HIGHER ORDER SHAPE REPRESENTATIONS FOR INCREASED QUALITY OF EXPERIENCE IN 3D INTERNET

Tor Dokken
SINTEF
P.O. Box 124 Blindern 0314 Oslo, Norway

Michał Koziuk
one2tribe Sp. z o.o.
Reguły, Bodycha 77, 05-816 Michałowice, Poland

Łukasz Kwoska
one2tribe Sp. z o.o.
Reguły, Bodycha 77, 05-816 Michałowice, Poland

Andre Stork
Fraunhofer IGD
Fraunhoferstraße 5, D - 64283 Darmstadt, Germany

Ewald Quak
SINTEF
P.O. Box 124 Blindern 0314 Oslo, Norway

ABSTRACT

The user quality of experience of current 3D Internet technology lags far behind the one of installed 3D applications. We believe that a broad uptake of '3D Media Internet' depends on a significant improvement of the user quality of experience, which should not significantly differ from the quality offered by similar 3D services installed on the user's device. The approach we propose combines compact shape representations with the use of the parallel computational performance of GPUs, multi-core CPUs and many-core CPUs.

KEYWORDS

Geometry, Computer Graphics, Distribution, Rendering.

1. INTRODUCTION

The user quality of experience of current 3D Internet technology lags far behind the one of installed 3D applications. We believe that a broad uptake of '3D Media Internet' depends on a significant improvement of the user quality of experience, which should not significantly differ from the quality offered by similar 3D services installed on the user's device. The approach we propose is motivated by three mega trends:

- Many different shape representations exist, not all well suited for performant 3D Internet applications.
- The variety of end user visual devices is growing, especially mobile devices with Internet access.
- In recent years we have seen a shift from single core to multi/many-core devices (CPUs and GPUs).

The idea is that an infrastructure should be developed consisting of 3D content distribution and adaptation strategies plus parallel processing capabilities to provide a user with the best possible balance of:

- Visual fidelity when presenting large 3D objects,

- Interactivity in 3D Internet applications, and
- Network load.

To achieve this we propose to introduce non-traditional model representations such as higher-order surfaces (e.g., subdivision surfaces) and procedural models to leverage their increased possibilities of adaptation and flexibility. The idea is to separate 3D model representation from graphics processing and thus to open up the possibility to distribute 3D web applications between server and client to best facilitate the use of the actual network, server and client resources. This will allow:

- Realization of an open embodiment of the approach on a wide range of client platforms ranging from handheld devices to high-end gaming PCs,
- Leveraging of different shape representation schemes and their benefits for 3D Internet applications,
- Exploitation of many- and multi-core CPUs and programmable graphics processors (GPUs) for increased performance and visual quality.

The paper will in Section 2 address the state-of-the-art and current trends for relevant hardware and software aspects of mobile and stationary devices, followed by Section 3 addressing browsers and 3D standards and technologies for the Internet. Section 4 addresses higher order shape representations with emphasis on subdivision surfaces and the Generative Modelling Language (GML). Distributing 3D content between client and server, and parallelizing rendering is the topic of Section 5. The paper concludes with Section 6, where we describe how to distribute rendering for the 3D Internet.

The institutions represented by the authors intend to pursue the outlined approach in the coming years, preferably within the scope of the ICT-program of the EU's seventh framework program.

2. HARDWARE AND SOFTWARE MARKET TRENDS FOR 3D DEVICES

2.1 Processor Evolution to Many-core CPUs - GPUs

CPUs. The computational engine of modern computers, the central processing unit (CPU), is becoming highly parallelized. The sustained growth in transistor density (Moore's law) allows chip designers to put more and more processor cores on a single silicon die. Until 2004, we experienced a growth in performance through growth in core clock frequency. Consequently most commodity processors only included one processor core. Recently we have seen only little increase in core clock frequency. To offer increased performance, current symmetric multi-core commodity chips have up to four processor cores on one die.

