Research on GPU Parallel Methods of Real Time Rendering

Yahui Zhu[1, a], Xiangnian Huang[2, b] and Yajun Zhang[3, c]

[1]School of Computer and Software Engineering, Xihua University, Chengdu 610039, China;

[2]School of Computer and Software Engineering, Xihua University, Chengdu 610039, China;

[3]School of Computer and Software Engineering, Xihua University, Chengdu 610039, China.

[a]zyhxhu@163.com, [b]huangxn@vip.qq.com, [c]zhangyafei1314@qq.com

Abstract. Currently, Graphics rendering have a very important role with the rapid development of computer graphics hardware. We are in order to meet requirements of the large complex scenes in real-time virtual reality interactive nature, with high precision and high efficiency. This paper, based on reading a large number of domestic and foreign references, reviewed GPU real time ray tracing render algorithm parallel of large-scale dynamic scene computing concepts, research status and research hot topics, and finally summarized the research difficulties and development direction of exist technical difficulties and prospected.

Keywords: Computer graphics hardware, GPU, Parallel computation, Rendering.

1. Introduction

The computer information technology, and the computer graphics theory and technology by with their rapid development, there are need of massive scene data using of computer technology, which with three-dimensional visualization is a form of manifested, it is an inevitable trend that rending development. People demand in dynamic scenes in real-time rendering technology to ensure the real-time outside also create a very realistic and convincing virtual natural environment has become increasingly intense. Large-scale real-time rendering of complex scenes is very extensive application, in the military, aerospace, medicine, geology, exploration, three-dimensional game, and other fields' industrial computer-aided design as a supporting role is significant. The CPU heavy burdens are growing problem of large-scale scene rendering, while the computer GPU graphics processor unit is developing rapidly, which are support for more complex calculations and programming modes. Through the GPU programming to play its high strength parallel computing power to solve the three-dimensional real-time rendering speed bottleneck, we can do improve the efficiency of the parallel algorithm, of which will be a very valuable of applications research contents.

2. Analysis of the Subject research

2.1. The Research Present Situation.

In recent years, the scientific visualization, the computer animation and the virtual reality are the three most active development direction in the field of computer graphics, which those the core technologies are realistic three-dimensional graphics display.

The current three-dimensional real-time rendering techniques in virtual reality: which is a very important valuable work that research on medical (tomography), entertainment (film production, advertising design), military and aerospace, and other field's e-commerce and CAD.

Improving the quality and real-time rendering of complex scenes, real-time graphics rendering core technology is the future of all 3D applications.

The domestic market, although relatively hot 3D applications, while rendering technology has developed relatively backward, and through the three rendering techniques (rasterization, ray tracing, radiometric) gradually promote the rapid development of 3D gaming industries, which makes rendering a research hotpots[1].

We are in pursuit of a more realistic graphical effects, by which are faced with large-scale complex dynamic scenes in real-time rendering algorithms computing is increasingly complex and important technology. In performance, and GPU computing to support increasingly complex, high intensity parallel computing power to solve the three-dimensional real-time rendering, improving the efficiency of the algorithm even more impressive.

According to the current needs of the practical application, the use of information-rich intuitive 3D scene instead of a simple two-dimensional images will be the development trend in the field of computer graphics.

Currently, the conducting high-quality rendering of 3D scene is still a very time-consuming calculation process, CPU also began to develop in the direction of dual-core and even multi-core, but still cannot meet the real-time requirements of practical application [2]. Domestic and foreign scholars have the research focus shifted to the multi-core processors, which the use of multi-core parallel computing.

GPU has a high-strength data parallel computing power to bring a breakthrough in computing performance beyond graphics rendering, and applies physics simulation, signal processing, global illumination, biological computation, geometric computing and data mining have been made a better research results[3].

The real-time rendering of large-scale complex scenes require computing is growing, as demand of large-scale and complex scenes to solve the high-quality 3D scene rendering is still a very time-consuming calculations. So we proposed an intractable global illumination algorithms based on GPU [5], in order to ensure that under the real degree of image proposed radiometric and ray tracing algorithms.

