Tensor-Based Human Body Modeling

Yinpeng Chen Zicheng Liu Zhengyou Zhang
Microsoft Research, Redmond, WA, USA
{yiche, zliu, zhang}@microsoft.com

Abstract

In this paper, we present a novel approach to model 3D human body with variations on both human shape and pose, by exploring a tensor decomposition technique. 3D human body modeling is important for 3D reconstruction and animation of realistic human body, which can be widely used in Tele-presence and video game applications. It is challenging due to a wide range of shape variations over different people and poses. The existing SCAPE model [4] is popular in computer vision for modeling 3D human body. However, it considers shape and pose deformations separately, which is not accurate since pose deformation is person-dependent. Our tensor-based model addresses this issue by jointly modeling shape and pose deformations. Experimental results demonstrate that our tensor-based model outperforms the SCAPE model quite significantly. We also apply our model to capture human body using Microsoft Kinect sensors with excellent results.

1. Introduction

3D human body modeling has numerous applications in Computer Vision, Graphics, and Multimedia. It plays an important role in tracking, reconstructing and animating human body. It provides users with immersive experience in Tele-presence and video game applications. The problem is challenging because the variation in 3D human body geometry (over different people and poses) is a complicated function over multiple shape and pose variables.

Among the early work in human body modeling [2, 13, 4, 3, 10], the SCAPE model [4] has been widely used in estimating human shape and pose as well as in reshaping human body in images and videos. It learns a pose deformation model from a subject with multiple poses and learns a shape model from many subjects with a neutral pose. However, the decoupling of shape and pose deformations in the SCAPE model has a major limitation - 3D meshes of different individuals change in the similar manner for the same pose change. Figure 1 demonstrates this problem. We fit the SCAPE model on a female subject at the neutral pose to estimate her shape parameters. Then the fitted SCAPE model is used to predict two poses (pose 1, 2 in the left column). The chest and shoulder areas in the predicted poses look far away from the ground truth (shown in the middle column), since the pose model in SCAPE is learned from a muscular male subject.

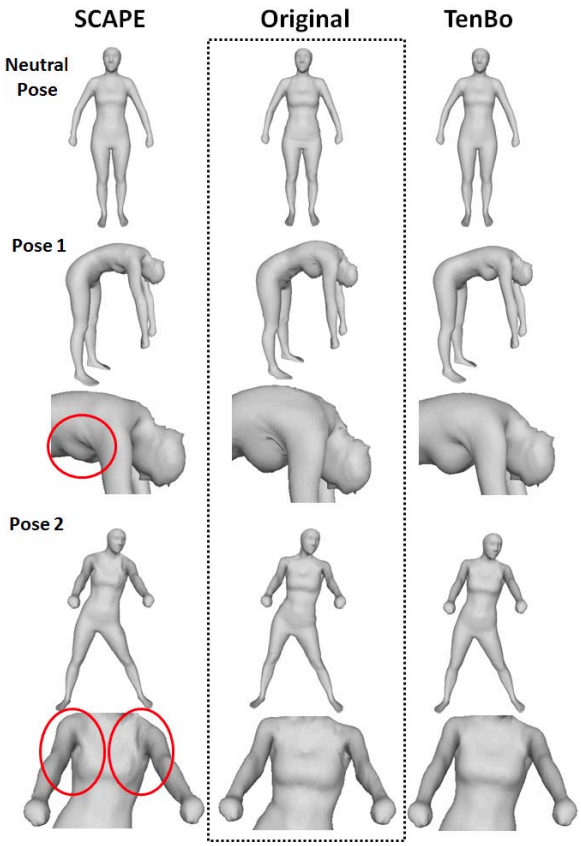

Figure 1. Comparison between the SCAPE model (left column) and the proposed TenBo model (right column). The ground truth is shown in the middle column. The shape parameters for both SCAPE and TenBo models are estimated from the original neutral pose data. They are then used to predict the 3D geometry under different poses. Two poses are shown in the figure, with close-ups to highlight the differences, especially in those circled areas.

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.21

In this paper, we propose a TENsor-based human BOdy model (abbreviated as *TenBo*) to address this problem. Using the tensor decomposition technique, we model the deformation as a *joint* function over both shape and pose parameters to preserve the dependency between them. Compared with SCAPE, TenBo model effectively leverages training data from multiple people under multiple poses to improve accuracy. Figure 1 (right column) clearly shows that the predicted poses based on the TenBo model is much closer to the ground truth. Experimental results show that our TenBo model outperforms the SCAPE model in predicting novel poses with 16.6% of error reduction. We also apply our TenBo model to capture human body using Microsoft Kinect sensors with excellent results.

This paper is organized as follows. We firstly discuss related work and overview our TenBo model. Then, we introduce mesh deformation definition and describe our TenBo model in details. The training of the TenBo model and fitting to the point cloud will be discussed next. Finally, we show experimental results and conclude this paper.

2. Related Work

3D Human Body Models: The early human body models, including [2] and [13], focused on modeling the shape variations in similar poses. Then [4, 3, 10] integrated both shape and pose variations to allow 3D animations for different people under different poses. Allen [3] used maximum posteriori estimation to learn a correlated model of identity and pose-dependent body shape variation. The optimization procedure is expensive due to the solving of nonlinear functions with high number of degrees of freedom. Hasler [10] used differential encoding to express shape/pose variations and used regression to incorporate shape and pose parameters. The SCAPE model [4] is a widely used model which decouples shape and pose deformations. However, due to the decoupling, the pose deformation model is shared by all individuals, i.e. the 3D meshes of different people change in the similar manner for the same pose change, which is not correct. In this paper, we address this problem by *jointly* modeling the shape and pose deformations to preserve their dependency.