CPUs for embedded devices. The majority (> 90%) of cell phones are powered by ARM processors. For devices not older than 3 years clock speeds range from 200MHz (Sony Ericsson W300) and 235 MHz (Nokia 6300) in ARM9 up to 528 MHz in HTC Dream's ARM11. This is comparable to stationary Pentium II processors from 10 years back. Memory capabilities are much more diverse. Embedded systems seem also to go multi-core, thus offering significantly increasing processing power for mobile devices.

GPUs. During the last decade Graphics Processing Units (GPUs) have evolved from being fixed functionality processors to become very powerful programmable many-core data stream processors. The performance growth of GPUs is reportedly better than the one for CPUs. Current GPUs offer an even higher number of parallel cores than CPUs do, e.g., 128 compared to 4. Albeit being programmable, certain restrictions have to be met to drive GPUs to their full performance. Typically the parallelization of algorithms needs to take GPU specificities info account.

GPUs for embedded devices. The most popular GPU for cell phones is PowerVR. Another one is GoForce by NVIDIA. A few GPUs for cell phones (such as ARM Mali and Imageon) are available, but they are still not very popular. Recently NVIDIA has adopted a completely new approach to the cell phone market. Instead of providing a GPU, NVIDIA created a new platform – Tegra. It is a SOC (System-On-Chip) or a COC (Computer-On-Chip), which enables other manufacturers to easily enter the mobile market.

The trend seems to be that the CPU evolves into a processing chip, combining both "fat" traditional cores and "thin" or "data parallel" cores. An example of such a CPU is the CELL BE having one "fat" core and eight "thin" cores. Both AMD and INTEL (Larrabee) have announced that they will produce heterogeneous multi-core processors, combining the data stream cores of GPUs and traditional CPU cores.

2.2 Current Trends for Mobile Devices

The market of mobile devices follows the pattern known from PCs – as time moves forward more and more resources become available within a standard-sized mobile device. This includes CPU power, runtime memory, storage memory, GPU capabilities, available network bandwidth, screen resolution, new interfaces, sensors, and so on. The challenge is to fit all the new functionality into a standard-sized device, while maintaining an acceptable battery lifetime.

At the same time mobile devices become ubiquitous due to lower costs, and the percentage of the population using a mobile device constantly increases. The increase in capabilities and popularity leads to the emergence of new types of mobile services and new paradigms, such as cloud computing, pervasive applications, mobile social applications, context aware applications, mobile multiplayer city games. In the same way it will soon be possible to develop 3D based mobile solutions for 'typical' devices.

3. BROWSERS AND 3D INTERNET STANDARDS AND TECHNOLOGIES

3.1 Web Browsers for Desktop and Mobile Devices

Multiple desktop web browser implementations exist. However, just a few are widely used. According to NetApplications (NetApplications, 2009), the leading browsers in April 2009 were: Internet Explorer (66%), Mozilla Firefox (22%), Safari (8%), Google Chrome (<2%), Netscape (<1%) and Opera (<1%). Some of these major players on the desktop web browser market also actively pursue the attention of mobile users. Examples of browsers for mobile devices are:

- Opera Mini (J2ME based) for devices with J2ME
- Opera Mobile (native code) for Windows Mobile, Symbian.
- Microsoft Internet Explorer Mobile for Windows Mobile
- Fennec (mobile Firefox) beta on Internet Tablet OS2008, expected for Windows Mobile and Symbian.
- In addition, specific browsers for mobile devices exist, e.g., Nokia's browser for S60.

3.2 3D Internet Standards

On March 24, 2009, the Khronos Group announced the initiative *Accelerated 3D on Web*, to create an open, royalty-free standard for bringing accelerated 3D graphics to the Web (Khronos Group, 2009). Standardization work started in April 2009 with a planned first public release within 12 months. This is timed well with our proposed approach, as we regard the integration of the distributed rendering framework into web browsers as a bottleneck. This new standard has the potential of simplifying integration into browsers.

COLLADA (Collada, 2009), a 3D standard by the Khronos Group, is a XML-based format for standardized encoding of visual scenes including: mesh geometry, shaders, materials, animation, kinematics, allowing multiple versions of the same asset. It is used in games and various digital graphics authoring software.

