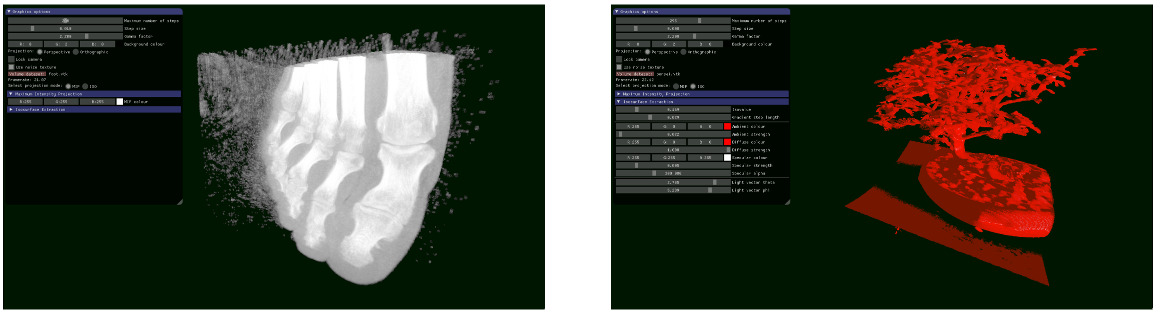
Volume rendering using GPU accelerated Raycasting

Magnus Ulimoen
Uppsala University



(a) Maximum Intensity Projection

(b) Isosurface extraction

Figure 1: Examples of volume rendering using two modes of the raycasting algorithm

1 Introduction

Volume rendering needs different approaches for rendering compared to meshes of polygons. A projection from the three dimensional data onto the screen is needed, taking into account data that obscurs itself.

In this work, raytracing is used with two different mapping modes. These rays are shot from the viewer through the volume, shading using either maximum intensity projection or based on an isovalue. The data visualised is from copmuted tomography.

2 Approach

The implementation follows the tecniques used in [1] implemented in OpenGL. To determine the ray starting and ending position, a cube is rendered to a texture, with RGB-channels set to the position in camera-space. This is done for both front- and backface. A quad is then drawn, a fragment shader samples the screen-space and iterates through the ray between the frontface and backface.

A simple way to visualise the data is extracting the largest value along the ray. And example of this mode can be found in Fig. 1a.

Another approach to visualise the data is thresholding the data and only render volumes with a value higher than the treshold. This forms a solid surface, with a normal that can be determined from the data[1]. Using finite differences the normal is determined, and the surface is shaded using Blinn-Phong shading. The result is included in Fig. 1b.

In order to allow render inside the objects, the volume is split into multiple smaller cubes that together forms the unit cube as inspired by [2].

References

[1] J. Kruger and R. Westermann, "Acceleration techniques for gpu-based volume rendering", in *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, IEEE Computer Society, 2003, p. 38.

[2] H. Scharsach, M. Hadwiger, A. Neubauer, S. Wolfsberger and K. Bühler, "Perspective isosurface and direct volume rendering for virtual endoscopy applications.", in *EuroVis*, vol. 6, 2006, pp. 315–322.

[1] Gradient of isosurface is proportional to the normal