Human Shape and Pose Estimation: The SCAPE model [4] has been used widely in human shape and pose estimation [8, 5, 6, 17, 20]. Guan [8] estimated human shape and pose from a single image using shading information. Balan [5] estimated human body shape under clothing. Weiss [17] scaned 3D human body from noisy image and range data by using silhouette objective. Freifeld [6] developed a 2D contour person model. In [9], Hasler presented a bilinear model of shape and pose to estimate 3D meshes of dressed subjects from images.

Other Applications: Guan [7] developed a DRAPE model to learn a deformable clothing model that uses SCAPE to represent underlying naked body. [19] and [11] allow users to reshape human bodies in images and videos.

Tensor Faces: Tensor based approaches have been successfully applied to face modeling [15, 16], which motivated us to extend them to human body modeling. Compared with TensorFaces, our TenBo model has less requirement on training data. TenBo allows each subject to perform only a small subset of the poses rather than the full set and allows large variations among different subjects to perform the same pose, while TensorFaces requires the same capture configuration for all subjects (e.g. the same expressions for all the subjects).

3. Overview

Each 3D human body mesh can be considered as a deformation from a reference mesh. Our TenBo model considers the deformation D as a joint function $D(\boldsymbol{v}, \boldsymbol{\theta})$ over shape parameters $\boldsymbol{v}$ and pose parameters $\boldsymbol{\theta}$ to integrate shape deformation (due to different persons) and pose deformation (due to different poses) using tensor technique. Compared with the SCAPE model, which separates the shape deformation $S(\boldsymbol{v})$ and the pose deformation $Q(\boldsymbol{\theta})$ as $D = S(\boldsymbol{v})Q(\boldsymbol{\theta})$, our TenBo model is able to preserve the dependency between the shape and pose deformations.

In addition, our TenBo model uses the training data (3D meshes from multiple subjects under different poses) more effective than the SCAPE model. The SCAPE model *only* uses one subject (with multiple poses) to train the pose model and *only* uses one pose (from multiple persons) to train the shape model. In comparison, our TenBo model uses multiple poses from multiple subjects to combine the shape and pose deformations together. Table 1 lists the difference between SCAPE and TenBo. Note that training the TenBo model only requires a few poses per subject rather than the full set of poses. It also allows large variations among different subjects to perform the same pose. This significantly simplifies the data capture process.

4. Mesh Deformation Definition

We use the same mesh deformation definition as in the SCAPE model [4]. Let us denote the reference mesh (i.e. a natural standing pose for the person who is selected as the reference subject) as $X = \{V_X, P\}$. X includes M vertices $V_X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M\}$ and N triangles $P = \{p_1, \ldots, p_N\}$. In this paper, we assume all meshes are registered (i.e. all meshes share the same triangulation P and number of vertices M, and the vertex correspondence is known).

The deformation for an arbitrary 3D body mesh $Y = \{V_Y, P\}$ indicates the difference between the mesh Y and the reference mesh X. The deformation is defined at the triangle level, e.g. three vertices $\boldsymbol{x}_{n,1}$, $\boldsymbol{x}_{n,2}$, $\boldsymbol{x}_{n,3}$ of the triangle p_n on the reference mesh X is deformed to $\boldsymbol{y}_{n,1}$, $\boldsymbol{y}_{n,2}$,

	SCAPE Model	TenBo Model
Model Difference	Consider shape and pose deformations $separately$	Consider shape and pose deformations $jointly$
Training	Use multiple poses from one subject and a neutral pose from multiple subjects	Use multiple poses from multiple subjects
Pose Deformation	Train on one subject	Train on $multiple$ subjects
Shape Deformation	Train on one pose	Train on $multiple$ poses

Table 1. Comparison between the SCAPE model and our TenBo model.

$\boldsymbol{y}_{n,3}$ on the mesh Y. Each vertex is a 3×1 vector including 3D coordinates. The deformation is applied in terms of the triangle's local coordinate system, by using the first vertex ($\boldsymbol{x}_{n,1}$ and $\boldsymbol{y}_{n,1}$) as the origin. The deformation of triangle p_n from X to Y is represented as the linear transformation of two edges ($\Delta\boldsymbol{x}_{n,1} = \boldsymbol{x}_{n,2} - \boldsymbol{x}_{n,1}, \Delta\boldsymbol{x}_{n,2} = \boldsymbol{x}_{n,3} - \boldsymbol{x}_{n,1}$) as follows:

$$\Delta\boldsymbol{y}_{n,q} = \boldsymbol{y}_{n,q+1} - \boldsymbol{y}_{n,1} = \boldsymbol{R}_{l[n]}\boldsymbol{D}_n\Delta\boldsymbol{x}_{n,q}, \quad q = 1, 2, \quad (1)$$

where $\boldsymbol{R}_{l[n]}$ is the rotation matrix (from X to Y) for the body segment $l[n]$ (e.g. torso, upper arm) that includes the triangle p_n, $\boldsymbol{D}_n$ is the non-rigid deformation matrix for the triangle p_n. Both $\boldsymbol{R}_{l[n]}$ and $\boldsymbol{D}_n$ are 3×3 matrices. Rotation matrices $\{\boldsymbol{R}_{l[n]}\}$ for all body segments determine the human *pose*, and deformation matrices $\{\boldsymbol{D}_n\}$ for all triangles determine the human *shape*. Note that $\boldsymbol{R}_{l[n]}$ is shared by all triangles belonging to the body segment $l[n]$. The calculation of $\boldsymbol{R}_{l[n]}$ and $\boldsymbol{D}_n$ for a given mesh Y and the 3D reconstruction of mesh Y using $\boldsymbol{R}_{l[n]}$ and $\boldsymbol{D}_n$ has been solved in the SCAPE model [4]. In this paper, we focus on modeling the non-rigid deformation matrices $\boldsymbol{D}_n$, which is the major difference between this paper and SCAPE.