Until now, there have only been a few other attempts to create an open standard for distributing 3D graphics over the Internet. The first was in the 1990s with VRML (Virtual Reality Modeling Language). However, the language (and its associated tools) did not gain a wide acceptance (Parisi, 2004). Recently similar activities have been re-established resulting in X3D (Web3D Consortium) - an open, XML-based ISO standard for describing 3D graphics. X3D is also a run-time environment, and consequently complementary to coding standards such as COLLADA. There are also other initiatives, e.g., 3DMLW (3D Markup Language for Web) and the ECMA standard U3D (Universal 3D).

3.3 3D Internet Technologies

Macromedia Shockwave 3D sends the 3D representation, and later incremental updates, only as needed to minimize the demand for network bandwidth during the interactive session. To adapt to low bandwidths when initializing the session, Macromedia Shockwave 3D uses algorithms to simplify textures and to make fewer polygons (Harris, 2009).

Java 3D API enables the creation of 3D applications and Internet-based 3D applets with Java's "write once, run anywhere" benefit. It is aimed at a wide range of 3D hardware and software platforms, from low-cost PC game cards and software renderers, to high-performance specialized 3D image generators. Java 3D is designed to scale as the underlying hardware platforms increase in speed over time.

Neither of these solutions is based on an open 3D standard, and neither offers, as far as we know, content adaptation across a wide range of platforms such as we are proposing.

4. SHAPE REPRESENTATION

The idea is to target accurate and compact client-independent 3D content representation, using an algorithmic approach for adapting a single version of the 3D content to the different client platform specificities. The objective will be to use and adapt current 3D shape representation approaches, not the development of new shape representations. We plan to focus on:

1. Semantic procedural shape descriptions, i.e., the Generative Modelling Language (GML, 2009)
2. Higher-order surface descriptions, e.g., subdivision surfaces and NURBS
3. Traditional triangle-based representations as usually found on the Internet

All types of shapes can have additional material information, especially textures. Shapes can also have fine surface details, which can be represented geometrically or as displacement maps.

The most popular shape representations for 3D content are:

- Triangulations are the dominant representation of 3D shapes within computer graphics applications.
- Subdivision surfaces brought 3D animation of complex shapes to a new quality level when introduced by Pixar in 1997 in the motion picture industry (DeRose, 1998)
- NURBS – NonUniform Rational B-spline surfaces used primarily in CAD systems (Farin, 2002).
- Procedural representation, as in GML, allows creation of complex models from a few parameters.
- Elementary surfaces including the sphere, cylinder, cone and torus.
- Implicit representations define the shape as the zero set of an equation $f(x,y,z) = 0$.
- Multi-resolution representation approaches combine the above shape representations to reduce bandwidth and memory requirements.

5. DISTRIBUTING 3D CONTENT AND PARALLELIZING RENDERING

The performance of real-time rendering has improved significantly in the last years, mainly due to many- and multi-core CPUs and programmable GPUs. However, realistic interactive rendering of large 3D models still requires the distribution of rendering subtasks (Repplinger, 2009). In a cluster of machines, the network is the bottleneck. Cluster adaptation techniques to reduce the network load are required. The rendering subsystem OpenSG provides distribution and load balancing for cluster-based rendering (Gierlinger, 2004; OpenSG, 2009; McCool, 2008).

Even in a non-networked setup of one heterogeneous client (multi-core CPU and GPU), best use of existing resources is vital. For content rendering local load balancing is vital. For content distribution the transfer of 'the right' shape representation and sets of data to the client is essential (coarse load balancing).

In parallel and distributed rendering one can distinguish two main streams that both need a good balanced approach in order to efficiently explore the available processing recourses:

- Distributed scene graph systems are used for interactive rendering of large scenes with limited rendering quality that cannot be rendered interactively otherwise (Voß, 2002).
- Parallel implementations of certain rendering algorithms, especially ray tracing (Carr, 2002; Chalmers, 2002; Parker, 1999; Georgiev, 2008: Pharr, 2005: Purcell, 2002; Seland, 2007; Wald, 2004), are used for speeding-up offline rendering techniques (Won-Jong, 2005) using global illumination methods (Schmitz, 2008) for very high quality.

