Visual Comput (2005) 21: 542–550
DOI 10.1007/s00371-005-0343-x

ORIGINAL ARTICLE

Sweep-based human deformation

Dae-Eun Hyun
Seung-Hyun Yoon
Jung-Woo Chang
Joon-Kyung Seong
Myung-Soo Kim
Bert Jüttler

Published online: 1 September 2005

D.-E. Hyun (✉)
Samsung Electronics, Suwon, Korea
danny.hyun@samsung.com

S.-H. Yoon · J.-W. Chang · J.-K. Seong · M.-S. Kim
School of Computer Science and Engineering, Seoul National University, Seoul, Korea

B. Jüttler
Institute of Applied Geometry, Johannes Kepler University, Linz, Austria

Abstract We present a sweep-based approach to human body modeling and deformation. A rigid 3D human model, given as a polygonal mesh, is approximated with control sweep surfaces. The vertices on the mesh are bound to nearby sweep surfaces and then follow the deformation of the sweep surfaces as the model bends and twists its arms, legs, spine and neck. Anatomical features including bone-protrusion, muscle-bulge, and skin-folding are supported by a GPU-based collision detection procedure. The volumes of arms, legs, and torso are kept constant by a simple control using a volume integral formula for sweep surfaces. We demonstrate the effectiveness of this sweep-based human deformation in several test animation clips.

Keywords Human modeling and deformation · Sweep surface · Vertex binding · Shape blending · Anatomical features · Volume preservation

1 Introduction

Virtual human models play an important role in computer animation, including real-time applications in virtual reality and computer games. Advanced anatomy-based simulation techniques produce very realistic modeling and deformation of virtual humans [3, 15, 21]. However, they do not yet support real-time applications, for which vertex blending is now widely accepted as the method of choice [12–14, 20]. In this paper, we propose a sweep-based approach as an alternative that combines the advantages of these existing techniques. Important anatomical features and volumes are preserved, while computational cost remains about the same as vertex blending.

This paper was motivated by the recent results of Allen et al. [1, 2] in which 3D human models are reconstructed from range scan data and then animated, parametrized, and processed further for various applications. We have also adapted the extended linear blending scheme of Mohr and Gleicher [14] to become a sweep-based shape blending scheme. In [1, 14], the deformation of the human models is based on example data acquired from many different poses; our approach requires only a single pose. Additional poses can then be generated automatically. The user can also modify the shape of the model in a particular pose or during motion by editing and interpolating the underlying sweep surfaces.

Our sweep-based approach is similar to the human limb modeling and deformation technique proposed by Hyun et al. [9] and Karla et al. [10]. The main difference is that we represent a whole body, not just arms or legs, using sweep surfaces with elliptic cross-sections. The body shape is then precisely reconstructed from these sweep surfaces using displacement maps. One technical challenge is how to combine displacement surfaces around the shoulder and the hip, where different surfaces meet. We address this by applying a smooth shape blending scheme to the displacements from different surfaces.

We also support certain features of the anatomy-based approach, such as elbow-protrusion, muscle-bulge and skin-folding as an arm bends. These effects are re-

Fig. 1. 'Victoria' in a ballet motion

alized using a GPU-based collision-detection procedure based on the work of Govindaraju et al. [7]. The volume of a human model is preserved under deformation using a simple volume integral formula for sweep surfaces [17].

The main contributions of this work are summarized as follows:

- **Sweep-based control** of the human body for motions such as bending and twisting of the arms, legs, torso and neck. This approach provides highly effective control handles for animating and designing human motion.
- **Real-time performance** of the deformation algorithm. Our approach exhibits real-time capabilities for human models of reasonable complexity, e.g., those with less than 70K triangles, making it feasible for improved animation of human characters in virtual games and similar applications.
- Preserving **anatomical features** of the human body. If needed, the description can be extended to cover as many anatomical features as desired. Elbow and knee protrusions as well as non-penetrating skin folding are realized using a **GPU-based collision detection** procedure.
- **Volume preservation** of arms, legs, and torso as they deform. This is based on a simple integral formula for the swept volume of a moving ellipse.

2 Related Work

Hyun et al. [9] and Karla et al. [10] represent deformable arms and legs using sweep surfaces. However, they describe no specific method of connecting the limbs to the main body. In the current paper, we attack this more challenging problem by smoothly connecting arms and legs to shoulders and hip using a shape blending technique.