In 2013, Parallel Visualization and Computer Graphics Conference [4], which pointed out that the real-time ray-tracing rendering algorithms research focused on the increase in computing speed and the real degree. While domestic proposed ray and triangle intersection rapid parallel algorithm research based on GPU and the real-time KD-Tree and dynamic light scenes for real-time ray tracing based on Opens, etc. and other studies for large-scale scenarios ray tracking solution provides the foundation.

2.2 The Purposes and Significance.

We hope to render engine that optimized analysis modes, by which fully surrounded GPU parallel computing capabilities to conduct research. At present in order to ensure the authenticity of the image, what a commonly used method is compared radiometric algorithms, focusing on generating ray tracing algorithm by means of parallel GPU calculation to real-time rendering, which become more smooth and natural scenes.

By means of ray tracing invisible light sources culling efficiency improvements within the scene, while ours to design the entire rendering software system, to take advantage of GPU parallel computing power for large-scale dynamic lighting scenes do real-time rendering, and to achieve the research goals of established reduce CPU burden.

We hope that by analyzing a variety of mature 3D game engine, the core part of the 3D rendering engine light sources---complex dynamic scene rendering techniques are discussed [5]. We design and implement some of our own programs, to do the development of our commercial game engine so that it brings some inspiration.

The domestic development of rendering engine technology that itself is relatively falling, which increasingly unable to meet people requirements that pursuit reality and real-time of large-scale complex dynamic between scenes visual and virtual interaction, we research and realization of an efficient rendering engine is a very important business sense.

Under the impetus of the game industry, computer graphics processor unit (GPU) performance is development by times speed, in order to pursue a more lifelike graphical effects, GPU support for more complex calculations, becoming more and more power to solve the problems that in real time of large-scale complex scenes, which has a very important significance.

Get through the GPU programming to play its high intensity parallel computing power to solve the three-dimensional real-time rendering speed bottleneck, which will improve the efficiency of the algorithm, will be a very valuable of applications research contents.

How to use the current GPU Programming Capabilities Programming and high intensity parallel computing capabilities, what's the pursuit of more efficient performance, lower power consumption is one of the current and future research hotpots.

Without prejudice to the quality of three-dimensional rendering and the degree of algorithm complexity, and we studied how to efficiently make use of the scene division algorithms based on GPU and real-time three-dimensional rendering method [6]. Now, Researchers so that in order to improve the graphics rendering speed, to solve real-time three-dimensional scene rendering in terms of speed, quality and scene complexity increasingly prominent contradiction.

The use of GPU parallel research effectively promote real-time ray tracing rendering, which constantly put under large scale scenes in real-time ray-tracing rendering solutions[7].

Playing to high-performance parallel computing power of the GPU to solve the efficiency problem of three-dimensional real-time rendering [8], what is a broad application prospect for work.

3. The Design of GPU Parallel Methods

3.1. Real Time Rendering Contents.

The premise is not obvious reduce the three-dimensional rendering quality, and reduce the complexity of calculation, to achieve real-time high simulation display of the scene. We research and adopt some division processing methods and graphics rendering algorithms can be used in large scale dynamic lighting scene, while the parallel computing based on GPU, combined with real-time rendering performance optimization solutions to enhance the three-dimensional rendering speed, to solve the three-dimensional real-time rendering between speed, quality and scene complexity increasingly prominent contradictions, these will be the main content of our study.

We do three-dimensional real-time rendering pipeline optimization processing based on GPU and parallel computing performance optimization processing. For the rasterization, The ray tracing and the radiometric algorithms, we do comparative analysis of their advantages and disadvantages, to optimize the use of GPU ray tracing algorithm, in which making the secondary light intersection computing, it is combined with parallel PKD-Tree improved dynamic scene classification method based on GPU that to accelerate real-time rendering of large-scale dynamic lighting scenes. We analysis the main characteristics of calculations resource consumption of ray tracing, from the aspects of mathematical formulas, and research kid-tree to improve ray tracing rendering algorithms, improving the efficiency of light removed from the perspective of parallel GPU computing.

