The Visual Computer manuscript No.
(will be inserted by the editor)

Parametric Editing of Clothed 3D Avatars

Yin Chen · Zhi-Quan Cheng · Ralph R. Martin

Abstract Easy editing of a clothed 3D human avatar is central to many practical applications. However, it is easy to produce implausible, unnatural looking results, since subtle reshaping or pose alteration of avatars requires global consistency and agreement with human anatomy. Here, we present a parametric editing system for a clothed human body, based on use of a revised SCAPE model. We show that the parameters of the model can be estimated directly from a clothed avatar, and that it can be used as a basis for realistic, realtime editing of the clothed avatar mesh via a novel 3D body-aware warping scheme. The avatar can be easily controlled by a few semantically meaningful parameters, 12 biometric attributes controlling body shape, and 17 bones controlling pose. Our experiments demonstrate that our system can interactively produce visually pleasing results.

Keywords Parametric human · editing · 3D avatar · clothing

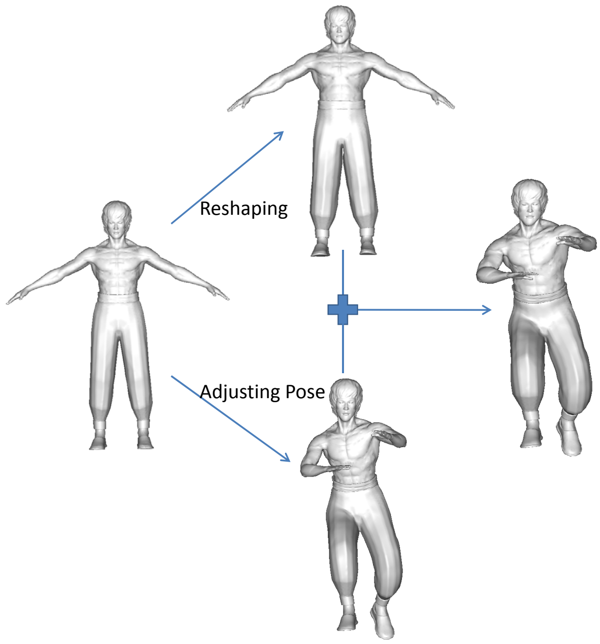

Fig. 1 Parametric editing of a 3D mesh of Bruce Lee, including shape alteration to add weight, adjustment to a kung fu pose, and simultaneous shape and pose change.

1 Introduction

Editing 3D avatars of human characters is an important task in film animation, games and other applications. Key goals are to provide a natural editing system with realtime computation and intuitive use.

Yin Chen
PLA University of Science and Technology, Jiangsu, China.

Zhi-Quan Cheng
Avatar Science Company, China.

Ralph R. Martin
School of Computer Science & Informatics, Cardiff University, UK.

Unfortunately, editing clothed 3D avatars by means of control parameters is difficult. The first issue is how to choose intuitively meaningful parameters, use them to control the avatar's surface. Secondly, editing tasks such as subtle reshaping or pose alteration of even unclothed bodies requires care to ensure global consistency and agreement with human anatomy. It remains a tedious task even for skilful professional users, and it is all too easy to produce implausible results which look unnatural. Thus, for example, when increasing height, other body parameters such as chest and waist measurements should change in an appropriate way; when turning the head to look back over the shoulder, the upper-body should also twist appropriately at the same time. Thirdly, when the model is clothed, full-body editing is even more difficult. Not only must the body change shape appropriately, but the fabric should also behave in a natural way, while only being loosely attached to the body. Fourthly, it is well-known that non-rigid editing is intrinsically a non-linear problem lacking simple solutions [1].

The aim of the research in this paper is to make clothed 3D avatar editing tractable for unskilled users. Our approach is based on a revised parametric SCAPE (Shape Completion and Animation of PEople) model [2], which effectively constrains the parameters describing the avatar's surface. Unlike the original SCAPE model [2], our revised SCAPE approach represents variations in body shape and pose separately, using semantically meaningful parameters, providing intuitive control and at the same time permitting real-time computation.

We show how the parameters of this revised SCAPE model can be estimated directly from a mesh representing a clothed avatar. Using a novel body-aware warping scheme, changes to the parameters can be used to drive realistic deformation effects in realtime for the clothed 3D avatar. Editing the parameters can be used to effectively control the clothed surface.

The contribution of this paper is an easy-to-use system for plausible, realtime, shape and pose adjustment of a 3D clothed avatar (see Figure 1). Reshaping (i.e. changes to body shape) can be carried out by changing a few biometric shape parameters such as weight, while pose is determined directly from transforms computed by manipulating 'bones' approximating the human anatomy. We demonstrate that our system is effective for 3D human bodies in a variety of poses, with a variety of body shapes and garments.

2 Related Work

We now consider various 3D human body editing methods and related work; see [1,3,4] for comprehensive reviews of 3D object deformation.

Several approaches have been used to describe the space of 3D human shapes. The pioneering morphable models in [5] can describe variations in full-body shape across different people, but only for persons in similar poses. Such methods provide a direct way to explore the human shape space via intuitive controls which directly use a small set of semantic attributes such as weight and height. More advanced parametric body models [2,6–8] encode human body variability in terms of both pose and shape. SCAPE [2] is a particularly useful parametric deformation model, which separately accounts for shape variation between different people, and changes in pose. However, as Anguelov himself notes, directly using SCAPE for full body reconstruction is time-consuming, and is far too slow for realtime practical applications. We employ this model as the basis for parametric editing of 3D avatars, but modify it to meet the dual requirements of realtime performance and intuitive editing.