Mohr and Gleicher [14] proposed an extended linear blending scheme which adds joints to the interior of a link so as to make the link flexible. If we keep on adding such joints, we end up with infinitely flexible links, which are similar in effect to our sweep surfaces. There remains the issue of controlling infinitely many degrees of freedom. Our approach, based on B-spline interpolation, makes it considerably easier to modify the shape of the controlling sweep surfaces using a few "key" ellipses.

Allen et al. [1, 2] present algorithms that reconstruct, deform and parametrize human body shapes from range scan data. In [1], subdivision surfaces were used for the representation of skeleton-based global shape changes. Starck et al. [8, 19] control the deformation of a polygonal human model using a simplified mesh. In bending and twisting motions, sweep surfaces are easier to control than subdivision surfaces or polygon meshes. To demonstrate this, we present animation test clips in which virtual 3D human models walk and dance.

Singh and Kokkevis [18] propose a surface-oriented FFD for controlling the skin deformation of characters. They use a polygon mesh as a driver for the skin deformation. Our approach is closely related to theirs. Our improvement is in using control sweep surfaces which fit tightly to the surface of the model and thus control skin deformation more faithfully.

Many previous authors [3, 5, 15, 21] have presented anatomy-based physical simulation techniques that produce very realistic modeling and deformation of humans and animals. We do not propose full-scale anatomical simulation with our simplified representation. Nevertheless, we can support in real-time some important features, such as muscle-bulge, elbow-protrusion, skin-folding, and volume preservation. The real-time performance is the main advantage of our method over previous methods that rely on physical simulation.

Our algorithm works for arbitrary human models based on boundary representations, such as polygonal meshes, scan data, point clouds, subdivision surfaces, B-spline surfaces, etc. Thus it is relatively easy to adapt our shape control scheme to conventional techniques such as those of Allen et al. [2] and Seo and Magnenat-Thalmann [16] for manipulating human models.

3 Sweep-based human modeling

We approximate human arms, legs, torso and neck using sweep surfaces, which are then used as control structures for human body deformation. The vertices of a human model are bound to the sweep surfaces and then follow the transformations of the sweep surfaces. We apply a shape

blending scheme to the areas where different sweep surfaces meet.

3.1 Sweep surface of a moving ellipse

Sweeps are a powerful paradigm for representing 3D free-form objects based on simple procedural rules [6]. Examples include translational sweeps and rotational sweeps. General sweeps of 2D cross-sections are known as generalized cylinders. In this paper, we consider elliptic 2D cross-sections that approximate the cross-sectional shapes of various human body parts. An important advantage of using ellipses is the existence of a simple volume integral formula for their swept volumes.

When a standard ellipse of variable size $E_t(\theta) = (a(t)\cos\theta, b(t)\sin\theta, 0)^T$ is moving under rotation $R(t)$ and translation $C(t)$, it generates a sweep surface $S(\theta, t) = R(t)E_t(\theta) + C(t)$. The transformed ellipse $\hat{E}_t(\theta)$ is precisely described as follows:

$$\begin{aligned}\hat{E}_t(\theta) &= R(t)E_t(\theta) + C(t)\\ &= \begin{bmatrix} r_{11}(t) & r_{12}(t) & r_{13}(t)\\ r_{21}(t) & r_{22}(t) & r_{23}(t)\\ r_{31}(t) & r_{32}(t) & r_{33}(t)\end{bmatrix}\begin{bmatrix} a(t)\cos\theta\\ b(t)\sin\theta\\ 0\end{bmatrix} + \begin{bmatrix} x(t)\\ y(t)\\ z(t)\end{bmatrix}. \quad (1)\end{aligned}$$

The ellipse $\hat{E}_t(\theta)$ lies on a moving plane $P(t)$, which is determined by a point $C(t)$ and a unit normal vector $N(t) = (r_{13}(t), r_{23}(t), r_{33}(t))^T$, which is the third column of $R(t)$.

Figure 2 shows how a moving ellipse generates a sweep surface. Note that the first two columns of $R(t)$ form the major and minor axis directions of the moving ellipse, and that the normal vector $N(t)$ is not exactly in the same direction as the tangent vector $C'(t)$ of the trajectory curve $C(t)$.

When the sweep surface $S(\theta, t)$ has no self-intersections, it bounds a volume of magnitude

$$\int a(t)b(t)\pi \left\langle N(t), C'(t)\right\rangle dt,$$

where $a(t)b(t)\pi$ is the area of the moving ellipse $\hat{E}_t$ (see [17]). This integral formula greatly simplifies the volume preservation procedure discussed in Sect. 4.4.

