3D Semantic Parameterization for Human Shape Modeling: Application to 3D Animation.

Christian Rupprecht
Department of Computer Science, CAMP,
Technische Universität München, Germany
christian.rupprecht@cs.tum.edu

Olivier Pauly
Institute of Biomathematics and Biometry,
Helmholtz Zentrum München, Germany
Computer Aided Medical Procedures,
Technische Universität München, Germany
pauly@cs.tum.edu

Christian Theobalt
Graphics, Vision & Video,
Max Planck Institute, Germany
theobalt@mpii.de

Slobodan Ilic
Department of Computer Science, CAMP
Technische Universität München (TUM)
slobodan.ilic@cs.tum.edu

Abstract

Statistical human body models, like SCAPE, capture static 3D human body shapes and poses and are applied to many Computer Vision problems. Defined in a statistical context, their parameters do not explicitly capture semantics of the human body shapes such as height, weight, limb length, etc. Having a set of semantic parameters would allow users and automated algorithms to sample the space of possible body shape variations in a more intuitive way. Therefore, in this paper we propose a method for re-parameterization of statistical human body models such that shapes are controlled by a small set of intuitive semantic parameters. These parameters are learned directly from the available statistical human body model. In order to apply any arbitrary animation to our human body shape model we perform retargeting. From any set of 3D scans, a semantic parametrized model can be generated and animated with the presented methods using any animation data. We quantitatively show that our semantic parameterization is more reliable than standard semantic parameterizations, and show a number of animations retargeted to our semantic body shape model.

1. Introduction

With the use of 3D laser scanners, static 3D human body shapes can be captured at different poses to enable the creation of statistical human body models covering a large variety of shapes and poses. However, it is not possible to continuously scan 3D human bodies performing some continuous motion. Therefore, marker based human body tracking systems are used to capture human motions that can then be applied to 3D body shapes for creating realistic 3D animations. Even though statistical human body models, like SCAPE [4], provide excellent results and are applied to many problems, their parameters have only statistical meaning and do not capture explicitly semantics of the human body shape like height, weight, limb length, *etc*. Providing a set of semantic parameters would allow users and automated algorithms to sample the space of possible body shape variations more intuitively. Using such sampling, creating large and realistic datasets becomes much easier than by capturing raw data using acquisition platforms. In many machine learning applications (pose estimation, activity recognition *etc*.) a large set of training data is required for the algorithms to work reliably. Manual acquisition of this data, *e.g*. scanning people using laser scanners, measuring their semantic parameters (height, weight, age, sex etc.), capturing their motion while doing different activities and removing background and noise from the ac-

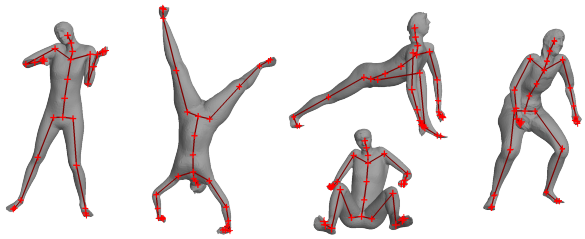

Figure 1. Human meshes in different poses produced by our framework

quired images is a tremendous amount of work. Often those databases contain thousands to millions of entries as learning algorithms generally perform better when more training data is available. Such databases were already created by MSR and used for their human pose estimation method based on randomized classification [16] and later regression trees [7], however not many details are given about the construction of the database.

Therefore, in this paper we propose a framework that enables the construction of such annotated 3D shape datasets based on a set of *semantic* parameters and animate them using arbitrary human motion capture data. This greatly reduces the amount of manual work and can be used in machine learning algorithms where large amount of data are required for good generalization. Our contributions is a reparametrization of the statistical human body models from the body model itself, *i.e*. without requiring manual annotation of large corpora of shapes and a new way of learning that parametrization based on regression forests that performs much better than previous approaches based on simple linear regression. We used a simple retargeting technique similar to cyclic coordinate descent [13, 18], which allows applying arbitrary animation to our body shape models. The presented methods are not limited to human data sets. Any set of 3D scans can be used to generate a semantic parametrized model, which can then be animated with the presented methods using any skeleton/animation data. In the reminder of this paper we first review related work, then present our two contributions and show evaluations.