5. Tensor-based Human Body Model (TenBo)

Our TenBo model includes two parts - (a) model for an individual body segment (e.g. forearm, upper arm), and (b) model for the whole body. We first introduce a tensor-based method to model the deformation of an individual body segment. Then we will discuss how to integrate *local shape vector* (refer to shape parameters for a body segment) into *global shape vector* (refer to shape parameters for the whole body). We use $\boldsymbol{s}_l$ to denote the local shape vector on the l^{th} body segment and use $\boldsymbol{v}$ to denote the global shape vector. We assume that a human body has L segments, and denote the number of triangles on the l^{th} segment as n_l ($\sum_{l=1}^{L} n_l = N$).

5.1. Model for an Individual Body Segment

We model the deformation for each body segment as a *joint* function over both shape and pose parameters using tensor technique. We rearrange the deformation matrix $\boldsymbol{D}_n$ column by column as a 9×1 vector for every triangle on the l^{th} segment (including n_l triangles) and group all vectors as a $1 \times 1 \times 9n_l$ tensor, denoted as $\boldsymbol{d}_l$. The segment deformation $\boldsymbol{d}_l$ is modeled as a joint function over the local shape vector $\boldsymbol{s}_l$ ($I_t \times 1$) and the joint angle vector $\boldsymbol{\theta}_l$ ($J_t \times 1$) as follows:

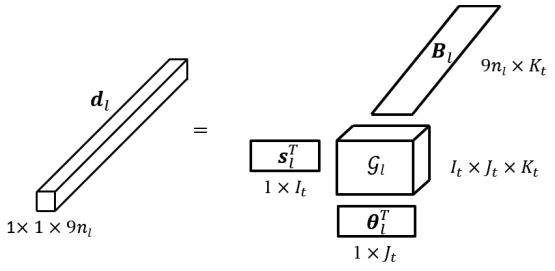

Figure 2. Tensor decomposition for the deformation of a body segment .

$$\boldsymbol{d}_l(\boldsymbol{s}_l, \boldsymbol{\theta}_l) = \mathcal{G}_l \times_1 \boldsymbol{s}_l^T \times_2 \boldsymbol{\theta}_l^T \times_3 \boldsymbol{B}_l \quad (2)$$

where $\mathcal{G}_l$ is a tensor core ($I_t \times J_t \times K_t$), $\boldsymbol{B}_l$ is a deformation basis matrix ($9n_l \times K_t$), and $\times_n$ refers to the n-mode multiplication [12]. The local shape vector $\boldsymbol{s}_l$ encodes the shape of the l^{th} segment using I_t parameters. The joint angle vector $\boldsymbol{\theta}_l$ includes joint angles from the two nearest joints of the l^{th} segment (e.g. wrist and elbow joints for forearm segment) and a constant bias. Each joint has 3 joint angles. Therefore, $\boldsymbol{\theta}_l$ has 6 joint angles and 1 constant ($J_t = 7$). The deformation basis matrix $\boldsymbol{B}_l$ includes K_t deformation bases, which represent the deformation of the l^{th} segment in a low dimensional space. Figure 2 shows the diagram of the tensor model. Essentially, the core $\mathcal{G}_l$ encodes the relationship between the deformation ($\boldsymbol{d}_l$) and the shape/pose parameters ($\boldsymbol{s}_l$, $\boldsymbol{\theta}_l$), where the deformation is represented as a linear combination of deformation-basis vectors:

$$\boldsymbol{d}_l^v(\boldsymbol{s}_l, \boldsymbol{\theta}_l) = \sum_{i=1}^{I_t}\sum_{j=1}^{J_t}\sum_{k=1}^{K_t} \mathcal{G}_l(i,j,k)\boldsymbol{s}_l(i)\boldsymbol{\theta}_l(j)\boldsymbol{B}_l(:,k) \quad (3)$$

where $\boldsymbol{d}_l^v$ is the vector permutation of $\boldsymbol{d}_l$ ($9n_l \times 1$ vector). The weight of basis $\boldsymbol{B}_l(:,k)$ is determined by the core $\mathcal{G}_l$ and the shape/pose parameters ($\boldsymbol{s}_l$, $\boldsymbol{\theta}_l$). Each segment has $I_t J_t K_t$ parameters in $\mathcal{G}_l$ and $9K_t n_l$ parameters in $\boldsymbol{B}_l$, which need to be trained. After training is completed, we can compute the deformation $\boldsymbol{d}_l$ based on the local shape parameters $\boldsymbol{s}_l$ and the joint angles $\boldsymbol{\theta}_l$, which are estimated for a specific person with a specific pose.

5.2. Model for the Whole Body

The local shape vectors $\boldsymbol{s}_l$ from different body segments are highly correlated to the global shape vector $\boldsymbol{v}$ that encodes the shape of the whole body (e.g. tall person has longer arms and legs). We model this correlation using linear transform as:

$$\boldsymbol{s}_l = \boldsymbol{A}_l \boldsymbol{v}, \tag{4}$$

where $\boldsymbol{s}_l$ includes I_t local shape parameters for the l^{th} segment, $\boldsymbol{v}$ includes I_v global shape parameters, and $\boldsymbol{A}_l$ is a transform matrix ($I_t \times I_v$) for the l^{th} segment. By replacing the local shape vector $\boldsymbol{s}_l$ with the global shape vector $\boldsymbol{v}$, eq. (2) can be rewritten as:

$$\boldsymbol{d}_l(\boldsymbol{v}, \boldsymbol{\theta}_l) = \mathcal{G}_l \times_1 \boldsymbol{v}^T \boldsymbol{A}_l^T \times_2 \boldsymbol{\theta}_l^T \times_3 \boldsymbol{B}_l. \tag{5}$$

Combining all body segments, the entire TenBo model has $L(I_t J_t K_t + I_t I_v) + 9K_t N$ parameters ($I_t J_t K_t$ parameters in $\mathcal{G}_l$, $I_t I_v$ parameters in $\boldsymbol{A}_l$, $9K_t n_l$ parameters in $\boldsymbol{B}_l$). TenBo has the same order of complexity as the SCAPE model which has $9(I_v + 7)N$ parameters.