The SCAPE model has been applied to estimate human shape and pose parameters from various kinds of data, including multiple 2D images [9–11], a single image [12, 13], video [14], and 3D point clouds [15,11]. Chen et. al [16] proposed a tensor-based human body subspace model and showed its superiority for representing pose deformation. It gives excellent results when used with a Kinect to capture the human body. The approach used in each case is to define a cost function relating the input data points (on the silhouette if using 2D data) to a template mesh, and to formulate the problem as minimizing this cost with respect to the SCAPE model. We also use such a framework to estimate the parameters of our revised SCAPE model from the 3D mesh of an avatar.

Having acquired SCAPE model parameters, the methods in [13,14] provide the most similar editing tools to our system; they and we consider how to change the shape of a clothed person. However, they consider reshaping in 2D, for an image [13] or video [14]. Our system operates on a 3D mesh, overcoming limitations caused by view-dependent occlusion, and other issues that arise in 2D. Furthermore, as well as changing the avatar's shape, our system allows the user to also change its pose in an easy way.

3 Overview

Our system has two main components (see Figure 2). Offline, a preprocessing step carries out parametric body fitting to determine the parameters of our revised SCAPE model. Changing the parameters of the revised SCAPE model allows changes to the pose and body shape of the clothed 3D avatar via an online body-aware warping scheme.

During preprocessing, we firstly build a revised SCAPE model $M(\beta, \theta)$, where β denotes the shape parame-

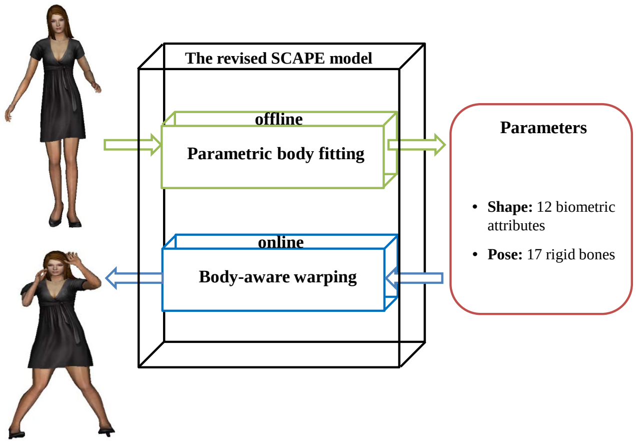

Fig. 2 System framework. Preprocessing: body fitting to the revised parametric SCAPE model. Online: reshaping and pose adjustment by changing the parameters of the SCAPE model, via the 3D body-aware warping scheme.

ters and θ denotes the pose parameters. In our revised SCAPE model, the shape and pose parameters are meaningful, unlike the unintuitive parameters of the original SCAPE model [2]. In our editing system, β represents biometric attributes such as height and weight, while θ captures the transformations of rigid 'bones' embedded in body parts such as the upper and lower arms and legs. Given an avatar A, we compute the unclothed instance $M^* = M(\beta^*, \theta^*)$ of this model which best describes the shape and pose information of A.

Using the computed SCAPE instance, the original 3D clothed avatar can readily be reshaped or put into a new pose with our easy-to-use system. During online editing, the parameters β^* and θ^* determining the shape and pose of M^* can be changed to create a performance. These details are then transferred from M^* to the 3D avatar A via a novel body-aware warping scheme. This is done using a graph G linking the unclothed mesh M^* and the clothed mesh A; its nodes are points sampled on the surfaces of M^* and A, and nearby nodes are connected by graph edges.

An example of editing is shown in Figure 3, which also illustrates the underlying unclothed SCAPE instance.

4 Preprocessing

During preprocessing, the parameters of the revised SCAPE model are computed for a given avatar, and its unclothed instance is also reconstructed at the same time. The warping graph is also built offline, linking the unclothed instance and the clothed avatar surface.

