Volume-preserving Mesh Skinning

Wolfram von Funck[1], Holger Theisel[2], Hans-Peter Seidel[1]

[1]MPI Informatik, Saarbrücken, Germany
[2]Visual Computing Group, University of Magdeburg
Email: {wfunck,hpseidel}@mpi-inf.mpg.de, theisel@isg.cs.uni-magdeburg.de

Abstract

We present a straightforward, yet effective approach to volume-preserving mesh skinning. After each skinned mesh deformation, a volume correction is applied which moves the vertices along an automatically constructed displacement field. The volume correction is exact and is computed directly by a closed-form solution. The animator can control the volume correction interactively to achieve custom effects like foldings or muscle bulging. We demonstrate the usability of our approach in several examples, especially the physically plausible deformation of characters. The volume preservation is also able to deal with extreme articulations without distortion artifacts.

1 Introduction

Mesh skinning is a standard technique which is widely used in Computer Graphics, especially in the context of character animation. The idea is to bind a mesh to a skeleton whose joints can be transformed in order to obtain a smooth non-rigid deformation of the surrounding mesh. The deformation of each mesh vertex is computed as a weighted blend of the joint transformations. While the method is intuitive and fast, producing convincing and physically plausible deformations can be quite challenging. Common problems are unnatural volume changes in deforming joint regions – examples are the well known 'collapsing joint' and 'candy wrapper' effects.

We present a technique which automatically preserves the volume for arbitrary skeletal mesh deformations. The volume-preservation can be adjusted manually by defining weights on the mesh surface. That way, the animator can precisely control how the volume is preserved to get more realistic deformations. The presented method has a number of advantages over existing volume-preserving deformation appraoches:

- The volume preservation is exact.
- The volume correction is obtained directly by a closed-form solution.
- Even during strong articulations the volume is preserved without distortion artifacts.
- No additional volumetric structures like control meshes, tetrahedral meshes or implicit deformers are required.
- The animator has precise control over the volume preservation. It is possible to define per-vertex how the volume should be preserved.

2 Related Work

Volume preservation is an important aspect in many application areas, especially in the context of shape deformation, where it has been addressed in a variety of research papers over the last years.

[7, 3] propose methods to preserve volume during free form deformations and [10] model objects using volume-preserving free form tensor-product solids. [6] use local volume controllers to guarantee volume conservation of implicitly described soft substances.

In the context of multiresolution modeling, [5] use volumetric elements between different multiresolution levels whose volumes are kept constant during deformation. That way, detail features are deformed more naturally, however the global volume is not preserved. [11] present a method for volume preservation of multiresolution meshes which is based on a quadratic minimization subject to a linearization of the volume constraint. [8] present a gradient domain technique which is able to preserve volume by using a nonlinear volume constraint. The subspace technique requires a coarse control mesh which encloses the actual mesh and is used to handle deformation energy and constraints efficiently.

Figure 1: Volume-preserving mesh skinning can be used to create natural-looking deformations. Even during extreme articulations the volume is exactly preserved without distortions.

[9] introduce a shape- and volume-preserving mesh editing technique where the mesh is represented by moving frames. The frames are scaled during deformation such that the volumetric shape properties are preserved.

[1] describe a space deformation technique which is able to preserve volume locally and globally, but is restricted to a tool with spherical influence that can be translated in space. [12] introduce a similar method based on constructing and integrating divergence-free vector fields. Here, the influence region can be defined using arbitrary implicit functions and the deformation is able to perform translations and rotations. While this method preserves volume and avoids self-intersections automatically, strong deformations, like bending an object by more than 90 degrees, produce heavy distortions as demonstrated in [13]. Here, the authors add more control to the vector-field-based method and moreover decrease distortions by defining the deformation influence on the mesh surface. [2] extend the rotational deformation of [12] to character skinning to get volume-preserving skeletal deformations. However, the method is not able to preserve volume during strong articulations without severe distortions. For this kind of deformation, the authors propose to gradually decrease volume-preservation to reduce the distortions.

