# Human Body Modeling Based on Pose and Shape Space

Baidar Nima[1,a) ]    Ikehata Satoshi[1]    Aizawa Kiyoharu[1]    Sakazawa Shigeyuki[2]

**Abstract:** In this paper, we propose a method to create a 3D model of a human body using commercial depth cameras. Due to the cheaply available commercial depth cameras, various systems that use 3D scan can now be realized at natural environment. However, due to the poor quality of raw depth data, instead of using the depth input directly to create a 3D model, we try to extract the pose and the shape of a person,and then use these information to deform a template model. We use a parametric body model called SCAPE. SCAPE body model employs a low dimensional model of shape and pose dependent deformation that is learnt from the database of range scans of human bodies. In our research, we try to estimate the parameters to model the human body using the Kinect depth information.

## 1. Introduction

3D model of human is required by various digital applications such as animation in movies and games, or motion analysis for medical purposes or even sport activities. This often requires using commercial 3D scanners which tend to be highly expensive, and limits the 3D scanning in terms of its uses and place. Moreover, because it is almost impossible for a person to not move during scanning process, 3D modeling methods for non-rigid items often result in errors. Various topics related to human motion capture is an actively researched topic and ability to do so will open ways to various other applications in the field of animations, surveillances, fashion, health-care,etc.

Generally speaking, capturing the shape of an individual stands as a problem mainly due to the non-rigid nature of the body. In addition, in any single image, only half of the body is visible, i.e. if a person stands straight facing the camera, information from the back part is missing. Other than that, various poses results in self occlusion, i.e. a part of body is hidden due to occlusion from some other parts. Because the human body is non-rigid, even if the person tries not to move, the appearance changes between two frames, Needless, to mention that human appearances changes during different forms of movements. Moreover, due to clothing it becomes very difficult to accurately estimate human body shape.

Human body modeling often makes use of the model, in which, various rigid body parts are linked in the kinematic chain. For example, to move an arm a total of six degrees of freedom must be controlled. (three degrees of freedom at the shoulder, one at both the elbow and and two at the wrist joint). This is just considering the major joints, if we go on considering the semi movable joints and muscles then the degree of freedom rises dramatically.

In this paper, we propose a data driven method to model a human body using the depth camera. Our system is capable of performing a full body scan in a natural environment. We use the SCAPE model which is a 3D body model which accounts for changes in pose and variation in shape between humans. SCAPE model is learnt from database of scans of various human bodies. Since the whole set of SCAPE database is not available, we use 3D scans provided by MPI Informatik [1] in order to learn the deformation model.

## 2. Related Works

High cost scanners such as [2] provide a high quality and accurate 3D models of human body. But these scanners tend to be highly expensive costing hundreds of thousands of dollars. For example, the Cyberware body scanner costs $ 240,000 [2].

Before the availability of commercial depth cameras, multiple synchronized cameras were often used to estimate the 3D model [3], [4]. These works use silhouettes from a multi-view image sequence to recover the skeleton and the 3D surface. This kind of system cannot be realised in natural environment. Moreover, it requires knowledge to create the setup and do camera synchronization. We can overcome this using commercial depth cameras, however only partial view of a person is provided and makes it difficult to create a 360° model.

With the commercial 3D cameras being available, more researches [5], [6] have used 3D cameras for scanning of human bodies.[5] uses multiple Kinect cameras to capture the human body. A special set up is required so that the depth cameras don't interfere each other. Non-rigid registration is used to bring the partly scanned body parts into alignment and finally, create a whole 3D human model. The system is still expensive compared to using a single depth camera and a special setup is requires which makes it troublesome for general people.

Our approach is very similar to [6]'s work. However, they make use SCAPE model as us. However, we go a step further to provide a direct control over these parameters. In their approach it is very difficult to change a attribute without changing

[1] University of Tokyo,Department of Information Science and Technology
[2] KDDI R&D Laboratories, Inc
[a)] nima@hal.t.u-tokyo.ac.jp

some other attributes. For example, changing height will incur a change in body weight. We try to solve for this in our work.

