human body. After inter-part alignment, the error is decreased by important amount. Finally, border smoothing reduces the error slightly but improves the appearance of the body.

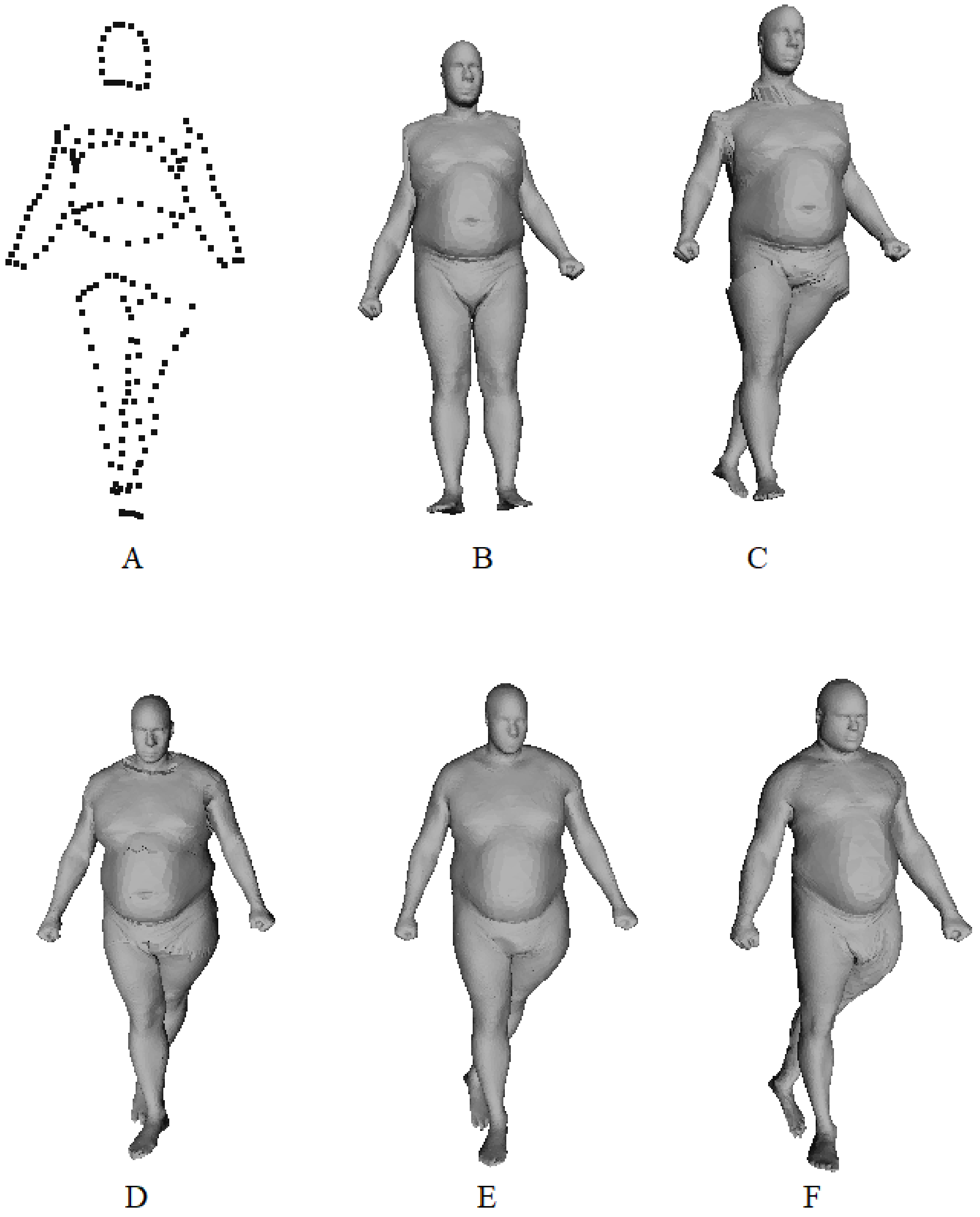


Figure 3.8: Sample Result. A: Input points, B: Result after moving base points (3.5.1), C: Result after Rigid Body Part Alignment (3.5.2), D: Result after Inter-parts Alignment (3.5.3), E: Result after Border Smoothing (3.5.4), F: Ground-truth result

CHAPTER 4

EXPERIMENTS AND RESULTS

The proposed approach has been evaluated with input body points which are in different poses and sizes. In order to make the evaluation easy and accurate, the input points are selected from the ground truth body meshes. However, the real-world applications can be possible such as obtaining the input points directly from 2D images and feed them to our method to produce 3D model of 2D image. To do so, there exists various studies in the literature such as [6]. Although [6] works with faces, it can be possible to extend their work so that the method produces 3D body points by training their deep learning model with 3D body models. In addition, [32] design a cascaded coupled-regressor approach by integrating a 3D point distribution model in order to estimate the 3D landmarks. [7] also estimates 3D landmark on human bodies by using sensor arrays which are combined with deep learning method which maps measured capacitance from the sensors to the deformed geometry.

Inputs points, ground truth result and outputs of each steps, explained in previous section, are shown in Figure 3.8 and Figure 4.1 for two different-shaped inputs. The final result of our algorithm has been compared with the results of ARAP-based algorithm introduced in [33]. The error metric for comparison is the average Euclidean distance between the output points and the corresponding ground truth points divided by average edge length of the ground truth mesh , formulated as follows:

$$\varepsilon = \frac{\sum_{i=1}^{n} \|\mathbf{I}_{\mathrm{i}} - \mathbf{G}_{\mathrm{i}}\|}{n\tau} \tag{41}$$

where n is the number of points in meshes, τ is the average edge length of the ground-truth mesh, $\mathbf{I}_{\mathrm{i}}$ is ith point of the output mesh and $\mathbf{G}_{\mathrm{i}}$ is ith point of the ground-truth mesh.

The constrained points are initialized with the ground-truth matches of the input points for ARAP method. Sample input points, the base body for ARAP method and ARAP-based result of the input points are shown in Figure 4.2. Some comparisons have been shown in the following figures. In the figures, the order of the bodies from left to right is as follows; input points, ARAP-based result, our Laplacian-based result, the ground-truth result. In addition, each body mesh in each figure is colored by the error with respect to the the ground truth. The error for each point of the mesh is the distance to the closest point in the ground truth mesh after the output mesh is rigidly aligned to ground truth mesh using Iterative Closest point (ICP) algorithm. The average error for each result produced by base method and two different variant of the proposed method are compared at Table 4.1. One of the variants interpolates the input points by linear method and the other does this by spline interpolation. The results show that the proposed method with spline interpolation performs better than ARAP-based method. In addition, non-linear point completion slightly improves the reconstruction accuracy compared to the linear point completion method. All errors in the tables and the figures are calculated using Equation 41 which is the relative error with respect to average edge length in order to make the errors understandable. The execution times for each input are shown in Table 4.2.

As shown in Figure 4.4, our method reconstructs the front part of the body better than ARAP method does although both methods performs similar for the back part of the body. The difference between methods becomes more clear in Figure 4.5 and Figure 4.6 as our method produces more accurate results for most of the body parts. Despite the deformation is higher than the other samples in Figure 4.7, both method reconstructs similar outputs. This means both methods are affected by the deformation in the same way.

The number of points for each body part is also experimented in this study. The input points are completed up to 10,20,30,40,50 and 100 points by starting different numbers of points and the resulting body are generated from that points. The starting points are the different percentages (namely 30%, 40%, 50%, 60%, 70%) of the target number of points. The comparison of the different number of completed points are

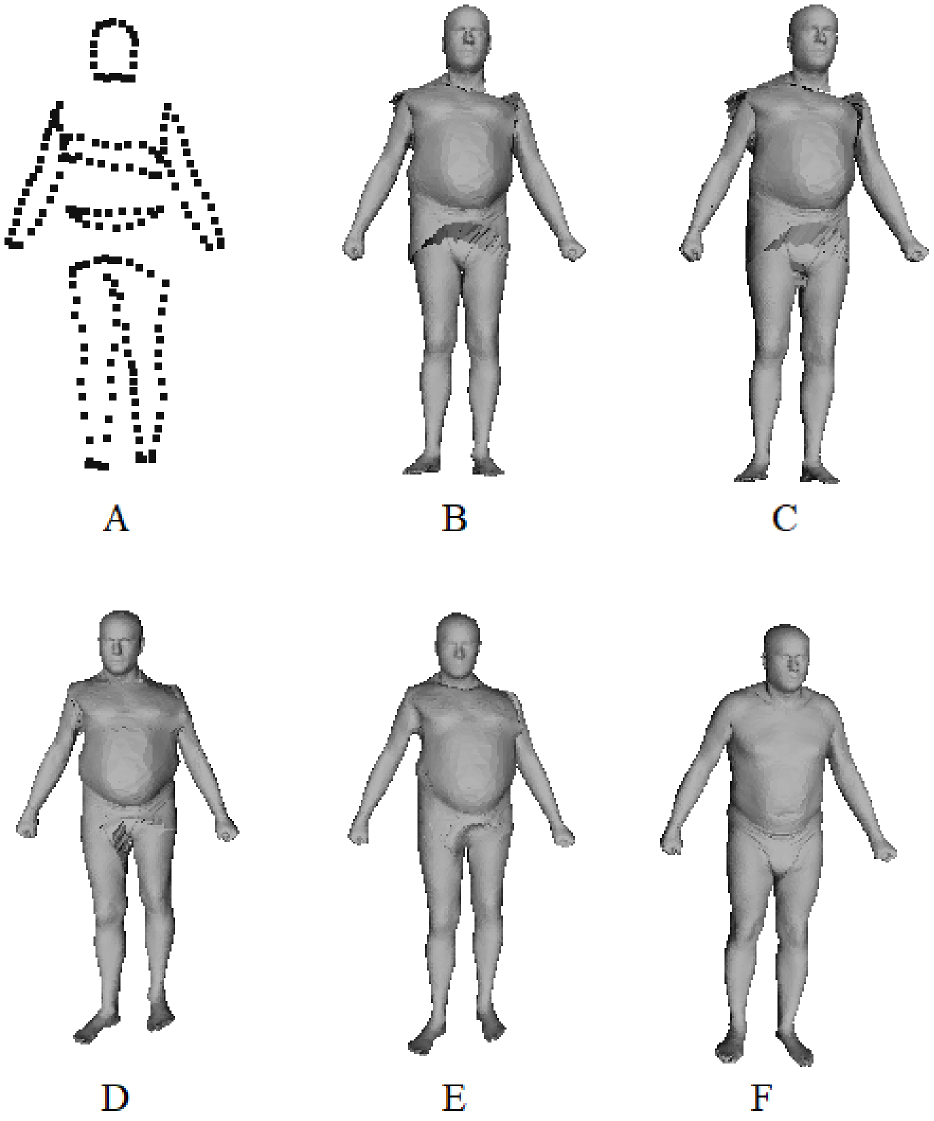


Figure 4.1: Sample Result. A: Input points, B: Result after moving base points (3.5.1), C: Result after Rigid Body Part Alignment (3.5.2), D: Result after Inter-parts Alignment (3.5.3), E: Result after Border Smoothing (3.5.4), F: Ground-truth result

shown in Table 4.3. Most of the cases, the error decreases as the number of points increases up to 40 points. After 40 points, the error almost remains same although the number of points increases. Furthermore, after 50%, increasing the starting points percentage does not change the error significantly. Considering the performance issues, we select 40 points as the most efficient number of input points. We also come

Table 4.1: Average Error by Methods

Figure No	Our Method with Linear Interpolation	Our Method with Spline Interpolation	ARAP-Based Method
4.4	1.146	**1.125**	1.736
4.5	1.146	**1.103**	1.23
4.6	1.170	**1.121**	1.70
4.7	**1.359**	1.333	1.154
4.8	1.759	**1.617**	2.284
4.9	1.552	**1.429**	1.853
4.10	1.178	**0.870**	0.973
4.11	0.899	**0.812**	0.933
4.12	1.06	1.02	**1.01**
4.13	1.528	1.403	**0.98**
4.14	1.098	**1.045**	1.183
4.15	1.275	**1.254**	1.303
4.16	1.047	1.067	**1.033**
4.17	1.070	**0.971**	1.007
4.18	2.106	**1.876**	1.981
4.19	1.773	1.738	**1.610**
4.20	1.775	**1.743**	1.96
4.21	1.342	**0.98**	1.00
4.22	0.974	**0.955**	0.961
4.23	1.183	**1.176**	1.281
Average	1.322	**1.232**	1.359

to conclusion that starting from merely 20 user input points (50% of 40) is sufficient.

We also tried our method with noisy data. To do so, each input point is moved in random direction with random amount of distance which can be at most the maximum edge length. Sample outputs with noisy input are shown in Figures 4.24, 4.25, 4.26. As shown in the figures, our method is robust to noise in the input as it produces body models which are very similar the ones produced from noiseless input points. As the

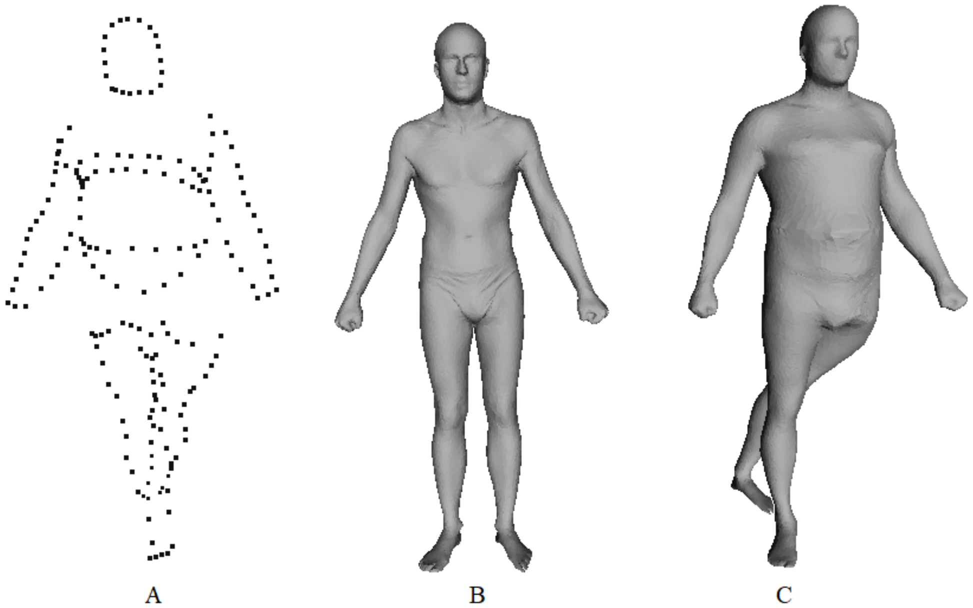


Figure 4.2: Sample result of ARAP-based method. A: Input points, B: Base model for ARAP method, C: ARAP-based result with ground-truth initialization

noise level increases, the average error of the output increases.

