## **Paper Analysis Project**

Delyar Tabatabai - CS557

**Deferred Shading**

Shawn Hargreaves

Mark Harris

2005

1. **What is the general theme of the paper you read? What does the title mean? What are they trying to do? Why are they trying to do it? (I.e., what problem are they trying to solve?)**

In the “Deferred Shading” paper, the author is trying to address the challenge of real time lighting mostly in the gaming development. In the games, we usually see a lot of objects and many light sources. So it is computationally expensive to calculate the shading throughout the whole game scene.

So in summary, the author is trying to propose and apply the lighting as a 2D postprocess, using buffers as input. Deferred Shading addresses this challenge by decoupling the lighting calculations from the geometry and material calculations, and performing them in a separate pass.

The title of this paper, “Deferred shading” means allowing for a large number of lights to be efficiently rendered, as the lighting calculations only need to be performed once per pixel rather than once per light. Furthermore, Deferred Shading also allows for more advanced rendering effects such as transparency, reflections, and screen-space effects to be handled more efficiently and with greater flexibility.

All in all, the author is trying to mention how the complexity of “Deferred shading” is lower than Single Pass Lighting and Multipass Lighting.

In single-pass lighting, we have one for loop over the objects in the scene and we render a mesh and apply all lights in one shader per each object. But it has some limitations when it comes to having many light sources.

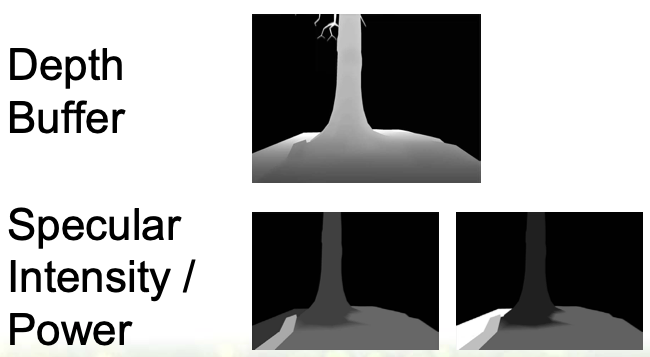
On the other hand, in Multipass lighting, we have nested for loops. In the outer for loop, we are going over each light source and for each light source we have another for loop which loops over the objects affected by the light. This means that the algorithm is of O(n^2) which is not optimum and we will have a lot of CPU work.

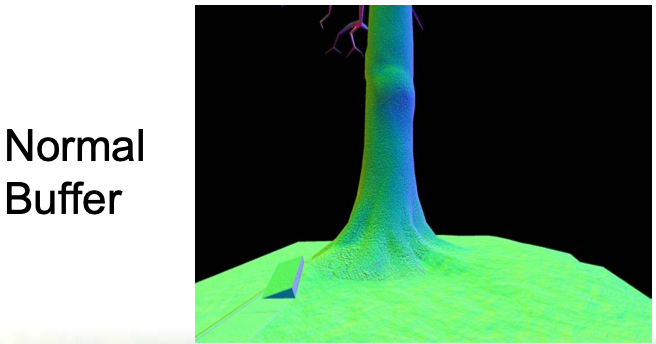
However, by introducing “Deferred Shading”, we have two separate for loops in the algorithm which is more efficient. One looping on the objects, and the other one looping on the lights.

1. **You *can* copy and paste images or graphs from the paper into your paper, but not as a way to have to write less to hit the 5-page requirement.**

In deferred shading, different types of information about each pixel in the scene are stored in multiple buffers, known as the G-buffer. Each G-buffer contains a different type of information, such as the world space position, normal, albedo, and specular components of each pixel in the scene.

Here are some pictures from the paper showing how different information about the scene is stored in different buffers.







Then the author introduces “fat framebuffers”. In deferred shading, a framebuffer is a data structure that holds the color and other information for each pixel in the scene. Framebuffers contain multiple channels or textures, such as G-buffers. By using fat framebuffers, we will be able to allocate enough memory to store all of the necessary G-buffer information without requiring any additional texture lookups or other computational overhead.

The use of G-buffers in deferred shading allows for more efficient lighting calculations, as the lighting calculations can be performed on a per-pixel basis without the need for multiple rendering passes or other computationally expensive techniques.

1. **Who are the authors? Where are they from? If you can tell, what positions do they hold? Can you find out something about their backgrounds?**
2. **What did the authors do?**

The authors of this paper are Shawn Hargreaves and Mark Harris. At the time the paper was written, Shawn Hargreaves was working as a Lead programmer on the racing games MotoGP and MotoGP2 at *Climax* racing and *Microsoft*. This paper was introduced in a GameDevelopers Conference. The interesting thing about him is that he has a BA in Music but pursues computer graphics as his main career. He is a software engineer and video game developer now at Microsoft for 17 years. He is famous for the frameworks such as XNA that he developed for video game development. Shawn Hargreaves has authored multiple books on game development, such as "Programming Role-Playing Games with DirectX" and "XNA Game Studio 2.0 Programming".

The other author of this article is Mark Harris. Mark Harris is a computer graphics researcher and software developer. He is the Chief Technologist in the Professional Visualization division at NVIDIA and holds a Ph.D. in Computer Science from the University of North Carolina at Chapel Hill. Harris has contributed significantly to graphics hardware and software in academia and industry, with a focus on parallel computing, real-time rendering, and high-performance visualization.

1. **What conclusions did the paper draw?**

All in all, the paper “Deferred Shading” is drawing several important conclusions about the benefits of using the deferred shading method in computer graphics instead of other methods.

Deferred shading is very efficient and can be used in the game scenes or any other graphics scenes which are very complicated and have a large number of light sources and objects and textures.

On the other hand, by using the fat framebuffers, we can store a variety of data types, such as normal vectors, secular coefficients and depth information in separate buffers that can be combined in different ways to make complex lighting effects.

Another conclusion that can be drawn, is that deferred shading can be used in modern computer hardwares. By using the parallel processing capabilities of GPUs, it is possible to achieve high levels of performance and scalability when rendering complex scenes with large numbers of lights. This makes deferred shading a highly practical technique that can be used to create visually stunning games and applications.

1. **What insights did you get from the paper that you didn't already know?**

I personally have never thought about how complex the process of shading can be and each object being added to the scene can affect a lot of the lighings.

I think the most important message of this paper is the importance of using efficient techniques for rendering complex scenes in real-time applications such as games and interactive graphic applications.

It is also very interesting how important the collaboration of industry and academia is. Study of algorithms and their time complexities are very important in the gaming and computer graphics industry.

This paper made me eager to learn more about the role theoretical computer science plays in computer graphics.

1. **Did you see any flaws or short-sightedness in the paper's methods or conclusions? (It's OK if you didn't.)**

This paper seems pretty famous in the field of computer graphics and widely cited in the realm of real-time rendering. I do not have enough information to criticize the paper, but I think it would have been interesting if they could prepare a table and compare the rendering time of the algorithm in different machines and different light sources and object numbers.

1. **If you were these researchers, what would you do next in this line of research?**

If I was going to continue this line of research, I would have been eager to know if there is more room to make this algorithm more efficient and would have collaborated with theoretical computer scientists. I think this approach would be very efficient in terms of time complexity but it would be interesting to study it from memory aspects too.

One interesting area that can be worked on in this area is when we have lighting that is not fixed. For example if our light sources keep moving, we have to calculate the shading per each movement and this might not be efficient and ruins the efficiency of the “Deferred shading” method. So I think it is interesting to study this situation too.