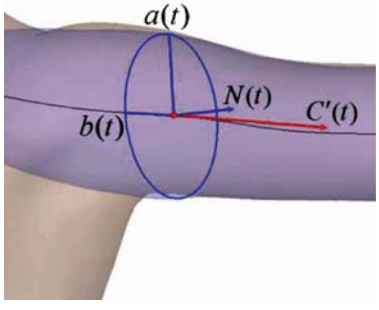

Fig. 2. Sweep surface generated by a moving ellipse

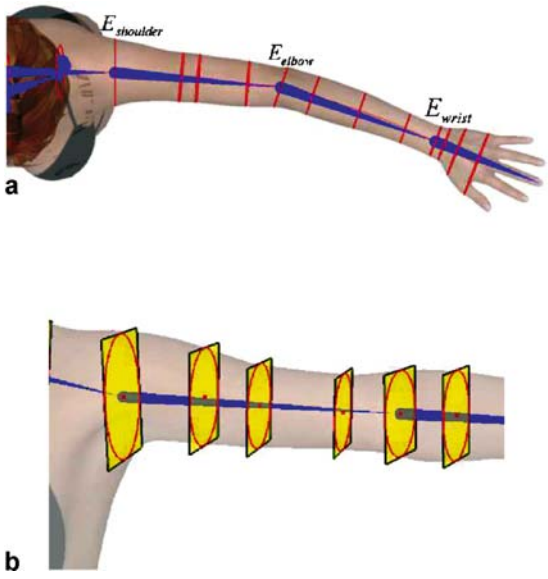

Fig. 3. Key ellipses for the left arm

3.2 Control sweep surface generation

Control sweep surfaces are generated by interpolating a set of "key" ellipses that tightly fit the cross-sections of human arms, legs, torso and neck. One key ellipse is assigned to each joint, and its center is fixed to that joint. The size and orientation of each key ellipse is determined by cutting the polygonal model through a plane orthogonal to the lower link of the joint and fitting the ellipse to the resulting cross-section while its center is still fixed to the joint. Additional key ellipses are assigned to intermediate locations on a link (see Fig. 3(a)). Each key ellipse is fitted to a cross-section of the model, cut through a plane orthogonal to the link (see Fig. 3(b)). Intermediate key ellipses have centers at the centroids of their cross-sections, which may not lie on the link.

The center positions of key ellipses are interpolated by a cubic B-spline curve $C(t)$ using a chord length parametrization for knot spacing. Their orientations are represented as unit quaternions, q_i, $i = 0, \ldots, n$, and linearly interpolated as follows:

$$Q_i(t) = (1-t)q_i + tq_{i+1}, \quad \text{for } t_i \le t \le t_{i+1}.$$

Note that the quaternion curve $Q_i(t)$ is unnormalized. We use a normalized curve $\frac{Q_i(t)}{\|Q_i(t)\|}$ to form a 3D rotation matrix $R(t)$, each term of which is a rational quadratic function of t. The half-axis lengths $a(t_i)$ and $b(t_i)$ of the key ellipses $\hat{E}_{t_i}$ are interpolated by the quadratic B-spline functions $a(t)$ and $b(t)$. The sweep surface $S(\theta, t)$ is constructed from $C(t)$, $R(t)$, $a(t)$, $b(t)$ using Eq. 1.

3.3 Binding a polygonal mesh to sweep surfaces

A sweep surface is a one-parameter family of ellipses. Each vertex V on a human polygonal model is bound to an instance of a moving ellipse, and follows the transformation of that ellipse. A vertex V is bound to an ellipse $\hat{E}_t(\theta)$ if it is contained in the moving plane $P(t)$:

$$< V - C(t), N(t) >= 0.$$

Solving an equation for each vertex V is time-consuming. Thus we sample the planes $P(t_j)$ at discrete intervals; and we bound each vertex V in-between two planes $P(t_j)$ and $P(t_{j+1})$ to an ellipse $E_{t_V}(\theta)$, where $t_V = \frac{d_{j+1}t_j + d_j t_{j+1}}{d_j + d_{j+1}}$, and d_j and d_{j+1} are the distances of V from the two planes.

