Ray Casting for Collision Detection in Haptic Rendering of Volume Data

Roman Vlasov[*] Karl-Ingo Friese[†] Franz-Erich Wolter[‡]
Leibniz Universität Hannover, Germany

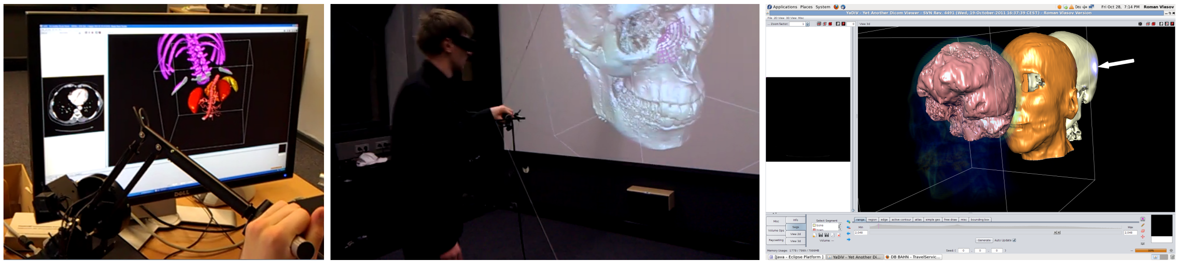

Figure 1: *Phantom (left) and INCA 6D (middle) haptic devices used for the haptic rendering. The test data sets are shown.*

1 Introduction

A haptic exploration adds an additional dimension to working with 3D data: a sense of touch (figure 1). This is especially useful in areas such as medical training and pre-surgical planning, entertainment, CAD and others. Each haptic rendering frame consists of three stages: collision detection, collision response and force-feedback generation. In order to feel the 3D data smoothly, an update rate of at least 1 kHz is required [Brooks Jr. et al. 1990]. Unaddressed practical problems for almost all haptic rendering methods are that no guarantees for collision detection could be given and/or that a special topological structure of objects is required. Here we present an approach which does not have these drawbacks. Furthermore our algorithm has nearly constant time complexity independent of data resolution and does not require any additional precomputed structures. We focus on volumetric voxel data, since that is the direct input from the scanning devices. Other data types could be transformed to this one, if necessary.

2 Our Approach

The key technique of our haptic rendering pipeline is the ray casting applied on volumetric voxel data. It has its roots in computer graphics (see e.g. in [Engel et al. 2004]). The idea of the original method is to numerically evaluate the volume rendering integral along a cast ray in a straightforward manner. But in our case, for a point in the virtual world following the position of the manipulator, we perform the ray casting from its last position to the current one. In more detail, we are going along the ray with 1-voxel step. If there is an obstacle on the way then *true* and a collision information are returned. *False* is returned otherwise. See the figure 2 for details. We use 1-voxel step, because a minimum possible thickness of an object is also one voxel. To further speed up the computations, we firstly create a list of objects that are determined as *collision candidates*. For that, we check if the ray from the last position to the current one collides with the axis-aligned bounding box of each object. If so, then the object is a candidate. The detailed collision detection is performed for these candidates only. Additionally, we impose a reasonable upper limit on the maximal movement of the user-controlled virtual point between two haptic frames.

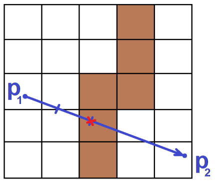

Figure 2: *The ray from the previous position* p_1 *to the current one* p_2 *is cast with 1-voxel step until an obstacle is found or* p_2 *is reached*

The approach described above is used in our *slide over a surface* joint collision detection and response stage of the haptic rendering pipeline, which was incorporated into the YaDiV virtual reality system [Friese et al. 2011].

For tests, real medical tomography data sets of up to 520x512x512 voxels were used (figure 1). For just the point-object collision detection the haptic update rate during the *peak load* is about 750 kHz on our moderate high-end user PC. For the joint collision detection and response approach the value is about 160-170 kHz. Those results are very good because we achieve update rates much higher than the required minimum of 1 kHz. This makes it possible to add support for object-object collisions and a more sophisticated collision response, such as FEM-based elastic deformations. Additionally, we are looking into the possibility to speed up the calculations drastically (up to 10-100 times) by employing GPUs.

This research is sponsored by a grant provided by Siemens/DAAD Postgraduate Programme (DAAD - German Academic Exchange Service).

References

BROOKS JR., F. P., OUH-YOUNG, M., BATTER, J. J., AND KILPATRICK, P. J. 1990. Project grope - haptic displays for scientific visualization. *ACM SIGGRAPH Computer Graphics 24(4)* (August), 177–185.

ENGEL, K., HADWIGER, M., KNISS, J. M., LEFOHN, A. E., SALAMA, C. R., AND WEISKOPF, D. 2004. Real-time volume graphics. *ACM SIGGRAPH 2004 Course Notes*.

FRIESE, K.-I., BLANKE, P., AND WOLTER, F.-E. 2011. Yadiv – an open platform for 3d visualization and 3d segmentation of medical data. *The Visual Computer 27*, 129–139.

---

[*] e-mail:rvlasov@gdv.uni-hannover.de
[†] e-mail:kif@gdv.uni-hannover.de
[‡] e-mail:few@gdv.uni-hannover.de

I3D 2012, Costa Mesa, CA, March 9 – 11, 2012.
ISBN 978-1-4503-1194-6/12/0003 $10.00