Once we finish training the TenBo model, we can apply it to estimate shape parameters $\boldsymbol{v}$ and pose parameters $\boldsymbol{\theta}_l$ using a 3D point cloud of a human body surface as input. Furthermore, we can generate animations for any subject (assuming shape vector $\boldsymbol{v}$ is available) with different pose sequences. Next, we discuss how to learn the TenBo model.

6. Learning the TenBo Model

The TenBo model is learnt based on a training dataset that includes 3D human body meshes from multiple subjects (each subject has one or multiple poses). We assume that all meshes are registered. Let us denote the number of subjects as I, the total number of poses as J and denote the number of poses for the i^{th} subject as J_i ($\sum_i J_i = J$). The training process includes two steps: preprocessing and optimization.

6.1. Preprocessing

In preprocessing, for every mesh in the training dataset, we compute the rotation matrix $\boldsymbol{R}_l$ and the joint angle vector $\boldsymbol{\theta}_l$ for every body segment, and compute the deformation matrix $\boldsymbol{D}_n$ for every triangle (see calculation details in the SCAPE model [4]). Then, we rearrange $\boldsymbol{D}_n$ to generate segment deformation tensor $\boldsymbol{d}_l$ (a $1 \times 1 \times 9n_l$ tensor). We denote the deformation tensor and the joint angle vector for the j^{th} pose for the i^{th} subject as $\boldsymbol{d}_l^{i,j}$ and $\boldsymbol{\theta}_l^{i,j}$ respectively.

6.2. Optimization

The goal of training is to search for the optimum tensor core $\mathcal{G}_l$, shape transform matrix $\boldsymbol{A}_l$ and deformation basis matrix $\boldsymbol{B}_l$ for every body segment as well as the global shape vector $\boldsymbol{v}_i$ for every training subject to minimize L_2 distance between the actual deformation $\boldsymbol{d}_l^{i,j}$ and the deformation generated by the tensor model (eq. (5)) for all training meshes. The cost function is as follows:

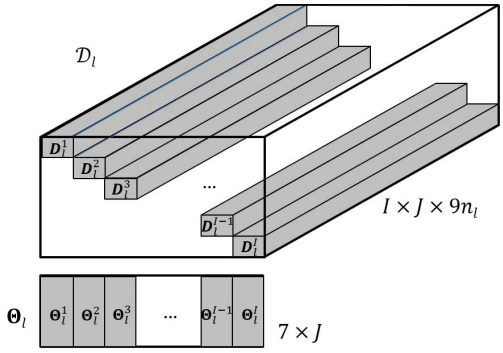

Figure 3. The deformation tensor $\mathcal{D}_l$ and joint angle matrix $\boldsymbol{\Theta}_l$.

$$h(\mathcal{G}_{1..L}, \boldsymbol{A}_{1..L}, \boldsymbol{B}_{1..L}, \boldsymbol{v}_{1..I}) = \sum_{i=1}^{I} \sum_{j=1}^{J_i} \sum_{l=1}^{L} ||\boldsymbol{d}_l^{i,j} - \mathcal{G}_l \times_1 \boldsymbol{v}_i^T \boldsymbol{A}_l^T \times_2 (\boldsymbol{\theta}_l^{i,j})^T \times_3 \boldsymbol{B}_l)||^2, \tag{6}$$

where $\boldsymbol{v}_i$ is the global shape vector for the i^{th} subject. This function can be simplified by the following steps. Firstly, we group the deformation tensors and joint angle vectors of different poses for every subject as $\mathcal{D}_l^i = [\boldsymbol{d}_l^{i,1}, \ldots, \boldsymbol{d}_l^{i,J_i}]$, $\boldsymbol{\Theta}_l^i = [\boldsymbol{\theta}_l^{i,1}, \ldots, \boldsymbol{\theta}_l^{i,J_i}]$. $\mathcal{D}_l^i$ is a $1 \times J_i \times 9n_l$ tensor and $\boldsymbol{\Theta}_l^i$ is a $7 \times J_i$ vector. Then, we combine $\mathcal{D}_l^i$ and $\boldsymbol{\Theta}_l^i$ from all subjects into a deformation tensor $\mathcal{D}_l$ and a joint angle matrix $\boldsymbol{\Theta}_l$ respectively as shown in Figure 3. $\mathcal{D}_l$ is a $I \times J \times 9n_l$ tensor and $\boldsymbol{\Theta}_l$ is a $7 \times J$ matrix, where J is the total number of poses ($\sum_i J_i = J$). Each horizontal slice of $\mathcal{D}_l$ corresponds to a subject, and each lateral slice of $\mathcal{D}_l$ corresponds to a pose. We group $\mathcal{D}_l^i$ along the diagonal direction and set entries off the diagonal as zero. We use a tensor $\mathcal{W}_l$ to indicate the non-zero pattern in $\mathcal{D}_l$. $\mathcal{W}_l(i,j,k)$ equals 1 if $\mathcal{D}_l(i,j,k) \neq 0$, 0 otherwise. Finally, we group the global shape vectors for all subjects as a shape parameter matrix $\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_I]$. $\boldsymbol{V}$ is a $I_v \times I$ matrix (I_v is the number of global shape parameters and I is the number of subjects). Therefore, the cost function eq. (6) can be rewritten as:

$$h(\mathcal{G}_{1..L}, \boldsymbol{A}_{1..L}, \boldsymbol{B}_{1..L}, \boldsymbol{V}) = \sum_{l=1}^{L} ||\mathcal{W}_l * (\mathcal{D}_l - \mathcal{G}_l \times_1 \boldsymbol{V}^T \boldsymbol{A}_l^T \times_2 \boldsymbol{\Theta}_l^T \times_3 \boldsymbol{B}_l)||^2, \tag{7}$$