3 Approach

Let us consider a three-dimensional triangle mesh with vertex positions $\mathbf{P} = [\mathbf{p}_1, ..., \mathbf{p}_n]$ and a triangulation $T \subseteq \{1, ..n\}^3$. In this notation, T contains for each triangle a triple of vertex indices. If the surface is a closed manifold, we can compute the volume of the shape as the sum of the signed volumes of the tetrahedra formed by each triangle and the origin:

$$\text{volume}(\mathbf{P}) = \frac{1}{6} \sum_{(i,j,k)\in T} \mathbf{p}_i \cdot (\mathbf{p}_j \times \mathbf{p}_k) \qquad (1)$$

In above formula '$\cdot$' denotes the scalar product and '$\times$' the cross product. Given a deformed mesh with vertex positions $\mathbf{P}' = [\mathbf{p}'_1, ..., \mathbf{p}'_n]$ and the same triangulation, we want to modify $\mathbf{P}'$ so that the volume of the deformed mesh is the same as the volume of the undeformed mesh. To do so, we define a displacement field $\mathbf{V} = [\mathbf{v}_1, ..., \mathbf{v}_n]$ which tells for each vertex position $\mathbf{p}'_i$ in which direction and how strong it has to be moved so that the original volume is reached. The displacement field can be freely defined as long as it changes the volume of the shape. The exact scaling of $\mathbf{V}$ can be computed automatically, as we will see below. Formally, we want the following condition to hold:

$$\text{volume}(\mathbf{P}' + \lambda \cdot \mathbf{V}) = \text{volume}(\mathbf{P}) \qquad (2)$$

λ is the factor by which $\mathbf{V}$ needs to be scaled. The remaining task is to find a λ which fulfills Equation 2. By insertion and reordering we get

$$\sum_{(i,j,k)\in T} (\mathbf{p}'_i + \lambda\mathbf{v}_i)((\mathbf{p}'_j + \lambda\mathbf{v}_j) \times (\mathbf{p}'_k + \lambda\mathbf{v}_k)) - \mathbf{p}_i(\mathbf{p}_j \times \mathbf{p}_k) = 0$$

$$\Leftrightarrow c_0 + \lambda c_1 + \lambda^2 c_2 + \lambda^3 c_3 = 0 \qquad (3)$$

with

$$c_0 = \sum_{(i,j,k)\in T} \mathbf{p}'_i(\mathbf{p}'_j \times \mathbf{p}'_k) - \mathbf{p}_i(\mathbf{p}_j \times \mathbf{p}_k),$$

$$c_1 = \sum_{(i,j,k)\in T} \mathbf{p}'_i(\mathbf{p}'_j \times \mathbf{v}_k) + \mathbf{p}'_i(\mathbf{v}_j \times \mathbf{p}'_k) + \mathbf{v}_i(\mathbf{p}'_j \times \mathbf{p}'_k),$$

$$c_2 = \sum_{(i,j,k)\in T} \mathbf{p}'_i(\mathbf{v}_j \times \mathbf{v}_k) + \mathbf{v}_i(\mathbf{p}'_j \times \mathbf{v}_k) + \mathbf{v}_i(\mathbf{v}_j \times \mathbf{p}'_k),$$

$$c_3 = \sum_{(i,j,k)\in T} \mathbf{v}_i(\mathbf{v}_j \times \mathbf{v}_k).$$

The cubic equation (3) has up to three real solutions λ[1]. Since we want to change $\mathbf{P}'$ as little as possible, we choose the solution whose absolute value is minimal.

In a nutshell, we can preserve the volume of arbitrary deformations in the following way:

1. Deform the mesh vertices $\mathbf{P}$ to get $\mathbf{P}'$.
2. Define a displacement field $\mathbf{V}$.
3. Compute scaling factor λ.
4. Set the final vertex positions to $\mathbf{P}' + \lambda\mathbf{V}$.

In the following section we will see how this approach can be applied to skeleton-based mesh skinning. We propose a method to automatically define $\mathbf{V}$ for arbitrary skeletal deformations and how to adjust $\mathbf{V}$ manually to get more complex volume-preserving deformations.

4 Volume-preserving mesh skinning

Standard mesh skinning usually works by binding a skeleton to a mesh where the influence of each joint has varying weights across the mesh vertices. The weights are set by the animator during the rigging process. The mesh is deformed by blending joint transformations with respect to these weights. Given k joints with transformation matrices $\mathbf{M}_1, ..., \mathbf{M}_k \in \mathbb{R}^{4\times4}$ and vertex weights $W_1, ..., W_k$ with $W_i = [w_{i,1}, ..., w_{i,n}]$, let the weights be defined such that they sum up to one for each vertex, i.e. $\sum_{i=1}^{k} w_{i,j} = 1$ for $j = 1, ..., n$. Each deformed vertex position is computed as a linear combination of the transformed bind positions:

$$\mathbf{P}' = \sum_{i}^{k} W_i \cdot \mathbf{M}_i(\mathbf{P}) \qquad (4)$$

[1]We use Cardano's method for solving cubic equations.