6. DISTRIBUTED RENDERING FOR 3D INTERNET

The idea is to separate 3D model representation and graphics processing to allow distribution of 3D web applications between server and client to facilitate better use of the actual network, server and client resources. An infrastructure has to be developed consisting of 3D content distribution, adaptation strategies and parallel processing capabilities to adapt to network conditions, server and client processing, and rendering capabilities. This is important to be able to strive for a best possible balance of network load, visual fidelity, and interactivity. To achieve new levels of flexibility and adaptation, it will be important to explore non-traditional model representations, such as compact higher order geometry, e.g., procedural representations, subdivision surfaces, hierarchical triangulations and NURBS. By this we achieve:

- A compact model representation on the server.
- The possibility to (partially) transfer the compact model to the client for rendering.
- Tessellation and rendering of models according to client specific parameters.
- Distribution of tessellation and parallelization of rendering on the server and/or client side for optimal performance according to the total resources available:
  - When sufficient client processing resources are available, perform tessellation and rendering on the client to reduce the required network bandwidth.
  - Split tessellation and rendering between the server and the client
  - Perform rendering on the server and stream the resulting images to the client.

Figure 1.1 shows different alternatives for the data communicated from the server to the client.

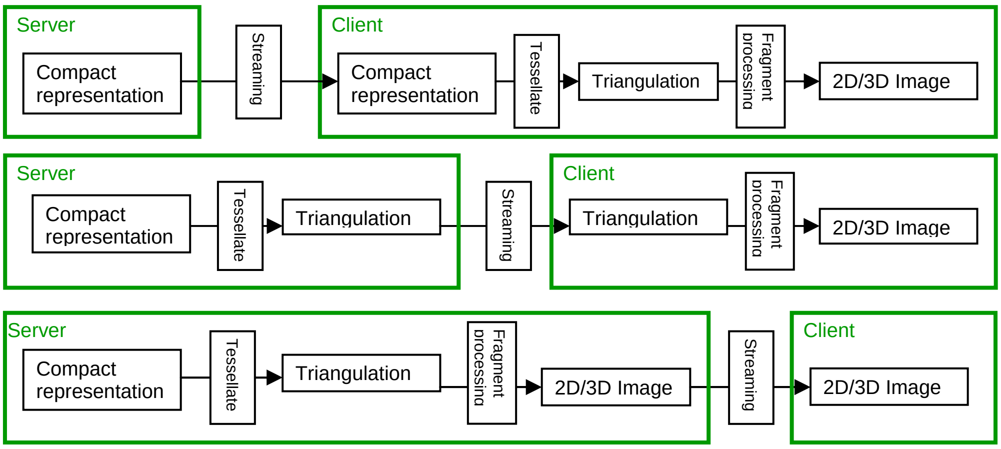

Figure 1. In the Case at the Top the Client has Sufficient Resources to Render Higher Order Representations. In the Middle Case the Server Converts the Higher Order Representation to a Triangulation and Streams this to the Client. In the Bottom Case the Complete Rendering is Performed on the Server.

6.1 Integration of 3D Content into Web-Browsers

The ease of integrating custom solutions with existing desktop browsers varies between browsers. Some like Mozilla Firefox offer a well documented API and support for plug-in developers. Others like Internet Explorer are harder to integrate into. In the case of mobile browsers, the possibilities for integration are even more limited, as plug-in interfaces (if they exist) are not as easily accessible as on desktop browsers.

To achieve a 3D Internet solution, it has to easily integrate into the most popular browsers both on the desktop and on mobile devices. Among the alternatives to consider are:

1. Exploit the standard *Accelerated 3D on Web* (Khronos Group, 2009).
2. Integrate with an existing popular plug-in, e.g., Flash. Until *Accelerated 3D on Web* was announced, integration through Flash seemed the way to go.
3. Use an existing 3D web technology.
4. Develop a dedicated plug-in for major browsers is not attractive as it will have limited use.