where '*' indicates the element by element product. In order to remove ambiguity in the deformation basis $\boldsymbol{B}_l$, we add an orthogonal constraint $\boldsymbol{B}_l^T \boldsymbol{B}_l = \boldsymbol{I}$. This optimization problem can be solved either by using a gradient descent based algorithm [1] or by using an iterative process in which each of the four parameters ($\mathcal{G}_l, \boldsymbol{A}_l, \boldsymbol{B}_l, \boldsymbol{V}$) is op-

timized separately, keeping the others fixed. We use the latter approach. $\mathcal{G}_l$, $\boldsymbol{A}_l$ and $\boldsymbol{V}$ can be easily solved since they have quadratic objective without constraint. Solving $\boldsymbol{B}_l$ is a little different due to the orthogonal constraint $\boldsymbol{B}_l^T\boldsymbol{B}_l = \boldsymbol{I}$. It is a standard orthogonal Procrustes problem which can be solved by using singular value decomposition (SVD).

Note that training a TenBo model needs three inputs: dimension of the local shape vector I_t, dimension of the global shape vector I_v and the number of deformation bases K_t. Any change in these values requires additional training process and results in a different TenBo model. However, in practice it is desirable to change the dimension of the global shape parameters I_v without re-training or loading a different model. To address this issue, we introduce principal shape components in the next section.

6.3. Identifying Principal Shape Components

Similar to the principal component analysis (PCA), the principal shape components keep the subspace with minimum cost (see cost function eq. (7)). Extracting principal shape components needs two changes in the optimization.

Firstly, we replace $\boldsymbol{A}_l\boldsymbol{V}$ with the local shape parameter matrix $\boldsymbol{S}_l$ ($\boldsymbol{S}_l = \boldsymbol{A}_l\boldsymbol{V}$) in eq. (7). $\boldsymbol{S}_l$ is an $I_t \times I$ matrix (I_t is the number of local shape parameters and I is the number of subjects in the training dataset). Similar to the optimization in the previous section, we can solve $\mathcal{G}_l, \boldsymbol{S}_l, \boldsymbol{B}_l$ to minimize cost $||\mathcal{W}_l * (\mathcal{D}_l - \mathcal{G}_l \times_1 \boldsymbol{S}_l^T \times_2 \boldsymbol{\Theta}_l^T \times_3 \boldsymbol{B}_l)||^2$ for all body segments.

Secondly, we group $\boldsymbol{S}_l$ for all body segments as $\boldsymbol{M} = [\boldsymbol{S}_1^T, \ldots, \boldsymbol{S}_L^T]^T$ ($\boldsymbol{M}$ is an $LI_t \times I$ matrix) and compute the singular value decomposition of $\boldsymbol{M}$ (i.e. $\boldsymbol{M} = \boldsymbol{U}_M\boldsymbol{\Lambda}_M\boldsymbol{V}_M^T$). Each row of $\boldsymbol{V}_M^T$ includes values of a principal shape component for all subjects. Therefore, we can change the dimension of global shape parameters I_v by selecting the first I_v rows of $\boldsymbol{V}_M^T$ as the global shape parameter matrix $\boldsymbol{V}$. Figure 4 shows the first four principal shape components. The first I_v columns of $\boldsymbol{U}_M\boldsymbol{\Lambda}_M$ includes the transform matrices $\boldsymbol{A}_l$ for all segments (i.e. $\boldsymbol{U}_M\boldsymbol{\Lambda}_M(:, 1:I_v) = [\boldsymbol{A}_1^T, \ldots, \boldsymbol{A}_L^T]^T$).

7. Fitting the TenBo Model to the Point Cloud

The fitting problem can be formulated as "*given a set of 3D points $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{Mp}$ captured from a human body surface, determine the global shape parameters $\boldsymbol{v}$ and the pose parameters (or joint angles) $\boldsymbol{\theta}$, such that the difference between the reconstructed 3D human body based on $\boldsymbol{v}$ and $\boldsymbol{\theta}$ and the original human body is minimum*".

We modify the shape completion algorithm in the SCAPE model [4] to solve the fitting problem. The SCAPE model assumes that the vertex correspondence between the point cloud and the reference mesh X is given. But we do not have this assumption in this paper. Therefore, we use ICP [18] to extract the correspondence. The fitting goal is

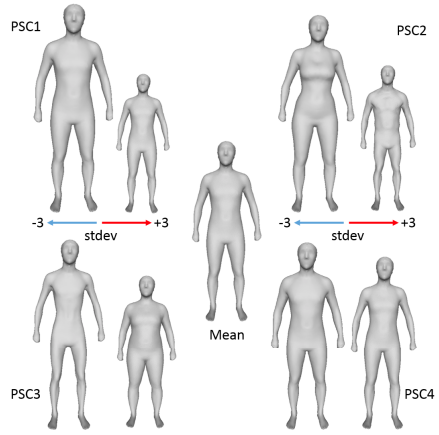

Figure 4. The first four principal shape components (PSC).

to solve $\boldsymbol{v}$ and $\boldsymbol{\theta}$ to minimize the cost:

$$g(\boldsymbol{v}, \boldsymbol{\theta}) = \min_{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M} \Big(\sum_{n=1}^{N} \sum_{q=1}^{2} ||\boldsymbol{R}_{l[n]}(\boldsymbol{\theta})\boldsymbol{D}_n(\boldsymbol{v}, \boldsymbol{\theta})\Delta\boldsymbol{x}_{n,q} - \Delta\boldsymbol{y}_{n,q}||^2 + w_z \sum_{m=1}^{M_p} ||(\boldsymbol{y}_{closest}(\boldsymbol{z}_m) - \boldsymbol{z}_m)||^2 \Big) \quad (8)$$

where $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M$ are the M vertices on the reconstructed 3D mesh, N is the number of triangles, $\boldsymbol{R}_{l[n]}, \boldsymbol{D}_n, \Delta\boldsymbol{x}_{n,q}$, and $\Delta\boldsymbol{y}_{n,q}$ refer to eq. (1), M_p is the number of points in the point cloud, $\boldsymbol{y}_{closest}(\boldsymbol{z}_m)$ is the closest vertex to $\boldsymbol{z}_m$ on the reconstructed mesh, w_z is a weight. The rotation $\boldsymbol{R}_{l[n]}$ is a function of joint angles $\boldsymbol{\theta}$ and the deformation $\boldsymbol{D}_n$ is a joint function of shape parameters $\boldsymbol{v}$ and joint angles $\boldsymbol{\theta}$.

