Real-Time Visualizations of Gigapixel Texture Data Sets Using HTML5

Charles-Frederik Hollemeersch, Bart Pieters, Aljosha Demeulemeester, Peter Lambert, and Rik Van de Walle

Multimedia Lab, Ghent University - IBBT,
Gaston Crommenlaan 8, B-9050 Ledeberg-Ghent, Belgium
charlesfrederik.hollemeersch@ugent.be
http://multimedialab.elis.ugent.be/

Abstract. With the recent standardization of WebGL as part of HTML5, new possibilities have arisen for graphically intensive web-based applications. This paper presents our gigapixel texture visualization system which runs entirely within the limitations of a standards-compatible browser. Compared to existing approaches, our system offers high-performance 3D texture visualization and streaming without any dedicated plugins. We show that real-time performance can be achieved (less than 12ms render time per frame) on current-generation desktop hardware for texture data sets of at least 15 gigapixels.

Keywords: Streaming, WebGL, Visualization.

1 Introduction

The introduction of the new HTML5 standard has enabled a new generation of web applications. Web applications are no longer constrained to DOM based user interfaces or proprietary plugins to generate rich visual output. Standardized technologies such as the 2D canvas and 3D WebGL allow graphics operations to be scripted directly from within JavaScript. By generating graphics entirely on the client side, the latency and bandwidth of the web application can also be significantly reduced compared to systems generating graphics in the cloud and streaming the results to the clients as a video stream.

Visualizing high-resolution image and texture data sets is a challenging problem which has many practical uses. For example, GIS, biology, archeology, heritage, and educational applications all have benefited from efficiently aquiring and accessing large image data sets [1]. In this paper we will show how accessing large texture data sets can be made possible within the framework of the HTML5 standard by using WebGL and other HTML5 features.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 621–623, 2012.

Table 1. Per-frame render times of our application in different browsers. Results were measured on a 2.4GHz Intel Core2 Quad CPU and a NVIDIA Geforce GTX 480.

		Google Chrome 12	Mozilla Firefox 5	Opera 11 Preview
Frame (ms)	Rendering	12	2.5	3.1*

* Note, the Opera 11 preview currently generates invalid results.

2 Visualizing Gigapixel Texture Data Sets Using HTML5

There are currently several web-based technologies that allow acessing GigaPixel images over the web[1,2]. However most of these technologies rely on the use of the Adobe Flash plugin to do the most graphics-intensive parts of their visualization work. In addition to this, all these visualizations are limited to 2D images. Furthermore, they rely on analytical approaches to determine the set of required data[2] which do not extend well to 3D visualizations. By adopting concepts and ideas from the high-performance computing world [3] it becomes possible to visualize large texture data sets applied on 3D models and geometry in real time on the latest versions of most major browsers. Our method works by offloading the most computationally expensive operations to the GPU using WebGL and uses algorithmic optimization to accelerate the remaining steps in JavaScript. Currently we have not done any extensive code-level optimizations.

Figure 1 shows a screenshot of our demo application. This application visualizes a large texture data set (122880×122880 pixels, around 75 gigabytes of uncompressed data) in real time in a standard browser without any custom plugins. Table 1 shows the average time it takes to draw the scene in the browser. These times include updating internal cache data-structures and generating the

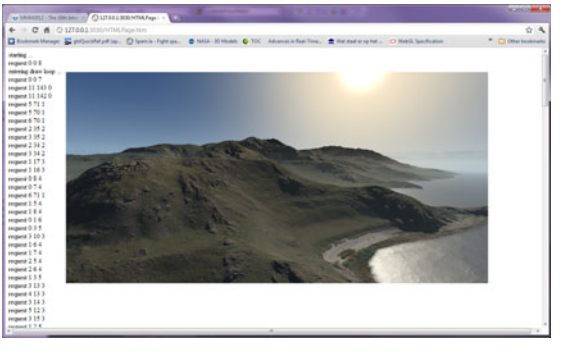

Fig. 1. A screenshot of the demo running in the Google Chrome browser. This is a visualization of orthopotography and height data made available by Utah's State Geographic Information Database (http://gis.utah.gov/).

[1] http://gigapan.org/
[2] http://www.yosemite-17-gigapixels.com/

necessary requests for image data. Note that Opera 11 Preview[3] currently gives invalid output. However on both Chrome and Firefox we can easily achieve real time framerates while still leaving enough CPU time available for other browser tasks. The speed difference between Chrome and Firefox are probably due to the fact that chrome sandboxes WebGL calls and translates all commands to DirectX, while firefox runs WebGL in the same process. However, a more detailed investigation is needed to confirm this.

3 Conclusions

We have shown that it is possible to create computational and data intensive applications using the latest generation of HTML5-based browsers. In particular in our demo we have shown that 3D visualizations using high-resolution textures are no longer limited to native applications requiring high-end computers. In the near future this technology will allow making large data sets available to a larger and less technically skilled public across a wide range of computing devices.

In the future we want to further optimize our system and extend it so it becomes a fully functional frontent for our collaborative editing tool [4]. This will allow a large number of users to not only visualize the data set but also modify and annotate it.

Acknowledgments. The research activities that have been described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWOFlanders), the Belgian Federal Science Policy Office (BFSPO), and the European Union.

References

1. Frenkel, K.A.: Panning for science. Science 330, 748–749 (2010)
2. Kopf, J., Uyttendaele, M., Deussen, O., Cohen, M.F.: Capturing and viewing gigapixel images. ACM Transactions on Graphics 26 (2007)
3. Hollemeersch, C., Pieters, B., Lambert, P., Van de Walle, R.: Accelerating virtual texturing using cuda. In: Engel, W. (ed.) Gpu Pro:Advanced Rendering Techniques, ch. 10.2, pp. 623–641. A K Peters (2010)
4. Hollemeersch, C.-F., Pieters, B., Demeulemeester, A., Cornillie, F., Van Semmertier, B., Mannens, E., Lambert, P., Desmet, P., Van de Walle, R.: Graphics for serious games: Infinitex: An interactive editingsystem for the production of large texture datasets. Comput. Graph. 34, 643–654 (2010)

[3] `http://labs.opera.com/news/2011/02/28/`