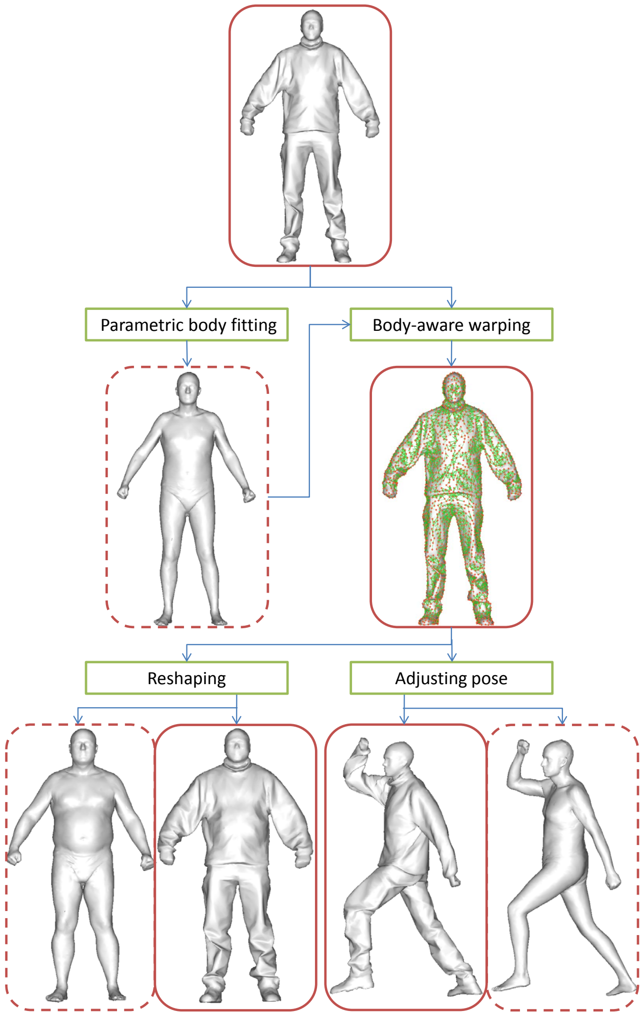

Fig. 3 Editing a clothed avatar. Given a clothed avatar (first row), we determine parameters of the corresponding unclothed body (left, second row) and build a warping graph relating the clothed and unclothed avatrs (right, second row). To change body shape, the unclothed body is deformed first (first box, third row); the nonrigid deformation is related to the warping graph. The warping graph is then used to transfer the deformation to the clothed avatar (second box, third row). A similar approach is used for pose adjustment (third and fourth boxes, third row). The underlying unclothed SCAPE instances are indicated by dashed boxes.

4.1 Learning the Revised SCAPE Model

The revised parametric SCAPE model represents variations in body shape and pose separately. It is a reduced model, i.e. having just a few parameters. Shape is modelled by a small vector β representing 12 biometric attributes, while pose is specified by transformations θ of 17 bones (parts such as hands are treated as a single rigid entity). Using a reduced representation is key to achieving high speed in SCAPE-based reconstruction. As the parameters are changed, the underlying

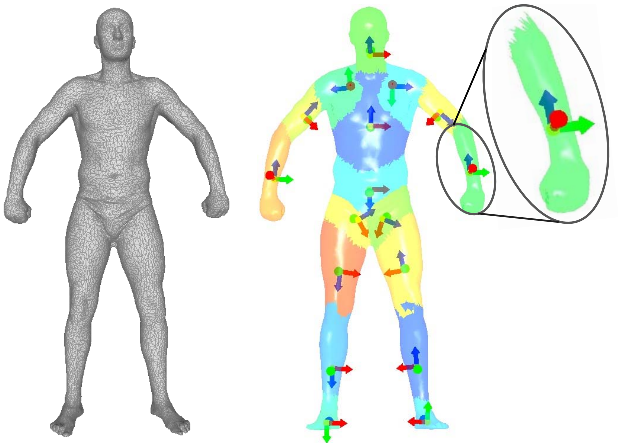

Fig. 4 Template mesh in the LBS representation. Left: template with 12,500 vertices and 25,000 faces. Right: weighted regions associated with its 17 bones.

unclothed instance can be rapidly determined, and the avatar updated, supporting interactive editing tasks in realtime.

The shape and pose space represented by our revised SCAPE model is data-driven, based on a database containing 2400 (almost) unclothed subjects in different 71 poses. We created this database using landmarked shape data purchased from CAESAR [17], and 71 poses of a single template, provided by [2]. Figure 4 illustrates the template T, which has $|V| = 12,500$ vertices and $|F| = 25,000$ faces. The database was then generated as in [2], by spanning the shape and pose dimensions for all subjects.

The revised SCAPE model is parameterized as follows. An individual in a given pose is obtained by matching the body template T to the data M^{sp} corresponding to a subject s in pose p in the database. Consider a triangle face f_k in T with vertices $(v_{k_1}, v_{k_2}, v_{k_3})$. Shape deformations S and pose deformations Q are applied in turn to transform it into its counterpart in M^{sp}. Deformations are applied to each triangle edge $e_{k_n} = v_{k_n} - v_{k_1}, n = 2, 3$, resulting in a deformed edge $e'_{k_n} = v'_{k_n} - v'_{k_1}$. The deformed mesh is reconstructed by finding the set $\mathbf{v}'$ of vertex locations $v'_1, \cdots, v'_{|V|}$ that minimizes the reconstruction error for the observed triangle edges:

$$\arg\min_{\theta,\beta,\mathbf{v}'} \sum_{k=1}^{|F|} \sum_{n=2,3} \|Q_k^{sp}(\theta) S_k^{sp}(\beta) e_{k_n} - e'_{k_n}\|^2. \quad (1)$$

Learning the shape deformation model. Solving the former minimization for the subject s provides a vector of $9 \times |V|$ elements containing the parameters of the shape matrices. The shape deformation matrices S^s can be expressed as a simple linear subspace, estimated by principal component analysis: $S^s(\beta) = U\beta^s + \mu$, where μ is the mean value, and U are eigenvectors. The parameter vector β^s of linear coefficients characterizes the shape of the particular subject s. In the revised SCAPE model, the parameters of shape space β^s are not arbitrary as in [2], but are 12 semantically meaningful biometric attributes: gender, height, weight, and size of chest, waist, hip, thigh, calf, body half height, shoulder, leg length, and arm length. These are related to the space by linear regression, following [5], and provide intuitive control over biometric variables.