The optimization is in the similar manner to the SCAPE model [4] except adding an ICP step in each iteration to search for the correspondence between the point cloud and the vertices on the reconstructed mesh. When using Microsoft Kinect, we can estimate the corresponding body segment $l[\boldsymbol{z}_m]$ for each point $\boldsymbol{z}_m$ using skeleton information. This is useful in searching for the closest vertex $\boldsymbol{y}_{closest}(\boldsymbol{z}_m)$ since it can significantly reduce the searching scope to the vertices on the body segment $l[\boldsymbol{z}_m]$.

8. Experimental Results

We compare our TenBo model with the SCAPE model [4] on the MPI Dataset [10] using cross validation. We also show 3D human body reconstruction using the depth map from Microsoft Kinect sensors.

8.1. Data Set

The MPI Dataset includes 520 full body 3D scans, captured from 114 subjects. 35 poses are predefined for subjects to perform. A neutral pose (natural standing pose) is

for everyone to scan. Only one subject (male) is scanned for all 35 poses, named as the *reference subject*. Other subjects perform either 10 predefined poses (randomly selected) or just the neutral pose. Each scan has 6449 vertices and 12894 triangles. All scans are registered. We choose 423 scans with good quality from 89 subjects. 50 subjects only have the neutral pose and the other 39 subjects have more poses.

8.2. Evaluation

We use leave-one-out cross validation to compare our TenBo model with the SCAPE model. A subject with more than two poses (except the reference subject) is selected as the validation data, and the remaining subjects are used for training both SCAPE and TenBo models. This process repeats until every subject with more than two poses (except the reference subject) has been used as validation data once. We keep the reference subject for training, since he is used to train the pose model in SCAPE. For each validation subject, the evaluation includes three steps: (a) training models (TenBo and SCAPE) using other subjects, (b) estimating the global shape parameters $\boldsymbol{v}$ for the validation subject under the *neutral* pose, and (c) predicting the 3D geometry of *non-neutral* poses for the validation subject based on the estimated shape parameters $\boldsymbol{v}$ and joint angles $\boldsymbol{\theta}$ (calculated in preprocessing, see Section 6.1). Predicting on novel poses shows the model generalizability. Accurate prediction requires (a) a good model to capture the relationship between the deformation and shape/pose parameters, and (b) accurate estimation of global shape parameters $\boldsymbol{v}$.

We use the average deformation error in the prediction over all validation subjects as the evaluation measure. The *deformation error* between a predicted mesh Y^r (including vertices $\{\boldsymbol{y}_1^r \ldots, \boldsymbol{y}_M^r\}$) and the original (or ground truth) mesh Y^o (including vertices $\{\boldsymbol{y}_1^o, \ldots, \boldsymbol{y}_M^o\}$) is defined as the average vector distance between triangle edges:

$$d(Y^r, Y^o) = \sqrt{\frac{1}{2N}\sum_{n=1}^{N}\sum_{q=1}^{2}||\Delta\boldsymbol{y}_{n,q}^r - \Delta\boldsymbol{y}_{n,q}^o||^2}. \quad (9)$$

This deformation error is approximately proportional to the difference in deformation matrix $\boldsymbol{D}_n$ because:

$$\Delta\boldsymbol{y}_{n,q}^r - \Delta\boldsymbol{y}_{n,q}^o \approx \boldsymbol{R}_{l[n]}(\boldsymbol{D}_n^r - \boldsymbol{D}_n^o)\Delta\boldsymbol{x}_{n,q}, \quad (10)$$

where $\boldsymbol{D}_n^r$ is the deformation matrix to generate the predicted mesh Y^r, $\boldsymbol{D}_n^o$ is the ground truth deformation matrix obtained from the original mesh Y^o. In our experiment, the deformation error is highly correlated to the average deformation matrix difference ($E_n||\boldsymbol{D}_n^r - \boldsymbol{D}_n^o||$) with Pearson correlation 0.99. We also calculated the Hausdorff distance which measures the global shape difference. The Pearson correlation between the deformation error and the Hausdorff distance is 0.74. Since the deformation error and the Hausdorff distance have the same trend in results, we only show the deformation error for the sake of brevity.

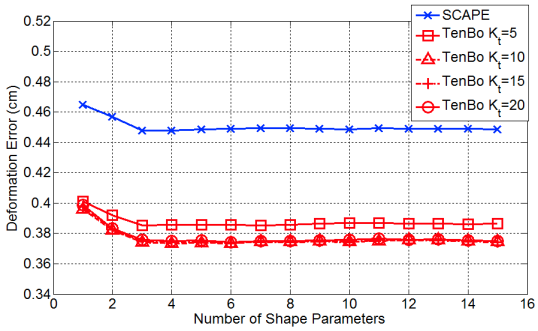

Figure 5. Prediction error over different number of global shape parameters I_v.

8.3. Cross Validation Results

In this section, we show the cross validation results for both TenBo and SCAPE models. When training the TenBo model, we heuristically choose the local shape dimension $I_t = 4$ due to the low dimensional shape of body segment. In the prediction step, we predict the 3D geometry for non-neutral poses for the validation subject using the global shape vector $\boldsymbol{v}$ (estimated under the neutral pose) and corresponding joint angles $\boldsymbol{\theta}$ for the non-neutral poses (computed in preprocessing). We estimate the global shape vector $\boldsymbol{v}$ using the neutral pose in two different ways: (a) using entire body scan as input, and (b) using sampled vertices as input.