2. Related work

Statistical human body shape models have been introduced for the first time in 2D by Changbo Hu *et al.* [10]. Later, Anguelov *et al.* [4] proposed a 3D human body shape and pose model called SCAPE (Shape Completion and Animation of People). In SCAPE two models a learned: the first one describes the human shape the other the human pose. The combination of both models results in a parametric human model that covers shape and pose. However no semantic mapping is described and the data used in their work is not publicly available. By modeling muscle deformations, animations are defined by the movement of different markers on the body. For our framework we use the work and data of Hasler *et al.* [9] which builds on top of SCAPE encoding shape and pose in the same way. Here again the combined model for pose and shape makes it difficult to integrate external animations. As they have released the data used to create the model, it is possible to recreate it leaving out the pose deformation part to produce humans all taking on the same pose but with different shapes.

A common animation technique, called *Linear Blend Skinning (LBS)* was described by Lewis *et al*. [14]. The animation is stored in terms of skeleton and a set of animation matrices, which transform the bones of the reference skeleton in each frame. The vertices of the mesh are attached to one or more bones by weights describing their degree of attachment to the bone. The final transformation for each vertex is a linear combination of the movement of the bones multiplied by their weights. This technique is very common in Computer Graphics applications as it is easy to compute and can also be implemented entirely on modern graphics hardware exploiting the high parallelism for additional speed [12]. In [1], Allen *et al.* present a more sophisticated animation technique by interpolation between preregistered scans. A human model is generated using principal component analysis on 250 registered human scans [2] similar to the methods used by Hasler *et al.* in a later publication [3]. Both of them, as well as recent Movie Reshape [11] where a standard kinematic skeleton is fitted to the shape of the average human and scaled with the deforming shape by expressing each joint in terms of the surrounding surface vertices, presented semantic mapping as a linear approximation on the whole PCA subspace. More precisely, they have measured people before scanning them and then learn a linear mapping between these semantic parameters and principal component vectors. By contrast we do not need any a priori semantic information and can extract and learn it directly from the data. Moreover, our approach permits to model non-linear relationships between the semantic parameters and the weights of the PCA vectors. We measure accuracy of the semantic mapping and compared it to the methods based on linear mapping. Our method, where the semantic mapping is learned from the statistical shape model, shows much better accuracy.

In the field of automatic animation of arbitrary meshes by a given skeleton, Liu *et al.* [15] fit a skeleton into an arbitrary mesh by creating a repulsive force field inside the mesh and choose the local minima as joint positions which are later refined. The mesh is then attached to the fitted skeleton according to the distance between the bones and the surface. Finding the vertex weights based only on distance produces problems in complex deformation areas like toes, shoulders and hips. Therefore, we relied on the work of Baran and Popović [5] as their algorithm takes an existing skeleton to find the best embedding. The best fit is found by optimizing both a discrete and a continuous error function. This is ideal as the skeleton will be given by the selected animation. However the skeleton has to be tagged with additional hints, marking some joints as symmetric, near the floor or inside big volumes which improves the results but has to be done by hand. Additionally the method for finding the skin attachment to the bones is more sophisticated as it takes into account the structure of the model instead of solely using the proximity of the bone. The vertex weights are found solving a heat equation on the surface of the body. One bone is heated to 1 degree while all others are forced

to 0 degrees. The resulting heat equilibrium on the skin (values between 0 and 1) are taken as weights for the bone heated to 1 degree.

3. Methods

Our method can be split into two parts: the creation of a realistic, semantically parametrized 3D human mesh and its animation using a motion capture database.

3.1. Human model and mesh generation

In the first step of our approach, the goal is to compute a parametrized human model that is able to produce 3D human meshes in a broad variety of shapes. Given as input a few semantic parameters describing the human body (such as height, weight, hip size, *etc*.), this model generates a 3d human mesh that can later be used in the animation step. In the current work, we propose to decompose this problem into two subtasks: (1) to generate a first low dimensional human model using PCA and (2), to learn a *semantic mapping* that relates the low-dimensional space spanned by the eigenvectors to human semantic variations. In the following, we start by describing the construction of a human PCA model.