Learning the pose deformation model. We model pose variation independently of body-shape variation. Pose deformation is represented by linear blending skinning (LBS), rather than articulated deformation as used in the original SCAPE formulation [2]. The main benefit of LBS representation is that we do not need to separate the rigid and non-rigid deformation as in [2], and instead handle pose deformation in a uniform way. LBS represents pose deformation Q using transformations θ of bones relative to the rest pose of template T, and skinning weights w. For a subject s, the weight $w_{k_n j}$ represents the influence of the j-th bone on the n-th edge of the k-th face. Each edge is associated with $|B| = 17$ bones in total, $e^r_{k_n}$ is the vector for the n-th edge on the k-th face in the rest pose, and θ^p_j is the transformation deforming the j-th bone in the p-th pose. The deformation is given by:

$$Q^{sp}_{k_n}(\theta) = \sum_{j=1}^{|B|} w_{k_n j} \theta^p_j e^r_{k_n}.$$

As in [18], the transformations and weights are determined for each subject in the database. Our experiments demonstrate that the weights vary little for different subjects, and hence in practice do not need redetermination for each new subject. Using an identical weight for all subjects provides surprisingly good reconstruction results, and simplifies the task of avatar editing. The only variables needed to represent the pose parameters Q are the rigid bone transformations θ.

Using the revised SCAPE model, we can now synthesize realistic meshes of various (unclothed) people in a broad range of poses. Given a particular set of biometric shape parameters β and rigid transformations θ, the corresponding unclothed instance can be generated using Equation 1. As shown in Figure 5, the user can intuitively control body shape via meaningful attributes such as height and weight. The two poses in Figure 5 are, for the purposes of making this figure,

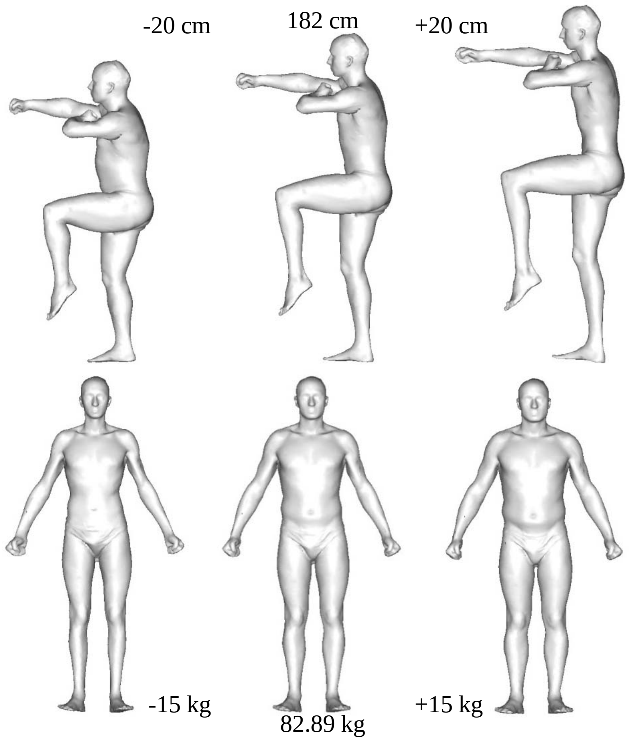

Fig. 5 Varying the template to change body shape and pose. Top: reducing and increasing height by 20 cm, bottom: reducing or increasing weight by 15 kg, using different poses in each case.

selected from the poses in the pre-built database, and transferred to the template.

4.2 Parametric Body Fitting

We now describe how we can edit a clothed avatar A. The first step is to estimate the shape and pose parameters for its corresponding SCAPE model, offline. That is, we find the SCAPE model instance $M^* = M(\beta^*, \theta^*)$, by minimizing the distance error between the deformed template T and A. Figure 3 illustrates such a SCAPE model instance produced by parametric body fitting. As indicated in Algorithm 1, an iterative approach is used to solve this problem.

Initialization. We first initialize the pose and shape parameters to suitable values; note that the avatar may be quite different from the template T in the SCAPE model. The pose parameters θ are manually coarsely adjusted to give a pose similar to the pose of A. The shape parameters β, such as height, weight, etc, are set to their mean values in the database.

Iteration. The solution is then iteratively computed. Each iteration includes three main steps:

Algorithm 1 Estimate pose and shape parameters for an avatar A

Input: template T and an avatar A
Output: β^*, θ^*

```
Initialization: set β and θ to appropriate initial values (see text);
i = 0;        //outer iteration counter
repeat
    update the point correspondences between T and A;
    k = 0;        //inner iteration counter
    repeat
        update T, with fixed β and θ;
        update θ, with fixed T and β;
        update β, with fixed T and θ;
        k++;
    until (k ≥ 20 or max_{j=1}^{n} ||v_j^k − v_j^{k−1}|| ≤ 0.001)
    i++;
until ( i = 10 or max_{j=1}^{n} ||v_j^i − v_j^{i−1}|| <= 0.005).
β* = β, θ* = θ;
```

1) *Fixing β and θ, update T to give the best correspondence between A and T.*

We build the closest correspondence set C between the unclothed body template T and the clothed avatar A as follows. For each vertex v of T, we find the closest vertex c in A. To discard incorrect correspondences, we use normal constraints. If the normal of v and the normal of c diverge by more than $30°$, we reject this correspondence.