Our method can generate body models do not exist in our database by processing an input whose some parts are edited manually while other parts remain same. As shown in Figure 4.27, points on belly and chest parts are moved outward from its center so as to create a human body which has wider belly and chest then the original one has.

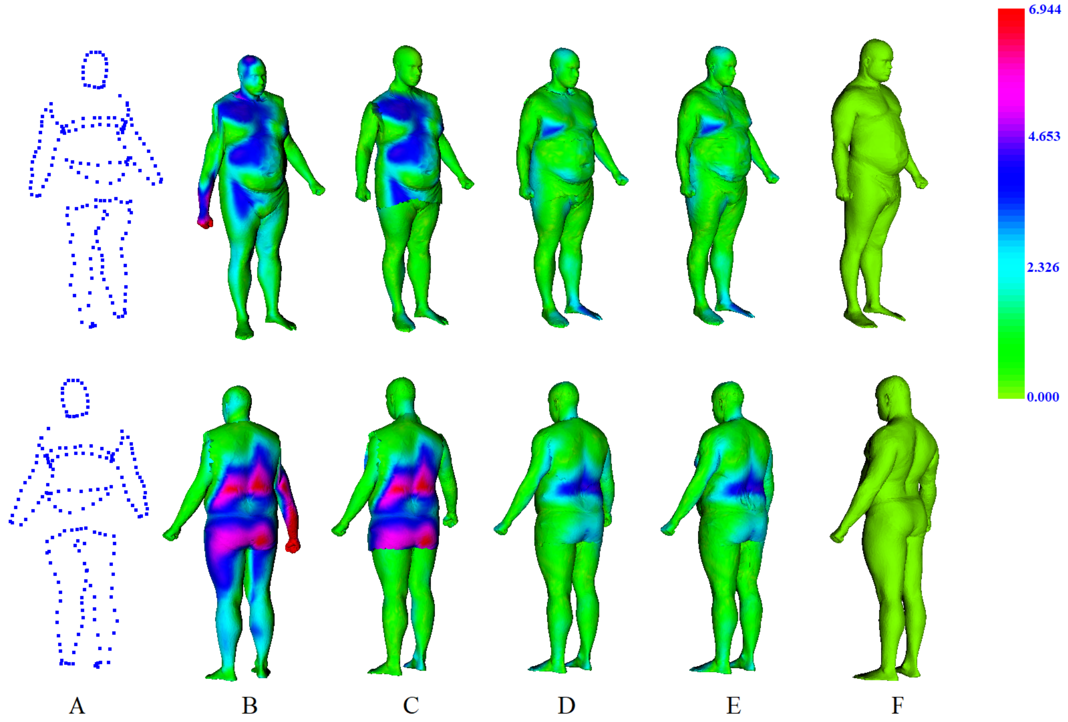


Figure 4.3: The comparison of outputs of intermediate steps. Top row is the front view, bottom row is the back view. A: input points, B: output of moving base points (3.5.1) , C: output of rigid body part alignment (3.5.2), D: output of inter-parts alignment (3.5.3), E: output of border smoothing (3.5.4), F: ground-truth. The average errors from column B to column E are 2.120, 1.291, 1.088, 1.076, respectively.

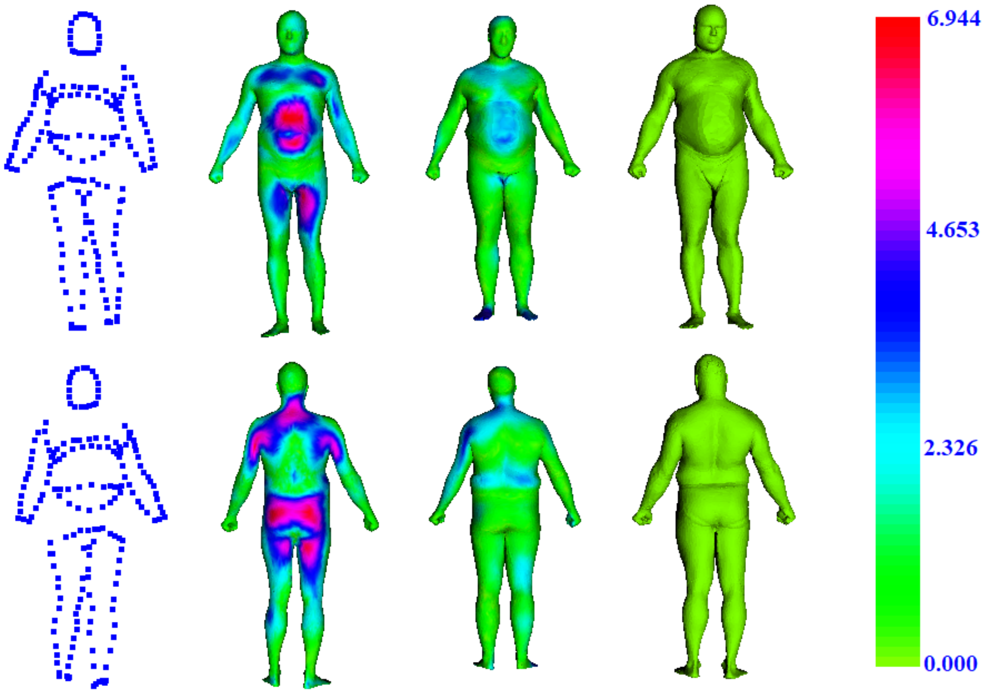


Figure 4.4: The comparison of results of Sample1. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 12, 40, 91, 79, 91, 91, 75 shown in A, respectively. The average errors of the second and third columns are 1.736 and 1.125, respectively. The average edge length of the ground-truth mesh is 0.0144.

Table 4.2: Execution Times by Methods (in seconds)

Figure No	Figure Matching (3.4)	Rotation Alignment(3.5.2)	Inter-part Alignment (3.5.3)	Border Smoothing (3.5.4)	Our Method Total	ARAP-Based
4.4	1.197	2.043	0.47	5.02	8.73	9.3
4.5	1.164	2.314	0.469	4.97	8.917	9.65
4.6	1.172	2.145	0.468	4.95	8.735	9.72
4.7	1.189	2.095	0.472	5.05	8.806	9.47
4.8	1.175	2.243	0.471	4.98	8.869	9.82
4.9	1.166	2.145	0.467	5.01	8.880	9.52
4.10	1.183	2.189	0.468	4.99	8.770	9.44
4.11	1.171	2.3	0.473	5.06	8.862	9.71
4.12	1.169	2.245	0.475	4.97	8.752	9.49
4.13	1.182	2.123	0.462	5.01	8.812	9.57
4.14	1.177	2.129	0.453	5.03	8.852	9.52
4.15	1.184	2.227	0.442	5.06	8.913	9.41
4.16	1.179	2.165	0.487	5.01	8.841	9.46
4.17	1.181	2.099	0.450	4.99	8.72	9.69
4.18	1.168	2.179	0.459	5.00	8.806	9.58
4.19	1.178	2.211	0.444	5.02	8.853	9.81
4.20	1.187	2.087	0.483	5.05	8.807	9.6
4.21	1.191	2.182	0.462	4.96	8.795	9.48
4.22	1.177	2.149	0.458	5.01	8.794	9.75
4.23	1.195	2.079	0.460	4.99	8.724	9.56

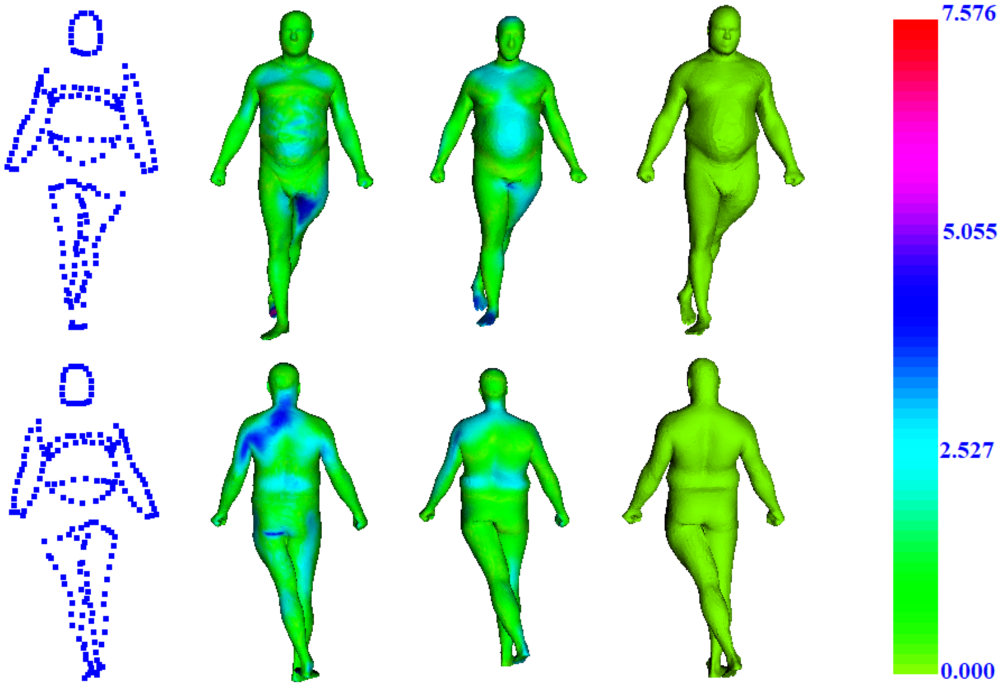


Figure 4.5: The comparison of results of Sample2. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 12, 40, 41, 29, 29, 29, 7 shown in A, respectively. The average errors of the second and third columns are 1.23 and 1.103, respectively. The average edge length of the ground-truth mesh is 0.0165.

Table 4.3: Errors by number of points after point completion

	10 Points	20 Points	30 Points	40 Points	50 Points	100 Points
30%	2.035	1.326	1.264	1.222	1.222	1.222
40%	1.39	1.354	1.25	1.18	1.183	1.181
50%	1.27	1.319	1.243	1.18	1.18	1.18
60%	1.25	1.25	1.243	1.153	1.151	1.151
70%	1.243	1.229	1.243	1.153	1.153	1.153

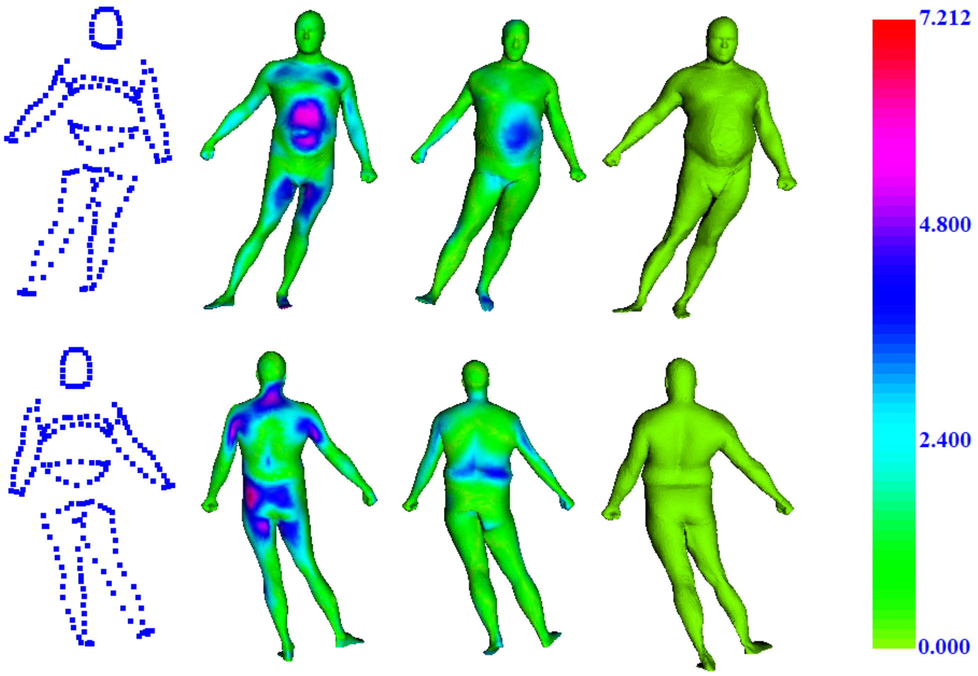


Figure 4.6: The comparison of results of Sample3. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 11, 40, 23, 79, 91, 91, 75 shown in A, respectively. The average errors of the second and third columns are 1.70 and 1.121, respectively. The average edge length of the ground-truth mesh is 0.0165.