3.4 Displacement maps

The vertex V is not exactly located on the ellipse $\hat{E}_{t_V}(\theta)$. We measure the relative displacement of V in the local frame of the moving ellipse, which is determined by the rotation $R(t_V)$ and translation $C(t_V)$. (Otherwise, the displacement would vary with the orientation of the sweep surface $S(\theta, t)$, which moves around in space as the human model swings its arms and legs.) Bringing the vertex V back to the coordinate system of the original ellipse $E_t(\theta)$, we have

$$R(t_V)^T (V - C(t_V)) = (a_V \cos\theta_V, b_V \sin\theta_V, c_V),$$

where $R(t_V)^T$ is the inverse of the rotation $R(t_V)$, and $c_V \approx 0$ is the error due to the linear interpolation of the binding parameter t_V. Ignoring the small component c_V, we represent the relative displacement $\boldsymbol{d}_V(\theta_V, t_V)$ of V from the moving ellipse $\hat{E}_{t_V}(\theta_V)$ as a 2D difference vector in the coordinate system of E_{t_V}:

$$\boldsymbol{d}_V(\theta_V, t_V) = ((a_V - a(t_V))\cos\theta_V, (b_V - b(t_V))\sin\theta_V, 0).$$

The original vertex V is approximately reconstructed as

$$V \approx S(\theta_V, t_V) + R(t_V)\boldsymbol{d}_V(\theta_V, t_V).$$

As the human model moves around in space and deforms its shape, the elements $S(\theta_V, t_V)$ and $R(t_V)$ will change continuously. However, the binding of V to the parameters (θ_V, t_V) and the displacement $\boldsymbol{d}_V(\theta_V, t_V)$ are fixed all the time.

A continuous displacement map $\boldsymbol{d}(\theta, t)$ may be constructed by approximating or interpolating a set of sample displacements $\boldsymbol{d}_V(\theta_V, t_V)$; this map defines the displacements at other locations. The original human model is then approximated by

$$V(\theta, t) \approx S(\theta, t) + R(t)\boldsymbol{d}(\theta, t).$$

Vertex binding is carried out at the "dress" pose, the initial position of the human model, at which the sweep surfaces have no self-intersections. Thus each vertex around a sweep surface will be bound to a unique moving ellipse $\hat{E}_t$. This binding is fixed for the rest of the deformation. As the arms and legs deform, the sweep surfaces and the polygonal model may start developing self-intersections. The interpenetration of deformed vertices with other body parts is prevented by a GPU-based collision detection procedure, and the skin starts developing creases and then folds. The details of this will be discussed in Sects. 4.3 and 4.4.

3.5 Blending sweep surfaces

Figure 4 shows a shoulder area, where vertices are bound to two different sweeps (one for the left arm and the other for the torso). When a vertex V is bound to the arm sweep, we have

$$V^a = S^a(\theta_V^a, t_V^a) + R^a(t_V^a)\boldsymbol{d}_V^a(\theta_V^a, t_V^a).$$

The same vertex V is similarly bound to the torso sweep:

$$V^t = S^t(\theta_V^t, t_V^t) + R^t(t_V^t)\boldsymbol{d}_V^t(\theta_V^t, t_V^t).$$

Initially, the two vertices V^a and V^t are located in the same position. As the shoulder angle changes, the two sweep surfaces deform quite differently and the two vertices V^a and V^t follow different paths. We compute the

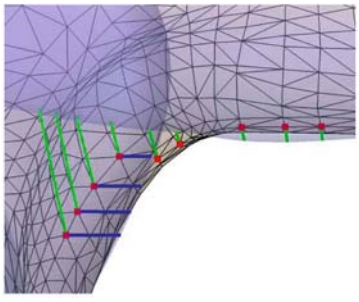

Fig. 4. Vertices bound to two different sweep surfaces

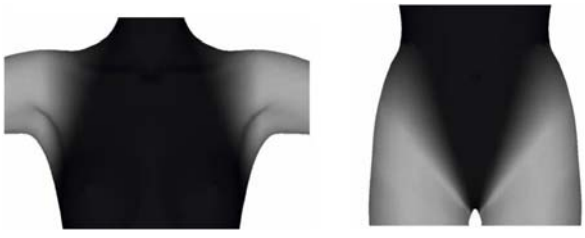

Fig. 5. Smooth transition of weight assignments

Fig. 6. Deformation of shoulder

final transformation of V as a convex combination of V^a and V^t:

$$V = (1 - w_V)V^a + w_V V^t,$$

where w_V is a weighting factor assigned to V. This weighting factor changes smoothly in the area where the different sweep surfaces meet.

Figure 5 shows how the body is segmented into separate regions (torso, arms, legs). The progression from one gray level to another represents the smooth transition of weight assignments. In this example, the weights in the transition area are determined by the relative distances from two bounding planes. Figure 6 shows the deformation of a human model in the region of the shoulder and armpit as the model lifts its left arm. The displacement is relatively large in the armpit. Nevertheless, the result of blending two sweep-based representations V^a and V^t of the polygonal model V produces a natural body deformation.