The real-time ray-tracing rendering algorithms are further improve, by us, that especially mining hardware parallel computing power. According to the independence of the rays emitted by each pixel on the screen, which the individual pixels can design a ray tracing processing unit, this way can improve ray tracing algorithms concurrency and thus improve the computation speed. The ray tracing with the massively parallel capabilities, thus we research object that how to use multi-threading to further hidden delays, combined with batch processing object unit and distributed performance optimization solutions to further improve rendering efficiency. We research on pre-rendered dynamic of ray tracing technology, produced random three-dimensional dynamic large-scale scenes, on the performance generated in real time rendering realistic images, to further improve the ray tracing rendering algorithm itself shortage. In this paper, we have Accelerated parallel rendering algorithms based on CUDA, which is the basis of multi-GPU clusters manner to the experimental research.

3.2. The Design Approaches and Technical Routes

GPU programmable graphics processor to offer the effectively support:

(1). Advanced illumination model --- arbitrarily complex illumination model;

(2). The per-pixel lighting --- per-pixel lighting effects;

(3). The process of texture --- procedural texture effects;

(4). CUDA general-purpose computing.

GPU programmable rendering pipeline show that:

Input data (vertex description) ->Rasterization processing->Global cache or texture cache->Pixel operations->Output data (image).

Rendering Languages about that:

In the vertex or fragment processors supported are: Direct X Advanced Rendering language HLSL; OpenGL advanced rendering language GLSL; NVIDIA's CG language.

An example shows that the large-scale dynamic of light and shadow scenes, which rendered as basic reference route. We set up a large-scale virtual dynamic lighting scene, where to simulate a three-dimensional scene objects and character animations, ground scenes and indoor scenes, etc. Follow: the first step is to generate depth maps, the second step is generated rendering. This test route technical methods be adopted to simulate that the following figure.

