Real-time Volume Shading for Deformable Model

Satoshi Yano † ∗ Megumi Nakao † Kotaro Minato †

Fig. 1: Lung volume deformation with GPU-based shading

This presentation proposes real-time volume shading methods for deformable modeling. Although several volume manipulation techniques have been developed, volume shading on time-varying, volumetrically rendered images was still difficult problem. In order to perform volume deformation, we apply FEM-based modeling of tetrahedral grids and visualize deformation results by using texture-based volume rendering. The proposed approach does not update 3D texture or voxels, and achieves fast volume shading by per-grid gradient estimation and interpolation on GPUs.

A normal vector is needed for the shading. Generally, a gray-level gradient method defines gradient voxels from 3D texture, which are used for normal vectors. However, since reconstructing voxels takes high computational costs, the method is not available for interactive volume deformation. In our framework, two data structures: initial volume data and its tetrahedral grid, are prepared. Next, gradient volume texture is created. This process is pre-computationally done. In real-time simulation stage, interactive volume deformation is performed through the following algorithms.

1. Grid deformation and update vertex position.
2. Define the difference of gradient at each vertex of the grid.
3. Create proxy geometry with interpolated gradient difference from the grid.
4. Render shaded volume deformation results by gradient interpolation and shading on GPUs.

Without reconstructing 3D texture, the gradient of each vertex of the grid is calculated and interpolated between the vertices fast on GPUs. We approximate a virtual plane at each vertex and estimate per-vertex gradient using the Least-Square method (See a Fig.2). Before and after the deformation, the per-vertex gradients are calculated. The gradient difference \( \Delta \mathbf{N} \) is utilized as change of normal vector and interpolate them between vertices on GPUs.

We confirmed volume visualization results of time-varying gradient difference via volume deformation. (see Fig. 3)

\[ \Delta \mathbf{N} = (N'x - Nx, N'y - Ny, N'z - Nz) \]

Fig. 2: Virtual plane definition per vertex using Least-squares method. The gradient difference \( \Delta \mathbf{N} \) is utilized for transformation of normal vector and interpolated on GPU.

Fig. 3: Volume visualization of gradient difference. The vector is represented using RGB color value and overlaid on the volumetrically rendered image. (a) Volume deformation of left-side lung and (b) volumetrically rendered gradient difference, (c) another deformation and (d) its rendered gradient.

References

\(^{†}\)Graduate School of Information Science, Nara Institute of Science and Technology. Nara, Japan  
\(^{*}\)satoshi-y@is.naist.jp