## 3. An overview of SCAPE

We borrow concept of deformable body model called SCAPE(Shape Completion and Animation of PEople) that was proposed back in 2005[7]. SCAPE models human body in two spaces. The pose space and the shape space. The pose deformation model captures the changes in the body as a function of pose. For example, the orientation of legs changes when a person walks. The beauty of this model lies in the fact that it captures the non rigid deformation arising due to the change in the pose of a person. For example, muscles bulging in the biceps when a arm is bent, or the changes in the underarms during shoulder movement. These kind of changes are captured under pose deformation model. The shape deformation model captures changes in the shape such as height and weight across different individuals.

### 3.1 Model Overview

The objective is to model the deformations that transforms template mesh $X$ to example meshes $Y$ present in the pose and shape data set. The SCAPE model uses triangle based deformation [8] rather than using the direct vertex based deformation. In the SCAPE model triangle deformations are given by sequence of linear transformations

( 1 ) Rigid transformations R resulting from change in pose, such as change in orientation

( 2 ) Non-rigid transformations Q resulting from change in pose, such as bulging of muscles

( 3 ) Non-rigid transformations S resulting from change in shape between different subjects

For a given triangle $t$ of the source mesh $X$ containing the vertices $(x_1, x_2, x_3)$, which corresponds to triangle containing the points $y_1, y_2, y_3$ we find a $3 \times 3$ deformation matrix $\mathbf{M}_t$ such that

$$\mathbf{M}_t(\vec{x}_{t,k}) = \vec{y}_{t,k}, \qquad k = 2, 3 \tag{1}$$

where $\vec{x}_{t,k}$ represents the edge vector $\vec{x}_{t,k} = x_{t,k} - x_{t,1}$.

Since, $\mathbf{M}$ is given by sequence of linear transformations, we can write

$$\mathbf{M}_t = \mathbf{R}_{p[t]}\mathbf{S}_t\mathbf{Q}_t \tag{2}$$

$p[t]$ specifies the body part to which a particular triangle $t$ belongs to. Matrix $\mathbf{R}_{p[t]}$ is body part specific, i.e. all the triangles belonging to same body part have same $R$.

### 3.2 Creating a new mesh

Once the rigid rotation for all body parts $R$, and non-rigid deformation matrices $Q$ and $S$ are estimated, these matrices can be used to recover the example mesh using the template mesh. Similarly, for any given $R, Q$ and $S$ a new mesh can be created using template mesh, such as

$$\arg\min_{y_1,\cdots,y_N} \sum_t \sum_{j=2,3} ||\mathbf{R}_{p[t]}\mathbf{S}_t\mathbf{Q}_t(\vec{x}_{j,k} - \vec{y}_{1,k})||^2 \tag{3}$$

This optimization process estimates the best position for the vertices such that resulting mesh will have consistent edge vectors. Since all the deformations are local, global translational degree of freedom remains over all triangles.

### 3.3 Rigid Deformation due to pose

Human body can be modeled as an kinematic chain of articulated body parts. We assume that during a change in pose, these body parts rotate freely around their joints. This causes the triangles in the body part to undergo changes in orientation. Hence, we need to solve for the rotation matrix for each body part. Since, the correspondences between the vertices in template mesh and example mesh is known, following equation can be used to solve for the rotation matrix, partwise.

$$\arg\min_{\vec{y}_1,\cdots,\vec{y}_N} \sum_t \sum_{2,3} ||\mathbf{R}_{p[t]}\vec{x}_{t,k} - \vec{y}_{t,k}||^2 \tag{4}$$

Rotation matrix in itself is not enough to capture the deformation caused due to the change in pose. We need the non-rigid deformation to account for the deformation such as changes in underarms or muscle bulging, etc.

### 3.4 Non-Rigid Deformation due to pose

Estimating the non-rigid deformation induced due to pose consists of two steps. First, non-rigid deformations are learnt from the pose dataset. Secondly, we try to model the non-rigid deformation as a linear function of $\mathbf{R}$.

