Remote Interactive Direct Volume Rendering for Intra-operative Application

P. Hastreiter[1], K. Engel[2], B. Tomandl[3], C. Nimsky[4], R. Fahlbusch[4], T. Ertl[2]

[1]Neurocenter, University of Erlangen-Nuremberg
Department of Neurosurgery, Schwabachanlage 6, 91054 Erlangen
Email: hastreiter@nch.imed.uni-erlangen.de
[2] Visualization and Interactive Systems Group, University of Stuttgart
[3] Division of Neuroradiology, University of Erlangen-Nuremberg
[4] Department of Neurosurgery, University of Erlangen-Nuremberg

Abstract. Interactive and high–quality direct volume rendering provides a comprehensive spatial understanding of complex structures within volume data. This is achieved with 3D texture mapping which is still limited to high-end graphics hardware. In order to access this type of visualization intraoperatively, the system presented in [1] was applied within the operating room. In this context, navigation within CTA data was evaluated in case of intracranial aneurysms. The suggested framework efficiently combines local desktop computers and remote high-end graphics hardware. Thereby, standard visualization capabilities which are available at the clinic are exploited to a maximum. In addition, remote specialized hardware is accessed providing 3D representations of high resolution and quality. Based on different 2D and 3D functionality the integration of the suggested remote visualization strategy is efficiently assisted. Different examples of intraoperative application demonstrate the value of this approach.

Keywords: Volume Rendering, Remote Steering, Intraoperative

1 Introduction

A fast and comprehensive visualization is fundamental for the analysis of tomographic image data. This requires a 2D approach in order to inspect local information on consecutive slice images. In addition to that supplementary assistance is obtained with 3D approaches which easily allow to convey the spatial relation of structures. Compared to indirect strategies which use polygonal models [2], methods of direct volume rendering proved to be superior since the resulting semi-transparent representations better convey the available information of the data [3]. Furthermore, interactive adjustment of transfer functions for color and opacity values circumvents time-consuming pre-processing in many cases of practical importance [4].

The application of 3D visualization within a clinical environment requires interactivity and high rendering quality. This is achieved with efficient graphics hardware. However, a major drawback is the general availability of such systems.

Even if the graphics capacity of low-end PCs increases considerably there is a need of more specialized 3D graphics hardware which provides high performance IO bandwidth and above all high image quality. In order to access this graphics capacity inexpensive network computers are used which gain increased attention.

Due to the restricted availability of powerful graphics hardware direct volume rendering is currently restricted to pre-operative planning independent of the applied rendering strategy. In order to provide a comprehensive visualization for intraoperative application a system was applied within the operating room which combines low-end desktop computers with remotely located high-end graphics hardware and which considers interactive steering of the rendering process [1].

2 Methods

The applied framework as shown in Figure 1 consists of a high-end visualization server which is used for direct volume rendering. The obtained images are directly transferred to one or more clients providing a user interface in order to display the rendered images and to control the visualization process. As an advantage, the client needs only low-end equipment without any special requirements.

On the client side the system was developed with Java in order to support arbitrary platforms. Overall, the 2D application is built on Java2D providing efficient classes for manipulation. As a result, the data can be inspected with a slicing tool which allows for the interactive inspection of the data in axial, coronal and sagittal directions. The supported features are slicing and zooming based on synchronized panels in the main directions. Selecting a volume position on one slice panel will show the corresponding orthogonal slices in the other two panels. While zooming into the volume data nearest-neighbor or bilinear filtering can be applied. Furthermore, the mapping of volume data to RGB values can be interactively manipulated using the transfer function control panel. A rectangular subregion of the volume data for 3D inspection is selectable.

Optionally, a Java3D viewer on the client side allows to visualize a specified sub-volume with limited rendering quality having selected an interesting subregion of the volume data using the slicing tool. Java3D supplies support for 2D and 3D texture mapping. Since low-cost graphics adapters currently do not supply hardware support for 3D texture mapping, object-aligned slices and 2D texture mapping are used for rendering. Various tools support this process providing a fast overview of the data. This includes 3D transformations and control of transfer functions which allow to change the mapping of volume data to RGBA values. Additionally, orthogonal clip planes and isosurface reconstruction are provided as presented in [5]. Selecting a threshold the isosurface is integrated into the texture-based volume visualization. In order to ensure intuitive investigation of the data the 3D visualization and the slicing tool are synchronized in both directions. In consequence, when slicing through the volume data the correspondent position of the slice is shown in the 3D view. On the other hand a slice probe can be moved in the 3D view and the correspondent 2D slice is displayed in the slicing tool. Because of the limited amount of texture memory on low-end graphics adapters down sampling of the volume data can be applied.

