Investigating 3D Object Movies for Web-Based Medical Visualization

F. Rößler[1], M. Nenov[1], S. Iserhardt-Bauer[1,2],
P. Hastreiter[2], T. Ertl[1]

[1] Institute for Visualization and Interactive Systems (VIS), Universität Stuttgart, Stuttgart, Germany
[2] Neurocenter, Dept. of Neurosurgery, University of Erlangen-Nuremberg, Erlangen

Ansprechpartner: friedemann.roessler@vis.uni-stuttgart.de

Abstract:

Standardized medical visualization is an important instrument for reproducible and comparable analysis of medical volume datasets. Therefore, different views of a dataset are pre-computed and e.g. provided by still images or videos. While videos contain fixed camera animations, the alternative 3D object movies permit free and interactive navigation along predefined camera positions. In this work we present a new object movie format and a corresponding java-based viewer, which we developed with respect to the needs of medical visualization. It allows for example the combination of several sub-movies to emphasize different regions or different aspects of a dataset. We have integrated the generation of object movies to an existing web-service for standardized medical visualization, where volume datasets can be uploaded and then videos and object movies are rendered in parallel.

Keywords: Medical Visualization, Object Movies, Web-Based Technologies

1. Purpose

Modern medical 3D visualization environments provide high interactivity and flexibility, but the definition of the visualization parameters is a manual and time-consuming task and the results do highly depend on the expert knowledge and experience of the user. Thus, it is extremely difficult to achieve reproducible visualizations for inter-patient and inter-study comparison. To overcome these drawbacks we have developed a web-service [1] for the standardized visual analysis of pathologies, which is currently applied to the diagnosis of intra-cranial aneurysms. The standardized process performs all tasks of segmentation, visualization and documentation of the data. The documentation is obtained by automatically generated video sequences of the volume rendered medical data set. Since the video generation is an extensive process and originally took about half an hour, we have ported the service to a GPU-Cluster, in order to perform the rendering and video encoding in parallel [2]. By this, the overall processing time was reduced to a couple of minutes.

While a clinical study [3] has shown the applicability of the web-service in clinical routine, the video-based 3D visualization has still some limitations. A major drawback is the fixed camera path and the restriction to linear navigation along the video's time line. Thus, it is difficult to get an impression of the spatial relationship between interesting details of the data set. Further on, there exist no direct links between the different video sequences. An alternative for offline medical 3D visualization are the so-called object movies. Here, images are taken at fixed camera positions on a spherical hull around the visualized object (see Figure 1 a) and specialized viewers allow the free navigation between these camera positions.

There exist several different formats and viewers for 3D object movies. The most common is QuickTime VR [4] in combination with the proprietary QuickTime Viewer. Alternatively, java based solutions like the javaC3D Object Movie Toolkit [5] are available. Object movies have already been applied to medical data for educational purposes. For instance "Bones of the Skull" [7] of the Hardin Library of the Health Sciences is an interactive learning tool for the anatomy of the human skull based on QuickTime VR movies. Mélin-Aldana and Scirotino [8] used QuickTime VR for an interactive atlas of pediatric liver pathologies. Thereby, both examples are based on photographies of real world objects.

We found that the existing formats and viewers lack of flexibility and do not fit to our demands on the applicability for the integration to our web service. Furthermore, they are difficult or impossible to extend. Therefore, we developed a new format and an accompanying java-based viewer, which are especially adapted to the requirements of standardized medical visualization.

2. Methods

For the development of the new *medical object movie* format and viewer we pursued two objectives: On the one hand all visualizations of an analyzed dataset should be integrated and inter-linked in a single movie and on the other hand the access and interaction via the web should be fast and intuitive. In the following two sections the basic ideas and concepts of the design and implementation are detailed with respect to these two aspects.