Here, '$\cdot$' denotes the componentwise scalar multiplication and $\mathbf{M}_i(\mathbf{P})$ is the transformation of all points $\mathbf{P}$ by joint matrix $\mathbf{M}_i$.

Since our volume preservation is global, we need to carry out the deformations at each joint one after another. We do this in hierarchical order, i.e. we start at the root joint and traverse recursively along the children. For each joint deformation, we apply our volume correction algorithm and thus can perserve the volume locally at the corresponding joint. After each deformation, $\mathbf{P}$ is replaced by $\mathbf{P}'$. To simplify matters, we consider in the following only one deformation at one joint, i.e. we assume that $\mathbf{P}$ and the bone transformations $\mathbf{M}_i$ are initialized properly.

First, we construct an initial displacement field $\mathbf{U}$ in bind pose which will be used to construct $\mathbf{V}$ for each joint deformation. Given start position $\mathbf{a}_i$ and end position $\mathbf{b}_i$ of each bone in bind pose, we can define a 3D vector field

$$\mathbf{d}_i(\mathbf{x}) = \mathbf{x} - (\mathbf{a}_i + \text{clamp}(\frac{(\mathbf{b}_i - \mathbf{a}_i)\cdot(\mathbf{x} - \mathbf{a}_i)}{\|\mathbf{b}_i - \mathbf{a}_i\|^2})(\mathbf{b}_i - \mathbf{a}_i)) \qquad (5)$$

with

$$\text{clamp}(\delta) = \begin{cases} 0 & \text{if} \quad \delta < 0 \\ 1 & \text{if} \quad \delta > 1 \\ \delta & \text{else.} \end{cases} \qquad (6)$$

Visually, $\mathbf{d}_i(\mathbf{x})$ is the vector from the closest point on bone segment $(\mathbf{a}_i, \mathbf{b}_i)$ to $\mathbf{x}$, as illustrated in Figure 2a.

Now we define $\mathbf{U}$ as the blended combination of all fields $\mathbf{d}_i$:

$$\mathbf{U} = \sum_{i}^{k} W_i \cdot \mathbf{d}_i(\mathbf{P}) \qquad (7)$$

That way, we get a smooth vector field pointing away from the skeleton (Figure 2b), which is ideal for our volume correction algorithm. We can now define the deformed field

$$\mathbf{U}' = \sum_{i}^{k} W_i \cdot (\mathbf{M}_i(\mathbf{P} + \mathbf{U}) - \mathbf{M}_i(\mathbf{P})) \qquad (8)$$

analogously to the skinning equation (4)[2]. Finally, the displacment vector field for the current deformation is defined as

$$\mathbf{V} = S \cdot \mathbf{U}' \qquad (9)$$

[2]$\mathbf{M}_i(\mathbf{P} + \mathbf{U}) - \mathbf{M}_i(\mathbf{P})$ is used to transform $\mathbf{U}$ without translation.

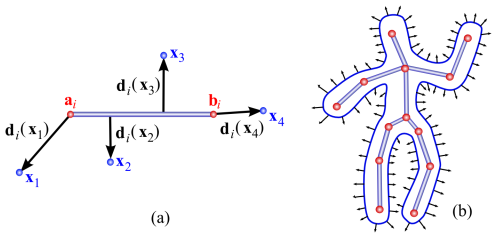

Figure 2: (a) The vector field $\mathbf{d}_i$ evaluated at a number of sample points $\mathbf{x}_j$. (b) We obtain $\mathbf{U}$ by blending the fields $\mathbf{d}_i$ at the mesh vertices.

where $S = [s_1, ...s_n]$ defines a weighing factor at each vertex which basically describes how much the volume correction should affect this vertex. As we will see in Section 5, S can be defined manually on the vertices to get complex, user defined deformations. However, there is also an automatic method to generate useful weights at a joint. Given the current joint j and its parent joint i, we simply set $S = W_i \cdot W_j$. In most cases, S has large weights near the joint and decreasing weights away from the joint. That way, $\mathbf{V}$ is only non-zero close to the joint and thus the volume is corrected locally in that region.