We use the pose dataset to train the non-rigid pose deformation model. All the models in the set belongs to the same person, hence the triangle deformations are induced due to the change in the pose and the shape factor can be left out. A smoothness constraint is added which requires that the neighbouring triangles undergo similar deformation. We can estimate $mathbf Q_t^i$ for each mesh $Y^i$ in the pose dataset using least squares solving techniques and solve for all the $\mathbf{Q}_t$ in a body part at once.

$$\arg\min_{\mathbf{Q}_1,\cdots\mathbf{Q}_T} \sum_t^T \sum_{j=2,3} ||\mathbf{R}_{p[t]}\mathbf{Q}_t\vec{x}_{t,j} - \vec{y}_{t,j}||^2 \tag{5}$$

This concludes the first part, learning the non-rigid deformation for all training data sets. Next, we assume that the non-rigid deformation can be expressed as linear function of rigid-deformations $\mathbf{R}$. Our objective is to predict $\mathbf{Q}$s from the pose, or from the rigid rotations more specifically. Given any new pose(not present in the training data set), the model should be able to predict the deformation matrices $\mathbf{Q}_k$.

We learn a linear regression function $a_k$ for each triangle which predicts the transformation matrices $\mathbf{Q}_k$ as a function of the rotations at the nearest joints.

Let $N_{p[t]}$ be the list of body parts connected to a particular body part. Then, we calculate relative joint rotation $\Delta\mathbf{R}_{(p[t],c)}$ for each joint between body part $p[t]$ and $c, c \epsilon N_{p[t]}$, using the absolute body part rotation $\mathbf{R}_{p[t]}$ and $\mathbf{R}_c$ as below.

$$\Delta\mathbf{R}_{p[t],c} = \mathbf{R}_{p[t]}^T\mathbf{R}_c \tag{6}$$

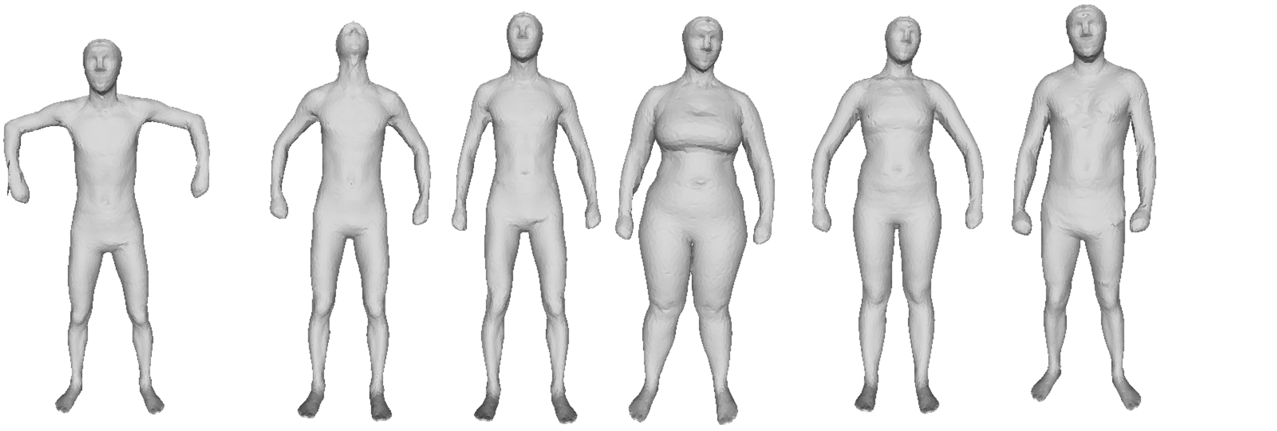

**Fig. 1** Training Data. Two from pose data set, Template mesh, three from shape dataset,two female and a male (left to right)