Using the correspondence set C, we deform T to a new state on each triangle edge $e_{k_j} = v_{k_j} - v_{k_1} (j = 2, 3)$:

$$\arg\min_{\mathbf{v}'} \sum_{k=1}^{|F|} \sum_{j=2,3} ||Q_{k_j} S_k e_{k_j} - e'_{k_j}||^2 + \sum_{i=1}^{|C|} (\alpha_{point} ||v'_i - c_i||^2 + \alpha_{plane} ||n_{c_i}^T (v'_i - c_i)||^2), \quad (2)$$

where v' is the new position of each vertex of T after deformation, and v is the old position from the previous iteration. α_{point} and α_{plane} are set to 0.1 and 1 respectively in all experiments.

2) *Fixing β and T, update the pose parameters θ.*

We estimate the pose θ by computing the bone transformations which deform T into the pose which best approximates that of A. This is done by iteratively solving the following optimization problem:

$$\arg\min_{\theta} \sum_{k=1}^{|F|} \sum_{j=2,3} ||Q_{k_j}(\theta) S_k e_{k_j} - e'_{k_j}||^2 + \omega \sum_{b_1, b_2} ||\theta_{b_1} - \theta_{b_2}||^2, \quad (3)$$

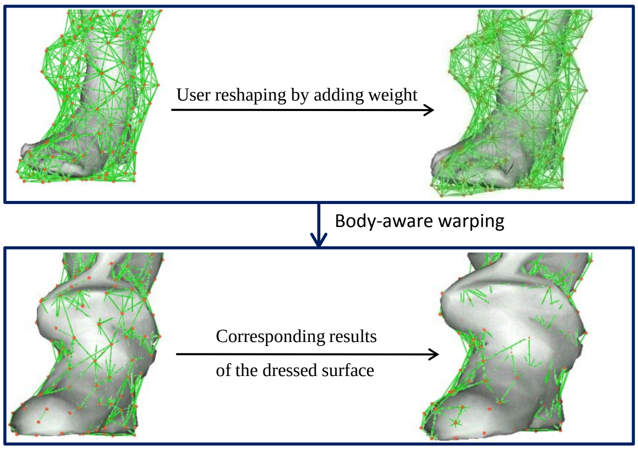

Fig. 6 Body-aware warping scheme. The weight of the example in Figure 3 is increased, changing the avatar surface, and thus the footwear too; the right foot in shown. The warping graph is shown as red nodes and green edges.

where bones b_1 and b_2 are adjacent bones. The second constraint is used to ensure that neighboring bones do not transform too differently from each other. In our experiments, we set $\omega = 0.1$.

3) *Fixing θ and T, update the shape parameters β.*

We estimate β by deforming T to the shape which best approximates A. Thus, β is updated by solving the following optimization problem:

$$\arg\min_{\beta} \sum_{k=1}^{|F|} \sum_{j=2,3} ||Q_{k_j} S_k(\beta) e_{k_j} - e'_{k_j}||^2. \quad (4)$$

Termination. This process causes T to gradually deform to agree with A. Iteration stops either after reaching a maximum number of iterations (experimentally set to 10) or when the residual fitting error is small enough. The final computed values of (β, θ) are output as the parameter values (β^*, θ^*), for the SCAPE model instance M^* of the clothed avatar A.

5 Parametric editing

Editing the clothed 3D avatar is performed by changing the parameters (β^*, θ^*) of M^*, the former controlling the shape, and the latter the avatar's pose. The parametric changes are transferred to the avatar's surface (including the attached garments) by our 3D body-aware warping scheme.

Figure 6 considers the right foot of an example in which the user has increased the weight of the model in Figure 3. This edit causes the foot to become fatter, which in turn affects the clothed surface too.

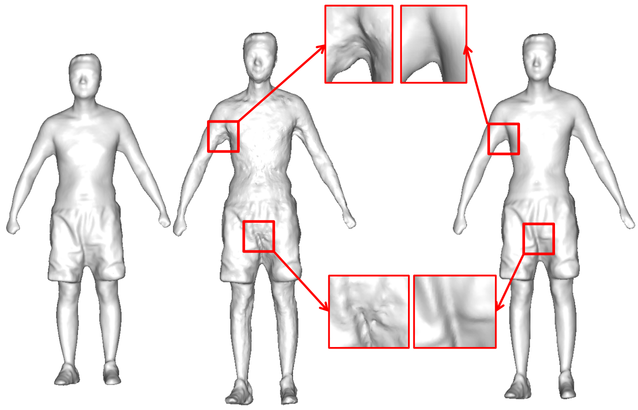

Fig. 7 Reshaping approach comparison, after increasing the height of a model. Left: visual artifacts when using the approach in [13]. Right: result using our method.

5.1 3D Body-Aware Warping

Our 3D body-aware warping scheme allows rapid propagation of changes from the revised SCAPE model to the clothed 3D avatar surface, via a graph G which links the fitted model M^* and the avatar A. When the parameters (β^*, θ^*) of the fitted model M^* are modified, this causes M^* and G to deform, which in turn deforms the 3D clothed avatar A accordingly.

Zhou [13] showed how to successfully transfer warping from a 3D mesh to a 2D image. Points are first sampled on the boundary of the original image space I and fitted template M. Then some edge pairs $P = \{(e_i, g_i)\}$ with $e_i \in R^2$ and $g_i \in R^3$ are edge vectors in I and M respectively connecting two adjacent boundary samples. Note that e_i and g_i lie along or perpendicular to the bone axis since deformation mostly happens in these two directions. The length changes of the g_i are transferred to e_i, thereby changing the original image I.