On the server side the system is based on OpenInventor with a render engine using 3D texture mapping hardware [6]. It was developed out of a stand-alone application [7] which was adapted according to a previously suggested framework [8]. After rendering the images to non-visible frame buffer (PBuffer), they are compressed and transferred to all connected Java clients. Using a Java user interface the client offers the same look and feel as the server application based on OpenInventor. In this way there is no difference with respect to the interface whether local or remote 3D graphics hardware is used. Besides, all features of the server application are accessible. All events sent from the client application in order to manipulate the visualization are directly managed by OpenInventor. Thereby, the remote control process allows to manipulate the visualization result interactively.

3 Results

The suggested framework of interactive remote direct volume rendering was investigated within the operating room in order to evaluate the value of intraoperative navigation within CTA data sets (compare Figure 2 and Figure 3). In this context, the system was applied in 5 cases of intracranial aneurysms.

The applied CTA data sets had an image matrix of 512x512 pixels with 100-200 slices. All remote visualization was performed at the Visualization and Interactive Systems Group in Stuttgart using a SGI Octane2 (R12000, 400MHz) with MXE and VPro graphics hardware and 128 MB graphics memory. The client to this visualization server was located at the operating room of the Department of Neurosurgery. The connection was established via Internet with an average communication rate of 10 Mbits/sec. For a comparison, the renderings were also performed at the Neurocenter which is part of the Department of Neurosurgery in Erlangen. In this case the same type of visualization server and a connection of a local area 100 MBit TCP/IP network was applied.

In the beginning of the data exploration a pseudo-3D navigation on a slice by slice basis is performed using slices with orthogonal orientation based on the low-end graphics capabilities of the client. Since the Java2D implementation is specialized for 2D imaging it provides interactive manipulation comparable to a native C++ implementation.

After transferring the volume data to the remote visualization server, volume rendering is performed and the resulting images are immediately sent back to the client. In order to accelerate this process every image is compressed using run length encoding. Applying a view-port of 600x600 pixels a frame rate of 3 fps is achieved in case of the Internet connection. Reducing the size of the transferred images by a factor of 4 during manipulation leads to an acceleration by a factor of 5. Alternatively, delay times due to data encoding and transmission over the Internet connection are overcome by rendering the data with the local rendering approach based on Java3D. Thereby, a reduced image quality is obtained. However, the performance increases to 5-10 fps. In comparison to this the local area network provides a rate of 5 fps which is increased by a factor of 3 if the size of the images is reduced.

4 Discussion and Conclusion

Since the applied visualization strategy determines the impact of an envisaged 3D analysis, the tomographic data was formally transferred to the computer science department in order to produce high-quality renderings using 3D texture mapping. Although the result images were of high quality their static nature was a major limitation. Therefore, the suggested approach provides a convenient way to introduce direct volume rendering to intra-operative application allowing for interactive manipulation. As a consequence the spatial orientation is considerably improved compared to static representations. Aligning the orientation of the projection according to the surgeons direction of view leads to even better navigation. For the future further developments are envisaged to fully integrate the visualization in the operating process. Above all, integrating remotely located high-end graphics capabilities allows to share expensive hardware components more efficiently.

References

1. K. Engel, P. Hastreiter, B. Tomandl, K. Eberhardt, and T. Ertl. Combining Local and Remote Visualization Techniques for Interactive Volume Rendering in Medical Applications. In *Proc. Visualization*, pages 449–452. IEEE Comp. Soc. Press, 2000.
2. S. Nakajima and H. Atsumi and A. Bhalerao and F. Jolesz and R. Kikinis and T. Yoshimine and T. Moriarty and P. Stieg. Computer-assisted Surgical Planning for Cerebrovascular Neurosurgery. *Neurosurgery*, 41:403–409, 1997.
3. B. Kuszyk and D. Heath and D. Ney and D. Bluemke and B. Urban and T. Chambers and E. Fishman. CT Angiography with Volume Rendering : Imaging Findings. *American Jour. of Radiol. (AJR)*, pages 445–448, 1995.
4. P. Hastreiter, C. Rezk-Salama, K. Eberhardt, B. Tomandl, and T. Ertl. Functional Analysis of the Vertebral Column based on MR and Direct Volume Rendering. In *Proc. MICCAI*, Lect. Notes in Comp. Sc., pages 412–421. Springer, 2000.
5. K. Engel and T. Ertl. Texture-based Volume Visualization for Multiple Users on the World Wide Web. In *5th Eurographics Workshop on Virtual Environments*, 1999.
6. B. Cabral, N. Cam, and J. Foran. Accelerated Volume Rendering and Tomographic Reconstruction Using Texture Mapping Hardware. *ACM Symp. on Vol. Vis.*, pages 91–98, 1994.
7. O. Sommer, A. Dietz, R. Westermann, and T. Ertl. An Interactive Visualization and Navigation Tool for Medical Volume Data. *Computers & Graphics*, 2:233–244, 1999.
8. K. Engel, O. Sommer, and T. Ertl. A Framework for Interactive Hardware Accelerated Remote 3D-Visualization. In *Proc. VisSym*, pages 167–177. Joint Eurographics and IEEE TCVG Symp. on Vis., 2000.

