Re: Deep G-Buffers for Stable Global Illumination Approximation

Ferit Tohidi Far

January 16, 2019

Abstract

G-buffers can be used to efficiently render images with large amounts of light sources. This is possible thanks to a process called "deferred rendering". Using g-buffers, we are only able to compute local illumination. By using deep g-buffers instead we can approximate global illumination, which is way more efficient than traditional global illumination methods like pathtracing, while of course not being physically accurate.

Keywords *g-buffer, deep g-buffer, pathtracing, global illumination, deferred shading, deferred rendering*

Contents

1 Global Illumination

Global Illumination is a lighting effect that is achieved by not only computing direct light, but also indirect light, meaning that it is neccesary to take into account how light reflects and carries information (in the most basic case: color).

1.1 Physically correct methods

In order to generate physically correct images, which is a requirement for creating photorealistic images, we need to solve the rendering equation

$$L_o(\omega) = L_e(\omega) + \int_\Omega f(\omega,\omega')L_i(\omega')cos(n,\omega')\delta\omega'$$

where

$L_o(\omega)$ is the outgoing light in direction ω,
$L_e(\omega)$ is the emmited light in direction ω,
$f(\omega,\omega')$ is the BRDF[1],
$L_i(\omega')$ is the incoming light from direction ω'
and $cos(n,\omega')$ is lambert reflectance[2] .

The most popular method for achieving this is pathtracing [PHC14].

1.1.1 Pathtracing

Pathtracing solves the rendering equation by

1.2 Computational difficulties of physically correct methods

Since we have to take into account every ray of light and its reflections, the computational difficulty becomes apparent [FP].

2 Deferred Rendering

2.1 How deferred rendering handles lighting more efficiently

The goal of deferred rendering is to postpone the shading stage. Instead of shading right away, we compute necessary geometry buffers (g-buffers) and cache them for later use. For all practical purposes, g-buffers have to at least consist of a frame-buffer, a normal-buffer and a z-buffer. Using only these g-buffers, we can now compute the shading separately.

2.2 Deferred Shading

3 Geometry-buffer (g-buffer)

Each geometry-buffer stores information of some sort for each individual pixel, meaning that they are all two-dimensional arrays.

3.1 Frame-Buffer

The frame-buffer stores color values.

3.2 Z-Buffer

The z-buffer stores depth values.

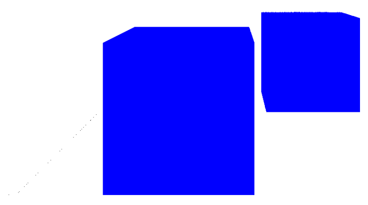

Figure 1: Example of a frame-buffer (also called image-buffer)

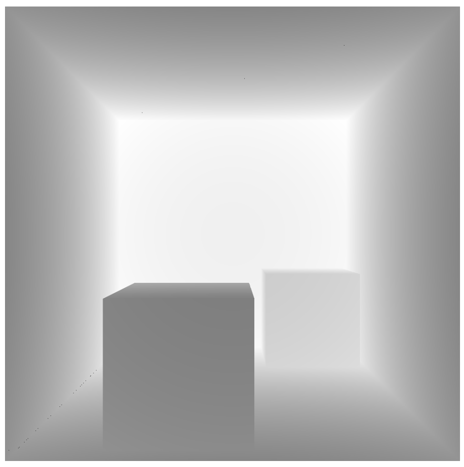

Figure 2: Example of a z-buffer

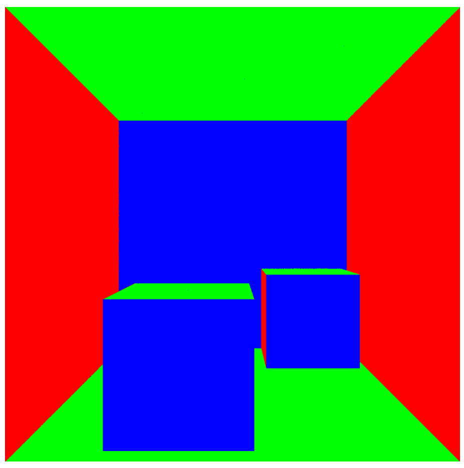

Figure 3: Example of a normal-buffer

3.3 Normal-Buffer

The normal-buffer stores surface-normals. Note that there are more possible buffers to choose from, but the three that were mentioned are the most essential in every g-buffer.

3.4 Computing local illumination using g-buffers

After collecting all g-buffers we can now work on illumination. To do this we need to define some light sources. We distinguish between three types of lights: Point-lights, spot-lights and directional-lights [FP].

4 Deep G-Buffer

4.1 Concept

Store multiple g-buffers [MM16].

4.2 How Deep G-Buffers improve performance

5 Visual effects

The following are visual effects that are sought after, but hard/impossible to achieve physically correctly without using computationally expensive methods like pathtracing.

5.1 Ambient occlusion

Ambient occlusion essentially describes how much shading the ”in-betweens” of a 3d object gets. This effect can be efficiently approximated by using a method called - ironically - screen space ambient occlusion (SSAO). This method basically runs an edge detector over the z-buffer and paints those edges black. Since it only runs over the z-buffer it is considered screen space.

5.2 Color bleeding

Color bleeding

5.3 Soft shadows

6 Performance and Output Comparison

6.1 Deep G-Buffers vs Pathtracing

References

[FP] Francisco Fonseca Fabio Policarpo. “Deferred Shading Tutorial”. In: ().

[MM16] D. Nowrouzezahrai D. Luebke M. Mara M. McGuire. “Deep G-Buffers for Stable Global Illumination Approximation”. In: *High Performance Graphics* (2016).

[PHC14] Wojciech Jarosz Per H. Christensen. “The Path to Path-Traced Movies”. In: *Foundations and Trends R in Computer Graphics and Vision* 10 (2014).