Joint rotation can be represented using exponential map coordinates [9]. Let **M** denote any $3 \times 3$ rotation matrix, then it can be expressed in 3D vector form computed from the following formula.

$$t = \frac{\theta}{2sin\theta}\begin{bmatrix} m_{32} - m_{23} \\ m_{13} - m_{31} \\ m_{21} - m_{12} \end{bmatrix}, \quad \theta = cos^{-1}\left(\frac{tr(M) - 1}{2}\right). \quad (7)$$

where $t$ represents the axis of rotation, and $\theta$ represents the magnitude of rotation. We create a column vector $\Delta r_{p[t]}$ such that all the adjacent joint rotations are pushed into the vector. $\Delta r_{p[t],c}$ is the representation of $\Delta R_{p[t],c}$ in the exponential map coordinate system.

Let $q_{t,ij}$, $i, j = 1, 2, 3$ represent the nine elements of matrix $\mathbf{Q}_t$. We express $q_{t,ij}$ as linear function of the relative rotation of the adjacent joints.

$$q_{t,ij} = a^T_{t,ij}\begin{bmatrix} \Delta r_{p[t]} \\ 1 \end{bmatrix}, \quad i, j = 1, 2, 3 \quad (8)$$

$a^T_{t,ij}$ is a $7 \times 1$ regression vector, and a constant term is added to $\Delta r_{p[k]}$ . Without this term, zero change in pose will assign null matrix to **Q**.

Now, we learn the regression vector $a_{t,ij}$ from the **Q**s learnt from the pose dataset and solving for the following least squares problem.

$$\arg\min_{a^i_{t,ij}} \sum_i \left( {a^i_{t,ij}}^T \begin{bmatrix} \Delta r_{p[t]} \\ 1 \end{bmatrix} - q^i_{t,ij} \right)^2 \quad (9)$$

Given an arbitrary pose whose rotation matrix is given by $\mathbf{R}^{new}_{p[t]}$, non-rigid deformation per triangle can be computed using Eq. 8.

### 3.5 Non-Rigid Deformation due to shape

The deformation matrix responsible for change in shape is represented as $\mathbf{S}_t$. In order to calculate the non-rigid deformation due to shape, we first calculate the deformations induced due to change in pose. Then, we try to estimate the deformation matrix accounting for the remaining deformation. After calculating the shape deformation matrix for each body in the shape data set , we sought to create a low dimensional model that accounts for the variations in the body shape.

Similar to the calculation of the non-rigid deformation matrix $Q_t$, we calculate the deformation matrix $S_t$.

$$\arg\min_{\mathbf{S}_1,\cdots\mathbf{S}_T} \sum_t^T \sum_{j=2,3} ||\mathbf{R}_{p[t]}\mathbf{S}_t\mathbf{Q}_t\vec{x}_{t,j} - \vec{y}_{t,j}||^2 \quad (10)$$

Next, we create a low dimensional model that captures the variation in body shapes among different individual. For every mesh in the shape data set $Y^j$, we create a vector of size $9 \times N$ where $N$ is the total number of triangles in a mesh. The body shape deformations $S_t$ for all the triangles is concatenated into this single column vector $\vec{s}^j$, and a new matrix is created such that the concatenated vector becomes the column vector of matrix. $\mathbf{S}^{shape} = \left[\cdots, \vec{s}^j, \cdots\right]$. Our mesh contains 12,000 triangles, the body shape is specified using 108,000 parameters. We use PCA to find a reduced dimension subspace that is able to characterize the variations in the body shapes.

$$\hat{\vec{s}} = \mathbf{U}\vec{\beta}^j + \vec{\mu} \quad (11)$$

Here, $\mu$ is the mean of the shape deformation, the columns of the **U** are the principal components given by PCA. The $\beta$ represents the PCA coefficients.

## 4. System Overview

The objective of our system is to capture the shape(height,weight,etc) and pose(standing,sitting,etc) of a person in the depth map. Commercial depth camera is used to capture the subject. We use a parametric body model to represent the human body. Furthermore, available tracking algorithms are used to estimate the parameters required to deform the template model. The objective of our system is to capture the variation in body shapes and is unable to capture the clothing. Moreover, the subject is required to wear tight fittings in order to accurately estimate the parameters.

