Soft-tissue Deformation for In Vivo Volume Animation

Taehyun Rhee[1] J.P. Lewis[2] Ulrich Neumann[1] Krishna Nayak[1]

University of Southern California[1] Stanford University[2]

{trhee/uneumann/knayak}@usc.edu zilla@computer.org

Abstract

Articulated body animation with smooth skin deformation is an important topic in computer graphics. This paper presents a pipeline that extends articulated body deformation to the volume graphics domain. The pipeline consists of in-vivo volume scans, kinematic joint estimation, volumetric joint weight computation, soft-tissue volume deformation, and direct volume rendering. The result is a fully articulated body volume driven by intuitive joint control that respects rigid deformation of the bone structures and produces smooth deformations of both the skin surface and the interior soft tissue regions.

1. Introduction

Volume graphics has many advantages, since it represents the 3D information of both surface appearance and the underlying interior structure. Especially, volume data obtained from medical image scans (e.g. MRI or CT) includes the bones, muscles, tendons, skin, and other anatomical structures. Transparent volume rendering can successively visualize inner organs without loosing the overall context of the subject.

Until recently, volume animation has been a difficult issue however, simply due to the addition of the time dimension to the already voluminous data. Although volume animation is now somewhat practical, articulated volume animation requires additional considerations. First, as joints move, the hard tissue (bones) should not be deformed, while the soft tissue volume and skin surface should be deformed continuously and without tearing. Second, while volume data of articulated body regions can be obtained from in vivo scans of living subjects, the use of in vivo data presents difficulties that are not present when the data are obtained from cadavers or other sources, such as motion artifacts and other limitations discussed in section 3. Third, the manual weight tuning required in many skinning algorithms is generally impractical in the case of volume deformation, both due to the much larger number of weights (one per voxel) that would be required and because simplified "animation skeletons" cannot be adopted in the volume case where the actual skeleton is present in the data.

In this paper, the issues described above are successively addressed, resulting a complete pipeline for rapidly producing an animatable volume model of an articulated body driven by standard intuitive joint control. The generated articulated volume model correctly produces both rigid deformations of interior bones (hard tissue) and appropriate deformation of the skin surface, while providing smooth deformation of the interior soft tissue regions. Volume animation of an articulated subject has potential uses in applications such as medical education, virtual medicine, bio mechanics, volume manipulation, and other areas.

2 Related work

Recently, volume animation has received attention in computer animation and volume graphics. However, the works have been mainly devoted to free-form volume deformation [3, 13] and have not been designed for articulated body animation involving high range of motion.

Gavani et al. [4] developed a volume animation system for articulated objects. An arbitrary pose is produced by transformed skeletal points and the whole volume is regrown by the volume reconstruction. The method is successfully integrated into a traditional animation system such as Maya, and the volume is manipulated by abstract skeletal control. Although the method produces volume animation of articulated bodies, they have not considered realistic soft-tissue deformations arising from motion. Therefore, their method can produce erroneous artifacts such as bone bending in some postures. Also, since the skeleton is produced by volume thinning and the joints are simply approximated, the constructed joints cannot guarantee proper movements of bones. We entrust the detail review of the volume animation and skin deformation to the appropriate related papers.

3 In vivo volume scan

In vivo scanning of complex subjects such as the human hand is challenging. CT scanning can provide 3D imaging with fine spatial resolution [8] and straightforward segmentation, but involves exposure to harmful ionizing radiation.

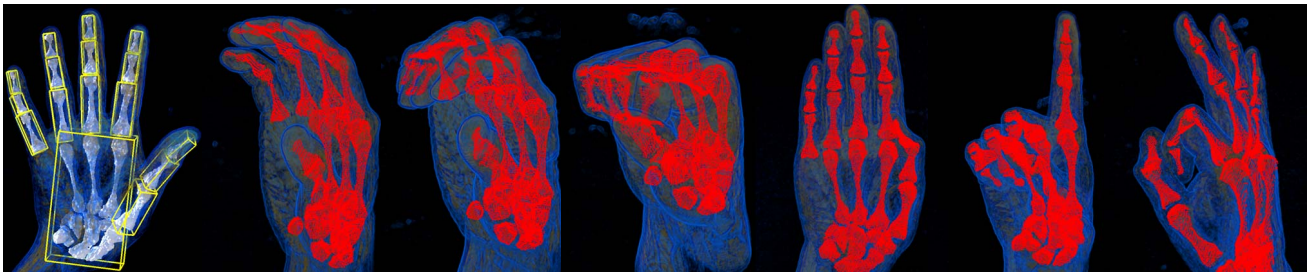


Figure 1. Bone registrations across poses: the left most image is the base volume including marching cube meshes of extracted bones and their OBBs (yellow boxes); The base bones are then registered (red points) to the other six poses.