5 Applications

Using the automatic displacement field construction described in the previous chapter, it is possible to directly preserve the volume of skinned meshes without user interaction. Figure 3 shows an example of a bar-shaped mesh which is skinned using a simple skeleton. Using our method, the volume is uniformly and smoothly preserved around the joint regions which gives the impression of deforming a real incompressible solid.

In some cases, uniform volume preservation is undesirable, e.g. when the arm of a human character is bent, we don't want volume preservation at the hard elbow but rather at the soft muscle and tissue areas near that joint. The animator can model this with our approach by painting the weights S by which $\mathbf{U}'$ is multiplied (see Equation 9) interactively on the vertices. He is not restricted to use positive weights, also negative weights can be used to achieve interesting effects like muscle bulging or foldings, as demonstrated in Figure 4.

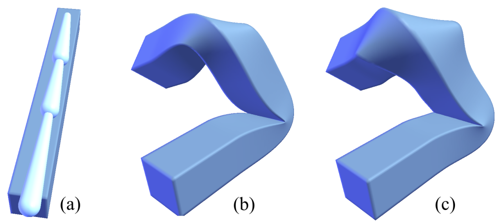

Figure 3: (a) A bar is skinned using a skeleton consisting of three bones. (b) The deformed mesh using standard mesh skinning and (c) using volume-preserving mesh skinning.

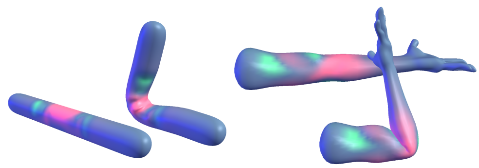

Figure 4: The animator can paint positive (pink) or negative (green) weights interactively on the surface. That way, soft tissue deformations and foldings can be modeled.

Figure 1 demonstrates how this technique can be used to model realistic skin and muscle deformations. Figure 5 shows a comparison between traditional mesh skinning and volume-preserving mesh skinning. In Figure 5b, where standard mesh skinning was used to twist the arm of a model at the shoulder joint, the 'candy wrapper' effect is clearly visible in the shoulder region. This unnatural loss of volume can be prevented with volume-preserving mesh skinning, as shown in Figure 5c, where the shoulder region deforms more naturally without volume change. Note that, while the method is able to preserve the volume, it is not able to prevent triangle distortions which are typical for the candy wrapper effect, resulting in wrinkles in the shoulder region.

All models shown in the images and the accompanying video were posed in real-time. The deformation of a mesh consisting of 20,000 triangles takes about 28 milliseconds per joint on a 2GHz notebook.

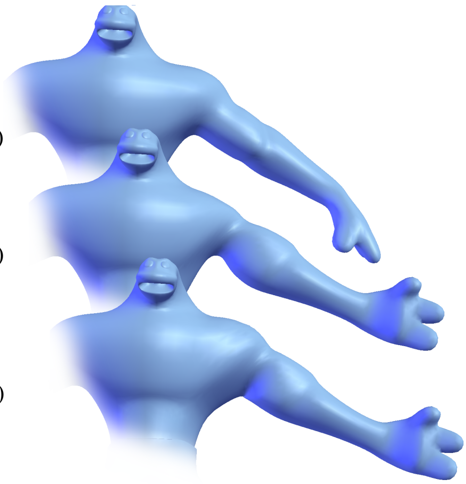

Figure 5: (a) A character in bind pose. (b) Using traditional mesh skinning, the twisting deformation produces a 'candy wrapper' effect in the shoulder region, while volume-preserving mesh skinning (c) deforms the shoulder naturally.

6 Conclusions

We introduced a straightforward, but nevertheless effective method to perform mesh skinning with volume-preservation. The key contributions are:

Exact volume preservation: The accuracy of volume preservation often depends on the mesh resolution or the numerical accuracy of the algorithm [12, 2, 9, 11]. Our approach can handle arbitrary coarse meshes and does not rely on a numerical approximation, but preserves the volume exactly by a closed-form solution.

Closed-form solution: In contrast to previous volume preservation methods, our approach neither requires non-linear opimizations [8] nor numerical vector field integration [12, 2]. Especially the vector field integration methods have the drawback that large deformations require longer integrations [12]. Our solution is obtained directly, no matter if the deformation is small or large.