## 5. Sensor

We use Microsoft Kinect [10] to capture the input data. However, models created using laser scanners are used to create the template model. The Kinect consists of an IR camera, a RGB camera and a IR laser projector that projects specific patterns of laser. It captures a 2D color image and 3D depth map at the rate of 30fps. The images and depth data can be captured using official MicrosoftSDK or other available libraries such as OpenNI[11]. These cameras can be operated at natural environments. And in general performance is not affected by lightning conditions.

## 6. Fitting the model to depth input

The SCAPE is a parametric model, described by a set of pose parameters $\theta$ which accounts for the orientation of the body parts, global position of body parts, shape coefficients $\vec{\beta}$. Our objective is to find a way to estimate these parameters from a given depth input.

### 6.1 Estimation of the pose deformation parameters

We make use of the generally available algorithms for pose tracking such as the one from the Microsoft SDK[12]. We make no rigorous claims about these tracking algorithms. We understand that these algorithms have limitations and may not be able to give accurate results for poses that incurs self occlusions. From the standard tracking algorithms we can track 14 joint positions Fig.2. We need to pay attention that these joints may not necessarily refer to anatomical joints. For example there is no joint at the centre of the torso or at head. These joints specify the centre of mass for such body parts. Moreover, the joints available from these tracking algorithms are not identical to the joints present in our template mesh or training dataset. However, this is not much of a problem for two main reasons:

(a) We are just interested in orientation of body parts and not the position of joints in itself.

(b) These joints can be used indirectly to infer the orientation of the joints in the template mesh.

Now we try to find the local rigid rotation such that input body parts would align with the template mesh.

### 6.2 Orientations

OpenNI and MicrosoftSDK1.5(and further) provided the joint orientation information for the joints tracked by the tracking algorithm. The joints orientations provided by OpenNI are global orientations relative to the standard T-pose, i.e. if the user stood in an ideal T-pose , all joint orientations will be identity matrices Fig.2. To be even more precise about how the arm is oriented in this ideal T-pose, the upper arm should be twisted in a way so that if the elbow is flexed, the lower arm should bend forwards towards sensor. Nevertheless, it should be noted that OpenNI has certain limitations to track the joint orientations. For example, when a person stands in a standard T pose, and rolls the arms, it would be difficult to notice the movements. Furthermore, the depth value at the joints would not change much, therefore OpenNI cannot distinguish such movements. Moreover, orientation at endpoints(head,wrists,ankles) of the articulated chain are not available. Therefore, in our implementation we assume that the orientation of endpoint is same as that of it's parent. However, the orientation provided by OpenNI cannot be used directly in our model. We need to find the relative rotation from template pose to the given pose. We capture a pose that is similar to that of template. While it is difficult to capture a pose that is exactly the same as the template pose, we choose a pose that minimizes the error. We find the inverse of the orientation for the pose that is similar to the template pose. Using this orientation, we compute the relative rotation from this captured template pose to any given pose.