3.1.1 Low dimensional human mesh model through PCA

Based on the work of N. Hasler *et al.* [9], we propose to use approximately 200 full body 3D scans to generate a parametrized model of the human body. By registering all scans to a template model consisting of 6449 3D vertices, a semantically unified description is created. This permits pointwise comparison of every pair of scans, and ensures that corresponding vertices lie at the same semantic position on all scans. To create a generative human model, a principal component analysis (PCA) is performed, thereby reducing the dimensionality from $6449 \times 3 = 19347$ to 228:

$$M = \overline{M} + \sum_{i=1}^{m} \alpha_i a_i \qquad (1)$$

where $M \in \mathbb{R}^{6449\times 3}$ is the final mesh parametrized by 3D vertex coordinates, generated as a linear combination of the mean mesh $\overline{M}$ and the principal components a_i with different weights α_i. With this model a 3D human shape can be fully described by the set of weights α_i. While principal components permit to describe main shape variations of the 3D human mesh over a population, those are difficult to interpret in terms of human semantics. Though it would be possible to inspect visually the variations of individual weights for the eigenvectors, it would not be possible to describe what effect this parameter has on the model. To solve this problem, we propose to learn a *semantic mapping*, to model the relationship between intuitive parameters like height, weight, waist size, *etc*. and the PCA weights. It can be understood as a function $\mathcal{S}(\varsigma) = \alpha$, where $\alpha \in \mathbb{R}^m$ is the vector of all α_i a set of all PCA coefficients controlling the shape and ς is a vector containing all the intuitive semantic parameters. In the following, we describe how to automatically measure human semantic parameters and to learn such a semantic mapping.

3.1.2 Measuring semantic parameters

In order to learn a semantic mapping, we need to create a large training set of corresponding human PCA weights and semantic parameters. Using the eigenvectors α_i and their standard deviation σ_i computed from the principal component analysis, a large set of 10,000 human meshes is generated by randomly sampling with $3\sigma_i$ for each factor. The higher sampling range was chosen to produce a greater variety of meshes. As by construction, vertices on the meshes are always on the same semantic position, we can define and measure semantic parameters on the average model and deform the measurement together with the mesh. Previous methods used manual annotation of shape examples which is cumbersome. We create the annotations at arbitrary density directly from shape instances of the PCA model and save the annotation time. For this work, six measurements were chosen: height, breast size, waist size, hip size, leg length and shoulder size. All of them can be described on the mean model by specifying the vertices that lie on the line of the measurement. For example, all vertices that define the waist build a ring around the body. The length of this waist measure can then easily be computed by calculating the distance from one vertex to its neighbor and adding it to the total distance. After a full round, the circumference of the waist is determined. For further precision, all vertices are projected to a plane with the up vector of the coordinate system (pointing from the feet to the head) as normal vector to the plane. This prevents a false length increase when the vertices are slightly moved up and down by the PCA deformations. For values like height and leg length only the height entry in the vertex position is used so the real height is measured even when the vertex moves to the side. Figure 2 shows the described measurements first on the mean mesh and then on three other randomly generated meshes. The vertices defining the lines have moved correctly and still define the same body property. In the next section, we detail how to learn our semantic mapping from this dataset using regression forests.

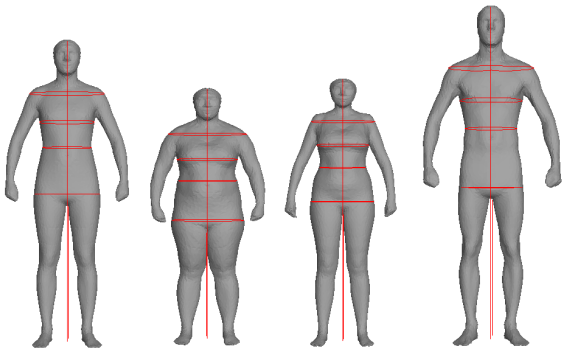

Figure 2. Automatic measurements of height, breast size, waist size, hip size, leg length and shoulder size shown on the average model and on three random examples