8.3.1 Using Entire Body Mesh for Shape Estimation

Using the entire body scan as input, the deformation tensor $\boldsymbol{d}_l$ and joint angles $\boldsymbol{\theta}$ are available (calculated in preprocessing). The global shape vector $\boldsymbol{v}$ can be estimated by solving eq. (5). Figure (5) shows the prediction error for the SCAPE model and four TenBo models over different number of global shape parameters I_v. The four Tenbo models have different number of deformation bases (i.e. K_t, see Section 5.1). Clearly, the TenBo model outperforms the SCAPE model (by 16.6%). If we separate female and male subjects, TenBo outperforms SCAPE for both female (by 17.5%) and male subjects (by 14.5%). We observe that the prediction error is flat when using more than 4 global shape parameters. This is likely because the mesh resolution is low and the mesh alignment is not perfect. We also observe that the fitting quality does not improve when $K_t \geq 10$. In the rest of this section, we use 4 global shape parameters ($I_v = 4$) for both SCAPE and TenBo models, and use 10 deformation bases ($K_t = 10$) for TenBo.

We also compared TenBo with Hasler's algorithm [10]. We use linear regression to train pose functions when implementing Hasler's algorithm. Compared with Hasler's model which has prediction error (0.77cm) in cross validation, TenBo has less prediction error (0.37cm).

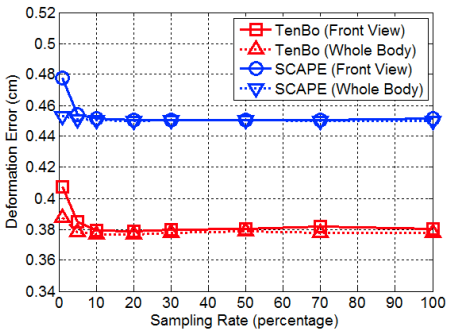

Figure 6. Prediction error over different sampling rates.

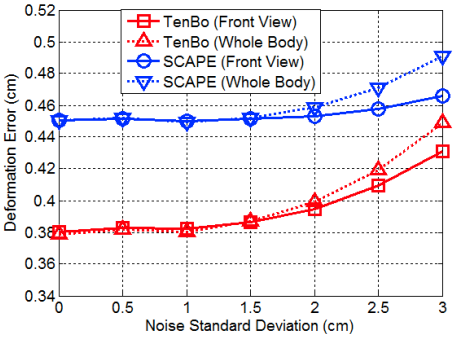

Figure 7. Prediction error over different adding noise.

8.3.2 Using Sampled Vertices for Shape Estimation

Next, we study the effect on shape estimation with different number of vertices to simulate when point clouds are available. For each validation subject, we randomly sample vertices on the 3D mesh of the neutral pose. We conduct random sampling in two ways: (a) sampling vertices of the whole body, and (b) sampling vertices on the frontal side. Different from using the entire mesh, we can not compute the deformation and joint angles directly from the sampled vertices. Therefore, we use the fitting algorithm in Section 7 to estimate both shape and pose parameters. The prediction part is the same as using the entire mesh as input.

Figure 6 shows the prediction error for both SCAPE and TenBo over different sampling rates. We have three observations - (a) TenBo model outperforms the SCAPE model, (b) prediction error converges when the sampling rate is above 10%, so we do not need a lot of data to estimate body shape, and (c) sampling the whole body helps when the sampling rate is lower than 10%. This means that the samples from the back help when the frontal samples are not enough to capture the shape.

Figure 7 shows the prediction error over different sampling noise. We sampled at sampling rate 50% and add zero-mean Gaussian noise along all x-y-z directions. The standard deviation σ of the noise changes from 0 to 3cm. Again, TenBo outperforms SCAPE. Both models are not visibly affected when σ is less than 1cm and then the prediction error increases as the noise increases. Using the frontal samples alone is less sensitive compared with using the samples from the whole body, since the confusion from both side makes things worse when noise level is high.

8.4. Run Time of the Fitting Algorithm

We test the run time of the fitting algorithm (Section 7) for both SCAPE and TenBo models. The fitting algorithm is run on a single desktop without GPU acceleration to fit 100 point clouds, which are generate by random vertex sampling on 100 3D body scans (at sampling rate 20%). We use the average shape at the neutral pose as the initial guess. On average, the SCAPE model takes 1.57 sec (45.9 iterations, 34.23 msec per iteration) and TenBo takes 1.78 sec (45.1 iterations, 39.49 msec per iteration) to fit one point cloud. Both models take similar number of iterations to converge. For each iteration, TenBo is more expensive only in computing deformation vectors. Note that both models have the same complexity in animation where shape parameters are known and only pose parameters change over time.

8.5. 3D Reconstruction Using Microsoft Kinect

We also apply the TenBo model to capture 3D human body using Microsoft Kinect. With the help of skeleton, the body segment correspondence for pixels in the depth map can be easily estimated. Then we can use eq. (8) to estimate both shape and pose parameters. Figure (8) show 3D reconstruction for three users under five poses. We can see that reconstruction is close to the user's shape and pose (using the color image on the left as the reference). These results show the capability of the TenBo model in real applications using the off-the-shelf capture device.

9. Conclusion

This paper presents a novel tensor-based 3D human body model (TenBo model). Compared with the popular SCAPE model which separates the shape and pose deformations, the proposed approach jointly models shape and pose deformations in a systematic manner. The experiments demonstrate the superior performance of the proposed TenBo model to the SCAPE model. Our TenBo model is capable of capturing the human body shape using the depth map and skeleton provided by Microsoft Kinect sensors and generating animations. In the future, we plan to extend our research in several ways - (a) speeding up the fitting algorithm to support real-time applications, and (b) integrating TenBo model and human pose estimation algorithm (e.g. inferring dense correspondences [14]) to improve shape/pose estimation using Microsoft Kinect sensors.