2.1. The Object Movie Format

The design of the *medical object movie* format was driven by the same ideas, that were already taken into account for the definition of the video sequences. There, one overview video of the whole dataset and several videos of dedicated sub-volumes in areas with a high risk for aneurysms are generated. The camera paths have been chosen with respect to the way clinicians usually examine individual patient data. First, the camera follows several predefined circular paths around the complete volume, for the purpose of orientation and in order to search for pathological regions. Then, the camera zooms in to take a detailed view from a closer distance. Occluding structures may be cut away by a clip plane.

To provide similar visual information as in the videos, a medical object movie is build of several parts (*sub-movies*), where a single sub-movie either shows the whole dataset or a sub-volume. For each sub-movie several modes can be applied, e.g. showing the dataset with and without a clip plane. To get the relation between the overview and the sub-volumes the movie parts are linked together and the positions of the sub-volumes are indicated in the overview by their surrounding bounding boxes.

Normally, an object movie contains views for predefined intervals of rotation (pan) and tilt angles and and stores them in a rectangular array (see Figure 1 b). This fact leads to a large number of images and thereby large consumption of disk space, especially if the investigated object should be visualized from a wide range of different angles. Since the size and generation time of the object movies should be limited, our new medical object movie format allows the combination of movies from arbitrary view areas around the object (see Figure 1 c). Further on, this provides the possibility to restrict the movies to interesting views of the data. Additionally, the format permits the attachment of zoomed views at any camera position to visualize details.

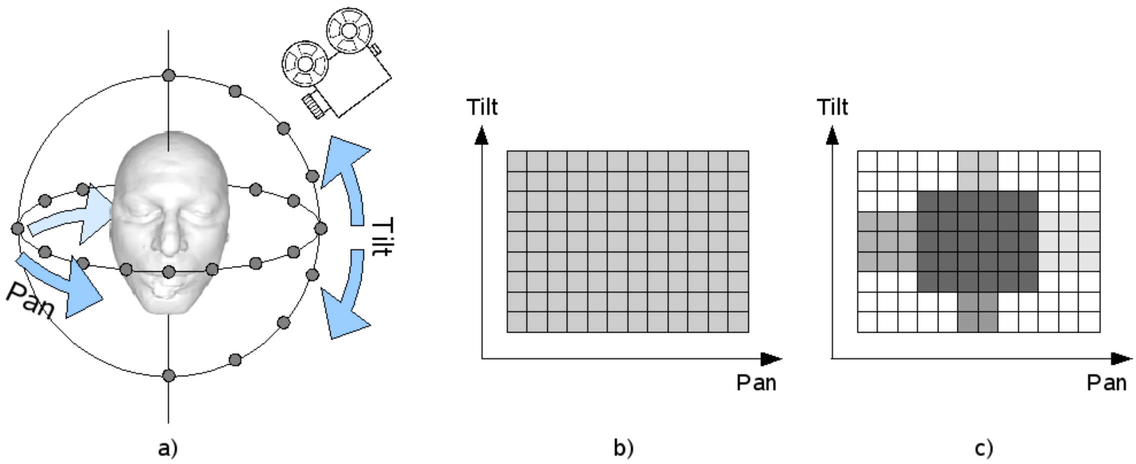


Fig. 1: Generation and storing of an object movie: (a) The acquisition of the views on a spherical hull. (b) Rectangular array for picture storage in a traditional object movie. (c) Combination of different view areas (different grey values) in the new medical object movie format.

A medical object movie consists of two files: a meta file in XML format describing the structure of the movie, and a file containing the images in an arbitrary compressed or uncompressed format. In the meta file the sub-movies are declared independently and can be inter-connected (e.g. overview and detail) by links. For each defined view a byte position is given, which addresses the associated image in the image file.

2.2. The Java Viewer

For the investigation of a medical object movie we implemented a java-based viewer which can either be run as stand-alone application or as an applet integrated to a website. The object movies can be loaded from any web location defined by an URL or from local disc. First, the meta file is read, the movie structure is analyzed and the viewer's GUI is accordingly configured. Then the image download for the actual shown sub-movie is started and the images are stored in a client-side cache. The interactive exploration of the movie can be started immediately. If a requested view is not actually stored in the cache, this view and the views in the surrounding area are downloaded privileged.