4 Sweep-based human deformation

Human figures in different poses are automatically generated at an interactive speed as the user controls the skeleton of a human model using inverse kinematics or motion capture data. The sweep surfaces deform as the joint angles change. The user can further edit the sweep surfaces to improve the visual realism of the shape deformation. Anatomical features are also supported by a GPU-based collision detection procedure. Volumes are preserved during shape deformation by automatically rescaling cross-sectional ellipses where the volume integral increases or decreases.

4.1 Sweep surface deformation

Sweep surfaces deform to follow changes to their key ellipses. Each key ellipse assigned to a joint has its center fixed to that joint, and its orientation changes with the joint angle as a function of the relative orientation of the two links connected to the joint. Other intermediate key ellipses follow the rigid motion of the link on which they are located. The trajectory curve $C(t)$ and the orientation curve $Q(t)$ are recomputed in each frame to reflect the changing positions and orientations of key ellipses. The scalar functions $a(t)$ and $b(t)$ are not updated at this stage. Section 4.4 discusses how to modify these scalar functions, so as to preserve the volume of the model under deformation. Figure 7 shows the deformation of a human model as it bends and spreads out its legs. This type of deformation is extremely difficult to carry out using conventional mesh processing techniques, because there will be many vertices clustered together in the crotch area of a human model.

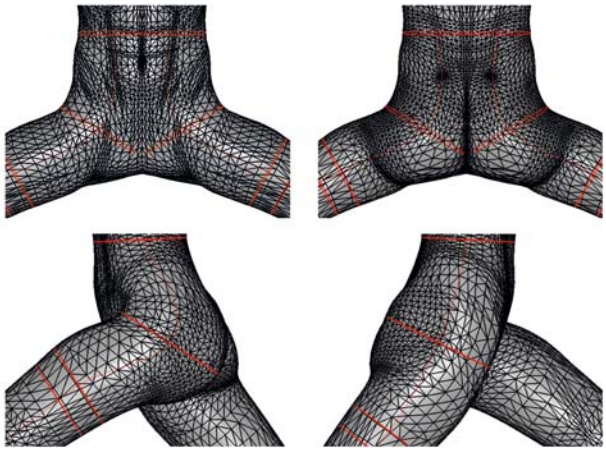

Fig. 7. Deformation as the legs bend and spread out

4.2 Editing a sweep-based deformation

At extreme joint angles, a sweep-based deformation may not appear quite as natural as a real human body in the same pose, because the sweep-based deformation does not consider the physical properties of skin and incorporates no anatomical knowledge. We allow the realism of such deformations to be enhanced by enabling the user to edit the control sweep surfaces.

The user can edit deformed human shapes by changing shape parameters for the key ellipses that generate the control sweep surfaces. The position, orientation and scaling of key ellipses can all be changed and the human

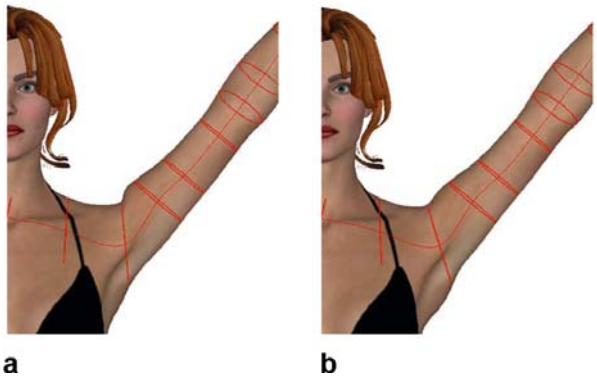

Fig. 8a,b. Editing sweep deformations: **a** an automatically generated shoulder and **b** the result of editing one of its key ellipses

model will deform to follow these changes. Figure 8(b) shows the result of changing the orientation of only one key ellipse from Fig. 8(a) to improve the deformation. The orientation of the key ellipse is a function of the shoulder joint angle. Thus the change of orientation at an extreme angle modifies this function smoothly using spline interpolation.

4.3 Anatomical features

Some anatomical features are very significant in the visual realism of body deformations: for example, elbow-protrusion, muscle-bulge, and skin-folding as an arm bends. We emulate these effects using a GPU-based collision-detection procedure.

Figure 9(a) shows the result of collision detection between the elbow-bone and the skin surface (shown in red), and also the result of detecting a self-intersection of the skin surface (shown in light green). Figure 9(b) shows similar results for knee bending. The polygon mesh around the collision region must deform according to the type of collision and its penetration depth. Our implementation of collision detection is based on the work of Govindaraju et al. [7], where GPU-based hardware programming was used to improve performance. For the computation of penetration depth, we use a software implementation.

