2012 International Conference on Image, Vision and Computing (ICIVC 2012)
IPCSIT vol. 50 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V50.33

High Realistic Real-Time Rendering for Night Scene

Hao Wang [+] , Xihou Li and Hongyu Yang

College of Computer Science
National Key Laboratory of Fundamental Science on Synthetic Vision, Sichuan University
Chengdu, China

Abstract. Aiming at characteristics of night scene, a high realistic real-time rendering method is presented. Deferred shading is employed to process multiple light-sources Lighting with improved HDR technology used in the post-processing stage, then produce glare effects. The experiment results show that, this rendering method for night scene can achieve both real-time performance and high realistic effects.

Keywords: night scene; deferred shading; HDR; glare; real-time rendering

1. Introduction

Night scene rendering is an important component of high realistic virtual scene rendering. Night scene has its own characteristics, which is different from scene during daytime, including:

a) Night scene needs multiple light-sources lighting;
b) Most parts of the scene are dim while others which are lighted are bright;
c) Highlight parts produce glare effects.

There are three major options for real-time multiple light-sources lighting[1], 1. use a single render pass; 2.use multiple render passes; 3.use deferred shading technique[2]. The first two solutions suffer from low efficiency and high computational complexity which is O(number of objects ×number of light-sources), while O(number of objects + number of light-sources) for deferred shading, it greatly improve the efficiency of rendering scene with multiple light-sources.

The second characteristic of the night scene is the intense contrast between the lighten and the dark, the luminance range of real scene (ratio of maximum and minimum luminance) usually distributes more than 5 levels, while 1 level for ordinary display devices, which results in lack of reality because of the confliction between them[3]. High dynamic range technique (HDR)[4] stores color information of the scene in floating texture of high range data information, not just [0,1], then map them into a low range with tone mapping algorithm, so as to achieve high realistic real-time rendering of the night scene. There are mainly two kinds of tone mapping algorithms: (1) Spatially Uniform, in which the entire image uses an identical transform function in the mapping process, is faster, however, to some extent ,it losses some color, luminance details [5]; (2) Spatial Varying, comparing to the former one, different transformations are used in different regions, which improves the quality. However, it costs high computational complexity, low rendering efficiency [6]. Previous method [7] proposed a tone mapping method for mapping high-dynamic range of luminance to the low range for display, which would not reduce the contrast degrees of the scene. Previous method [8] improved it with adaptive experience parameters in computation. They need to be improved for it is difficult to calculate the average luminance efficiently.

[+] Corresponding author. Tel.: + (13679015307).
E-mail address: (wanghaohord@126.com).

Bloom[9] process spreads color values of highlight parts of the scene to the surrounding pixels, resulting in blooming effects. Unfortunately, the highlight in the night scene will produce the glaring phenomenon, which cannot be generated by Bloom process, therefore, a glare generating process is absolutely necessarily.

This paper proposes a method for night scene rendering. Considering the characteristics of the night scene, we use deferred shading technology for multiple light-sources lighting, with improved HDR technology and glare generation algorithm in the post-processing stage to meet the needs of high realistic real-time rendering for night scene.

2. Rendering Method

A. Deferred Shading

Deferred shading technology improves the traditional graphics rendering pipeline, General architecture of deferred shading technique is shown in Figure 1 [10]:

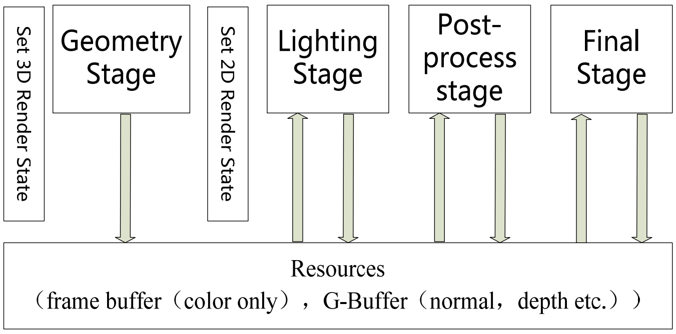

Fig. 1: General architecture of deferred shading technique

1. Geometry stage: implements a perspective projection camera, in 3D space, and it is responsible for feeding the G-buffer with information to be used in the following stages, such as color information and normal information. All subsequent stages operate in image-space and work with an orthographic projection camera, by using screen resolution dimensions.

2. Lighting stage: receives as input the contents of the G-buffer as well as light-sources information and it accumulates lighting into a texture buffer. Computational complexity of Lighting is O(number of objects + number of light-sources).

3. Post-process stage: in this stage, some post-processing passes are performed to enhance the image generated in the previous stage, such as HDR and glare generating process.

4. Final stage: the enhanced image is transferred to the main frame buffer in order to be displayed.

Deferred shading technique has some advantages against traditional pipeline:

•computational complexity (O (number of objects + number of light-sources)) lesser than traditional rendering techniques (O (number of objects ×number of light-sources));

• allows the application to take advantage of batching;

• allows a post-process stage in order to enhance the final result. In this paper, we use HDR process and glare generating algorithm.