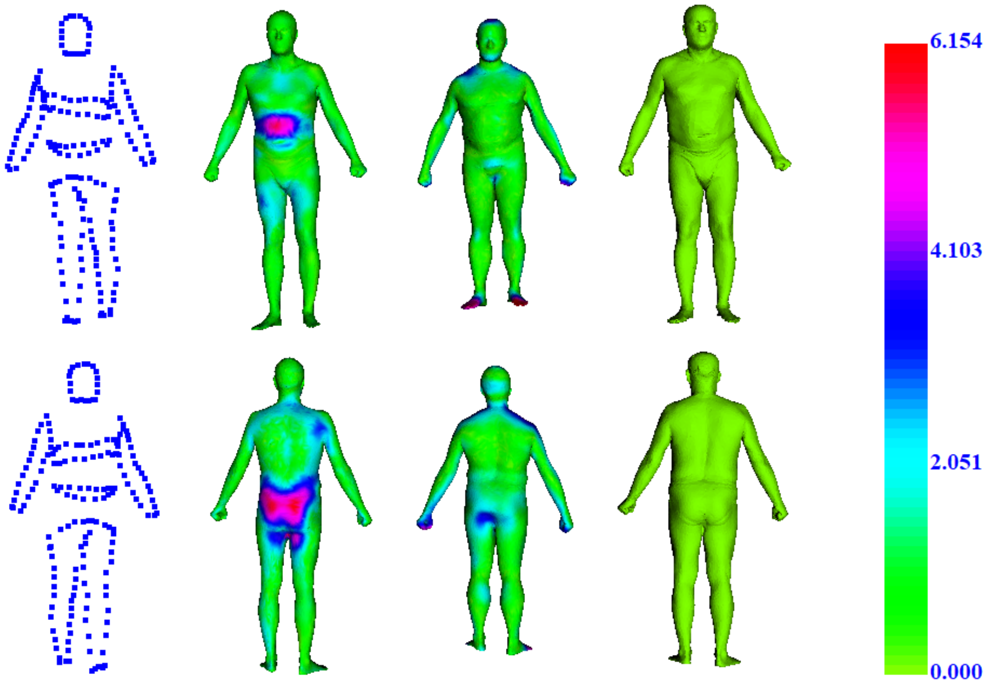


Figure 4.7: The comparison of results of Sample4. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 60, 58, 12, 12, 3, 3, 38 shown in A, respectively. The average errors of the second and third columns are 1.154 and 1.133, respectively. The average edge length of the ground-truth mesh is 0.0156.

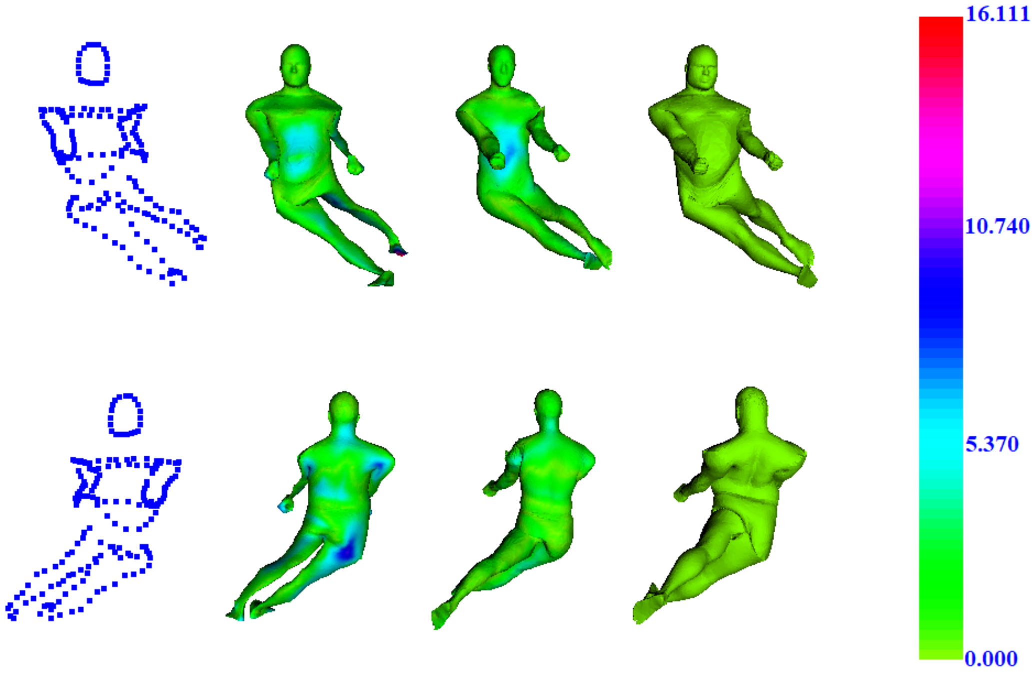


Figure 4.8: The comparison of results of Sample5. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 1, 12, 89, 89, 32, 32, 75 shown in A, respectively. The average errors of the second and third columns are 2.284 and 1.617, respectively. The average edge length of the ground-truth mesh is 0.0162.

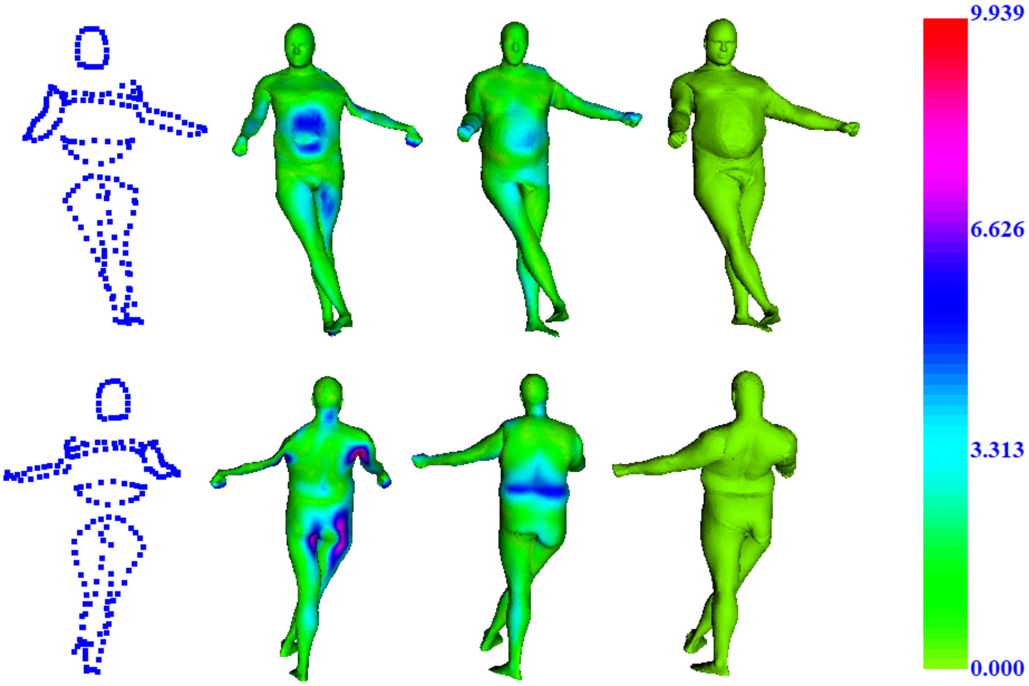


Figure 4.9: The comparison of results of Sample6. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 1, 71, 79, 79, 79, 79, 75 shown in A, respectively. The average errors of the second and third columns are 1.853 and 1.429, respectively. The average edge length of the ground-truth mesh is 0.0163.

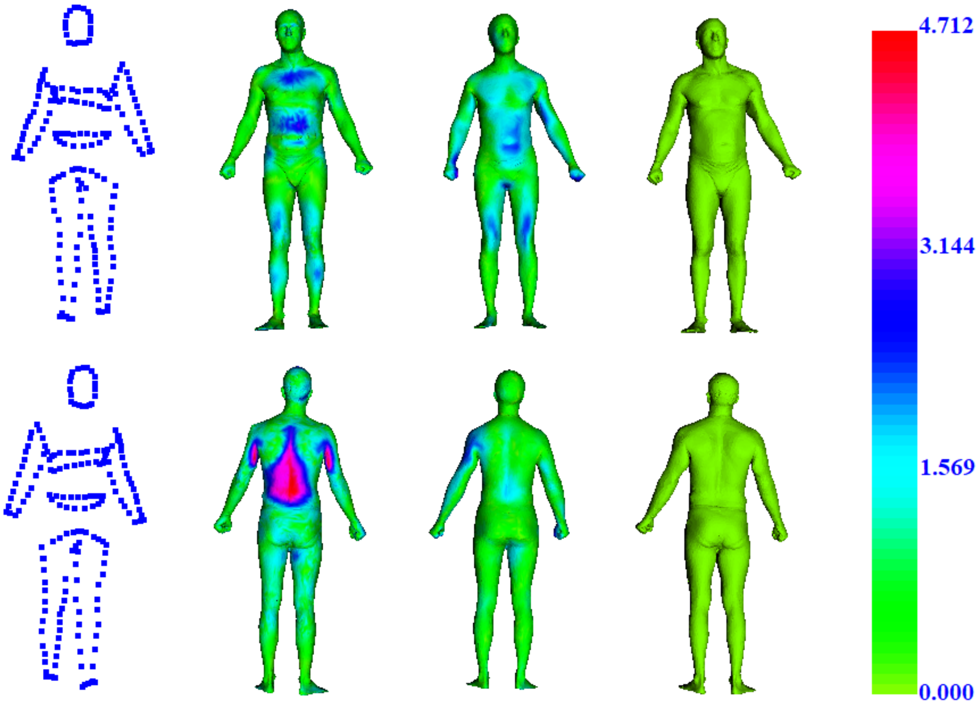


Figure 4.10: The comparison of results of Sample7. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 12, 81, 57, 57, 56, 56, 44 shown in A, respectively. The average errors of the second and third columns are 0.973 and 0.870, respectively. The average edge length of the ground-truth mesh is 0.0146.

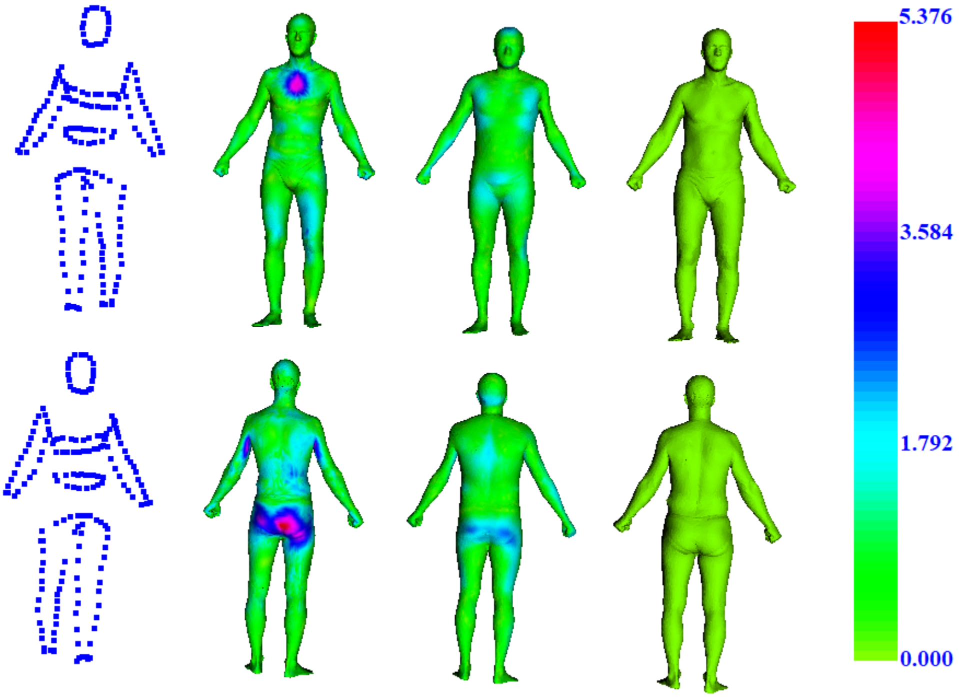


Figure 4.11: The comparison of results of Sample8. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 8, 80, 25, 25, 50, 50, 38 shown in A, respectively. The average errors of the second and third columns are 0.933 and 0.812, respectively. The average edge length of the ground-truth mesh is 0.0149.

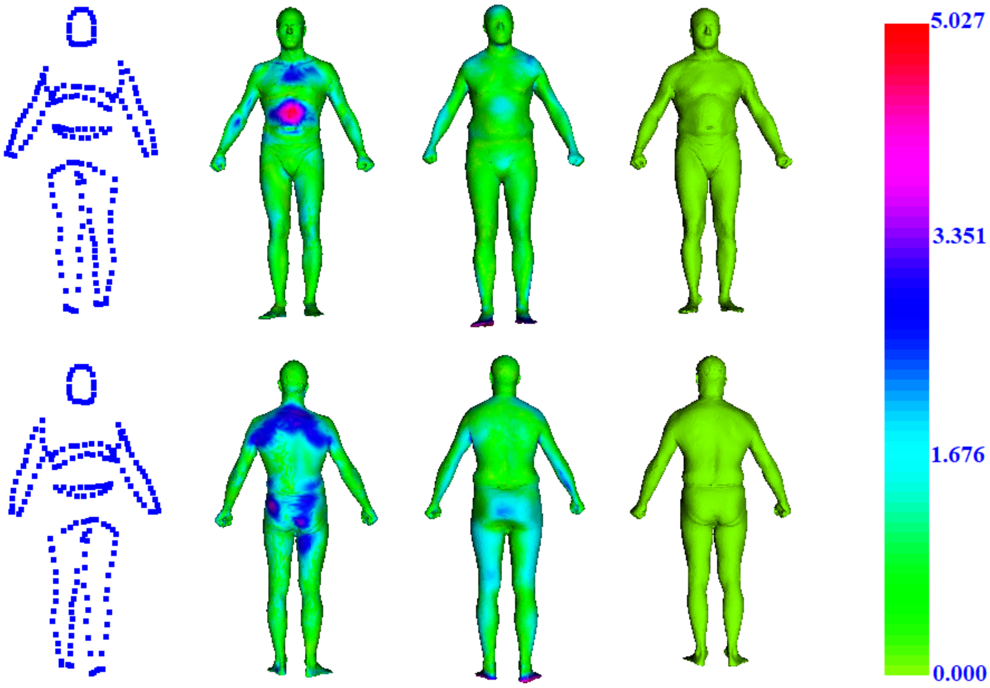


Figure 4.12: The comparison of results of Sample9. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 50, 36, 51, 51, 28, 28, 45 shown in A, respectively. The average errors of the second and third columns are 1.01 and 1.02, respectively. The average edge length of the ground-truth mesh is 0.0148.