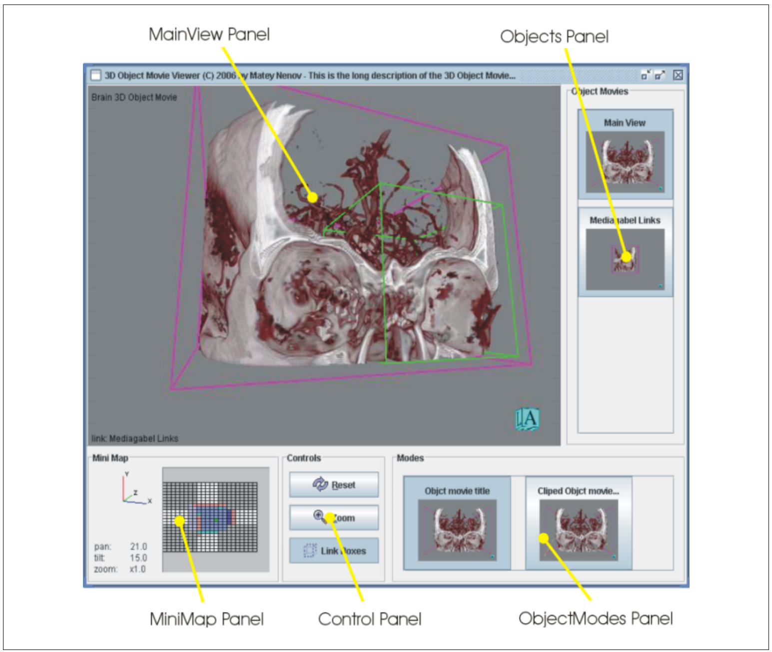


Fig. 2: Screenshot of the java viewer with labeled panels.

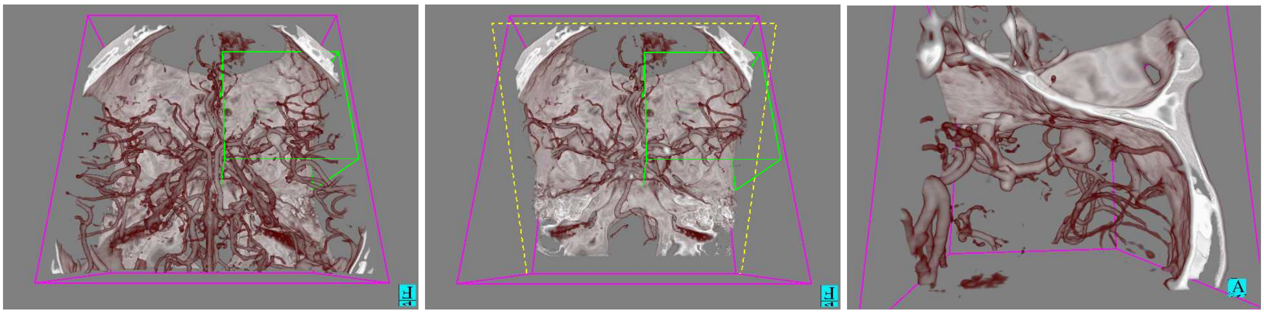
Figure 2 shows a screen shot of the java viewer with its different controls and panels. In the *MainView Panel* the actual chosen view is displayed. The camera can be moved by dragging the mouse, but only in the range of precomputed views. The actual camera position is shown in the *MiniMap Panel* in the left bottom. Here, the object movie is visualized as a rectangular grid, showing the availability and the caching state of the views by a color scheme. In the *Control Panel* different actions like reseting or switching to a zoomed view can be initiated. The *ObjectModes Panel* allows the switch between different modes of a single sub-movie. Which sub-movies exist, is shown in the right *Objects Panel*. A sub-movie can either be chosen by selecting the symbol in the *Objects Panel* or by clicking to the related area in the *MainView Panel*, which is indicated by a green bounding box.