It is natural to consider extending the method to the 3D case. Unfortunately, this does not work well, as shown in Figure 7: the resulting 3D avatar surface includes undesirable artifacts. The reason is that (e_i, g_i) are always coplanar in the 2D case, while this can not be ensured in the 3D case. As a result, changes in e_i transferred to g_i may result in changes in the other direction. For example, even if e_i perpendicular to the bone axis, changes in g_i may undesirably result along the bone axis. The problem arises in that [13] considers how to change the shape, but does not consider pose adjustment.

Instead, before editing the model, we build a warping graph G linking the avatar A and its revised SCAPE instance M^*. The graph nodes are points sampled on the surfaces of M^* and A, and nearby nodes are connected by graph edges. Consequently, changes to the shape and pose of M^* affect G, and hence are transferred to A by graph deformation. We utilize the embedded graph idea proposed in [19]. However, that work only builds the embedded graph on the avatar surface A, whereas we also use M^*, to form a relationship between A and the parameters of M^*.

To speed up the computation, we do not use all vertices of M^* and A as nodes of G, but instead, employ a subset computed by a simple 3D Poisson sampling method [20]. The final number of nodes is $\min(\tau, (|V_{M^*}|+|V_A|))$, sampled from M^* and A in proportion to $|V_{M^*}|$ and $|V_A|$, the numbers of vertices of M^* and A respectively; τ is experimentally set to 7500. We then find the edges of G by connecting each node to the 6 other nodes closest to it.

As the user interactively manipulates the parameters of M^*, changes to it are transferred to A by affine transformations of the warping graph. This is effected by solving the following optimization problem, using a linear system. It has two terms, one to ensure deformation consistency (E_c) and one to apply position constraints (E_p), as explained below.

$$\mathop{\arg\min}_{R_1,t_1,\ldots,R_m,t_m} \quad E_c + w_p E_p. \tag{5}$$

The affine transformation for graph node j is specified by a 3×3 matrix R_j and a 3×1 translation vector t_j. We find $w_p = 0.1$ works well in all examples we have tested.

The idea for the consistency term comes from the embedded graph-node deformation method of [19]. The node positions are given by g_j, $j = 1, \ldots, m$, while the set of neighbors $N(j)$ comprises all nodes that share an edge with node j. If nodes j and k are neighbors, they affect a common subset of the embedded shape. The consistency term E_c expresses the idea that neighbors should transform in similar ways:

$$E_c = \sum_{j=1}^{m} \sum_{k \in N(j)} ||R_j(g_k - g_j) + g_j + t_j - (g_k + t_k)||^2. \tag{6}$$

The position constraint term is formulated as:

$$E_p = \sum_{j=1}^{m} ||v_j - v'_j||^2, \tag{7}$$

where v'_j is the new position of vertex v_j in M^* mesh.

After calculating the affine transformation for each node in G, the warping graph is used to deform the 3D clothed avatar vertex by vertex. The influence of individual graph nodes is smoothly blended, so that the deformed position v'_i of each avatar vertex v_i is a weighted sum after application of the warping graph G's affine transformations. In this way, the user readily can edit the clothed avatar A by changing intuitive pose and shape parameters.

Fig. 8 Pose adjustment approach comparison. Left: input. Center: dual-quaternion skinning results [3]; note the artifacts indicated. Right: our results.

5.2 Implementation

Our implementation uses various strategies to improve efficiency. We utilize the cholmod library [21] to solve Equations 1 and 5. They are both sparse systems, which allows us to achieve realtime performance as editing proceeds.

During reshaping, the user changes the 3D clothed avatar by adjusting sliders corresponding to a small set of semantic parameters. Each slider allows a predefined maximum amount of change for the corresponding parameter, e.g. for height adjustment, this is ±50 cm. Suitable ranges are determined from the human shape space provided by the body database, ensuring that the resulting 3D body shapes are visually plausible.

For pose adjustment, the user can manipulate the bones directly. When the user drags a bone, we calculate the rigid transformation for this bone and how it affects the transformation of all neighboring bones.

6 Results

We have evaluated the performance of our editing system using a large set of examples in various poses, and with various shapes and garments. See Figures 1–3, 5, 7–11.

We used a desktop PC with a 3.4GHz CPU to assess performance. Our approach is extremely fast after initial model building. For example, an avatar mesh comprising 12,500 vertices and 25,000 triangles requires about 50 s for preprocessing but only 31 ms to compute each subsequent pose update. Preprocessing time is dominated parametric body fitting and building the 3D body-aware warping graph. The current version could be further optimized by use of a GPU-based implementation.

After coarsely fitting a revised parametric SCAPE model to the input data, the user can manipulate various sliders interactively to change body shape. Figure 7 demonstrates results after increasing an avatar's height; equivalent results using the method in [13] are also shown. The latter leads to clearly visible visual artifacts on the avatar's surface, unlike our approach.

We compare our pose adjustment results to those from the well-known dual-quaternion skinning (DQS) [3] approach. As shown in Figure 8, our system effectively avoids two significant kinds of deformation artifact: candy-wrapper artifacts (caused by local twisting), and bulging.