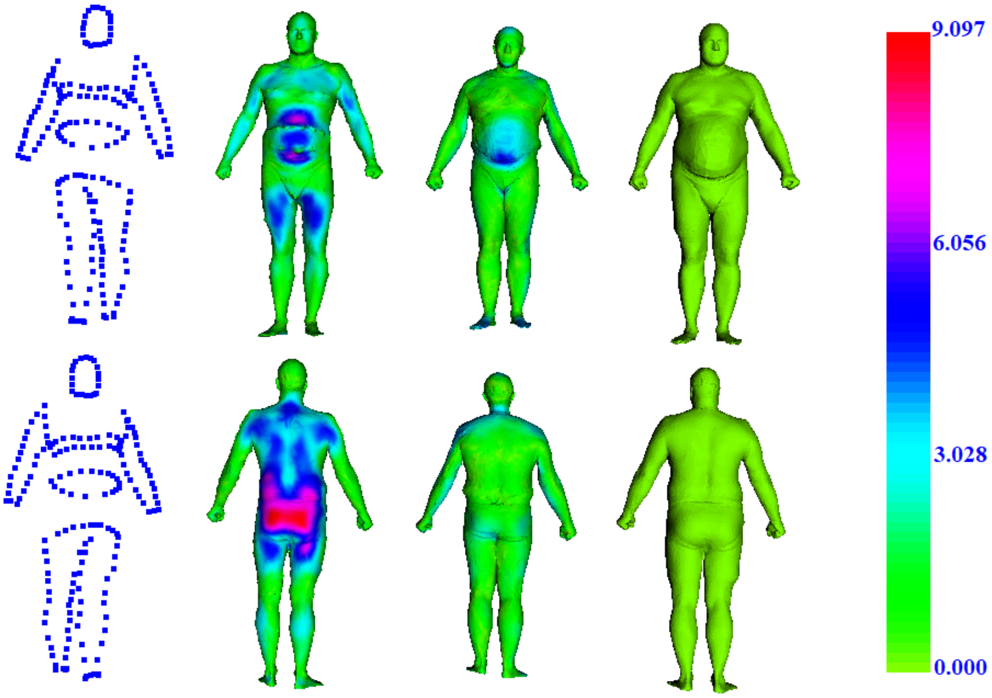


Figure 4.13: The comparison of results of Sample10. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 89, 38, 13, 13, 31, 31, 90 shown in A, respectively. The average errors of the second and third columns are 0.98 and 1.403, respectively. The average edge length of the ground-truth mesh is 0.0144.

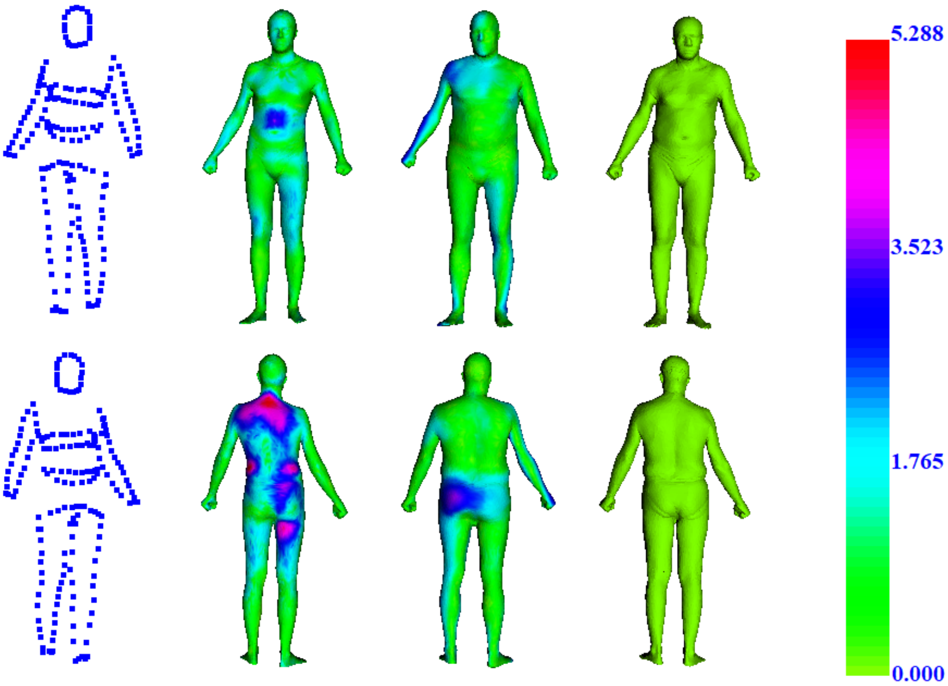


Figure 4.14: The comparison of results of Sample11. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 57, 68, 35, 35, 28, 28, 85 shown in A, respectively. The average errors of the second and third columns are 1.183 and 1.045, respectively. The average edge length of the ground-truth mesh is 0.0153.

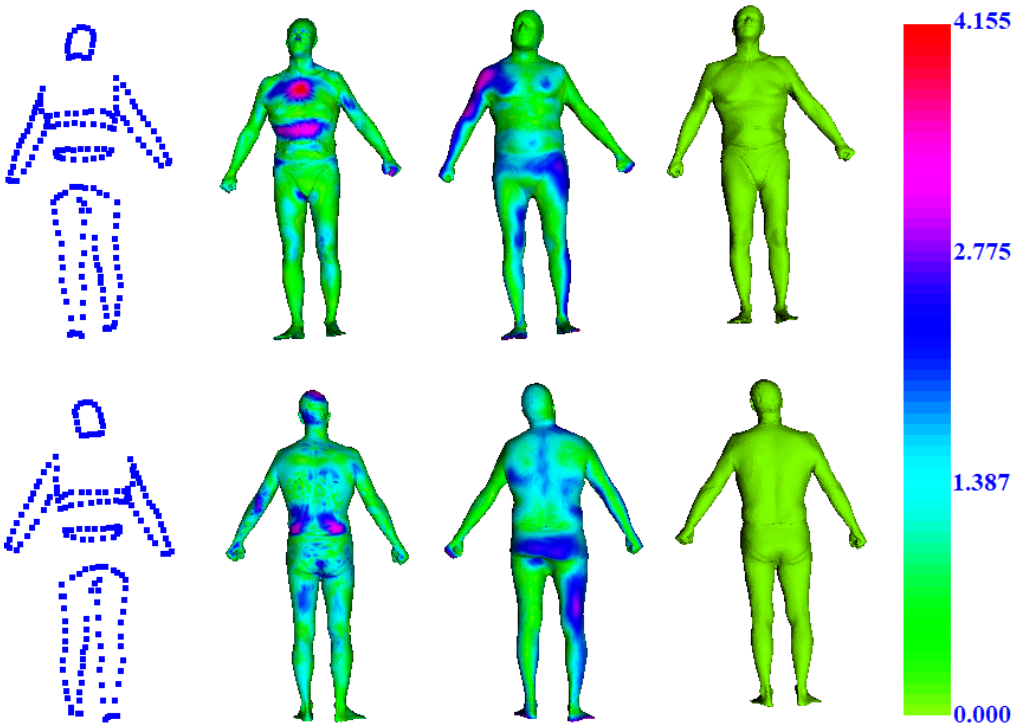


Figure 4.15: The comparison of results of Sample12. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 39, 9, 70, 70, 9, 9, 99 shown in A, respectively. The average errors of the second and third columns are 1.303 and 1.254, respectively. The average edge length of the ground-truth mesh is 0.0142.

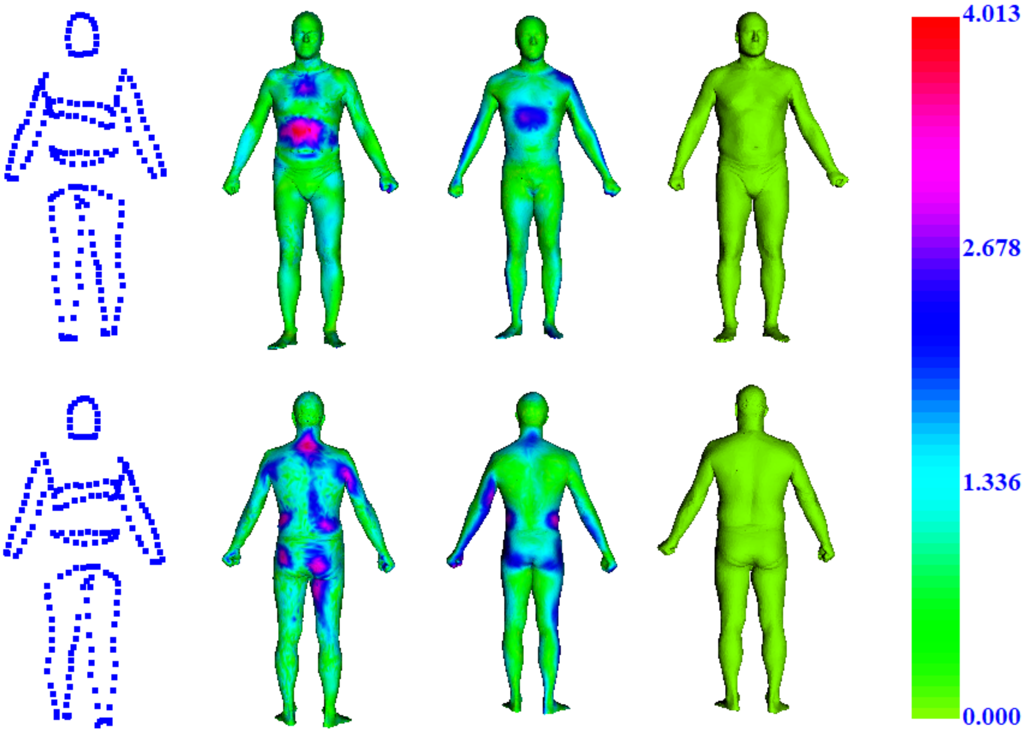


Figure 4.16: The comparison of results of Sample13. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 4, 25, 17, 17, 55, 55, 55 shown in A, respectively. The average errors of the second and third columns are 1.033 and 1.067, respectively. The average edge length of the ground-truth mesh is 0.0149.

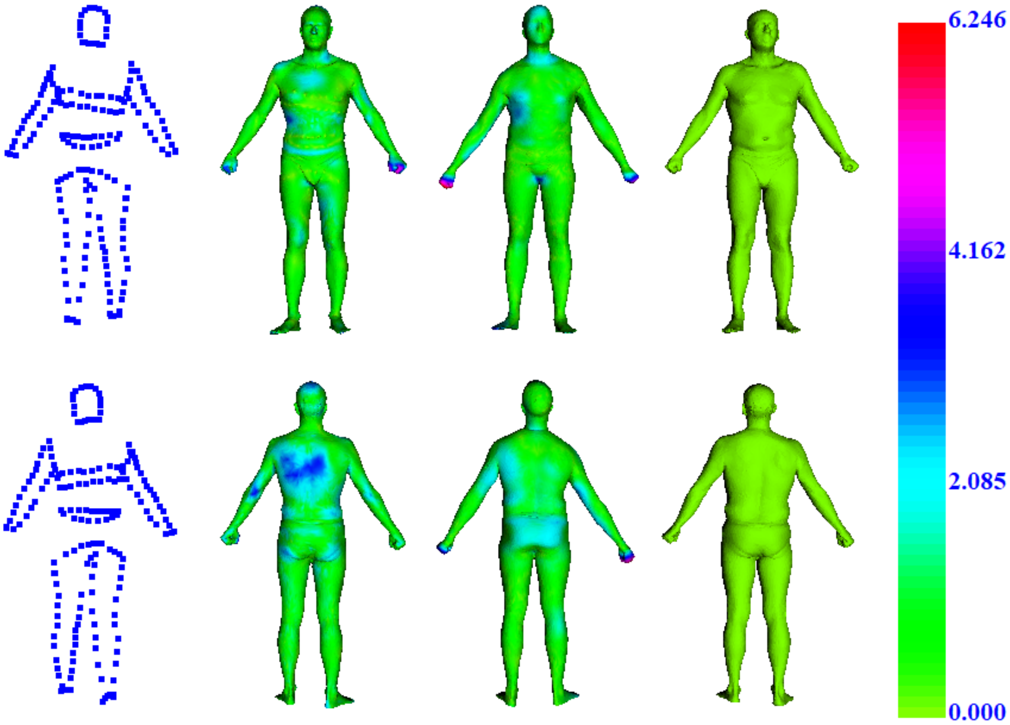


Figure 4.17: The comparison of results of Sample14. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 8, 8, 82, 82, 38, 38, 19 shown in A, respectively. The average errors of the second and third columns are 1.007 and 0.971, respectively. The average edge length of the ground-truth mesh is 0.0142.