Therefore, deferred shading can improve rendering efficiency, and achieve better rendering effects, corresponding to the characteristics of the night scene.

B. High Dynamic Range

In high realistic rendering, the human vision system (HVS) is an important concept. In the natural environment, the human vision system can identify the light intensity of 10-2 to 103, while the ordinary devices can only display 0-102, which is far below the real situation. In the night scene, great difference in luminance exists between the lighten and dark, the RGB channels of each pixel of ordinary display device distribute only between 0 and 255 in gray scale, though, which is not enough.

To achieve high realistic rendering for the night scene, we use high dynamic range technology. HDR employs actual physical parameters and functions to achieve a better range of color and luminance.

In this paper, tone mapping is processed with the texture generated by the first two steps of deferred shading. We improve the tone mapping algorithm by use of the characteristics of the night scene, to increase processing efficiency without sacrificing the rendering quality.

For any pixel located at coordinate (x, y), the luminance value Lw (x, y) is derived as follow:

$$L_w(x, y) = 0.2125R + 0.7154G + 0.0721B \quad (1)$$

Where R, G, B are RGB channels. This process generates a luminance texture from the color texture.

The average luminance value of all the pixels $\bar{L}_w$ is:

$$\bar{L}_w = \exp[\frac{1}{N} \sum_{x,y} \log_2(\delta + L_w(x, y))] \quad (2)$$

Where N is the total number of the pixels, δ is a small constant value to avoid the singularity that occurs if black pixels are present in the texture. It costs much to get the values of all the pixels in the render pipeline, so down-sampling is used.

Down-sampling is to sample the luminance texture every N*N pixels, and store the luminance values generated by bilinear interpolate in a smaller texture. The sampled value is approximate to average value of N*N region. Continue this process on the smaller texture until get a 1*1 luminance texture, in which, the value is approximate to the average value of the original texture. Example for down-sampling is shown in Fig.2.

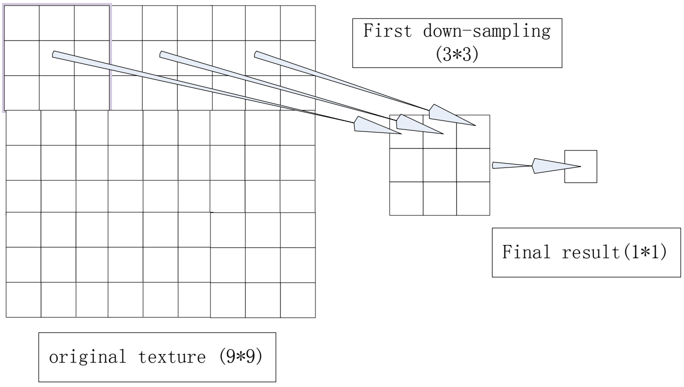

Fig. 2: Example for down-sampling.

Down-sampling costs less time and gains higher efficiency. 5 times is enough to get average value for a 1024*1024 pixel texture sampled every 4*4 pixels, the total number of processed pixels is 256*256+64*64+16*16+4*4+1*1=6.9905*104, much less than 1024*1024=1.048576*106 , which directly use (2).

Different from day-time scene, luminance value is close to 0 in many regions of night scene, so we can even define N as big as 16, average value generated by down-sampling is also approximate to the exact average value. The efficiency of calculation is improved vastly, since the number of processed pixels decrease to 4113 as in the above example.

Then, tone mapping operator is given by:

$$L(x,y) = \frac{key}{\bar{L}_w} L_w(x, y) \quad (3)$$

$$L_d(x,y) = \frac{L(x, y)(1 + L(x, y) / L_{\max}^2)}{1 + L(x, y)} \quad (4)$$

Where $\bar{L}_w$ is generated by down-sampling, key is set by experience, it adjusts the luminance, as shown in Fig.3. Lmax is the smallest luminance that will be mapped to pure white.   Ld(x,y)is the final result of tone mapping.

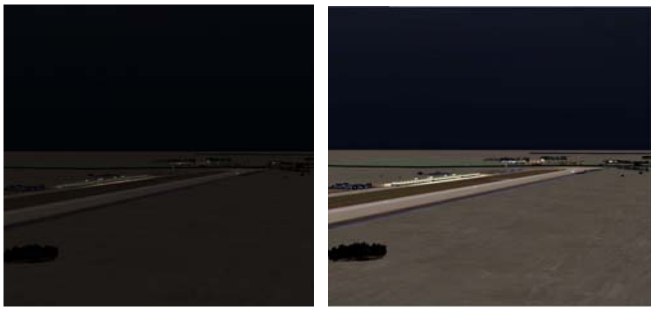

Fig. 3: Adjust luminance by key value. (Left, key value 0.18; right, key value 0.36)

We calculate the value of lighting in high dynamic range and store color values in the texture in Lighting stage of deferred shading. In post-process stage, we use the method presented in this section to map values to the low range for display, to simulate effects of night scene with high realistic, contrastively lighting.

C. Glare Generation

Bloom process can produce animating blooming effects, however, it cannot produce the glare effects of the light-sources at night. This section presents an algorithm to generate highlight glare effects.