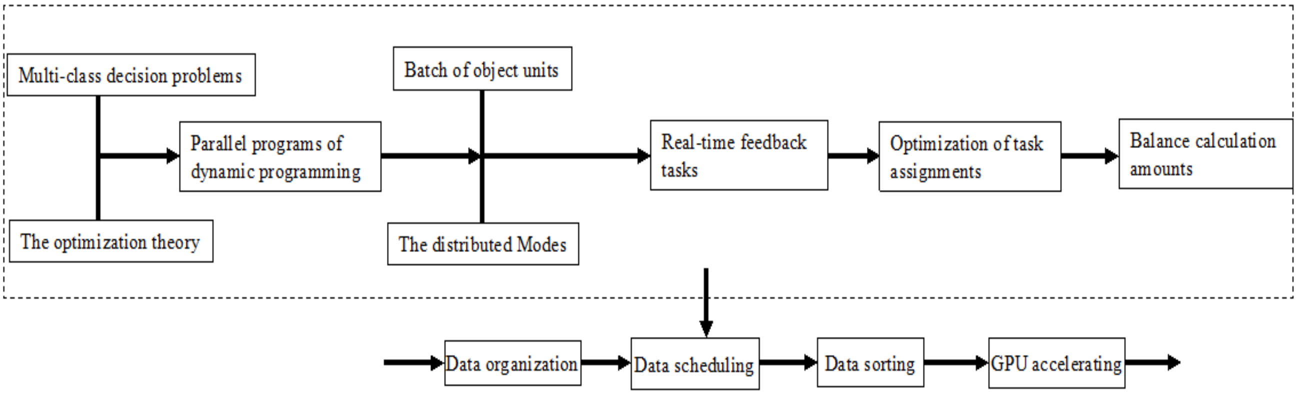

Fig. 1 The first step: Unit design of modes

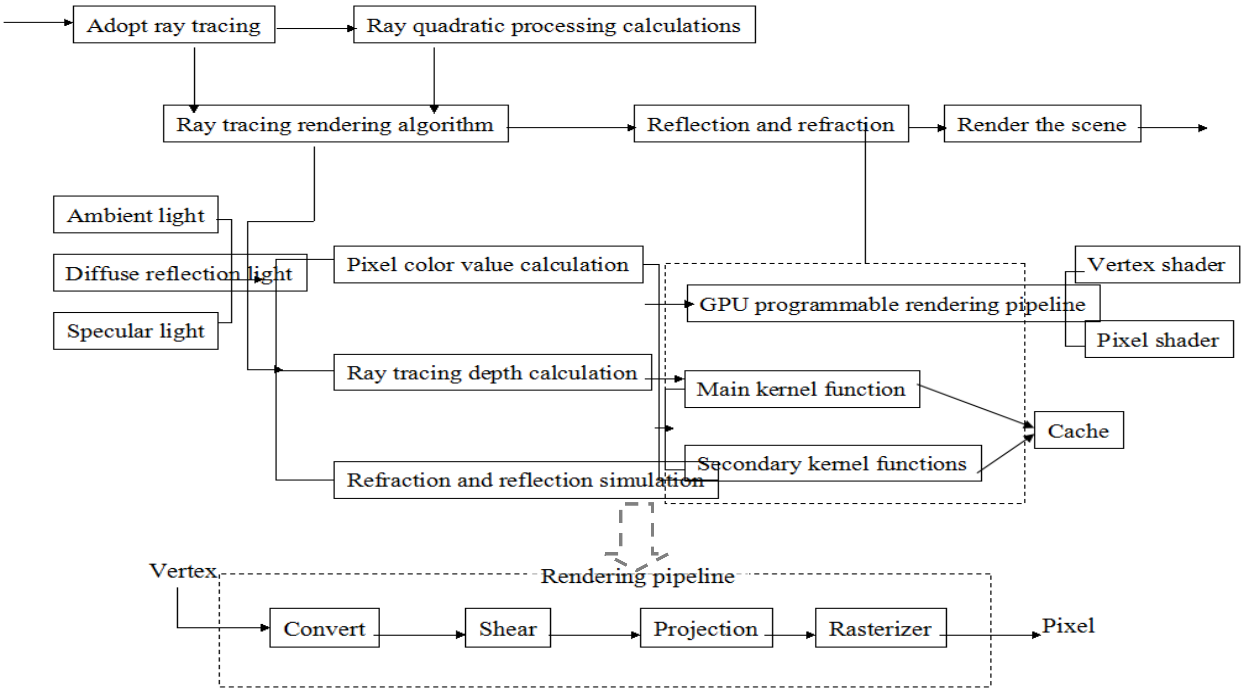

Fig. 2 The second step: Rending methods design of ray tracing

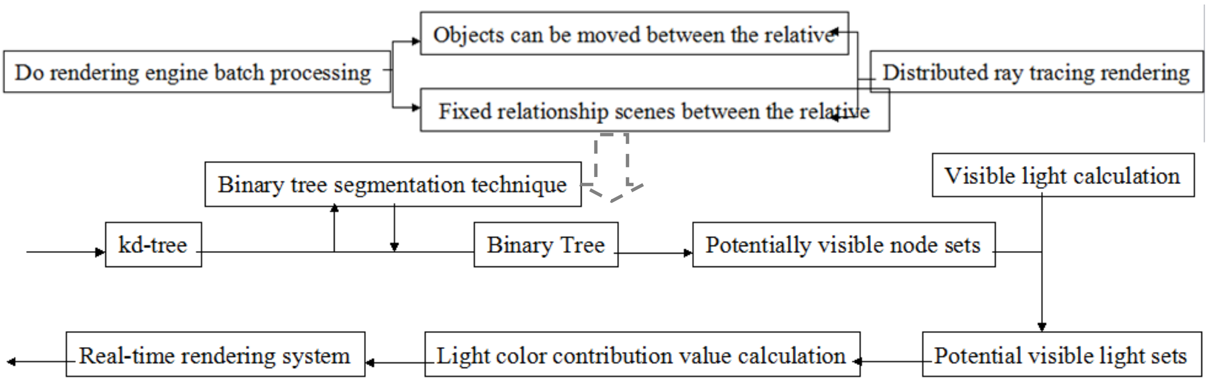

Fig. 3 The third step: Real-time rendering system design of kid-tree

The GPU technical framework and the principles of acceleration:

We are full use of extra-long pipelining technology, caching, parallel computing and GPU programmable rendering pipeline. Completed the following tasks:

(1). The local coordinate of input objects transform into the final clipping coordinates;

(2). Generating texture coordinates of each vertex;

(3). Generated colors of each vertex.

Multi-threaded parallel construct that PKD-Tree structure based on GPU acceleration method (shown in Fig. 4). This method is focused on solving the parallel computation in between frames and frames, which to parallel computing in the image plane between sampling points.