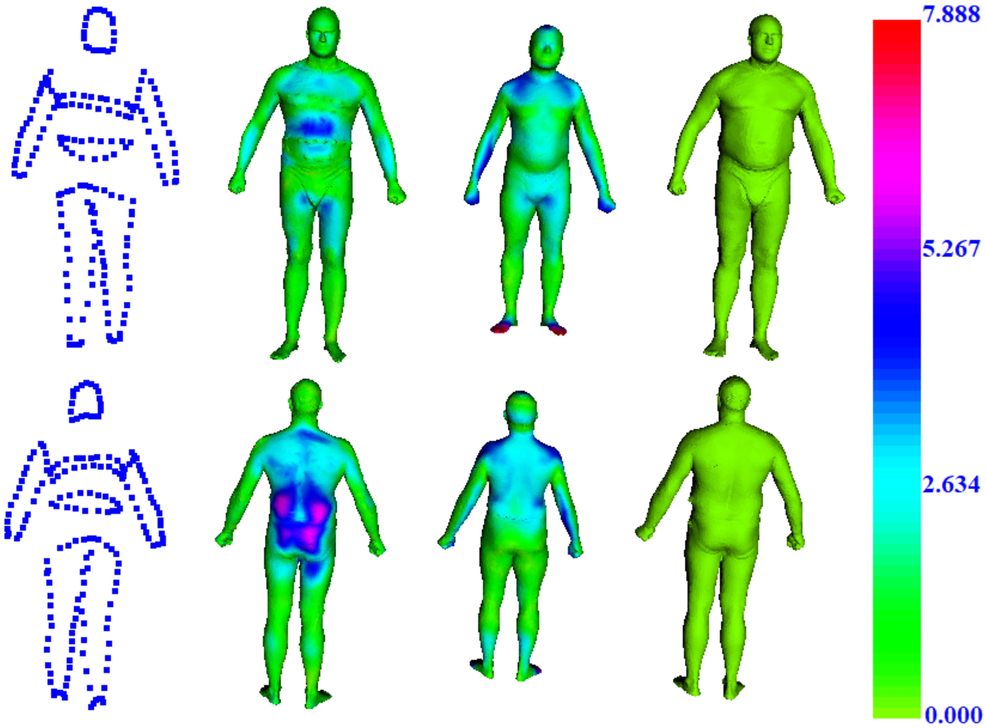


Figure 4.18: The comparison of results of Sample15. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 42, 51, 92, 4, 39, 39, 97 shown in A, respectively. The average errors of the second and third columns are 1.981 and 1.876, respectively. The average edge length of the ground-truth mesh is 0.0161.

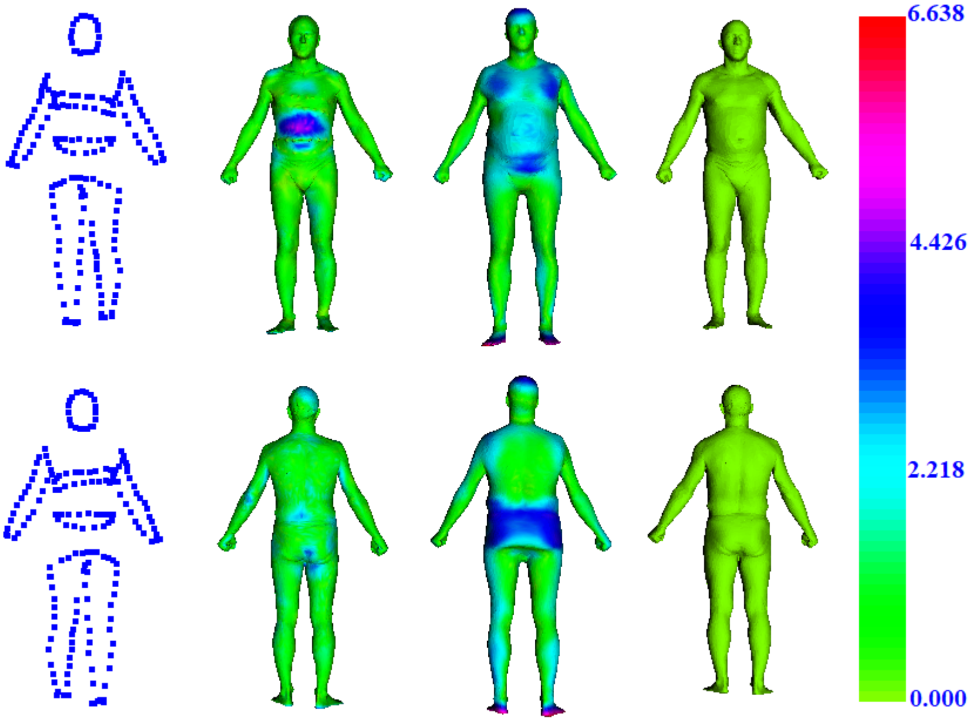


Figure 4.19: The comparison of results of Sample16. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 57, 84, 91, 91, 28, 28, 53 shown in A, respectively. The average errors of the second and third columns are 1.610 and 1.738, respectively. The average edge length of the ground-truth mesh is 0.0141.

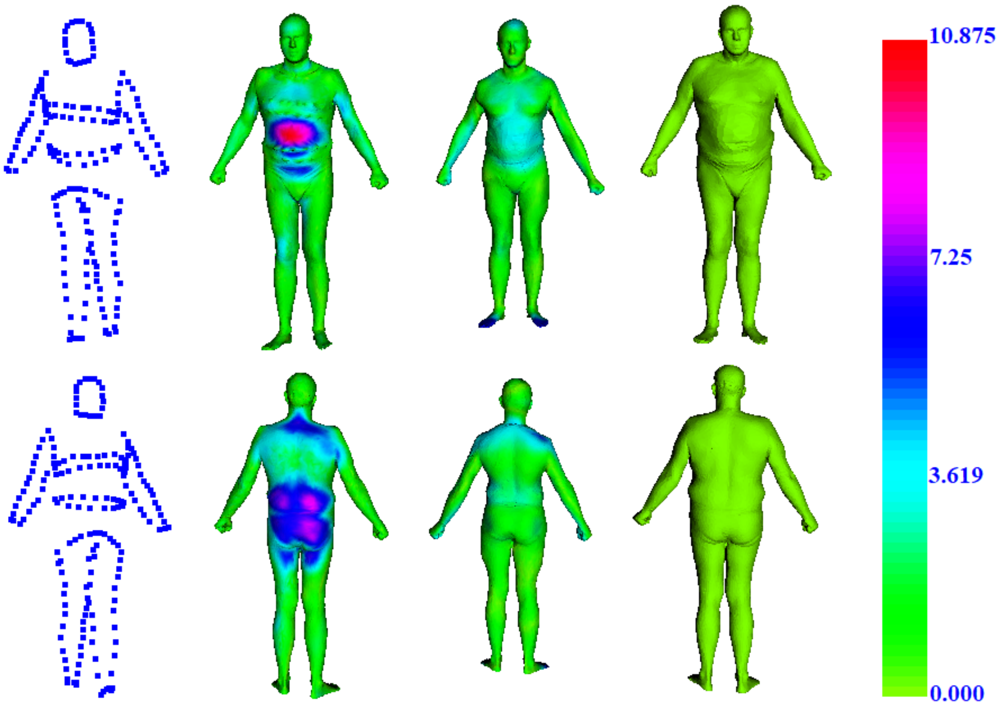


Figure 4.20: The comparison of results of Sample17. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 69, 30, 39, 39, 72, 72, 68 shown in A, respectively. The average errors of the second and third columns are 1.96 and 1.743, respectively. The average edge length of the ground-truth mesh is 0.0160.

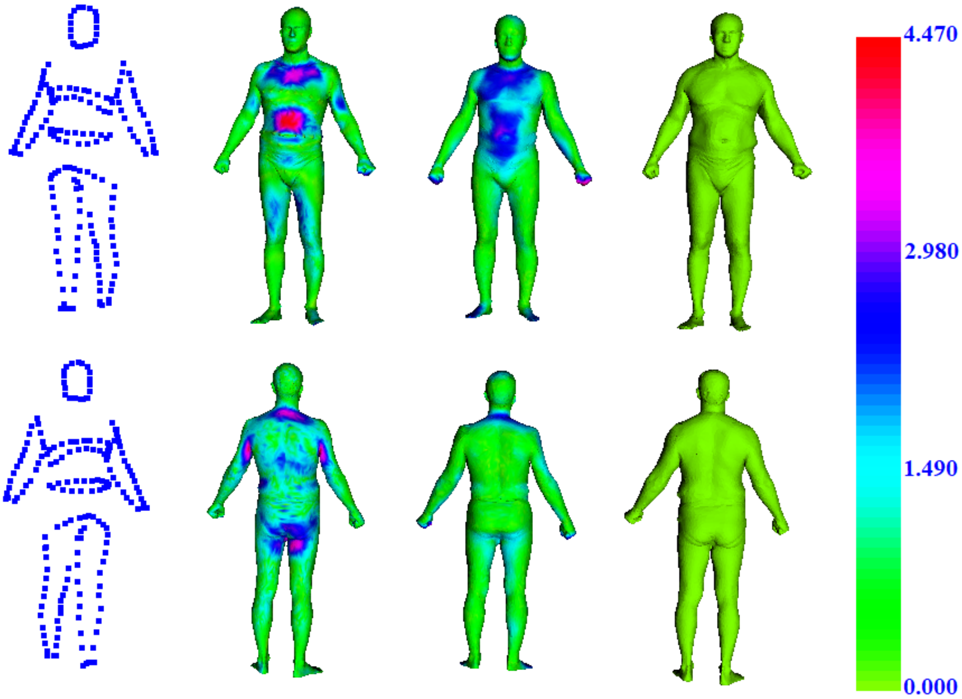


Figure 4.21: The comparison of results of Sample18. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 62, 7, 43, 98, 31, 31, 8 shown in A, respectively. The average errors of the second and third columns are 1.00 and 0.98, respectively. The average edge length of the ground-truth mesh is 0.0149.

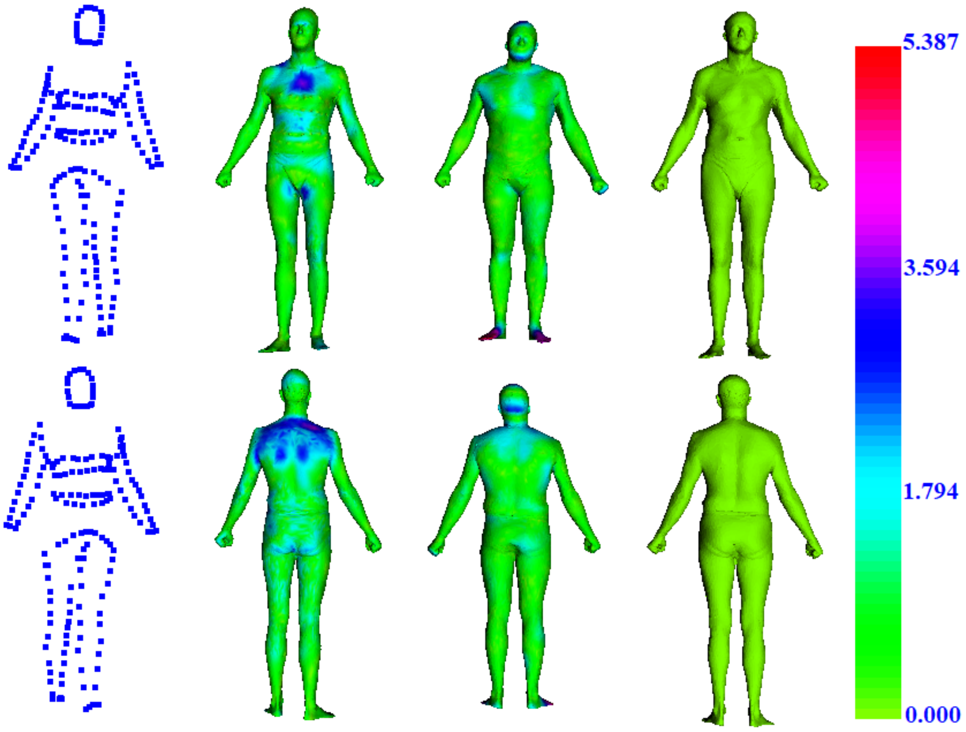


Figure 4.22: The comparison of results of Sample19. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 57, 75, 3, 3, 38, 38, 70 shown in A, respectively. The average errors of the second and third columns are 1.281 and 1.176, respectively. The average edge length of the ground-truth mesh is 0.0155.

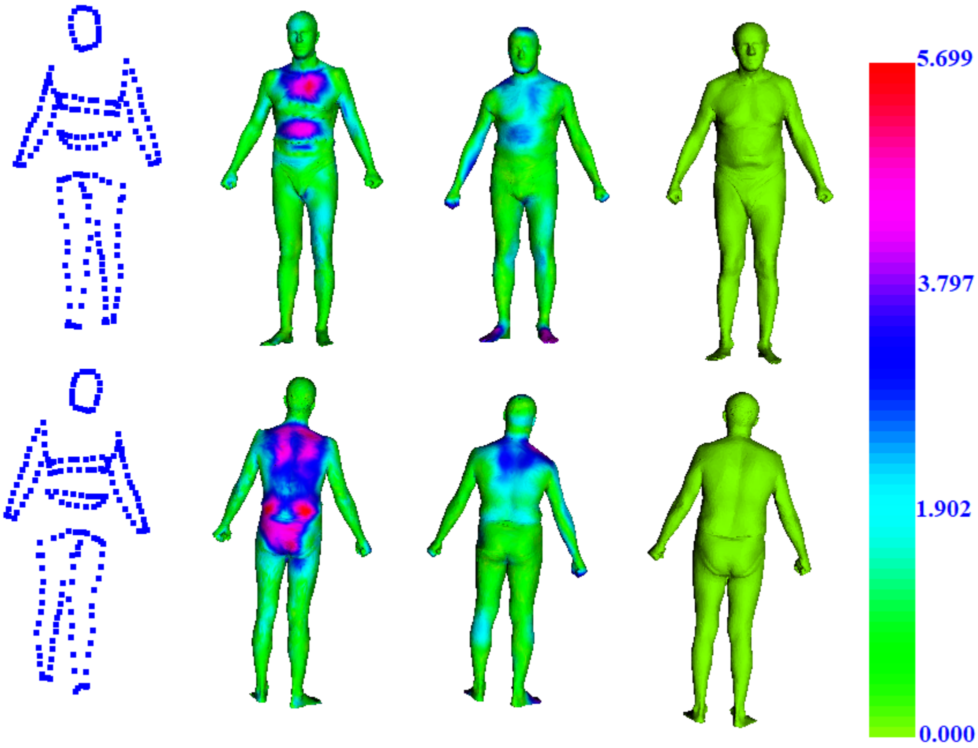


Figure 4.23: The comparison of results of Sample20. Top row is the front view, bottom row is the back view. From left to right, input points, ARAP-based result, result of our method and ground-truth are shown, respectively. The body parts, namely left arm, right arm, chest, belly, left leg, right leg and head, are taken from the bodies with number 89, 77, 82, 82, 67, 67, 74 shown in A, respectively. The average errors of the second and third columns are 1.359 and 1.232, respectively. The average edge length of the ground-truth mesh is 0.0153.