3. Results

We have integrated the object movie generation into our medical visualization web service for the analysis of intra-cranial aneurysms. Via the dynamic web-interface the user can upload a CT Angiography (CTA) dataset and choose which sub-movies and modes should be combined. Then the movie is generated in parallel on a GPU cluster equipped with eight nodes (see [2] for details). The parallelization of the rendering is intrinsic, since a medical object movie consists of several parts, which can be computed independently. After the generation of an object movie, it can be directly investigated via the viewer applet, which is integrated into the web-interface.

The structure of the generated object movies was designed with respect to the original video sequences (see [1]). They consist of an overview movie with two modes applied. One mode shows the dataset as a whole, for the second mode a clip plane is used to cut away occluding structures. Additionally, four sub-movies are attached, which visualize sub-volumes in regions with a high risk of intra-cranial aneurysms. For an intuitive navigation the overview movie contains links to the four sub-movies. Figure 3 shows three example views, of the overview without an with an applied clip-plane and of a sub-volume containing an aneurysm.

Depending on the chosen sub-movies and the complexity of the dataset an object movie consist of up to 4000 images and has a size of 50 to 100 MB, if JPEG 50 compressed images are used. The maximum generation time is about two minutes. The interactivity of the java viewer is on the hand bounded by the bandwidth of the network connection on the other hand by the achievable frame rate for client-side cached images. This depends on the size and the complexity of the images, since they are cached in their compressed stage and have to be decompressed bevor display. For a modern PC and a image size of 640 x 480 pixels the average frame rate is about 10 frames per second.



a) b) c)

Fig. 3: Three example views of an object movie of a CTA Dataset, which show two overview visualizations without (a) and with a clip plane (b), and a zoomed view of the greensub-volume with an aneurysm in the center.

4. Conclusion

We have presented the usefulness of object movies for the standardized visual medical analysis via the web. A new object movie format was developed, that especially addresses the needs of medical visualization. The computation of the object movies on a GPU-Cluster in parallel provides a fast feedback to the user and the applet viewer allows an online 3D examination of the patient data. First tests have shown the applicability of object movies for web-based medical visualization. They combine the advantages of interactive 3D visualization applications and offline visualization via video sequences. In the future we want to perform a complex user study, to further investigate these facts

Literatur

[1] Iserhardt-Bauer, S., Hastreiter, P., Ertl, T., Eberhardt, K., Tomandl, B.: Case study: medical Web service for the automatic 3D documentation for neuroradiological diagnosis. In Proceedings of IEEE Visualization 2001, 425-428

[2] Rößler, F., Wolff, T., Iserhardt-Bauer, S., Tomandl, B., Hastreiter, P., and Ertl, T.: Distributed video generation on a GPU-cluster for the web-based analysis of medical image data. In Proceedings of SPIE Medical Imaging 2007

[3] Tomandl, B., Hastreiter, P., Iserhardt-Bauer, S., Köstner, N., Schempershofe, M., Huk, W., Ertl, T., Strauss, C., Romstock, J.: Standardized evaluation of CT angiography with remote generation of 3D video sequences for the detection of intracranial aneurysms, Radiographics 23 (2003) e12

[4] Apple Inc., QuickTime VR, http://developer.apple.com/documentation/Quicktime/InsideQT_QTVR/insideqt_qtvr.pdf

[5] Littlebitsmultimedia, javaC3D Object Movie Toolkit V 1.0, http://www.littlebitsmultimedia.com/littlebits/html/objectmovie.htm

[6] Hardin Library for the Health Sciences, Bones of the Skull, http://www.lib.uiowa.edu/commons/skullvr/index.html

[7] Medlin-Aldan, H., Sciortino D.: Virtual reality demonstration of surgical specimens, including links to histologic features, Modern Pathology (2003), 9(16) 958-961