We chose to use MRI, because it is non-invasive and involves no ionizing radiation, and can capture the required 3D volume information. MRI scans of the human hand in various poses were performed with careful consideration of scan-time, signal-to-noise ratio (SNR), subject comfort, and potential for motion artifacts. Supporting casts were made to minimize subject discomfort and motion artifacts. These materials have virtually no moisture, and produce low MR signals.

4 Joint skeleton modeling

A joint skeleton model is required for realistic motion generation and intuitive joint control. The joint skeleton of an in vivo subject volume can be generated by anatomical analysis of the bone skeleton or bio-mechanical joint estimation from multiple samples [8, 10].

4.1 Hard Tissue (Bone) extraction

In case of in vivo MRI, image quality is highly limited. Furthermore, bone segmention from hand MRIs is particularly difficult and manual segmentation is unavoidable, because automatic segmentation using simple methods such as thresholding cannot be used [7, 10]. In order to minimize the manual work, only the bones of a single base volume are extracted using 2D graph cuts algorithm [12]. Then, the bones are registered to the other poses' samples.

4.2 Bone registration

In a pre-processing step, the iso-surfaces of the volume are extracted using the marching cubes algorithm. Voxels identified to be inside the volume are stored as the *representative volume* to speed subsequent processing. The oriented bounding box (OBB) of each bone is created from three eigenvectors of the covariance matrix of the volume [10]. The centers of the OBBs' base are assigned as the origin of the OBB coordinates of the bone. The OBBs of the bone and their representative volumes are transformed to the origin of the world coordinate frame. The extracted bones and OBBs of the base pose are shown in figure 1.

We wish to find T_j^k, the matrix that transforms b_j^0 (the OBB of bone j of pose 0 in the world center) to b_j^k (the OBB of bone j in pose k). Intensity based volume registration require manual initialization [7] and the transformation is denoted by $\tilde{T_j^k}$. The approximated bone volumes are registered to the MRI scan of the target pose by optimizing

$$\min_T \sum_{i=1}^{n} |I(v_i) - J(Tv_i)|^2 \tag{1}$$

where T is T_j^k initialized by $\tilde{T_j^k}$, n is the number of voxels in the representative volume, v_i is the location of the ith representative voxel, Tv_i is the transformed voxel location, $I()$ is the intensity value of the volume in the base pose 0, and $J()$ is the intensity value of the volume in the target pose k. A quasi-Newton methods is used as the optimization solver. The registered bones are shown in Figure 1.

4.3 Kinematic joint estimation

From the sample poses, the center of rotation (COR) c_j of joint j can be estimated by minimizing the error defined in equation 2, because the COR does not change under rotation when viewed in the parent joint's coordinates [8],

$$\min_{c_j} \sum_{k=1}^{m-1} |T_j^{0\to k} c_j - c_j|^2 \tag{2}$$

where, c_j is the COR of the joint j in the parent joint coordinates, m is the number of the sample poses (pose 0 will be excluded), and $T_j^{0\to k}$ is the relative (homogeneous coordinates) transformation of the joint j from the pose 0 to pose k after registering the parent bone of the pose k to the parent bone of the pose 0. The c_j of each joint is calculated by minimizing equation 2, and the result is shown in Figure 2.

5 Soft-tissue volume deformation

The Skeletal Subspace Deformation (SSD) algorithm is based on the weighted blending of affine transforms determined by each joint [9],

$$v_a = (\sum_{j=1}^{n_{joint}} w_j T_j) v_0 \tag{3}$$

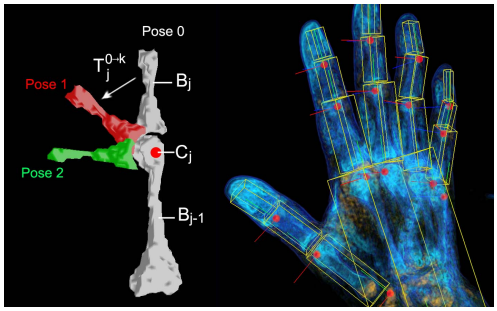


Figure 2. Joint Estimation: B_{j-1} is parent bone and B_j is transferred to pose k by matrix $T_j^{0\rightarrow k}$; Red dots in right image shows estimated CORs of each joint.

where n_{joint} is the number of joints, v_0 is a vertex in the base pose 0, v_a is a deformed vertex in an arbitrary pose a, T_j is a world coordinate transformation matrix to transfer joint j from base pose 0 to the arbitrary pose a, and w_j is a joint weight that defines the contribution of joint j's transformations. Although equation 3 is usually used for surface deformations, it can be extended to volume deformation if we substitute v from vertex to voxel and calculate the weight w_j for each voxel.