3.2. Learning the semantic mapping using regression forests

Our goal is to learn a function $\mathcal{S}$ relating an input x to a continuous output y based on a set of samples, where $x \in \mathbb{R}^n$ represent the n semantic values $\varsigma = (\varsigma_1, ..., \varsigma_n)$ measured from the mesh and $y = \alpha = (\alpha_1, ..., \alpha_m) \in \mathbb{R}^m$ represents the according weights for the eigenvectors that generate this mesh. To learn this highly non-linear mapping, we propose to use regression random forests. We show later that this results in a much better performance than the linear mappings on the whole space used in the related works. In [6], Criminisi *et al.* demonstrate that random forests are state-of-the-art learners that can be used for many different tasks such as classification, regression or density estimation. Recently, they have gained increased interest in computer vision, as they show impressive results in a wide range of applications as human pose estimation [16] or semantic segmentation [17]. Basically, a random forest consists of an ensemble of decorrelated decision trees that provides a piece-wise approximation of the function $\mathcal{S}$. Using a divide and conquer strategy, each tree first partitions the input space, *i.e.* subdivides the training data into consistent subsets using decision functions, and then models the mapping $\mathcal{S}$ locally in each part of the input space. More precisely, a tree consists of a collection of split nodes and leaves that contain, respectively, the decision functions and the local estimates of $\mathcal{S}$.

In this work, we base the decision functions on so-called axis-aligned splits, that subdivide the incoming training instances based on their value in one randomly selected dimension in the input space. In the leaves, we propose to model the semantic mapping locally by using a multi-dimensional linear function g:

$$g_{\mathbf{A},\mathbf{b}}(x) = \mathbf{A}x + \mathbf{b} \tag{2}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ is a matrix and $\mathbf{b} \in \mathbb{R}^m$ is a vector. In the following, we will detail briefly the training of our regression forest.

3.2.1 Forest learning

Given a set of training instances $(x, y) \in \mathcal{T} \subset \mathbb{R}^n \times \mathbb{R}^m$, learning is the process of choosing the best decision functions within the nodes and estimating the best model parameters in the leaves. In the present work, our training set consists of vectors x containing the semantic parameters such as height or hip size and their corresponding output y that contains the weights of the PCA eigenvectors for mesh construction. During the training phase, a tree is grown by recursively splitting the training data. As detailed in [6], at each node, the parameters of the decision function, in our case a dimension of the input space as well as a threshold, is chosen in a greedy fashion by maximizing an objective function called information gain. Recursive splitting stops when the maximum tree depth is reached or when the amount of training instances falls below a certain threshold, and a leaf node is created.

In a leaf, the parameters of the prediction function $g_{\mathbf{A},\mathbf{b}}(x)$ are estimated from the subset $\tau \subset \mathcal{T}$ of training instances that reach this leaf. In the case of our linear predictor model, we estimate the matrix $\mathbf{A}^*$ and the vector $\mathbf{b}^*$ by minimizing the squared difference between the line and the actual values in the sample pair (x, y):

$$(\mathbf{A}^*, \mathbf{b}^*) = \arg\min_{\mathbf{A},\mathbf{b}} \sum_{(x,y)\in\tau} (\mathbf{A}x + \mathbf{b} - y)^2 \tag{3}$$

3.2.2 Shape parameter prediction

Once the forest has been trained, prediction can be performed for an unseen input x by combining the output of the different trees. The input x traverses each tree, choosing recursively the left or right child node based on the evaluation of the decision function. Once a leaf node is reached, a prediction can be computed by using the stored linear model. The combined prediction y of the forest is averaged over all predictions y_t of all trees: $y = \frac{1}{T}\sum_{t=1}^{T} y_t$, where T is the number of trees in the forest.

To conclude this section, a full human mesh can be generated from few simple semantic parameters by using the previously trained regression random forest to estimate the factors α_i in equation 1 for the principal components with which the position of the vertices can be computed.

3.3. Human mesh animation

Given the semantic 3D human body shape, our goal now is to apply an animation from any available motion capture database. Here we use the CMU Motion Capture Database

as it provides a large variety of animations[1], but any available motion database can be used.