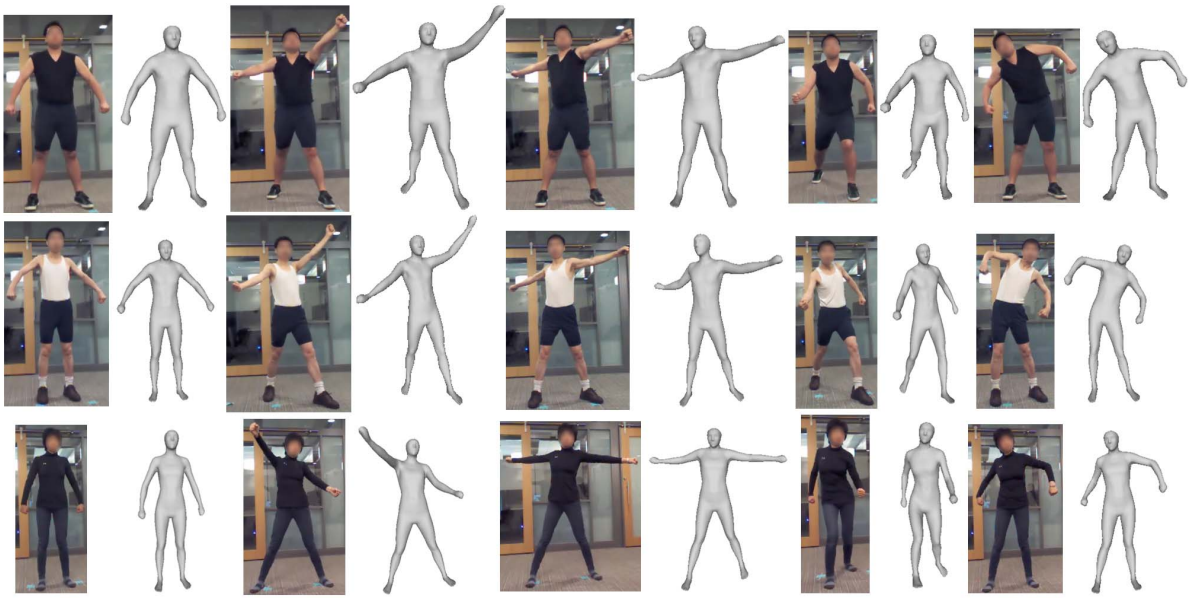

Figure 8. 3D reconstruction of human body using Microsoft Kinect sensors.

References

[1] E. Acar, T. G. Kolda, and D. M. Dunlavy. All-at-once optimization for coupled matrix and tensor factorizations. In *MLG'11: Proceedings of Mining and Learning with Graphs*, August 2011.

[2] B. Allen, B. Curless, and Z. Popović. The space of human body shapes: reconstruction and parameterization from range scans. In *SIGGRAPH*, pages 587–594, 2003.

[3] B. Allen, B. Curless, Z. Popović, and A. Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 147–156, 2006.

[4] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. *ACM Trans. On Graphics*, 24(3):408–416, 2005.

[5] A. Balan and M. J. Black. The naked truth: Estimating body shape under clothing,. In *European Conf. on Computer Vision, ECCV*, volume 5304, pages 15–29, oct 2008.

[6] O. Freifeld, A. Weiss, S. Zuffi, and M. J. Black. Contour people: A parameterized model of 2d articulated human shape. In *IEEE Conf. on Computer Vision and Pattern Recognition, CVPR*, pages 639–646, jun 2010.

[7] P. Guan, L. Reiss, D. Hirshberg, A. Weiss, and M. J. Black. Drape: Dressing any person. *ACM Trans. on Graphics (Proc. SIGGRAPH)*, 31(4):35:1–35:10, jul 2012.

[8] P. Guan, A. Weiss, A. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *Int. Conf. on Computer Vision, ICCV*, pages 1381–1388, sep 2009.

[9] N. Hasler, H. Ackermann, B. Rosenhahn, T. Thormählen, and H.-P. Seidel. Multilinear pose and body shape estimation of dressed subjects from image sets. In *CVPR*, pages 1823–1830, 2010.

[10] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel. A statistical model of human pose and body shape. In *Computer Graphics Forum (Proc. Eurographics 2008)*, volume 2, Mar. 2009.

[11] A. Jain, T. Thormählen, H.-P. Seidel, and C. Theobalt. Moviereshape: tracking and reshaping of humans in videos. *ACM Trans. Graph.*, 29(6):148:1–148:10, Dec. 2010.

[12] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl*, 21:1253–1278, 2000.

[13] H. Seo and N. Magnenat-Thalmann. An example-based approach to human body manipulation. *Graph. Models*, 66(1):1–23.

[14] J. Taylor, J. Shotton, T. Sharp, and A. W. Fitzgibbon. The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation. In *CVPR*, pages 103–110, 2012.

[15] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *European Conference on Computer Vision*, pages 447–460, 2002.

[16] M. A. O. Vasilescu and D. Terzopoulos. Multilinear subspace analysis of image ensembles. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 93–99, 2003.

[17] A. Weiss, D. Hirshberg, and M. Black. Home 3d body scans from noisy image and range data. In *Int. Conf. on Computer Vision, ICCV*, pages 1951–1958, Barcelona, nov 2011.

[18] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.

[19] S. Zhou, H. Fu, L. Liu, D. Cohen-Or, and X. Han. Parametric reshaping of human bodies in images. *ACM Trans. Graph.*, 29:126:1–126:10, July 2010.

[20] S. Zuffi, O. Freifeld, and M. J. Black. From pictorial structures to deformable structures. In *CVPR*, pages 3546–3553, 2012.