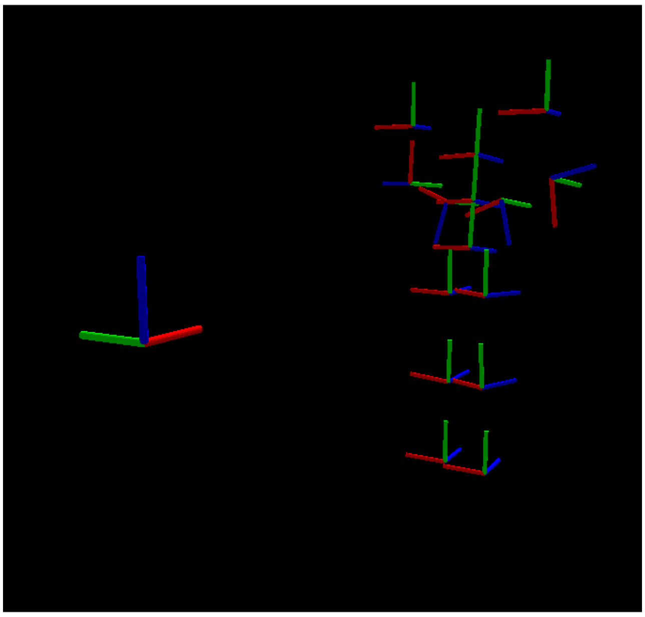

**Fig. 2** Joint Orientation from OpenNI

The relative rotation quaternion gives the parameters that acounts for the orientation of the body parts. Now, we have the relative orientation for each body part, we can use the regression vector learnt from the pose data set to infer **Q**s for the given pose using Eq 9.

### 6.3 Estimation of shape deformation parameters

Our objective is to estimate the $n$ PCA coefficients, so that the shape of the subject can be estimated using the principal components of the shape deformation matrix. We can simply change the template model by changing the PCA parameters. However, it is difficult to direct the change in only one direction. For example, the first PCA parameter changes a captures axis of variation from small to big, thin to fat, the second PCA parameter captures the variation in height, etc. The modification of a single PCA parameter will simultaneously modify other features that are naturally plausible. Such as change in height may lead to change in gender. So, even though principal component analysis helps to characterize the space of human body variation, it doesn't provide with a direct method to control specific variation. For example, changing a height of a person will change weight of a person. Therefore, we discuss a direct way to control the characteristics of body shape with specific parameters. We follow a method similar to [13]. Fortunately, the database provided to us [1] also contains the attributes for each scan such as height, weight, breast girth, waist girth, hips girth, leg length and other data for each scan. By learning a linear mapping we can control the PCA parameters and the physical attributes, we can get a greater control over the induced changes.

We create a linear mapping between these two spaces. The linear mapping can be represented by $(k) \times (n + 1)$ matrix $\mathbf{L}$ where $k$ is the number of PCA parameters and $n$ is the number of physical attributes, 7 in our database.

$$\mathbf{L}[f_1 \quad f_2 \quad \cdots \quad f_n \quad 1]^T = b \tag{12}$$

where $f_n$ are tha values of physical attributes, $b$ are the corresponding PCA coefficients.

$$\mathbf{L} = \mathbf{B}\mathbf{F}^{+} \tag{13}$$

$\mathbf{F}$ is the matrix whose columns contain physical attributes of each subject. Similarly, $\mathbf{B}$ contains PCA coefficients of each subject as its columns. $\mathbf{F}^{+}$ is the pseudoinverse of $\mathbf{F}$. Now, we can create an average person with given set of characteristics as shown in Fig.3.

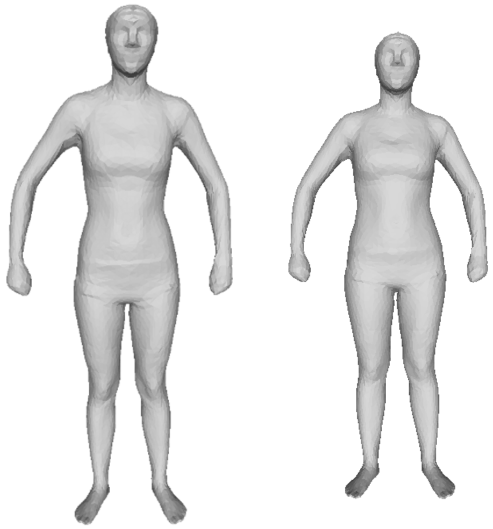

**Fig. 3** Both are creates using same weight but different height,difference in height is 20cm

Moreover, we can specify amount of change to any given model as

$$\Delta f = [\Delta f_1 \quad \Delta f_2 \quad \cdots \quad \Delta f_n \quad 0]^T \tag{14}$$

where $\Delta f_n$ is the targeted amount of change in an individual. For example, if we want to increase the height of a person by 10cm then $\Delta f_{height} = +10$ and so on. The $\mathbf{L}$ matrix can be used to calculate the resulting change in the PCA weight and can be added to the PCA weights. We can change the attributes of the individual such as increase in height, decrease in waist girth,etc. Fig**??**

$$\Delta b = \mathbf{M}\Delta\mathbf{f} \tag{15}$$

In our implementation, we produce an initial mesh with given weight and height. Then, we make further changes to the mesh depending on other attributes.