Skeletons from public motion capture databases differ in skeletal dimensions and joint configurations. In order to animate any arbitrary target body shape with arbitrary input motion data we need to solve two problems. We need to rescale and embed the motion data skeleton into our target shape, and we need to adjust the joint angles from the motion capture file to match the shape of the target model. Solving this problem is known as *retargeting*. A lot of work has been done in this area which is nicely summarized in a SIGGRAPH06 course of J. P. Lewis and Frederic Pighin [14]. However since our target application does not require perfect retargeted animation, but rather generation of larger database of animated 3D shapes to be used by the learning algorithms we use a very simplified retargeting algorithm that will be briefly described below.

3.3.1 Skeleton embedding

We rely on the Pinocchio system [5] for fitting the given skeleton inside the 3D mesh and compute Linear Blend Skinning for encoding how much this particular bone influences that vertex. In practice we alter Pinocchio system in order to produce good results with the output meshes from the semantic mapping and the target animation skeletons from the CMU motion capture database.

3.3.2 Skeleton remapping

The standard human skeleton provided with the Pinocchio library produces good results with the meshes from the semantic mapping. So instead of training the weights of Pinocchio's penalty functions to work with the CMU skeletons, we use Pinocchio's simple skeleton counting 17 bones. This ensures faster execution time, better embedding results and greater generalization. After the continuous optimization, the skeleton has to be replaced by the target skeleton from the animation. This replacement needs to be done by a skeleton matching algorithm, that is fast and robust for different target skeletons. Here, the two skeletons to be matched are considered as directed acyclic graphs with 3D position information in the nodes representing the skeleton joints. First, leaf nodes of both graphs are extracted, then head, hands and feet nodes are found in both sets of leaves. Afterwards, common parents of pairs of leaves, mainly corresponding to shoulder and hip joints, are matched in both graphs. In the final step joints between the leaves and parents are matched and unmatched joints, like thumbs, are placed inside the mesh such that their connected bones are scaled and rotated according to the transformation of their parent bones.

[1] http://mocap.cs.cmu.edu

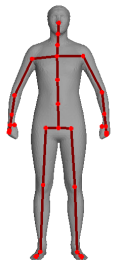

Figure 3. A complex skeleton embedded in the mesh.

This modified skeleton is now put back into the Pinocchio algorithm, so that the skin of the mesh can be attached to it. Pinocchio attaches the vertices of the model to the bones of the skeleton to make it possible to transfer motion from the skeleton to the mesh.

3.3.3 Animation retargeting

The actual skeleton to be animated is the one returned from the skeleton fitting algorithm described above. It may have different bone lengths and a different pose than the original skeleton from the CMU database. Thus, a new set of joint parameters associated to each bone has to be computed in order to produce correct animation of the new target shape. Finding this new joint parameters is known as retargeting in Computer Graphics and any sophisticated algorithm can be used. However, due to the less restrictive nature of our application that is the generation of a large database of various human body shapes and poses, we are not concerned about usual artifacts of retargeting, such as foot skating, and, therefore, perform a simple and efficient retargeting technique similar to Cyclic Coordinate Descent (CCD) [13, 18] but without enforcing explicit end-effector placement constraints.

4. Results

In this section, we evaluate our approach for mesh generation and semantic mapping in comparison to existing solutions.

4.1. Human PCA model

Given 228 different scans in the standard pose, we perform PCA resulting in 228 eigenvectors a_i and one mean shape $\overline{M}$. For the semantic mapping only the first 25 eigenvectors are used as the others mostly contained noise and do not improve the semantic mapping. The training of the regression forest is significantly faster as the output dimensionality is reduced from 228 to 25. To create a training set for the regression forests, the 25 principal components were randomly sampled with 3σ (σ being their standard deviation) generating $10,000$ random human meshes. Each mesh is then automatically associated to its corresponding

Our semantic mapping approach						
(in cm)	height	breast size	waist size	hip size	leg length	shoulder size
mean	-1.545e-07	0.5894	0.7111	0.1980	-1.0371	0.9387
variance	8.007e-13	4.7540	2.7176	0.9891	0.1279	33.5327

Table 1. Error in cm resulting from our proposed semantic mapping with regression random forests

Linear Semantic mapping					
(in cm)	height	breast size	waist size	hip size	leg length
mean	-1.9923	1.5960	-2.4732	-3.7087	-8.4568
variance	0.9175	160.7080	26.4817	32.2614	50.1143

Table 2. Error in cm resulting from the standard linear semantic mapping

semantic parameters using the methods described in section 3.1.2. As semantic parameters, we choose height, shoulder size, breast size, waist size, hip size and leg length.