Editing accuracy was evaluated using a bi-directional comparison involving three real persons (P_1, P_2, P_3), recorded by 3D capture providing anthropometric information at a given date and one year later. P_1 and P_2 were adults, who lost and gained weight respectively, P_3 was a boy who was initially 14 years old. The earlier data was treated as the reference model, and the latter was the target. For each scanned dataset, a revised SCAPE model was computed, and the corresponding shape and pose parameters obtained. We then performed forward and backward comparisons:

Forward comparison. We computed the L_2 distance error (normalised to the bounding box diagonal) between the target and the deformed reference after adjustment of pose and shape (taking into account the one-year change). Figure 9 shows P_1 as an example, including the reference, the target, and the deformed reference with color-coded L_2 distance error. The maximum L_2 distance error was 0.01, while the average distance error was 0.002. Editing the reference gave a result which was a good match to the target.

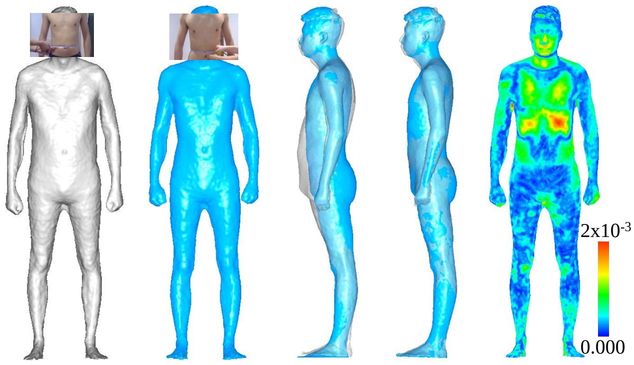

Fig. 9 Editing accuracy evaluated for subject P_1, measured at times one year apart. Left to right: reference, target one year later (head is masked by the waist-girth measuring picture), alignment of reference and target, match between deformed reference and target, and L_2 distance error.

Table 1 Differences in biometric values over a year for three real persons. In each pair, the first value is computed from the reconstructed model and the second is actual anthropometric information.

ID	Height (mm)		Weight (kg)		Chest Girth (mm)		Waist Girth (mm)		Hip Girth (mm)	
P_1	+2.0	+0.9	−3.9	−3.2	−2.2	−1.9	−9.3	−8.5	−2.7	−2.3
P_2	+1.1	+0.7	+4.0	+4.3	+4.0	+3.5	+20.6	+21.3	+14.5	+15.2
P_3	+80.0	+81.3	+12.3	+13.1	+33.9	+35.2	+48.8	+50.0	+41.7	+42.3

Backward comparison. Five biometric changes (height, weight, chest girth, waist girth, hip girth) were computed between the reference values and those for the constructed human models, and compared with actual differences measured for the corresponding real persons. Table 1 shows these biometric changes, demonstrating that the estimated anthropometric information give a good approximation to the real values.

Discussion. Parametric editing of a 3D clothed avatar is achieved using a 3D body-aware warping scheme, which uses a linking graph to couple the clothed avatar surface with its unclothed SCAPE instance. This approach greatly simplifies deformation transfer, and allows for efficient computation. Our method can also deal with human models in point cloud form. As shown in Figure 10, our system can still generate satisfying pose deformation results.

Figure 11 demonstrates a typical set of pose changes produced by our system, with a realtime transition between different poses. However, to achieve simplicity, this approach ignores the different physical properties of the body and garments, and on close examination, some mid-scale creases in the clothes do not plausibly change along with pose changes. In principle, a better choice is to separately edit the body and garment in a de-coupled way, and address editing of avatar bod-

Fig. 10 Pose adjustment of a point cloud human model. Left to right: input, fitted body, and four pose adjustment instances.

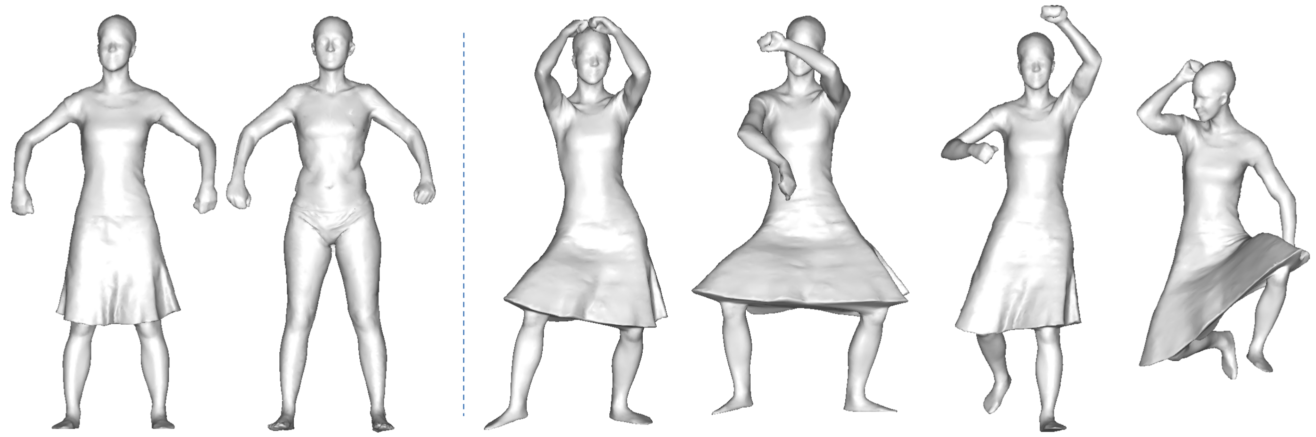

Fig. 11 Pose adjustment of a girl clothed a skirt. Left to right: input, fitted body, and four pose adjustment instances. The creases marked do not plausibly change as the pose is adjusted.

ies with realistic clothing as suggested by [22]. However, this is considerably more complex. Loose-fitting garments present difficulties for large pose changes in the current system. One further problem is that we do not take any steps to prevent global self-intersection of the deforming avatar; see [4]. Doing so would require collision detection and avoidance, which would add a significant computational overhead.

