Bachelor Project: Interactive Global Illumination
Project Plan and Description

Jonas H. Nielsen, s121751

Supervisors

Niels Jørgen Christensen

Jeppe Revall Frisvad

DTU Compute
Technical University of Denmark

1 Project Description

I wish to implement, document, analyse and discuss a reasonably fast screen-space method for global illumination. The base of this will be formed by the use of a deep G-buffer with minimal separation between layers as described in the 2014 paper "Fast Global Illumination Approximations on Deep G-Buffers." While GI encompasses a multitude of methods, my primary focus will be on multi-bounce diffuse-diffuse interreflectance. While physical correctness (i.e. of energy/light transfer) will be part of solving this problem, and at the very least will be discussed in the paper, the main goal will be to produce effects that look right, without necessarily being entirely correct. That is to say, the implementation will be a compromise between physical correctness and lowering the time to draw a frame.

Specifically and in order, the goals of the project are to 1) Make a very simple framework to work and perform tests in (using GL, GLEW, GLFW, GLM and AssImp,) 2) Implement a deep G-buffers rendering method 3) Implement a diffuse-diffuse interreflectance approximation using deep G-buffers. Ideally I would reach acceptable real-time drawing times of (20ms or less) on my most powerful testing platform (1.8 TFlOPS) for acceptable quality screen-space radiosity, but I would consider 50ms and less (20fps or more) an acceptable result. If there's still development time left after reaching these goals it would be used to implement other screen-space GI methods like SSAO and dealing with some of the limitations inherent to screen-space radiosity. For instance, if a brightly coloured diffuse object escapes the viewing frustrum, it could create a disruption in quality, since the colour of the scene could change drastically. A way to fix this could be to combine the screen-space method with a global one (such as virtual point lights.) Again, this only becomes relevant if there's time to spare for development, but it would be a nice addition to the project.

2 Project Plan

Period	Task
July 6th - July 20th	Framework Development
July 20th - July 31st	Implementing deep G-buffers
August 1st - August 31st	*No ECTS points*
September 1st - September 5th	Continue work on deep G-buffer implementation
September 5th - September 12th	Write Design and Implementation chapters on the Deep G-buffers implementation
September 12th - September 26th	Implement screen-space radiosity method using Deep G-Buffers
September 27th - October 4th	Write Design and Implementation chapters on the radiosity method and perform tests on performance
October 4th - October 31st	Buffer period. If either of the two other implementations are not done, I finish them here, if they are, this period can be used for the following: • SSAO • Occlusion testing for the radiosity method • Adding normal mapping to the G-Buffer generation • Optimisations • An AA method taking advantage of the deep G-Buffer • More in-depth testing
November 1st - December 5th	Writing remaining part of the report including introduction, results, discussion, conclusion and abstract. If parts of design or implementation are still unfinished these are finished as well.
December 6th	Delivery of project and report.