A different approach is used to emulate muscle-bulge, an effect which is demonstrated in Fig. 10. As the arm bends, the binding of skin vertices to underlying muscles is activated by increasing the weights influencing the muscles. The user can further edit the extent of muscle-bulge by changing the sweep parameters. Other anatomical features can also be emulated by adding sweep surfaces and binding skin vertices to these sweeps.

The sweep surfaces are generated so that they have no self-intersections when the body is in the initial "dress" pose. Self-intersections start to develop in the region near

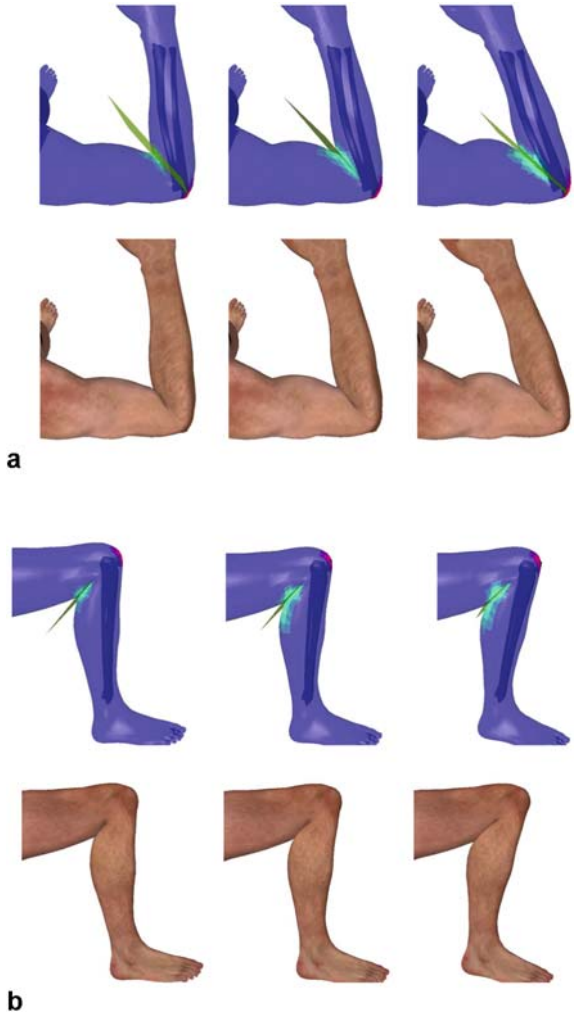

Fig. 9. Elbow and knee protrusions and skin-folding

a joint as the joint angle changes. A real human arm develops creases as it bends; and the skin folds as the arm or the leg bends further. We emulate these effects using a GPU-based collision detection procedure. Once all triangles in the region of self-intersection have been detected, we construct a bisecting plane for the self-intersection by fitting a plane (using least squares) to their vertices. The distance from this plane to each vertex gives the penetration depth of the vertex. By pulling the vertices back to the bisecting plane, we can generate skin-folding effects as an arm or leg bends. Figure 9 shows examples of skin-folding.

Elbow and knee protrusion effects are also realized using GPU-based collision detection. The penetration depth of each skin vertex in the collision region is computed by shooting a ray to the triangles of the bone in the collision region. The vertex is then pulled back to the exterior of the protruding bone.

Fig. 10. Muscle-bulge

4.4 Volume preservation

Botsch and Kobbelt [4] addressed a similar issue in bending polygonal models. They avoided severe folding by preserving local displacement volumes. In the case of human body deformation, the local volumes change quite dramatically as the result of complex interactions among muscles and bones. Thus we take a rather global approach that is based on a simple volume integral formula for sweep surfaces.