7 Conclusions

This paper has presented a new easy-to-use system for practical editing of 3D clothed avatars based on a variation of a parametric SCAPE model. Its parametric nature enables users to easily produce globally consistent and visually pleasing editing effects which are transferred to the avatar surface via a novel 3D body-aware warping scheme. Our experiments demonstrate that over a wide range of cases, our system allows simple and intuitive real-time editing. Reshaping is performed by manipulating a set of semantically meaningful shape attributes, while pose adjustment is performed by manipulating bones directly.

In the future, we wish to devise a more efficient human subspace model, since the current model needs to solve a large linear system which is still time consuming. One possible solution is to build a direct link between the final vertex positions and the original template vertex positions. It would also be a further improvement to decouple the clothes and the human body, and deform them individually. We can do this by building separate subspace models for the human body and clothes, and transfer shape or pose deformations to the body and clothes separately, as is done in [22]. Changes in human topology, e.g. when the hands are on the hips, affect the pose deformation process, and we propose to use a template with prior knowledge to refine clothed human models to solve this problem.

References

1. M. Botsch and O. Sorkine, “On linear variational surface deformation methods,” *IEEE Transactions on Visualiza-*

tion and Computer Graphics, vol. 14, no. 1, pp. 213–230, 2008.
2. D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "SCAPE: shape completion and animation of people," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 408–416, 2005.
3. L. Kavan, S. Collins, J. Žára, and C. O'Sullivan, "Skinning with dual quaternions," in *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games*, pp. 39–46, 2007.
4. R. Vaillant, L. Barthe, G. Guennebaud, M.-P. Cani, D. Rohmer, B. Wyvill, O. Gourmel, and M. Paulin, "Implicit skinning: Real-time skin deformation with contact modeling," *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 125:1–125:12, 2013.
5. B. Allen, B. Curless, and Z. Popović, "The space of human body shapes: Reconstruction and parameterization from range scans," in *SIGGRAPH*, pp. 587–594, 2003.
6. 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*, pp. 147–156, 2006.
7. F. Bogo, J. Romero, M. Loper, and M. J. Black, "FAUST: Dataset and evaluation for 3D mesh registration," in *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.
8. N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel, "A statistical model of human pose and body shape," *Computer Graphics Forum*, vol. 28, no. 2, pp. 337–346, 2009.
9. A. O. Bălan, L. Sigal, M. J. Black, J. E. Davis, and H. W. Haussecker, "Detailed human shape and pose from images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
10. A. O. Balan and M. J. Black, "The naked truth: Estimating body shape under clothing," in *10th European Conference on Computer Vision*, pp. 15–29, 2008.
11. A. Weiss, D. Hirshberg, and M. J. Black, "Home 3D body scans from noisy image and range data," in *International Conference on Computer Vision*, pp. 1951–1958, 2011.
12. P. Guan, A. Weiss, A. Balan, and M. J. Black, "Estimating human shape and pose from a single image," in *International Conference on Computer Vision*, pp. 1381–1388, 2009.
13. S. Zhou, H. Fu, L. Liu, D. Cohen-Or, and X. Han, "Parametric reshaping of human bodies in images," *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)*, vol. 29, no. 3, pp. Article No. 126, 1–10, 2010.
14. A. Jain, T. Thormählen, H.-P. Seidel, and C. Theobalt, "MovieReshape: tracking and reshaping of humans in videos," *ACM Transactions on Graphics (SIGGRAPH ASIA)*, vol. 29, no. 6, pp. 148:1–148:10, 2010.
15. N. Hasler, C. Stoll, B. Rosenhahn, T. Thormählen, and H.-P. Seidel, "Estimating body shape of dressed humans," *Computer Graphics*, vol. 33, no. 3, pp. 211–216, 2009.
16. Y. Chen, Z. Liu, and Z. Zhang, "Tensor-based human body modeling," in *CVPR*, pp. 105–112, 2013.
17. SAE, "Caesar 3-D anthropometric database, north american edition," 2013. http://store.sae.org/caesar/index.htm.
18. B. Le and Z. Deng, "Smooth skinning decomposition with rigid bones," *ACM Transaction of Graphics (SIGGRAPH ASIA)*, vol. 31, no. 6, pp. 199:1–199:10, 2012.
19. R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," *ACM Transactions on Graphics*, vol. 26, no. 3, 2007.
20. M. S. Ebeida, S. A. Mitchell, A. Patney, A. A. Davidson, and J. D. Owens, "A simple algorithm for maximal Poisson-disk sampling in high dimensions," *Computer Graphics Forum*, vol. 31, no. 2, pp. 785–794, 2012.
21. Y. Chen, T. A. Davis, W. W. Hager, and S. Rajamanickam, "Algorithm 887: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate.," *ACM Transactions on Mathematical Software*, vol. 35, no. 3, pp. 22:1–22:14, 2008.
22. P. Guan, D. Reiss, L.and Hirshberg, A. Weiss, and M. J. Black, "Drape: Dressnig any person," *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 31, no. 4, 2012.