During the shading process, each pixel can sample the values of the surrounding ones, while cannot modify them. So we sample the color of the surrounding pixels in 4 directions and increase the color value corresponding to the sampled color and distance between them if the sampled value is larger than a highlight value we defined beforehand. Process the following steps in each pixel shader, we can produce glare effects:

Step1. Define highlight value Chighlight, maximal sampling distance maxdis.

Step2. For each pixel, sample the pixels located in 4 directions(top left, top right, bottom left, bottom right) every other pre-defined distance. If the sampled color value is larger than Chighlight, go to step 3, or go to step 4, otherwise.

Step3. Spread the highlight color to the surrounding places as follow, then go back to step 2.

$$color = color + color_{highlight} * (1 - \frac{dis}{maxdis}) * colorFade \quad (5)$$

Where color is original color of pixel, colorhighlight is highlight color value, dis is the distance between two pixels, colorFade is color fading parameter, 0.3 is suggested. As the result shows, it spreads highlight color in 4 directions and the influence decreases as the distance increases.

Step4. If dis in 4 directions are all greater than maxdis, processing of this pixel is over, otherwise back to step 2.

3. Experiment Results and Analysis

We have implemented our method in C++ programming language, by using Shuangliu Airport model with multiple light-source. The program runs on a machine with Intel (R) Core (TM) 2 Duo E7500@2.93GHz 2.13GHz CPU, NVIDIA GeForce GTX 260 GPU and 2.0G RAM memory.Fig.4 shows the result which uses traditional render pipeline, while Fig.5 uses deferred shading with HDR process and glare generation algorithm is used in Fig.6.

Fig. 4: Rendered by traditional pipeline

Fig. 5: Use deferred shading with HDR process

Fig. 6: Use deferred shading with HDR and glare

Fig.4 shows that, the traditional rendering suffers low contrast, and lack of reality. Fig.5 shows that, comparing with Fig.4, bright things can be really bright, luminance details can be seen clearly. Fig.6 shows the glare effects, it produce an animation night scene.

Table 1 lists the average performance of these methods.

Table 1 Average frame per second

Render method	Average Frame/Second(FPS)
Traditional	77.6
Deferred shading	89.7
Deferred shading with HDR	76.3
Deferred shading with HDR and glare	61.1

Note that FPS of deferred shading was greater than FPS of tradition rendering for multiple light-sources lighting. And the rendering method presented by this paper is completely able to meet the requirements of real-time rendering.

4. Conclusions and Future Work

In this paper, we proposed a high realistic real-time rendering method for the night scene, in which deferred shading for multiple light-sources lighting, with improved high dynamic range process and glare generation algorithm were used. This rendering method for night scene can achieve both real-time performance and high realistic effects. In the future, we will try to improve our method to produce more vivid effects.

5. Acknowledgments

Supported by National “863” Plan Projects of China (No. 2009AA01Z332). We would like to thank Yanci Zhang for their support and enthusiasm.

6. References

[1] Leiting Chen，Hongbin Cai，Guojin Wang，Mingyun He, HDR-Based Deferred Lighting Technology, Journal of University of Electronic Science and Technology of China, 2007, 05:817-820.

[2] Hargreaves, Shawn, and Mark Harris. “Deferred Shading.” Presentation.

[3] http://download.nvidia.com/developer/presentation/2004/6800_Leagues/6800_Leagues_Deferred_Shading.pdf

[4] Cheng Ren, Tizhou Qiao, Shuling Dai, Real-time Simulation of Sky and Sun in Clear Day, Journal of Computer-Aided Design&Computer Graphics, 2009, 03:305-310.

[5] Green S，Cebenoyan C．High dynamic range rendering on the GeForce6800．

[6] http://http.download.nvidia.com/developer/presentations/2004/6800_Leagues/6800_Leagues_HDR.pdf

[7] Tumblin J,Hodgins J K,Guenter B K.Two methods for display of high contrast images.ACM Trans.Graph,1999,18:56-94.

[8] Pattanaik S N,James A F,Mark D,et al.A multiscale model of adaptation and spatial vision for realistic image display(Proceedings of SIGGRAPH'98).Orlando:[s,n.],1998.

[9] Reinhard E，Stark M，Shirley P．et a1. Photographic tone reproduction for digital images.ACM Trannsactiona on Graphics，2002，21(3)：267-276.

[10] Krawczyk G，Myszkowski K，Seidel H P．Perceptual effects in real-time tone mapping(Proceedings of the 21st Spring Conference on Computer Graphics), Budmerice, 2005.

[11] James G, O'Rorke J.Real-time glow. Fernando R．GPUGems，Programming Techniques．Tips and Tricks for Real-Time Graphics. Boston:Addison—Wesley, 2004:343-362.

[12] Fabio Policarpo, Francisco Fonseca. Deferred Shading Tutorial.

[13] http://www710.univ-lyon1.fr/~jciehl/Public/educ/GAMA/ 2007/Deferred_Shading_Tutorial_SBGAMES2005.pdf