Interactive High Fidelity Navigation in Archaeological Models

Luís Paulo Santos (psantos@di.uminho.pt)
Departamento de Informática
Universidade do Minho, Portugal

Abstract

Interactive navigation within virtual archaeological models using global illumination algorithms is a major challenge due to the huge computational requirements associated with these algorithms. We present a framework to achieve this goal, based on image space subsampling, parallel rendering, exploitation of spatiotemporal coherence and progressive refinement. Results, obtained with a prototype and a model of the roman town of Bracara Augusta, suggest that this is a promising approach.

1. Introduction

The ability to navigate through a model of an archaeological site allows researchers to visualize virtual reconstructions of long disappeared sites of interest. This ability becomes even more empowering for the researcher if the navigation is generated with global physically based illumination models and is interactive - in the sense that the user can walk through the model without significantly perceived delays. This paper presents a framework developed to allow such interactive navigation on ancient town models, specifically the roman town of Bracara Augusta.

Global illumination algorithms are too demanding to allow interactive rendering on current PCs. Brute force rendering of each frame is impractical and ingenious techniques are required to allow smooth navigation while presenting tolerable degradation of the image quality. We propose a technique based on parallel processing, asynchronous rendering and exploitation of spatiotemporal coherence, while progressive refinement is used to converge to high quality images whenever there is computing power surplus. The proposed framework is structured as a three-tier architecture: a parallel physically based renderer, a Shading Management Agent (SMA), which caches previously computed shading values and a light-weight visualization client. We present results related to parallel rendering and progressive refinement, using Bracara Augusta model as the case study.

2. System overview

To support interactive frame rates both spatial and temporal coherence is exploited. This approach relies on image space subsampling and object space caching of shading values. The two basic functionalities of a navigation system are decoupled: image visualization and rendering run asynchronously and at their own pace. An additional process is inserted between the renderer and the visualizer: a "Shading Management Agent" (SMA) computes visibility and decides whether shading samples are requested to the renderer or retrieved from a local cache of previously evaluated shading information [1, 2]. Spatial coherence is exploited by requesting shading samples only at visible triangles' vertices and interpolating among samples when displaying the image; interpolation is performed by the visualizer using its graphics hardware. The SMA determines which triangles are visible for a given view point and either requests vertices' shading values from the renderer or retrieves these values from the local cache, if available. By supplying the visualizer with only the visible portion of the geometry, the traffic volume between the SMA and the visualizer is reduced and the workload imposed upon the visualizer hardware is kept to a minimum, allowing this to be a light-weight client, such as a mobile device. Since the visualizer is supplied with the geometry, this is always accurately reproduced, even if the view point changes. Temporal coherence is exploited by caching previously computed shading samples and reusing them whenever the geometry becomes visible again. Caching occurs on object space to avoid reprojection artifacts. The SMA is responsible for maintaining this local cache. When the view point changes smoothly, as in ordinary navigations, the set of visible triangles also changes smoothly from frame to frame. Thus very fast feedback is given to the user by reusing triangles that are already stored on the visualizer or sending triangles from the SMA cache that are already shaded (because they have been used on past frames). Only triangles that have never been shaded must be requested to the renderer. The user can thus navigate on lower quality images, visualizing only the subset of shaded triangles, and converge to higher quality by stopping at a given view point. The second time a portion of the geometry becomes visible feedback will be much faster since all shaded information is stored on the cache.

Image reconstruction from a sparse set of samples is inaccurate where high spatial frequencies occur, such as at shadows boundaries. The geometry is represented on the SMA using a hierarchical subdivision mesh, which allows original triangles to be arbitrarily subdivided. Whenever there is available computing power, the SMA selects some triangles for subdivision and requests shading values from the renderer. These subdivisions and respective shading information are stored on the hierarchical mesh and sent to the visualizer. The image will thus converge to a high quality image, given enough computing time. As a triangle selection criterium for subdivision we use an object space technique that selects those triangles that have the largest luminance differences among their vertices and that project onto a significant number of pixels for the current view point.

The renderer, the SMA and the visualizer can be mapped onto different sets of machines. A point based renderer, such as path tracing or ray tracing, must be used to accept requests arbitrarily distributed across the image plane or the object space - we use a parallel version of Radiance [3]. The SMA is conceived as a "Rendering Service Provider": visualization clients can connect to it from remote locations. The SMA and the rendering processes can run on a cluster of workstations and be shared by multiple clients. The visualization client is a light-weight program with modest requirements. It is conceivable that in the near future it can run on mobile devices.

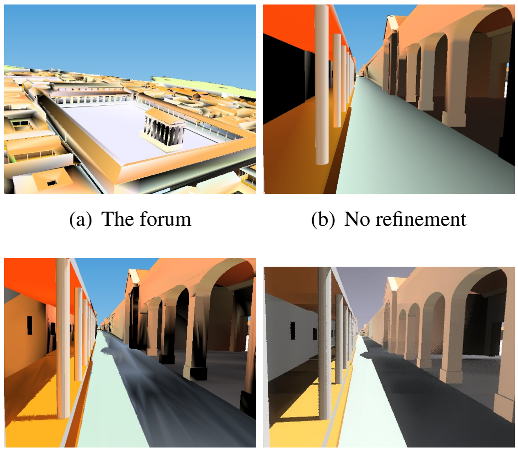
3. Results

The existing prototype allows interactive navigation and performs progressive refinement. Using a dual-Xeon machine for the SMA and 4 AMD Athlon based systems for the parallel rendering processes we obtained the timing shown in table 1 to render the view point shown in figure 1(a). These timings are for the original triangles' vertices, i.e., no refinement. The obtained speedup is far from linear, because the SMA requests shading values in an episodic fashion, rather than keeping the renderer constantly busy. Note that the user can navigate through the scene without waiting for all triangles to be rendered and profiting from cached values.

1 proc.	2 proc.	3 proc.	4 proc.
16806	13732	11327	10578

Table 1. Time (in msec) for 1 to 4 processes.

Figures 1(b) and 1(c) show another view of Bracara Augusta with and without progressive refinement; the reference image obtained with exhaustive ray tracing with Radiance is presented for comparison (figure 1(d)). The refinement algorithm is able to locate all visual discontinuities (shadow boundaries) in 20 seconds (no parallel rendering) and generating only 7701 triangles - the original geometry is shaded and displayed in 3,5 seconds (with 603 visible triangles) while rendering with Radiance takes 49,4 seconds. Exploitation of parallel processing will almost certainly speedup this result.



(a) The forum (b) No refinement (c) Refined image (d) Radiance image

4. Conclusion

A proposal for a framework enabling interactive walkthroughs with high fidelity global illumination on archaeological models has been presented. This approach is based on image space subsampling, asynchronous rendering, parallel processing and spatiotemporal coherence exploitation. The results achieved with the currently available prototype indicate that this a promising solution. Further work is required to improve parallel scalability and speedup, enhance tone mapping, migrate the visualizer to mobile devices, include specular BRDFs and support moving objects. These are our goals for the near future.

References

[1] P. Tole, F. Pellacini, B. Walter, and D. Greenberg. Interactive Global Illumination in Dynamic Scenes. *ACM Transactions on Graphics*, 21(3):537–546, 2002.

[2] B. Walter, G. Drettakis, and D. Greenberg. Enhancing and Optimizing the Render Cache. In *Proceeding of the Eurographics Workshop on Rendering Techniques*, pages 37–42, 2002.

[3] G. Ward. The RADIANCE Lighting Simulation and Rendering System. In *ACM SIGGRAPH'94*, pages 459–472, 1994.