In the end, our training data consists of $10,000$ meshes, each of them associated to a vector containing six semantic parameters. Figure 2 illustrate these semantic parameters shown directly on the mesh.

4.2. Semantic mapping

A randomized regression forest was trained on the previously generated data. In this scenario, we can automatically measure all semantic parameters directly on the mesh. So we propose to evaluate our semantic mapping approach by comparing the input semantic parameters with their values measured directly on the resulting mesh (after reconstruction with PCA). For instance, if we give a height of 2m as input, we can compute the error by measuring it directly on the resulting mesh. To find the best parameters for the random forest, the error was computed for many combinations of parameters. For the final application, 50 trees with depth 15 are chosen although more trees are generally performing better, the improvement was marginal beyond 50 trees. The number of trials for split planes used in randomize node optimization is 13 and the minimal number of samples in one leaf is set to 30 to find a reliably probabilistic line.

Results of our quantitative evaluation are shown in table 1. The best results are found for the height. This is explained by the fact that the height is captured by a single principal component from the PCA. So the height value can be directly encoded into the factor α_1 of the first eigenvector a_1 of equation 1.

The same evaluation is performed with a standard linear semantic mapping. The results are shown in table 2. Shoulder size was not used for this test as it was not a semantic parameter on their model. Comparing both tables shows the significant improvements by the new method. In all cases the mean error is closer to zero meaning it is more centered around the desired value. The variance and in consequence the standard deviation is greatly reduced.

Both tables 2 and 1 are visualized in figure 4.

Comparing both tables shows the significant improvements by the new method. In all cases the mean error is closer to zero meaning it is more centered around the desired value. The variance and in consequence the standard deviation is greatly reduced.

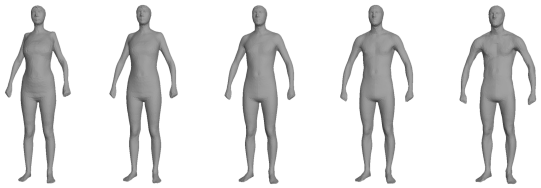

Figure 6. Successive increase of shoulder size from 100 cm to 140 cm with steps of 10 cm

4.3. Animation

In some cases Pinocchio has problems to find the exact position of degree two joints. Especially the elbow joints are sometimes placed too high in the upper arm or too low in the lower arm. For the case of generating artificial motion capture datasets this problem can be accepted as the pose estimation algorithms do not capture such fine details. Figure 5 shows different meshes animated in the same frame of a simple running , punching or slide flip animation. This figure shows how well different models are animated by the same skeletal animation. Although the shapes of all meshes differ greatly, the animation is applied nicely to all of them.

Figure 6 shows the performance of the semantic mapping on the example of shoulder size. By keeping every other semantic parameter constant, the shoulder size is increased from 100 cm to 140 cm in 5 steps of 10 cm.

Linear blend skinning as animation technique also comes with its well known limitations. As it is a simple linear interpolation of transformations sharp creases and bends produce artifacts that are not natural. In poses where the human skin would fold and touch itself linear blend skinning has problems. Figure 7 shows one problem that occurs when the shape of the motion captured human and the shape of the animated mesh are very different. The left arm of the mesh intersects the left leg. The captured person had much thinner upper legs so this movement is valid. However the animated mesh has rather large hips and legs so the self intersection occurs. This problem also persists the other way round.

Although in some extreme cases problems can be seen in the animation, the presented algorithms work well for the purpose of generating artificial training data sets. Additionally, the presented methods can also be used for different purposes such as generating crowds or custom characters in computer games.

