Time-Critical Volume Rendering

Han-Wei Shen and Xinyue Li
Department of Computer and Information Science
The Ohio State University
Columbus, Ohio 43210
E-mail: hwshen@cis.ohio-state.edu and xli@cis.ohio-state.edu

Although the speed of volume rendering has been significantly increased in the past decade due to the advent of efficient volume rendering algorithms and fast 3D texture mapping hardware, the size of an average volumetric data set also continues to grow. The amount of data from a large scale volume can easily overwhelm the underlying computational resources, and thus makes interactive rendering impossible. To overcome the problems, researchers have proposed to use hierarchical rendering algorithms for their effectiveness in trading image quality for speed. While effective, most of the existing hierarchical volume rendering algorithms rely on the user to make a run-time decision, which is often done by trial and error, to select appropriate levels of detail for rendering. However, for real-time applications such as virtual reality or computer assisted medical surgery, using trial and error to tune the performance is impractical as these applications require immediate responses and the performance requirement can change very frequently.

In this technical sketch we present a time-critical volume rendering algorithm with an aim to solve the aforementioned problem. The purpose of our algorithm is to alleviate the user's burden in searching appropriate levels of detail that can meet the performance goal. With our algorithm, the user only needs to specify a desired wall clock time for each frame, and the algorithm will adaptively render data of appropriate resolutions from the volume hierarchy to complete the computation just in time. In general, the run-time performance of hierarchical volume rendering is controlled by an error tolerance, which is used to choose appropriate resolutions of data from the volume hierarchy. A low error tolerance produces a better quality, while a high error tolerance allows a faster rendering speed. To allow the user to control the rendering time in a high level manner, having an automatic error specification algorithm that can guarantee the frame rate becomes crucial.

We tackle the problem of automatic error tolerance specification from a control point of view. The ultimate goal of control is to choose an error tolerance that will result in both a satisfactory rendering speed and a highest possible quality. To achieve this goal, we construct a feedback control system, which is to monitor the volume rendering performance and adjust the error tolerance whenever necessary. The object to be controlled is the volume rendering algorithm. The output of the control unit is the error tolerance. The goal of this feedback control system is to guarantee a desired response of the system , i.e., keeping the the actual and desired rendering time close. We use a divide-and-conquer algorithm to implement the feedback control system. The algorithm starts by subdividing the underlying volume into several blocks, hereafter called subvolumes, and splitting the total time budget into smaller shares. Each subvolume is assigned a share of the time budget. Given the subdivision, we feed each subvolume into our feedback control loop in a front-to-back depth order and perform a hierarchical rendering using the subvolume's corresponding branch in the volume hierarchy. For the first subvolume, as there is no error tolerance given by the user, an initial guess is used for the rendering. At the end of this rendering, the actual rendering time is fed back to the control unit. With the feedback information, the control unit updates the available computation time and revises a new time budget for each of the

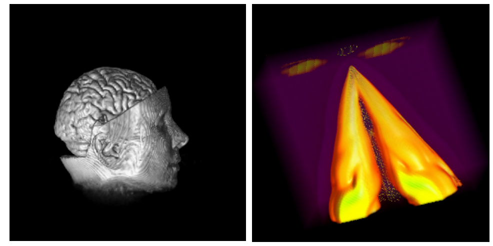

Figure 1: volume rendered images using our time-critical algorithm

remaining subvolumes. In addition, the control unit estimates a rendering time for the next subvolume based on the previous rendering result. After this, the control unit compares the estimated rendering time with the subvolume's allocated time budget and uses the amount of difference to compute a new error tolerance. This error tolerance is then used to render the next subvolume. This feedback control loop continues until all subvolumes are rendered or no more computation time is available.

The key component in our control system is a fuzzy controller that can choose an appropriate error tolerance based on the past experience so that the next subvolume can be rendered at the expected speed. The fuzzy logic controller is used because it is efficient and effective in modeling complex and nonlinear problems that are difficult or impossible to solve analytically. In our case, as the speed of volume rendering is a combination effect of 3D projection, transfer function, and visibility, the relation between the error tolerance and the rendering time is therefore nonlinear. In our feedback control loop, the fuzzy controller adjusts the error tolerance at the end of each feedback iteration based on the rendering result of the previous subvolume. This adjustment is done by first calculating the difference between the estimated rendering time and the budgeted time for the next subvolume, and then determining how much change is necessary. Intuitively, if the difference is high, a larger change to the error tolerance will be needed. Otherwise, the control unit will make a small or no change. We will provide more details about construction of the fuzzy logic controller unit in our presentation.

A unique feature of our time-critical algorithm is that it is possible to take into account the importance of the subvolumes and spend different fractions of the time budget to render different subvolumes. This is done by appropriately defining an importance function for the subvolumes, and use the importance function to distribute the available computation time among the subvolumes. We have designed a flexible time budget allocation algorithm that can take into account multiple factors such as the opacities, data errors, or gaze direction to control the rendering algorithm's interactivity.

We have conducted preliminary experiments for the proposed time-critical volume rendering algorithm and received very promising results. We integrated both the software based ray casting algorithm and 3D texture mapping hardware volume rendering method into our control system. We are able to guarantee the rendering to be completed within a 5% difference from the user specified rendering speed. In addition, we are able to accelerate both software and hardware volume rendering and receive high quality rendering results by allocating more computation time to render important regions. Figure 1 shows two images generated by our time-critical volume rendering algorithm.

Information about the presentation

Sketch Summary: In this technical sketch we present a time-critical algorithm for hierarchical volume rendering. The purpose of our algorithm is to alleviate the user's burden in searching appropriate levels of detail that can meet the performance goal. With our algorithm, the user only needs to specify a desired wall clock time for each frame, and the algorithm will adaptively render data of appropriate resolutions from the volume hierarchy to complete the computation just in time.

Speaker: Dr. Han-Wei Shen

Project Participants: Xinyue Li and Han-Wei Shen
Department of Computer and Information Science
The Ohio State University
Columbus, Ohio 43210
hwshen@cis.ohio-state.edu and xli@cis.ohio-state.edu

Special Requirement for Presentation Equipment: None