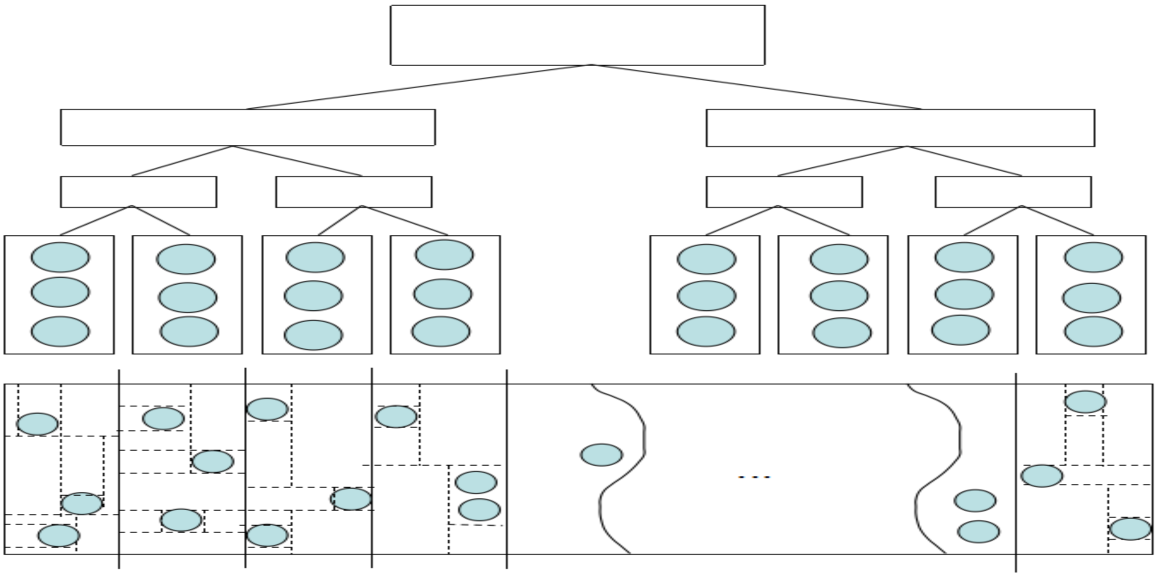

Fig. 4 Multi-threaded parallel construct PKD-Tree

Three-dimensional real-time rendering technology:

(1). The complexity reduced of the scene: Spatial data structure, cropping and removing method;

(2). The rendering speed Improved: Calculation method of light and shadow, Texture processing method;

(3). The technology of hardware supported: Programmable shaders, including vertex and fragment shaders;

(4).The programmable rendering pipeline optimization: Optimization methods of application stage, optimization methods of vertices and rasterization stages, performance bottlenecks detecting;

The ray tracing recursive algorithm:

We adopt CUDA technology to achieve recursive process, the whole algorithm is consists of the core kernel functions tree composed from the main kernel functions and second-class kernel functions, the entire kernel functions tree should be in accordance of depth-first approach(shown in Fig. 5), which calling second-class kernel functions to complete the rendering process.

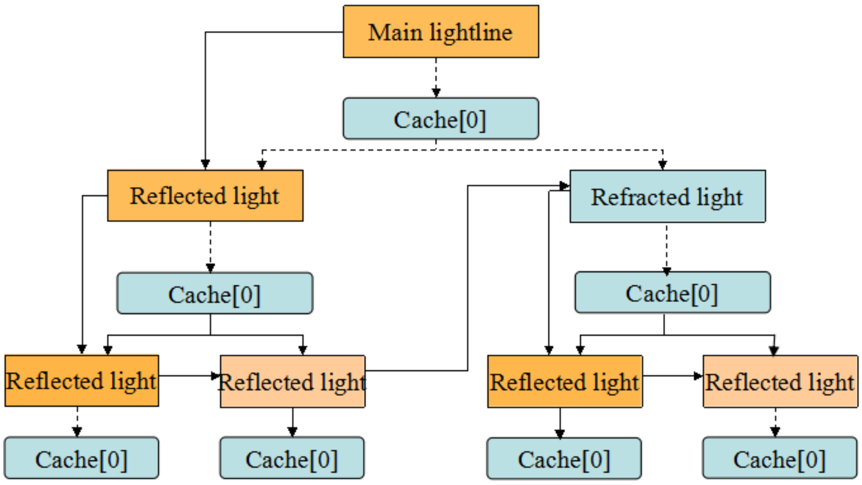

Fig. 5 In depth-first manner called the kernel function tree

The batch rendering method:

We introduction of the batch object units, So that the scene is divided by module, which according to the different properties divided into a class, and then going on the small module doing performance optimization. This method can reduce the cache storage time, which has improved real-time rendering speed.