Strong articulations: Strong articulations, like bending an arm by more than 90 degrees, are often problematic and lead to distortions near the joint area [13, 2]. Instead of turning off volume preservation for such deformations [2], our method can handle strong articulations without problem.

Surface-based volume correction: Volume-preserving mesh skinning does not require control meshes [8], implicit deformers [12, 2] or multiresolution representations [11]. The computation is carried out on the mesh surface and can be controlled on a per-vertex basis: The animator can specify a scalar field on the surface which steers the behavior of the volume correction during deformation. While vector-field based approaches like [2] model tissue variation by 3D implicit functions, our approach can handle arbitrary complex variations without performance impact since the variation is defined on the vertices. In fact, the runtime of our algorithm is proportional to the number of vertices times the number of joints that affect the deformation.

Our approach has some limitations compared to existing deformation techniques. In particular, it does not preserve local surface features and does not automatically prevent self-intersections. However, traditional mesh skinning, which is still a standard in character animation, doesn't have these features either: Feature preservation is usually not required for characters and self-intersections resulting from strong articulations are quite common during character posing [4]. For instance, the pose of the rightmost character in Figure 1 produces self-intersections which are not noticeable at all. In fact, automatic prevention of self-intersections and volume preservation at the same time during strong articulations is not possible without distortion artifacts up to now [13, 2]. We believe that it is a good compromise to keep the volume preservation and leave the prevention of self-intersections up to the animator.

In the future, we would like to investigate the applicability of our approach to other areas like facial animation, surface-based mesh editing and deformation via control meshes.

References

[1] A. Angelidis, M.-P. Cani, G. Wyvill, and S. King. Swirling-sweepers: Constant-volume modeling. In *Computer Graphics and Applications, 12th Pacific Conference on (PG'04)*, pages 10–15, 2004.

[2] Alexis Angelidis and Karan Singh. Kinodynamic skinning using volume-preserving deformations. In *SCA '07: Proceedings of the*

2007 ACM SIGGRAPH/Eurographics symposium on Computer animation, pages 129–140, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.

[3] F. Aubert and D. Bechmann. Volume-preserving space deformation. *Comput. and Graphics*, 21(5):6125–639, 1997.

[4] David Baraff, Andrew Witkin, and Michael Kass. Untangling cloth. *ACM Trans. Graph.*, 22(3):862–870, 2003.

[5] M. Botsch and L. Kobbelt. Multiresolution surface representation based on displacement volumes. *Computer Graphics Forum*, 22(3):483–491, 2003. (Proceedings Eurographics 2003).

[6] M. Desbrun and M.-P. Gascuel. Animating soft substances with implicit surfaces. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 287–290, New York, NY, USA, 1995. ACM Press.

[7] G. Hirota, R. Maheshwari, and M. Lin. Fast volume-preserving free form deformation using multi-level optimization. In *Proceedings Solid Modeling and applications*, pages 234–245, 1992.

[8] Jin Huang, Xiaohan Shi, Xinguo Liu, Kun Zhou, Li-Yi Wei, Shang-Hua Teng, Hujun Bao, Baining Guo, and Heung-Yeung Shum. Subspace gradient domain mesh deformation. *ACM Trans. Graph.*, 25(3):1126–1134, 2006.

[9] Yaron Lipman, Daniel Cohen-Or, Ran Gal, and David Levin. Volume and shape preservation via moving frame manipulation. *ACM Trans. Graph.*, 26(1):5, 2007.

[10] A. Rappoport, A. Sheffer, and M. Bercovier. Volume-preserving free-form solids. *IEEE Transactions on Visualization and Computer Graphics*, 2(1):19–27, 1996.

[11] Basile Sauvage, Stefanie Hahmann, and Georges-Pierre Bonneau. Volume preservation of multiresolution meshes. *Computer Graphics Forum*, 2007. Proceedings of Eurographics'2007.

[12] W. von Funck, H. Theisel, and H.-P. Seidel. Vector field based shape deformations. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, pages 1118–1125, New York, NY, USA, 2006. ACM Press.

[13] Wolfram von Funck, Holger Theisel, and Hans-Peter Seidel. Explicit control of vector field based shape deformations. In *Proceedings of Pacific Graphics 2007*, 2007.