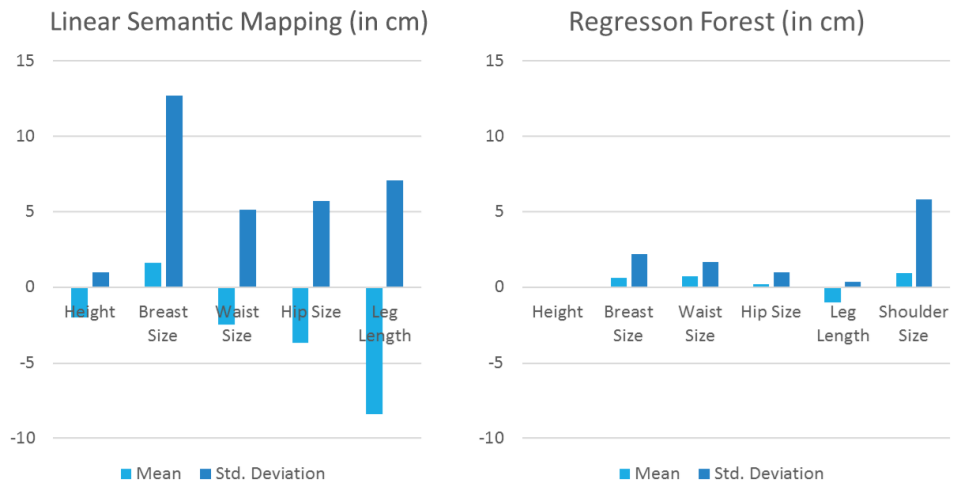

Figure 4. Visualization of the results in table 1 and 2

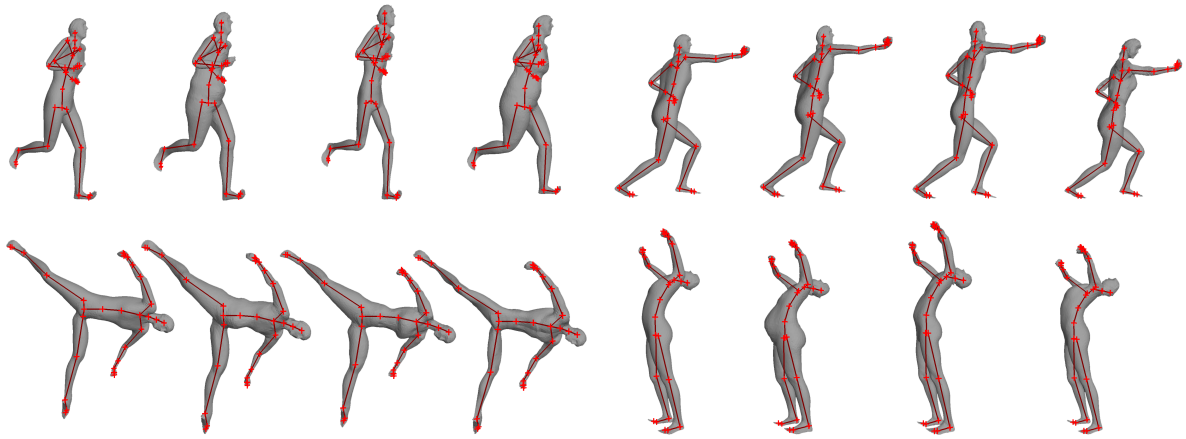

Figure 5. Different meshes in the same frame of a running, punching, side flip and pantomime animation.

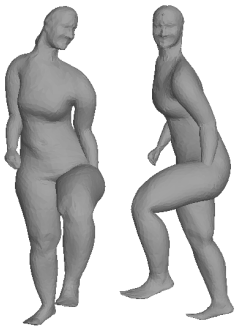

Figure 7. Self intersections can occur when the captured person and the mesh differ greatly in shape

5. Discussion and Conclusion

We presented a method to create a semantically meaningful parametrized human model from a set of full body 3D scans. The human meshes generated by the model can then be animated with any given skeletal animation data. The results of comparing the proposed semantic mapping from understandable and intuitive body measurements to the internal representation of the human body show a significant improvement to existing methods. Furthermore, we propose a framework for creating a large database of 3D human shape animations using a motion capture database. Our approach is not limited to human datasets and any set of 3D scans can be used to generate a semantic parametrized model. The arbitrary model can be animated with the presented methods using any skeleton/animation data that fits the general appearance of the mesh.