The distributed real-time ray tracing rendering technology:

We select two allocation methods are that real-time task allocation algorithm and real-time feedback task allocation algorithm, their advantages of good computing performance, good reliability, good price, optimized task allocation, stronger balanced amount of calculation.

The Potential visible light source sets algorithm based on kd-tree:

We do processing for binary trees by the two-difference Segmentation technique, collect its potential visible light source sets, and do calculating of the light source visible, after that find potential visible light source sets, then do calculating of the light source color contribution value, finally, adoption geometric computing of the selected corresponding light, on the screen system to complete the real-time rendering.

3.3. Characteristic and Innovation Points

(1). We introduce a new object unit in the object module that is batch object units, which the theory is that some of the objects with common features bundled together into GPU rendering. This unit simpler compared with more commonly used units can be reduced to the GPU calling, enhanced rendering performance.

(2). We are use of split-screen technique to assign the task models to the client, in order to take full advantage of the performance of batch processing and distributed computing, according to the feedback the calculated amount of client computers to distribute different computational tasks in different ways, which for ray-tracing rendering accelerated.

(3). The large-scale dynamic scenes not only taking into account the situation between both objects can be moved relatively, but taking into account the relative relationship between the fixed scene.

4. Conclusion

By reading a lot of domestic and foreign references, we have further study mainly about theoretical and technical methods of ray tracing rendering algorithms under large-scale dynamic scenes, at the same time improving its optimization program, which imposed more optimization programs that are able to reflect the sound and rapid rendering performance. At ensure the authenticity of scene while improve the real-time rendering will open up a wider road, we hope that our efforts and research energy on the application and promotion of virtual reality technology to start enormous guiding role. The main contribution of this paper to achieve results as follows:

(1). Through the GPU programming to play its high strength parallel computing power to solve the three-dimensional real-time rendering speed bottleneck, which improve the efficiency of the algorithm.

(2). In large-scale and complex dynamic scenes, we are under situation that to ensure the quality of three-dimensional rendering and improve the three-dimensional rendering speed, to solve the increasing prominent contradictions of three-dimensional real-time rendering between speed, quality and scene complexity.

(3). We are full use of the GPU's principle of maximize parallel implementation, the principle of maximum memory bandwidth, the principle of maximum instruction throughput, to improve the optimization performance of parallel computing, which can make the real-time rendering more smooth, more real's.

References

[1] Q. Lin, S.L. Dai, et al. A Black LOD Real-time Rendering Algorithm for Large Scare Terrain, Journal of Computer-Aided Design & computer Graphic, Vol. 25 (2013) No.5, p.708-713.

[2] Y.N. Zhang: Architecture of Real-Time Rendering Engine, ZTE Technology Journal, Vol. 19 (2013) No.6, p.48-53.

[3] S.Q. Chen: *GPU Efficiency Based Graphic Engine Investigation and Implementation* (Ph.D., Shanghai Jiaotong University, China 2013), p.46.

[4] Bruneton E, Neyret F, Holzschuch N: Real-time realistic ocean lighting using seamless transitions from geometry to BRDF, Computer Graphics Forum, Vol. 29 (2010) No.2, p. 487-496.

[5] Tomas Akenine-Moller, Eric Haines, Naty Hoffman: Real-Time Rendering. Third Edition. A K Peters, Ltd, Vol. 10(2008) No.7, p.11-13.

[6] G. Chen, B. Li, F.L. Tian, et al. Design and implementation of a 3D ocean virtual reality and visualization engine, Journal of Ocean University of China, Vol. 11 (2012) No.4, p. 481-487.

[7] L.J. Huang: *Research of Distrubuted Real Time Ray Tracing Render Base on kd-tree Technology* (Ph.D., Harbin Engineering University, China 2013), p.23.

[8] H.Q. Mao: *The Research on the 3D Real-time Rendering OPtimized Base on GPU* (Ph.d., Wuhan University, China 2010), p.31.