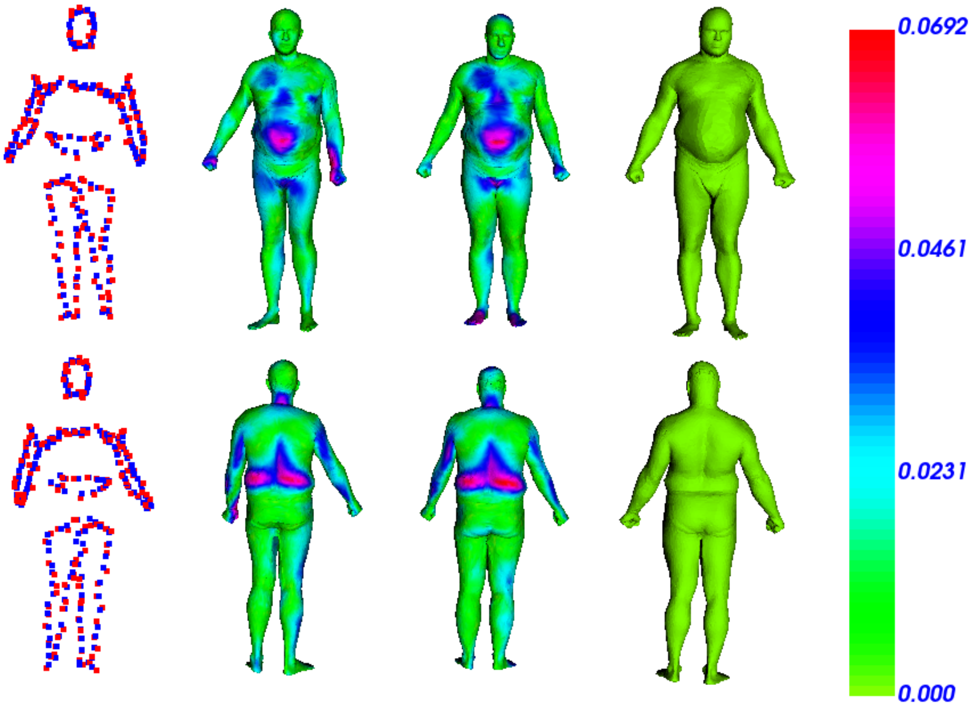


Figure 4.24: The comparison of results of noisy input. Top row is the front view, bottom row is the back view. From left to right, input points(noisy input colored as red, original input colored as blue), result of noisy input , result of original input and ground-truth are shown, respectively. Noise is generated from Gaussian distribution with standard deviation of 0.8. The average errors of the second and third columns are 1.243 and 1.125, respectively. The average edge length of the ground-truth mesh is 0.0144.

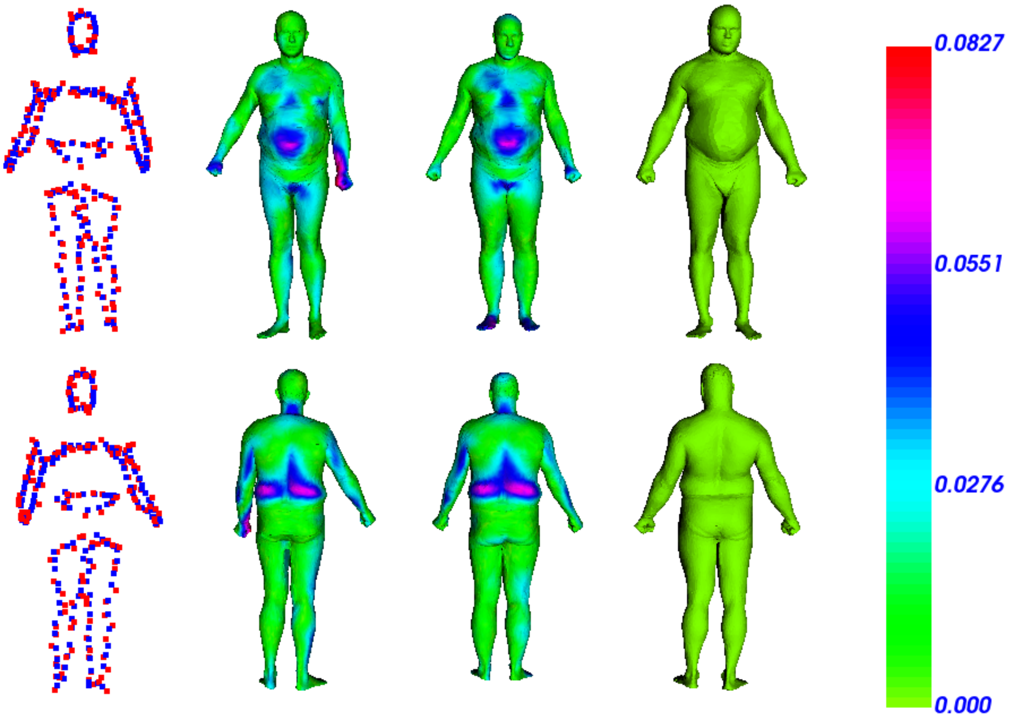


Figure 4.25: The comparison of results of noisy input. Top row is the front view, bottom row is the back view. From left to right, input points(noisy input colored as red, original input colored as blue), result of noisy input , result of original input and ground-truth are shown, respectively. Noise is generated from Gaussian distribution with standard deviation of 1. The average errors of the second and third columns are 1.285 and 1.125, respectively. The average edge length of the ground-truth mesh is 0.0144.

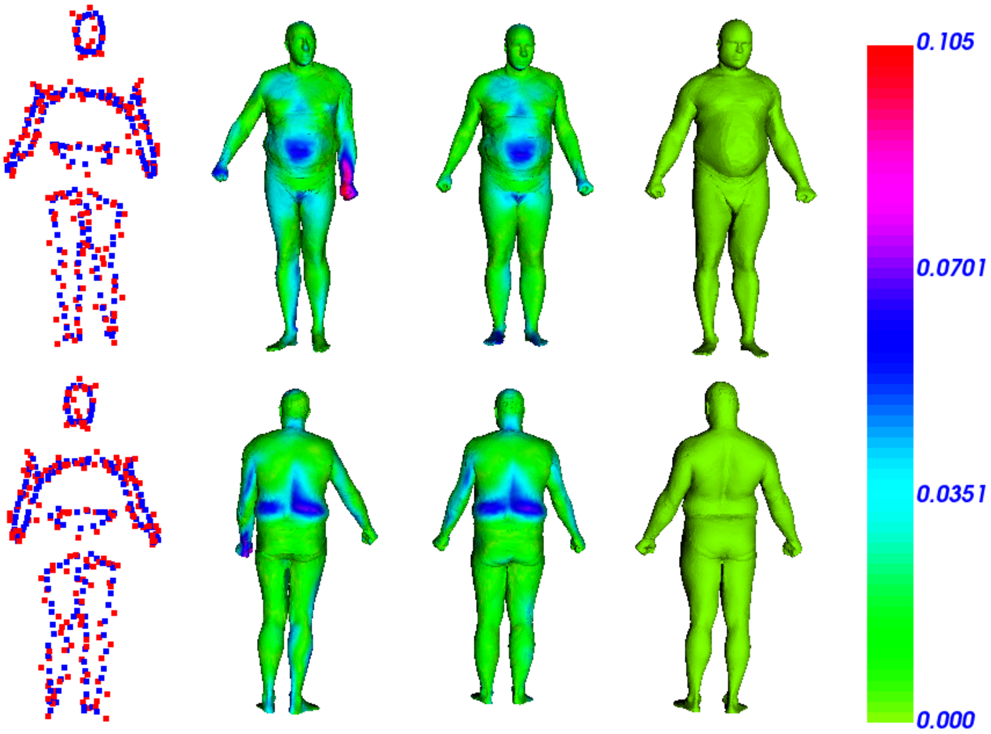


Figure 4.26: The comparison of results of noisy input. Top row is the front view, bottom row is the back view. From left to right, input points(noisy input colored as red, original input colored as blue), result of noisy input , result of original input and ground-truth are shown, respectively. Noise is generated from Gaussian distribution with standard deviation of 1.5. The average errors of the second and third columns are 1.458 and 1.125, respectively. The average edge length of the ground-truth mesh is 0.0144.

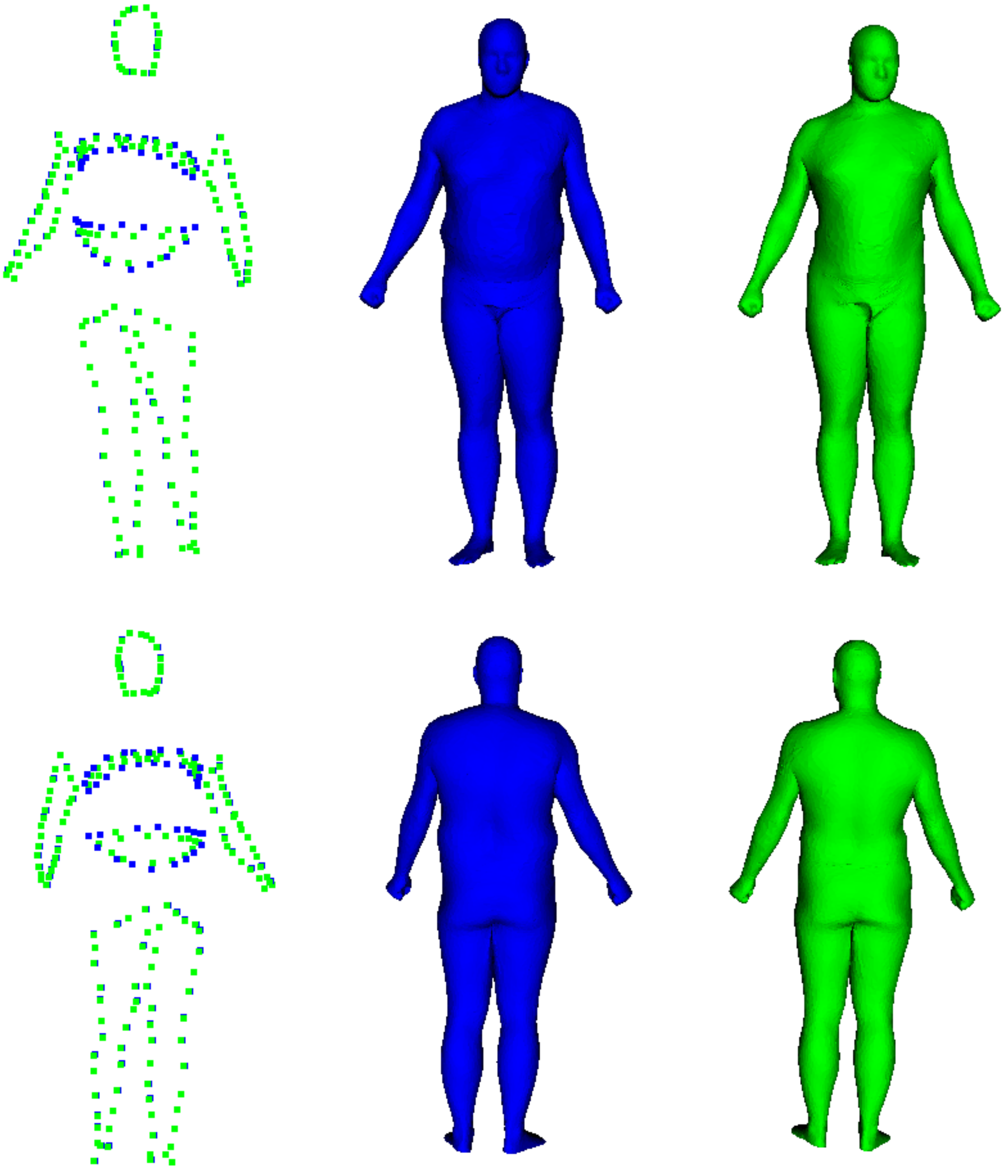


Figure 4.27: Illustration of generating new body by manual editing. Top row is the front view, bottom row is the back view. From left to right, input points(manually edited input colored as blue, original input colored as green), result of manually edited input and result of original input are shown, respectively.

CHAPTER 5

LIMITATIONS

The most important limitation of the proposed method is that the input points need to be on the contours of the corresponding body part. As the input points move away from the contours, the resulting reconstructed body becomes different from the ground-truth result. This is caused by the fact that the features are designed for defining a body part as a contour;therefore, given a non-contour point set, our method produces erroneous result. The erroneous results produced from the input points which are far from the contours of the body parts are shown in Figure 5.1 and Figure 5.2.