### 6.4 Measurements with Kinect

We use depth map from Kinect to infer the body measurements of the person. For example, we can get a a rough estimation of a person's height by taking the difference between the top pixel and bottom pixel of the user. We understand that there exists error in such estimation. However, for initial estimation this is enough. Similarly, we take the waist measurement of the subject. Unlike measurement of height, waist involves taking two different poses. Once, the subject is facing the camera and when the user faces his back towards the camera. This gives us an estimate of waist of the subject. Using these measurements, we create our mesh.

## 7. Results

We make a person stand in front of the depth camera such that the whole body, i.e. from head to foot is visible. We estimate the height of the person and make the person turn and then calculate the waist circumference. Since, the person has moved the pose of the person changes, i.e. position of the waist has changed. Nevertheless, we use this measure to create the mesh. Fitting the pose of the subject to the template mesh is shown in left of Fig. 4. Similarly, the right shows the mesh constructed from the pose and shape of the subject. We assume that the shape of a person does not change. Therefore, we make an estimate of shape only once. Later, the shape parameters are re-used while constructing new meshes for same person.

### Accuracy

Since we don't have ground truth to test the accuracy, we test for the accuracy of the given mesh, by comparing it with the RGB image provided by the Kinect. We overlap the produced new mesh in the image and see how accurate is the testing. Moreover, the quality of mesh itself is an indicator. The areas around the joint has not undergone smooth deformation, which will be focused in our future work.

## 8. Conclusion and Future Work

We have presented a method for estimating 3D human shapes using the depth cameras and existing motion tracking algorithm. However, spaces for further improvement lies in our work. First of all , the measurements we use from Kinect are not correct. For now, we just use the the height and waist measurements because this is easier to infer from standard tracking algorithms. For more accurate capture, other measurements such as breast circumference, hip circumference, etc. can be used. Since, the laser scans already provide us with these data, an estimate of these attributes will lead to more accurate mesh.

Secondly, after an initial estimate of a body shape, a more accurate body shape can be achieved by a direct comparison between