Although producing good results, the presented methods could be improved in some parts. Using the principal component analysis as first dimensionality reduction method may not be the best choice. If the initial set of human scans appears to fall into different dense groups in the high di-

mensional spaces a clustering method like the mean shift algorithm can be used to identify those clusters. The principal component analysis could then be used on each individual cluster to capture finer details from the scans. Adding more or different semantic parameters to the model is simple as they can be defined on the mean shape and those definitions generalize well to all possible meshes. Parameters like arm length, head size or biceps size could easily be added and would then produce a greater variety on the resulting meshes. As the semantic measuring is directly possible on the mesh it would be possible to use the output of the randomized forest as a starting point for a non linear minimization, that would lead to a fine tuned resulting mesh. The error between actual measurements on the result and the user input is minimized. The simplest approach for this would be gradient free minimization algorithms, although the gradient could be calculated on the principal components and then be used for more sophisticated minimization techniques. Changing the animation technique from linear blend skinning to more complex methods and or utilizing a sophisticated IK solver could remove some artifacts introduced during animation.

In future work, we will aim at dressing the resulting mesh using the recently published DRAPE (DRessing Any PErson) methods by Guan *et al.* [8] to produce more challenging data for pose estimation.

References

[1] B. Allen, B. Curless, and Z. Popović. Articulated body deformation from range scan data. *ACM Trans. Graph.*, 21(3):612–619, July 2002. 2

[2] B. Allen, B. Curless, and Z. Popović. The space of human body shapes: reconstruction and parameterization from range scans. *ACM Trans. Graph.*, 22(3):587–594, July 2003. 2

[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, Aire-la-Ville, Switzerland, 2006. Eurographics Association. 2

[4] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. *ACM Trans. Graph.*, 24(3):408–416, July 2005. 1, 2

[5] I. Baran and J. Popović. Automatic rigging and animation of 3d characters. In *ACM SIGGRAPH 2007 papers*, SIGGRAPH '07, New York, NY, USA, 2007. ACM. 2, 5

[6] A. Criminisi, J. Shotton, and E. Konukoglu. Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning. Technical Report MSR-TR-2011-114, Microsoft Research, Oct 2011. 4

[7] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 415–422. IEEE, 2011. 2

[8] 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

[9] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel. A statistical model of human pose and body shape. In P. Dutr'e and M. Stamminger, editors, *Computer Graphics Forum (Proc. Eurographics 2008)*, number 28 in 2, Munich, Germany, Mar. 2009. 2, 3

[10] C. Hu, Q. Yu, Y. Li, and S. Ma. Extraction of parametric human model for posture recognition using genetic algorithm. *Automatic Face and Gesture Recognition, IEEE International Conference on*, 0:518, 2000. 2

[11] A. Jain, T. Thormählen, H.-P. Seidel, and C. Theobalt. Moviereshape: Tracking and reshaping of humans in videos. In *ACM Transactions on Graphics (TOG)*, volume 29, page 148. ACM, 2010. 2

[12] L. Kavan, S. Collins, J. Zara, and C. O'Sullivan. Geometric skinning with approximate dual quaternion blending. In *ACM Trans. Graph.*, volume 27, page 105, New York, NY, USA, 2008. ACM Press. 2

[13] J. Lander. Making kine more flexible. *Game Developer Magazine*, 1:15–22, November 1998. 2, 5

[14] J. P. Lewis, M. Cordner, and N. Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '00, pages 165–172, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. 2, 5

[15] P.-C. Liu, F.-C. Wu, W.-C. Ma, R.-H. Liang, and M. Ouhyoung. Automatic animation skeleton construction using repulsive force field. In *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*, PG '03, pages 409–, Washington, DC, USA, 2003. IEEE Computer Society. 2

[16] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *In In CVPR, 2011. 3*, 2011. 2, 4

[17] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 4

[18] L.-C. Wang and C. C. Chen. A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. *Robotics and Automation, IEEE Transactions on*, 7(4):489–499, 1991. 2, 5