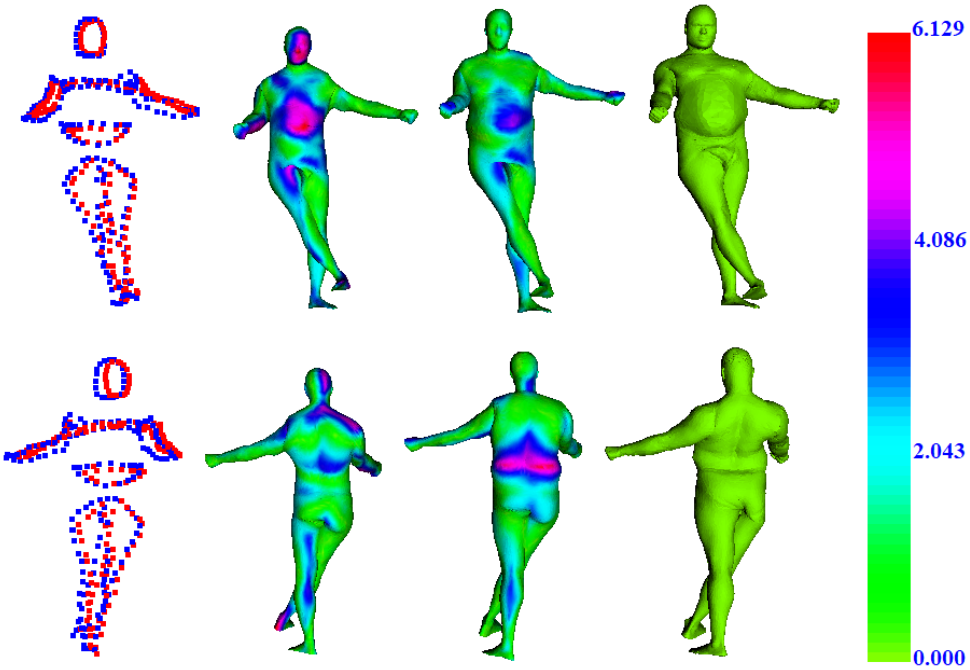


Figure 5.1: Comparison of the results produced from the input points which are far from the contours and from the input points which are around the contours. Top row is the front view,bottom row is the back view. From left to right, input points (red one represents the input points which are far from the contours and blue one represents the input points which are around the contours), result of red input points,result of blue input points and ground-truth are shown respectively. The average errors for model in the second and third column are 2.083 and 1.569 respectively.

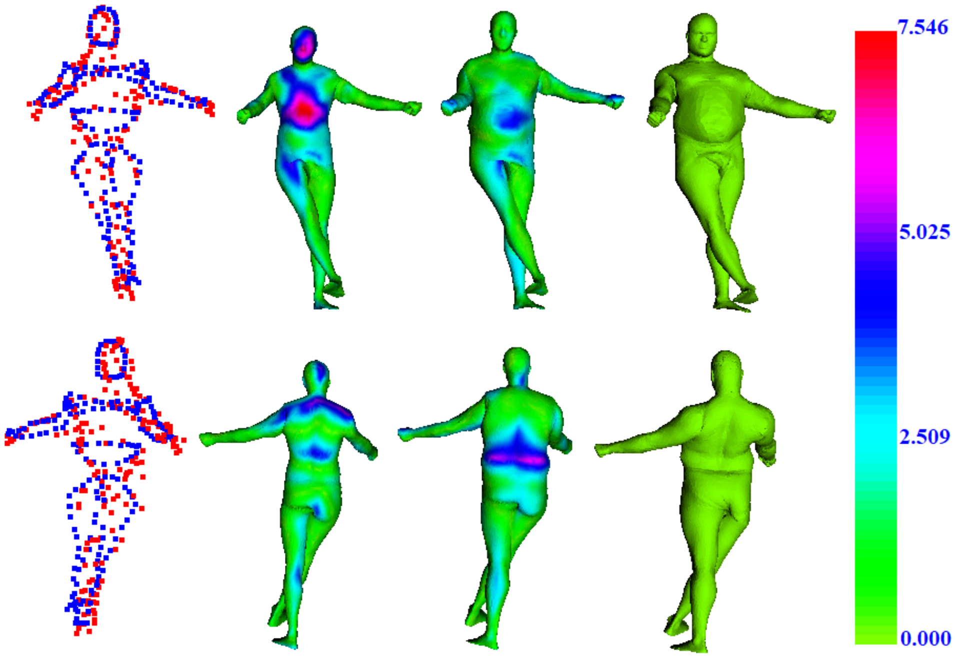


Figure 5.2: Comparison of the results produced from the input points which are selected randomly from each body part and from the input points which are around the contours. Top row is the front view,bottom row is the back view. From left to right, input points (red one represents the input points which are selected randomly from each body part and blue one represents the input points which are around the contours), result of red input points,result of blue input points and ground-truth are shown respectively. The average errors for model in the second and third column are 2.639 and 1.569 respectively.

CHAPTER 6

CONCLUSION AND DISCUSSION

In computer graphics, one of the most studied field is human body modelling. In order to create human body models, different approaches uses different inputs such as single RGB-D depth frame [10], multiple depth frames [15], morphable template models and landmarks [18], 3D point cloud [26]. Differently from previous studies, in this thesis, a data-driven method which reconstructs human body from sparse input points is proposed.

The evaluations have shown that proposed method is successful at generating human body from limited number of points in terms of both accuracy and execution time with respect to the base method. The proposed method is able to reconstruct bodies in different shapes and different orientations thanks to the novel point set features.

As a future work, our method can be extended so that the bodies are reconstructed from 2D images by combining our method and the methods like the one proposed in [6].

APPENDIX A

MODELS USED IN OUR ALGORITHM

In this chapter, all models in the database are shown in Figures A.1, A.2, A.3 and A.4. The set of model which used in this study is a small subset of the SCAPE database provided by [23] where each model has 12500 vertices and 2500 faces. In addition, the models in the dataset are in full-correspondence.

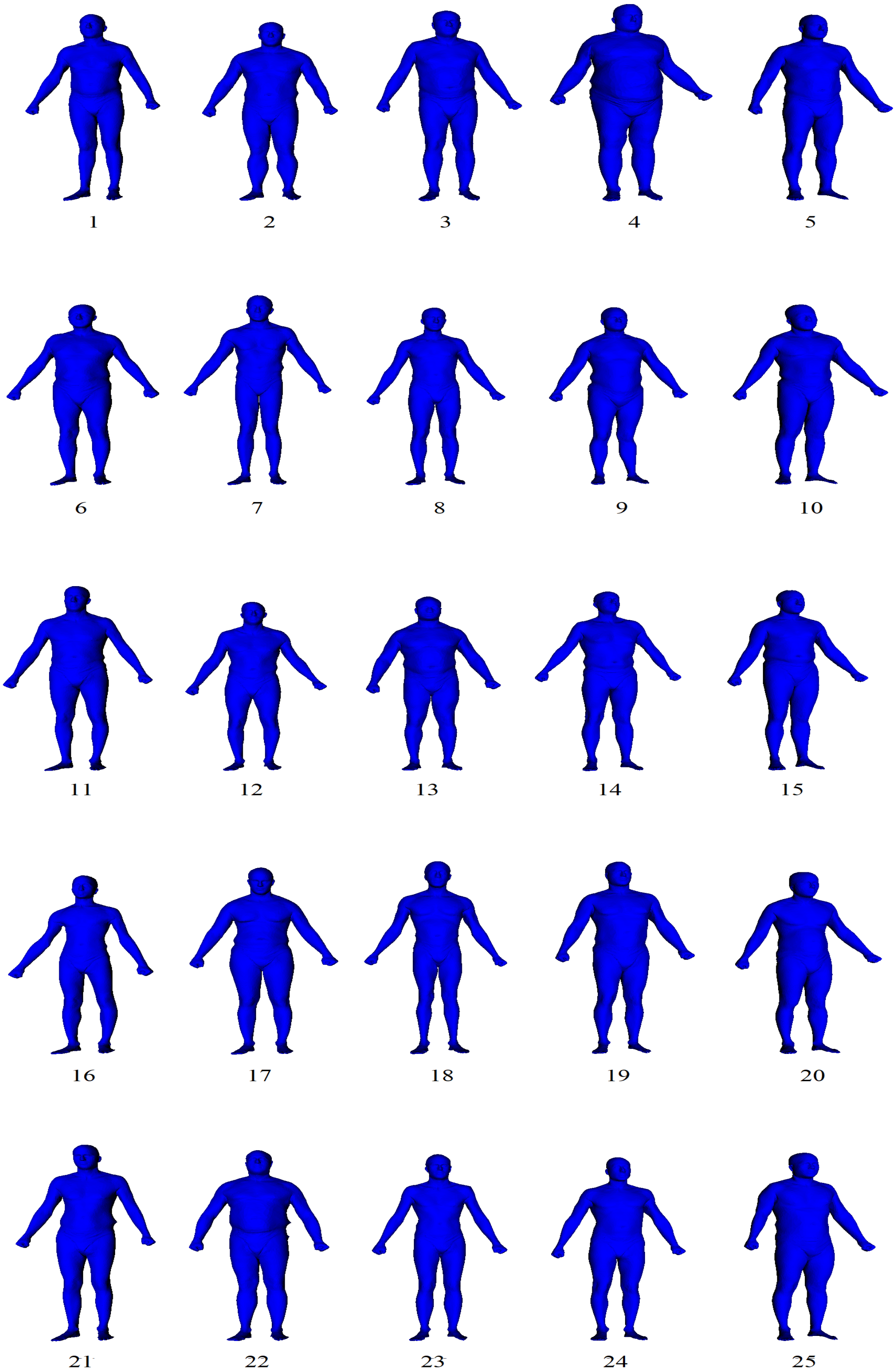
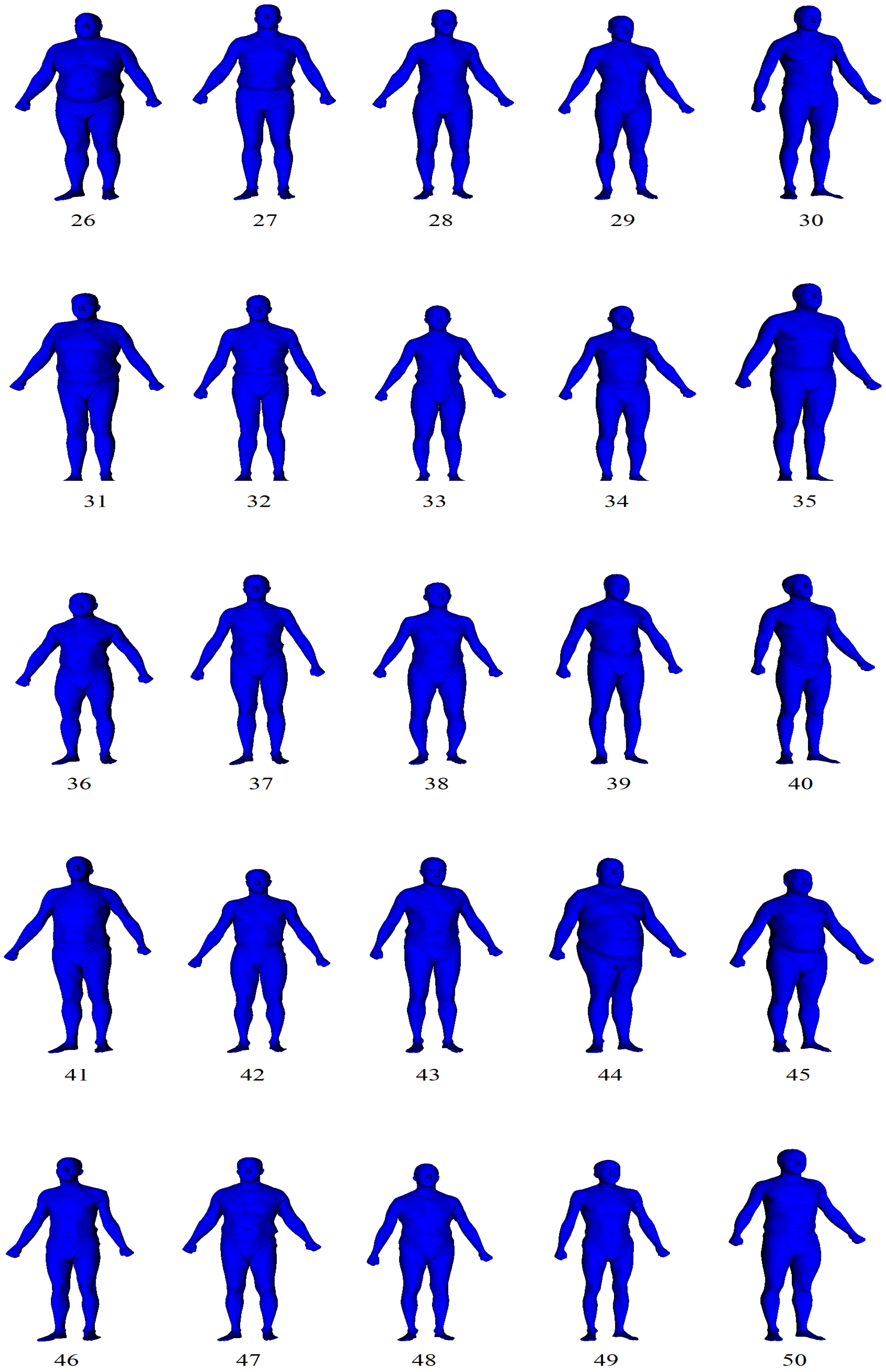