In Sect. 3.1, we formulated the volume of a sweep surface as

$$\int a(t)b(t)\pi \left\langle N(t), C'(t)\right\rangle dt,$$

where $a(t)b(t)\pi$ is the area of a cross-sectional ellipse $\hat{E}_t(\theta)$ of the sweep surface. In Sect. 4.1, the vector functions $N(t)$ and $C'(t)$ were modified as a result of the sweep surface deformation. Now we need to update the scalar functions $a(t)$ and $b(t)$, so as to preserve the volume of the deformed sweep surface. Let $\alpha_1, \cdots, \alpha_n$ and $\beta_1, \cdots, \beta_n$ denote the parameters that control changes of $a(t)$ and $b(t)$, which become $a(\alpha_1, \cdots, \alpha_n; t)$ and $b(\beta_1, \cdots, \beta_n; t)$. The volume integral is then represented as

$$I(\alpha_1, \cdots, \alpha_n, \beta_1, \cdots, \beta_n) = \int a(\alpha_1, \cdots, \alpha_n; t)b(\beta_1, \cdots, \beta_n; t)\pi \left\langle N(t), C'(t)\right\rangle dt.$$

Because the current volume I may be different from the initial volume I_0, the parameters $(\alpha_1, \cdots, \alpha_n, \beta_1, \cdots, \beta_n)$ are adjusted as follows:

$$(\Delta\alpha_1, \cdots, \Delta\alpha_n, \Delta\beta_1, \cdots, \Delta\beta_n) = \frac{I_0 - I}{\sum \left|\frac{\partial I}{\partial \alpha_i}\right|^2 + \sum \left|\frac{\partial I}{\partial \beta_i}\right|^2} \left(\frac{\partial I}{\partial \alpha_1}, \cdots, \frac{\partial I}{\partial \alpha_n}, \frac{\partial I}{\partial \beta_1}, \cdots, \frac{\partial I}{\partial \beta_n}\right).$$

The change in volume is large at the elbow as well as at the knee. As an arm or leg bends, the chord length of $C(t)$ increases substantially around the joint area. Initially, we reduce the size of the key ellipse assigned to the elbow or the knee joint. After two or three iterations, we consider other adjacent key ellipses as well and repeat the same procedure. Finally we consider the volume change I_C, which is the net increase due to bone-protrusion, muscle-bulge, and skin-folding. The target volume of the sweep surface for the next frame is set to $\hat{I}_0 = I_0 - I_C$, where I_0 is the initial volume of the sweep surface. Figure 11 shows the result of volume preservation as an arm bends. We assume that the extra volume due to the displacement map is constant and does not effect the volume change.

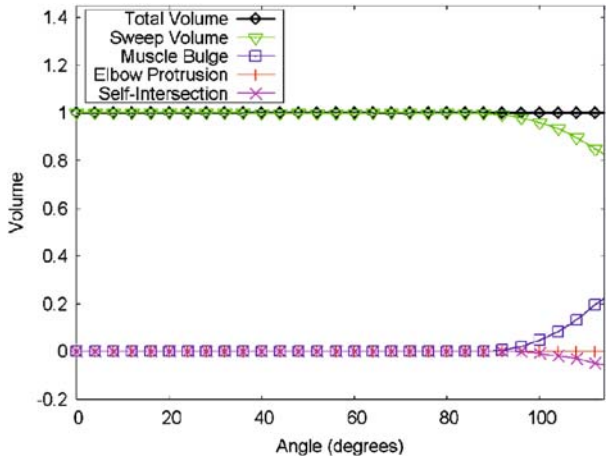

Fig. 11. Volume preservation as an arm bends

5 Experimental results

We have implemented our sweep-based human modeling and deformation algorithm in C++ on a P4-2GHz computer with a 1GB main memory and NVIDIA GeForce FX5900 GPU. Both the female model, "Victoria", and the male model, "Michael", were purchased from a commercial provider (http://www.daz3d.com). Each model has 72,712 vertices and 143,444 triangles. Motion capture data were re-targeted to these models using FilmBox [11].

Figure 12 shows snapshots from animation clips we have generated to demonstrate the effectiveness of our approach (http://3map.snu.ac.kr/hd.avi). Each animation was generated by our system at about 7 frames per second, including all rendering processes. However, this frame rate does not include the time spent on supporting special features such as anatomy or volume preservation.

In emulating the special features of an arm, our system generates 11–12 frames per second, not including the rendering time. A sweep-based arm deformation, including muscle-bulge, is applied to approximately 20,000 triangles in the arm. This takes about 30–35% of the total processing time for a bending arm. After that, GPU-based collision detection is applied to a smaller set of 4400 triangles, which takes about 60–65% of the processing time. Elbow-protrusion and skin-folding effects are computed using these triangles in a software implementation. This

Fig. 12a,b. Snapshots from animation clips: **a** "Victoria" in a ballet motion, and **b** "Michael" doing a 'techno' dance motion

process also includes the construction of the bisecting plane and computing the penetration depth of skin vertices. The performance for a leg model is slightly better, since the muscle-bulge effect is not considered in this case.