5.1 Joint weights from samples

In each sample pose k, we have following equations based on SSD:

$$\tilde{v}_k - e_k = (\sum_{j=1}^{n_{joint}} v_j w_j) \tag{4}$$

where $\tilde{v}_k$ is a particular vertex from pose k, the right hand side is the SSD deformation, v_j is v_0 transformed by T_j in equation 3, e_k is a displacement between the SSD deformation and $\tilde{v}_k$, and the other variables are as in equation 3.

If we have sufficient pose samples least square methods can estimate the weights that minimize the e_k after substituting the equations 4 into a linear matrix system [11]. The non-negative least square (NNLS) method can solve this problem [5, 11], and it determines positive weight values minimizing error in equation 4.

5.2 Volume weights for in vivo subject

The automatic determination of skinning weights in section 5.1 is expensive for volume data. In the case of a complex body having large DOF (e.g. the human hand), direct capture of a sufficient number of volume samples with adequate pose variability using MRI is expensive and difficult (for example, independently controlling the ring finger is difficult for some people).

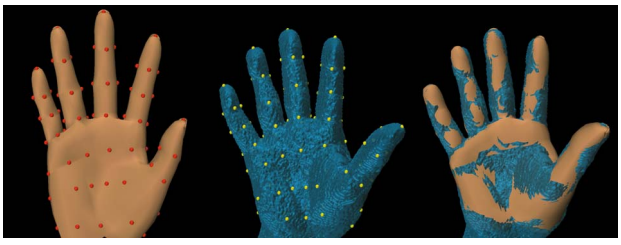
To solve this problem, first a generic polygonal surface model for the subject (e.g. the human hand) is created. A sufficient number of poses of the model are generated by general computer graphics software and conventional surface joint weights are calculated [section 5.1]. The surface joint weights determine the properties of surface deformation for the subject.

Then, the vertices of a generic polygon hand model are aligned to the volume hand iso-surface by scattered data interpolation using manually created sparse corresponding features. With the set of feature locations x_i and their values f_i we can find a function $R(x)$ which gives smooth interpolation of these features using radial basis functions (RBFs) [2]. Our RBFs are defined by manually created sparse corresponding features, and they interpolate the offset between the vertices of the surface model and corresponding voxels in the iso-surface of the volume. The thin plate spline (TPS) is used as the basis function for smooth deformation of the polygon mesh [1]. Figure 3 shows the polygon model in the base pose, the corresponding features, and the deformed polygon model fit to the iso-surface of the volume.

The volume data of the skeletal object can be divided into rigid-body (bone) and non-rigid (flesh) parts, and the bone volume should not be deformed by movement. From this observation, we can assign the joint weights of the voxels in the bone: For bone voxels, $w_j(v) = 1 \quad \text{iff} \quad v \in \text{bone}\, j$, otherwise 0. Then, the weights of the soft tissue volume can be interpolated by RBFs. The previously aligned mesh vertices as well as the voxels of the bone volume now become the feature locations for RBFs interpolation, and their joint weights are feature values to be interpolated. The number of features is $n_p + n_b$, where n_p is the number of the vertices in the surface polygon model and n_b is the number of voxels in the iso-surface of the bone volume.

6 Results

The robustness of methods is tested by an in vivo human hand volume. It produces a wide range of motion involving many joints with overlapping influence. In one human volunteer, MRI volumes from seven different poses were captured using a GE Signa Excite 3.0 T scanner with fast



(a) (b) (c)

Figure 3. Aligning a generic polygon mesh to the volume: (a) Source polygon mesh and corresponding features, (b) Target volume and corresponding features, (c) Polygon mesh warped to align to the volume surface.