Figure A.1: Models in our database



26 27 28 29 30

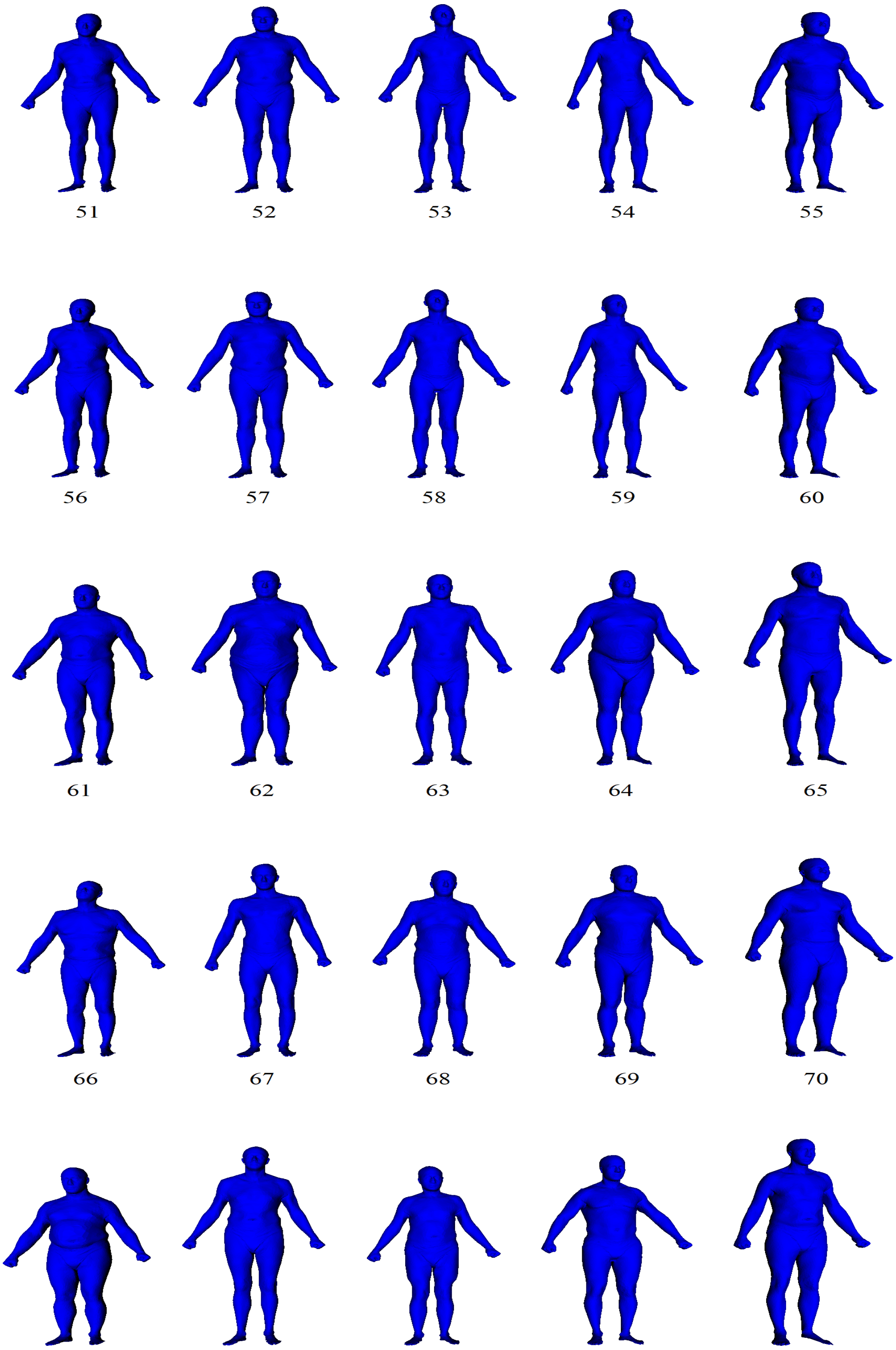
31 32 33 34 35

36 37 38 39 40

41 42 43 44 45

46 47 48 49 50

Figure A.2: Models in our database



51 52 53 54 55

56 57 58 59 60

61 62 63 64 65

66 67 68 69 70

71 72 73 74 75

Figure A.3: Models in our database

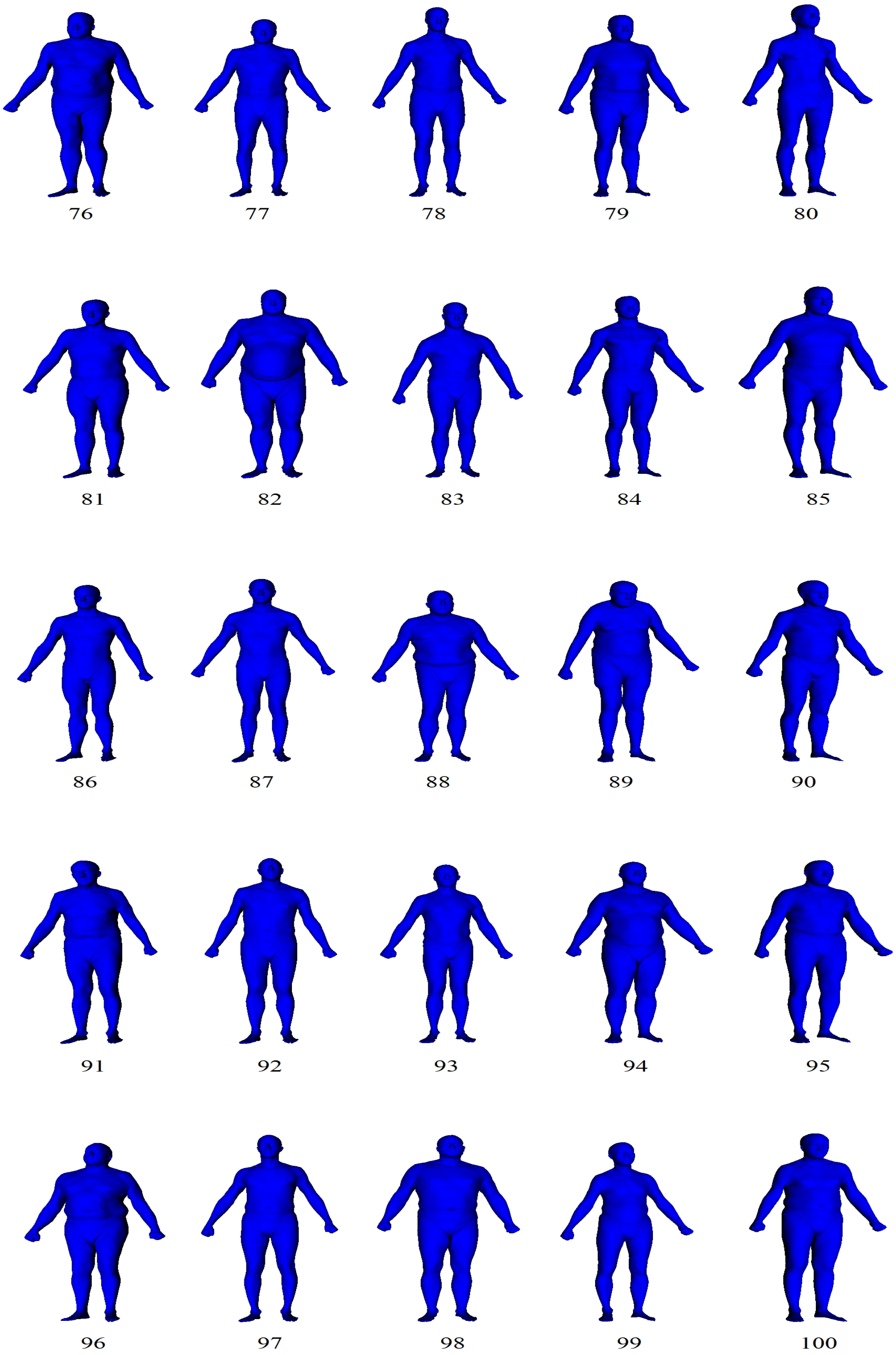


Figure A.4: Models in our database

REFERENCES

[1] P. R. Jones and M. Rioux, “Three-dimensional surface anthropometry: applications to the human body,” *Optics and Lasers in Engineering*, vol. 28, no. 2, pp. 89–117, 1997.

[2] N. Fuller, S. Jebb, M. Laskey, W. Coward, and M. Elia, “Four-component model for the assessment of body composition in humans: comparison with alternative methods, and evaluation of the density and hydration of fat-free mass,” *Clinical Science*, vol. 82, no. 6, pp. 687–693, 1992.

[3] M. Batouche, “A knowledge based system for diagnosing spinal deformations: moiré pattern analysis and interpretation,” in *[1992] Proceedings. 11th IAPR International Conference on Pattern Recognition*, pp. 591–594, IEEE, 1992.

[4] J. K. Hodgins, W. L. Wooten, D. C. Brogan, and J. F. O’Brien, “Animating human athletics,” tech. rep., Georgia Institute of Technology, 1995.

[5] P. Fua, A. Gruen, R. Plänkers, N. D’Apuzzo, and D. Thalmann, “Human body modeling and motion analysis from video sequences,” in *ISPRS Commission V Symposium*, Swiss Federal Institute of Technology, Institute of Geodesy and . . . , 1998.

[6] R. Zhao, Y. Wang, C. F. Benitez-Quiroz, Y. Liu, and A. M. Martinez, “Fast and precise face alignment and 3d shape reconstruction from a single 2d image,” in *European Conference on Computer Vision*, pp. 590–603, Springer, 2016.

[7] O. Glauser, D. Panozzo, O. Hilliges, and O. Sorkine-Hornung, “Deformation capture via soft and stretchable sensor arrays,” *ACM Transactions on Graphics (TOG)*, vol. 38, no. 2, p. 16, 2019.

[8] L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys, “Motion capture of hands in action using discriminative salient points,” in *European Conference on Computer Vision*, pp. 640–653, Springer, 2012.

[9] T. von Marcard, B. Rosenhahn, M. J. Black, and G. Pons-Moll, “Sparse inertial poser: Automatic 3d human pose estimation from sparse imus,” in *Computer Graphics Forum*, vol. 36, pp. 349–360, Wiley Online Library, 2017.

[10] S. Liang, I. Kemelmacher-Shlizerman, and L. G. Shapiro, “3d face hallucination from a single depth frame,” in *2014 2nd International Conference on 3D Vision*, vol. 1, pp. 31–38, IEEE, 2014.

[11] J. C. Gower, “Generalized procrustes analysis,” *Psychometrika*, vol. 40, no. 1, pp. 33–51, 1975.

[12] B. Allen, B. Curless, B. Curless, and Z. Popović, “The space of human body shapes: reconstruction and parameterization from range scans,” in *ACM transactions on graphics (TOG)*, vol. 22, pp. 587–594, ACM, 2003.

[13] I. Atmosukarto, L. G. Shapiro, and C. Heike, “The use of genetic programming for learning 3d craniofacial shape quantifications,” in *2010 20th International Conference on Pattern Recognition*, pp. 2444–2447, IEEE, 2010.

[14] E. Mercan, L. G. Shapiro, S. M. Weinberg, and S.-I. Lee, “The use of pseudo-landmarks for craniofacial analysis: A comparative study with l 1-regularized logistic regression,” in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 6083–6086, IEEE, 2013.

[15] E. Bondi, P. Pala, S. Berretti, and A. Del Bimbo, “Reconstructing high-resolution face models from depth sequences acquired in uncooperative contexts,” in *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 7, pp. 1–6, IEEE, 2015.

[16] A. Myronenko and X. Song, “Point set registration: Coherent point drift,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.

[17] P. Frasconi, L. Silvestri, P. Soda, R. Cortini, F. S. Pavone, and G. Iannello, “Large-scale automated identification of mouse brain cells in confocal light sheet microscopy images,” *Bioinformatics*, vol. 30, no. 17, pp. i587–i593, 2014.

[18] J. Achenbach, T. Waltemate, M. E. Latoschik, and M. Botsch, "Fast generation of realistic virtual humans," in *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*, p. 12, ACM, 2017.

[19] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Josa a*, vol. 4, no. 4, pp. 629–642, 1987.

[20] S. R. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Journal of Robotics and Automation*, vol. 17, no. 1-19, p. 16, 2004.

[21] Y. Guo, J. Zhang, L. Cai, J. Cai, and J. Zheng, "Self-supervised cnn for unconstrained 3d facial performance capture from an rgb-d camera," *arXiv preprint arXiv:1808.05323*, 2018.

[22] V. Blanz, T. Vetter, *et al.*, "A morphable model for the synthesis of 3d faces.," in *Siggraph*, vol. 99, pp. 187–194, 1999.

[23] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "Scape: shape completion and animation of people," in *ACM transactions on graphics (TOG)*, vol. 24, pp. 408–416, ACM, 2005.

[24] D. Anguelov, P. Srinivasan, H.-C. Pang, D. Koller, S. Thrun, and J. Davis, "The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces," in *Advances in neural information processing systems*, pp. 33–40, 2005.

[25] C. Miller, O. Arikan, and D. Fussell, "Frankenrigs: building character rigs from multiple sources," in *Proceedings of the 2010 ACM SIGGRAPH symposium on Interactive 3D Graphics and Games*, pp. 31–38, ACM, 2010.

[26] Q. Chen and V. Koltun, "Robust nonrigid registration by convex optimization," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2039–2047, 2015.

[27] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching," *Proceedings of the National Academy of Sciences*, vol. 103, no. 5, pp. 1168–1172, 2006.

[28] H. Maron, N. Dym, I. Kezurer, S. Kovalsky, and Y. Lipman, “Point registration via efficient convex relaxation,” *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 73, 2016.

[29] A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa, “Laplacian mesh optimization,” in *Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, pp. 381–389, ACM, 2006.

[30] O. Sorkine, “Differential representations for mesh processing,” *Computer Graphics Forum*, vol. 25, no. 4, pp. 789–807, 2006.

[31] M. Meyer, M. Desbrun, P. Schröder, and A. H. Barr, “Discrete differential-geometry operators for triangulated 2-manifolds,” in *Visualization and mathematics III*, pp. 35–57, Springer, 2003.

[32] A. Jourabloo and X. Liu, “Pose-invariant 3d face alignment,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3694–3702, 2015.

[33] T. Igarashi, T. Moscovich, and J. F. Hughes, “As-rigid-as-possible shape manipulation,” in *ACM transactions on Graphics (TOG)*, vol. 24, pp. 1134–1141, ACM, 2005.