7. CONCLUSION

We believe that a broad uptake of '3D Media Internet' depends on a significant improvement of the user quality of experience. This should not significantly differ from the quality offered by similar 3D services installed on the user's device.

REFERENCES

Carr N.A, 2002,. The ray engine. *In Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, Saarbrucken, Germany, pp. 37 – 46.

Chalmers A. et al., 2002*, Practical parallel rendering*. A. K. Peters, Ltd. Natick, MA, USA.

Collada, 2009, https://collada.org/, [25 March, 2009].

DeRose T. et al., 1998, T. Subdivision Surfaces in Character Animation, *In Proceedings of SIGGRAPH '98*, pp. 85–94.

Farin G, 2002. *Curves and Surfaces for CAGD, A Practical Guide*. Morgan Kaufmann Publishers, San Francisco, USA.

Gierlinger, T. and Prabhu P, 2004. Towards Load Balanced Computations using GPU and CPU. *In Proceedings of 2004 ACM Workshop on General-Purpose Computing on Graphics Processors (GP2)*, Los Angeles, California, p C-14.

Georgiev I. et al, 2008. Real Time Ray Tracing on Many-Core-Hardware. *In Proceedings of the 5th INTUITION Conference on Virtual Reality*, Torino, Italy. To appear.

GML, 2009, The Generative Modelling Language, http://www.generative-modeling.org, [25 March, 2009].

Harris T., 2009, How Shockwave 3-D Technology Works, http://computer.howstuffworks.com/shockwave.htm, [25 March, 2009].

Khronos Group, 2009, Accelerated 3D on Web, http://www.khronos.org/ [25 March, 2009].

McCool M., 2008, Scalable Programming Models for Massively Multicore Processors. *In Proceedings of the IEEE*, vol. 96, No. 5, pp. 816-831.

Net Applications, 2009. Browser Market Share, http://marketshare.hitslink.com/ [April, 2009].

OpenSG, 2009, http://www.OpenSG.org, [March 25, 2009].

Parker S. et al., 1999. Interactive ray tracing for volume visualization. *In IEEE Transactions on Visualization and Computer Graphics*, July-September 1999.

Parisi T., 2004. The Immersive Web(log) "Like a Phoenix From the Ashes: X3D and the Rebirth of Reason", http://flux.typepad.com/the_flux_papers/2004/08/project_phoenix.html, [25 March, 2009].

Pharr M., and Fernando R. 2005. *GPU gems2: programming techniques for high-performance graphics and general-purpose computation*. Addison-Wesley Professional.

Purcell T.J. et al., 2002. Ray Tracing on Programmable Graphics Hardware. *In ACM Transactions on Graphics*, Vol. 21, No. 3, pp. 703 – 712.

Repplinger M., Löffler A., Thielen M, and Slusallek P., A Flexible Adaptation Service for Distributed Rendering. *In Proc. of Eurographics Symp. on Parallel Graphics and Visualization 2009*, Munich, Germany, March 29-30, 2009.

Seland J. S. and Dokken T., 2007, Real-time algebraic surface visualization. *In Geometrical Modeling, Numerical Simulation, and Optimization: Industrial Mathematics at SINTEF*, Springer, Heidelberg, Germany, pp. 163-183.

Schmitz A. et a. 2008. Interactive Global Illumination for Deformable Geometry in CUDA. *In Graphics Forum*, Vol. 27, No. 7, pp 1979 – 1986.

Voß G. et al, 2002. A multi-thread safe foundation for scene graphs and its extension to clusters. *In EGPGV '02: Proc. of the Fourth Eurographics Workshop on Parallel Graphics and Visualization*, Blaubeuren, Germany, pp. 33–37.

Wald I., 2004, Realtime Ray Tracing and Interactive Global Illumination. PhD thesis, Saarland University.

Web3D Consortium, X3D, http://www.web3d.org/, [25 March, 2009].

Won-Jong L. et al., 2005. Adaptive and Scalable Load Balancing Scheme for Sort-Last Parallel Volume Rendering on GPU Clusters. *In Proceedings of International Workshop on Volume Graphics, 2005*, Stony Brook, New York, USA.