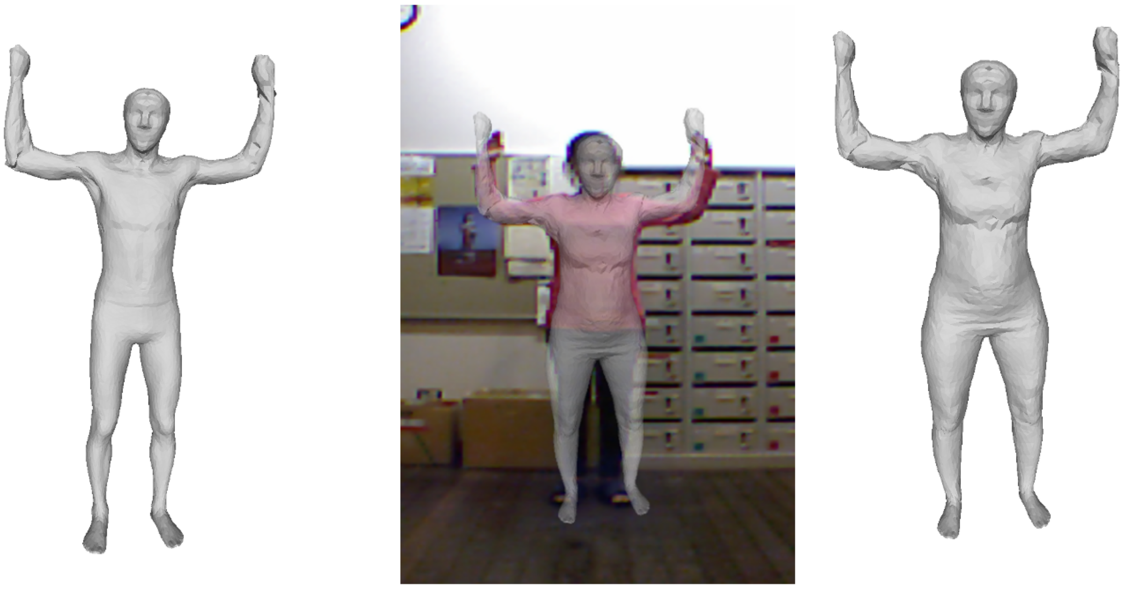

**Fig. 4** Comparision with the image and output mesh

the point cloud input from depth cameras and the mesh. For this, an optimization will be necessary which decreases the difference between the point cloud and the mesh. Our future work should address the problem.

We hope that realization of such a system will open doors to many applications in different fields such as in animation, games, or virtual dressing rooms. Our system can be used at normal environments. Moreover, compared to previous work, our approach strikingly decreases the number of parameters (to be estimated), which makes our system computationally cheap. Hence, for most of the part our system can be realised closer to real time.

## References

[1] Hasler, N., Stoll, C., Sunkel, M., Rosenhahn, B. and Seidel, H.-P.: A Statistical Model of Human Pose and Body Shape, *Computer Graphics Forum (Proc. Eurographics 2008)* (Dutr'e, P. and Stamminger, M., eds.), Vol. 2, No. 28, Munich, Germany (2009).

[2] Cyberware: scanners, Cyberware (online), available from ⟨http://www.cyberware.com/pricingdomesticPriceList.html⟩ (accessed 2013-01-23).

[3] Balan, A., Sigal, L., Black, M., Davis, J. and Haussecker, H.: Detailed human shape and pose from images, *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, IEEE, pp. 1–8 (2007).

[4] Gall, J., Stoll, C., De Aguiar, E., Theobalt, C., Rosenhahn, B. and Seidel, H.: Motion capture using joint skeleton tracking and surface estimation, *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, IEEE, pp. 1746–1753 (2009).

[5] Tong, J., Zhou, J., Liu, L., Pan, Z. and Yan, H.: Scanning 3d full human bodies using kinects, *Visualization and Computer Graphics, IEEE Transactions on*, Vol. 18, No. 4, pp. 643–650 (2012).

[6] Weiss, A., Hirshberg, D. and Black, M.: Home 3D body scans from noisy image and range data, *Computer Vision (ICCV), 2011 IEEE International Conference on*, IEEE, pp. 1951–1958 (2011).

[7] Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J. and Davis, J.: Scape: shape completion and animation of people, *ACM Transactions on Graphics (TOG)*, Vol. 24, No. 3, ACM, pp. 408–416 (2005).

[8] Sumner, R. and Popović, J.: Deformation transfer for triangle meshes, *ACM Transactions on Graphics (TOG)*, Vol. 23, No. 3, ACM, pp. 399–405 (2004).

[9] Ma, Y., Soatto, S., Kosecka, J. and Sastry, S. S.: *An Invitation to 3-D Vision: From Images to Geometric Models*, SpringerVerlag (2003).

[10] : Microsoft Corp (online), available from ⟨http://www.xbox.com/kinect⟩ (accessed ).

[11] OpenNI: *OpenNI User Guide*.

[12] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A. and Blake, A.: Real-time human pose recognition in parts from single depth images, *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, pp. 1297–1304 (2011).

[13] Allen, B., Curless, B. and Popović, Z.: The space of human body shapes: reconstruction and parameterization from range scans, *ACM Transactions on Graphics (TOG)*, Vol. 22, No. 3, ACM, pp. 587–594 (2003).