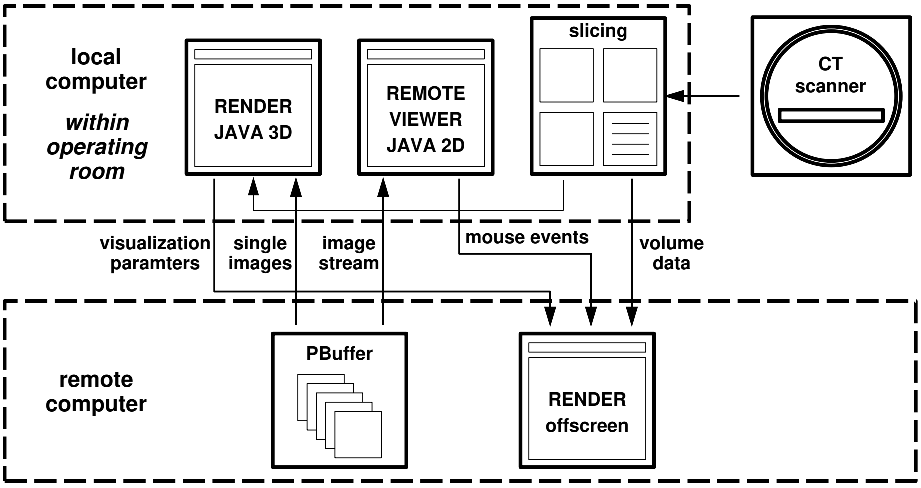

Fig. 1. General architecture of the visualization system.

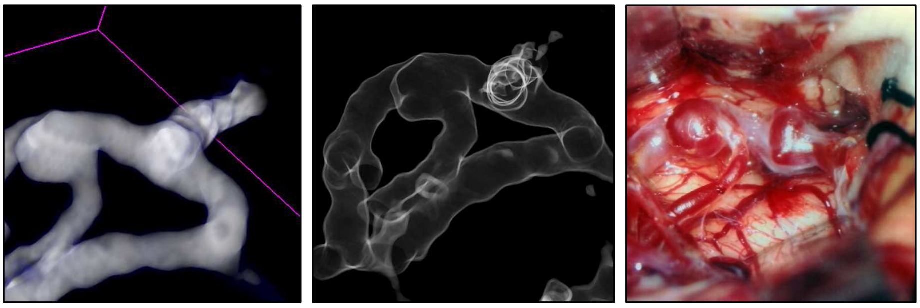

Fig. 2. Patient with 2 aneurysms of the medial cerebral artery (MCA). The visualization shows good correlation with the intraoperative view.

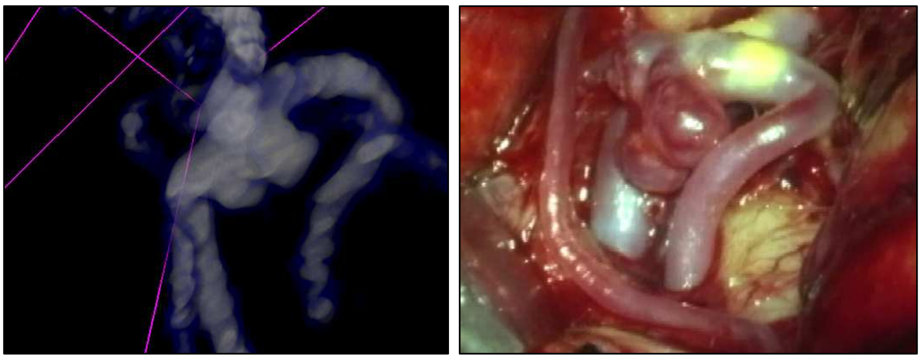

Fig. 3. Aneurysm of the left medial cerebral artery (MCA): the intraoperative view and the 3D visualization show the same viewing direction.