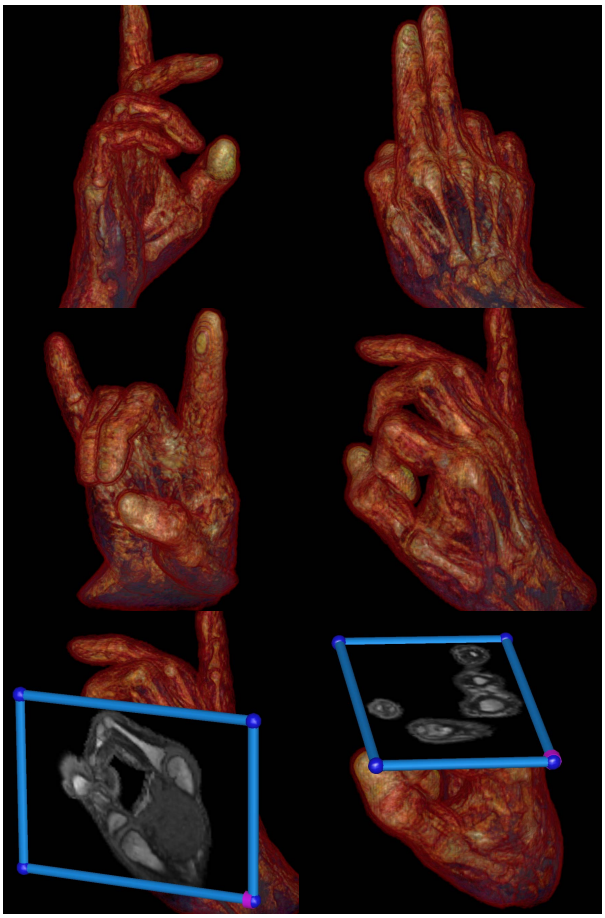


Figure 4. Animated MRI volumes created by volume SSD: the first and second row show various arbitrary postures constructed by volume SSD. The clipping planes in the third row show the interior of the deformed volume.

gradients, fast receiver, and a stock 3D fast gradient echo pulse sequence. The spatial resolution was $0.78 \times 0.78 \times 1.0$ mm^3 and scan-time was approximately 2.0 min per volume.

A generic polygonal surface model of a deformable human hand (16 Joints with 23 DOFs) is created by Poser. Sufficient independent pose samples (25 samples) are exported and joint weights for the human hand volume are computed [section 5]. The joint structure estimated by section 4.3 supports intuitive joint control and arbitrary poses of the volume hand are created by volume SSD. A real-time volume rendering program developed by Kniss et al. [6] is used for display. A clipping plane shows the interior of the MRI deformed to an arbitrary pose. The continuity and smoothness as well as anatomical adequacy is visually validated. The results are shown in figure 4.

Although the methods require complex preprocessing steps to handle in vivo volumes, the volume SSD time is just around 2.5 seconds to deform a large volume containing 623,415 voxels on a 3.4 Ghz Pentium 4 CPU with 2 Gbyte memory. The enclosing volume size is $255 \times 255 \times 100$.

7 Conclusion

In any arbitrary pose, our volume SSD algorithm can generate smooth deformation of the soft-tissue volume while preserving the hard tissue volume. Our pipeline successfully produces the volumetric counterpart of the previous skinning animation and contributes to related areas in both computer animation and volume graphics. Example based skinning methods can be considered as a future improvement to produce specific deformations of each layer.

References

[1] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE. Trans. Pattern Analysis Machine Intelligence*, 11:567–585, 1989.

[2] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In *SIGGRAPH 2001, Computer Graphics Proceedings*, pages 67–76, 2001.

[3] Y. Chen, Q. Zhu, A. Kaufman, and S. Muraki. Physically-based animation of volumetric objects. In *CA '98: Proceedings of the Computer Animation*, page 154, 1998.

[4] N. Gagvani, D. Kenchammana-Hosekote, and D. Silver. Volume animation using the skeleton tree. In *VVS '98: Proceedings of the 1998 IEEE symposium on Volume visualization*, pages 47–53. ACM Press, 1998.

[5] D. L. James and C. D. Twigg. Skinning mesh animations. *ACM Transactions on Graphics (SIGGRAPH 2005)*, 24(3), Aug. 2005.

[6] J. Kniss, G. Kindlmann, and C. Hansen. Simian. http://www.cs.utah.edu/ jmk/simian/.

[7] S. Komojima, N. Miyata, and J. OTA. Identification of position and orientation of hand bones from MR Images by bone model registration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004.

[8] T. Kurihara and N. Miyata. Modeling deformable human hands from medical images. In *Proceedings of the 2004 ACM SIGGRAPH Symposium on Computer Animation (SCA-04)*, pages 357–366, 2004.

[9] N. Magnenat-Thalmann, R. Laperrière, and D. Thalmann. Joint–dependent local deformations for hand animation and object grasping. In *Graphics Interface '88*, pages 26–33, June 1988.

[10] N. Miyata, M. Kouchi, M. Mouchimaru, and T. Kurihara. Finger joint kinematics from MR Images. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005.

[11] T. Rhee, J. Lewis, and U. Neumann. Real-time weighted pose-space deformation on the GPUs. *Computer Graphics Forum, Proc. of Eurographics 2006*, 25(3):439–448, 2006.

[12] C. Rother, V. Kolmogorov, and A. Blake. GrabCut: interactive foreground extraction using iterated graph cuts. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 309–314. ACM Press, 2004.

[13] R. Westermann and C. Rezk-Salama. Real-time volume deformations. *Comput. Graph. Forum*, 20(3), 2001.