6 Conclusions

We have shown that sweep surfaces provide an excellent control mechanism for deforming polygonal meshes that represent human models. Once a model has been approximated with simple sweep surfaces, it is relatively easy to control the body shape using a small number of sweep parameters. Anatomical features are supported at an interactive speed using a GPU-based collision detection procedure. It is still difficult to apply all these features to a whole human model at an interactive speed. In future work, we plan to improve the efficiency of collision-handling algorithms that support anatomical features. We will also investigate the feasibility of applying this approach to controlling the shape of other freeform geometric objects such as virtual non-human characters.

Acknowledgement The authors would like to thank the anonymous reviewers for their invaluable comments. We are also grateful to Jieun Lee and Seo-Yeon Moon for their help in the animation and visualization of the human models. This work was supported by the Korean Ministry of Information and Communication (MIC) under the Program of IT Research Center on CGVR.

References

1. Allen, B., Curless, B., Popović, Z.: Articulated body deformation from range scan data. ACM Transactions on Graphics, **21**(3):612–619 (2002)
2. Allen, B., Curless, B., Popovic, Z.: The space of human body shapes: reconstruction and parametrization from range scans. ACM Transactions on Graphics, **21**(3):587–594 (2003)
3. Aubel, A., Thalmann, D.: Interactive modeling of the human musculature. In: Proceedings of Computer Animation 2001, 167–73 (2001)
4. Botsch, M., Kobbelt, L.: Multiresolution surface representation based on displacement volumes. Computer Graphics Forum, **22**(3):483–491 (2003)
5. Chadwick, J., Haumann, D., Parent, R.: Layered construction for deformable

animated characters. Proceedings of ACM SIGGRAPH 1989, 243–252 (1989)
6. Foley, J., van Dam, A., Feiner, S., Hughes, J.: Computer Graphics: Principles and Practice, 2nd Ed. Addison-Wesley, Reading, Mass (1990)
7. Govindaraju, N., Redon, S., Lin, M., Manocha, D.: CULLIDE: interactive collision detection between complex models in large environments using graphics hardware. In: Proceedings of Eurographics/SIGGRAPH Workshop on Graphics Hardware 2003, 25–32 (2003)
8. Hilton, A., Starck, J., Collins, G.: From 3D shape capture to animated models. In: Proceedings of 3D PVT 2002, 246–257 (2002)
9. Hyun, D.-E., Yoon, S.-H., Kim, M.-S., Jüttler, B.: 2003. Modeling and deformation of arms and legs based on ellipsoidal sweeping. In: Proceedings of Pacific Graphics 2003, IEEE Computer Society, 204–212 (2003)
10. Karla, P., Magnenat-Thalmann, N., Moccozet, L., Sannier, G., Aubel, A., Thalmann, D.: Real-time animation of realistic virtual humans. IEEE Computer Graphics and Applications, 42–56 (1998)
11. Kaydara Inc. FiLMBOX Reference Guide (2001)
12. Kry, P. G., James, D. L., Pai., D. K. Eigenskin: real time large deformation character skinning in hardware. In: Proc. of ACM SIGGRAPH Symposium on Computer Animation 2002, 153–160
13. Alias-Wavefront Technology: Maya 5.0 Users Manual(2003)
14. Mohr, A., Gleicher, M.: Building efficient, accurate character skins from examples. ACM Transactions on Graphics, **22**(3):562–568 (2003)
15. Scheepers, F., Parent, R., Carlson, W., May, S.: Anatomy-based modeling of the human musculature. Proceedings of ACM SIGGRAPH 1997, 163–172 (1997)
16. Seo, H., Magnenat-Thalmann, N.: An example-based approach to human body manipulation. Graphical Models 66, 1–23 (2004)
17. Sheynin, S., Tuzikov, A., Vasiliev, P.: Volume computation from nonparallel cross-section measurements. Pattern Recognition and Image Analysis 13, 1, 174–176 (2003)
18. Singh, K., Kokkevis, E.: Skinning characters using surface oriented free-form deformations. In: Proceedings of Graphics Interface 2000, 35–42 (2000)
19. Starck, J., Collins, G., Smoth, R., Hilton, A., Illingworth, J.: Animated statues. Machine Vision and Applications 14, 4, 248–259 (2002)
20. Wang, X. C., Phillips, C.: Multi-weight enveloping: least-squares approximation techniques for skin animation. In: Proceedings of ACM SIGGRAPH Symposium on Computer Animation 2002, 129–138 (2002)
21. Wilhelms, J., Gelder, A. V.: Anatomically based modeling. Proceedings of ACM SIGGRAPH 1997, 173–180 (1997)