

Tutorial 12: A shadow mapping technique in OSG

Franklin Foping
`franklin@netcourrier.com`

May 11, 2008

Abstract

Shadows allow users to have a deep understanding of the spatial relationships between different objects making a given scene. There are several algorithms to compute shadow. In this tutorial, we are going to explain how to achieve a basic shadow mapping in OSG. This tutorial will also show another importance of the material object defined in the second tutorial and highlight the difference between directional and point light sources.

1 Shadow in OSG

There are many shadow algorithms shipped with the latest version of OSG (v 2.4): shadow mapping, shadow textures, volume shadowing and so forth. However, some of them are still under development. From my personal experience, I found the shadow mapping technique quite useful and very well written. Here is how to proceed to add shadow to your scene.

1. Create a Shadow Scene Node and set its shadow technique (lines 37-39).
2. Create a Root Node for your world as usual and add it to the Shadow Scene.

3. Define and add a light source object to the Shadow Scene Node (lines 42-47).

2 Materials revisited

Ok, you already know from the second tutorial that Material objects allow you to change the visual appearance of an object. You can setup the shininess of objects within this class. When OSG computes shadow maps, there are still some shadow alias. You cannot get rid of them but you can easily do is to hide them using the Material class. Go to the source code and understand what I achieve in lines 55-60. If you comment these lines, you will be able to see the aliases I was talking about.

3 Scene graph of our scene

The scene graph of our program is shown at the picture 3.

4 Results

The results of our scene is illustrated in the sketch 4.

5 Do-it-yourself

Here is your task list:

1. Comment the line 60, something weird will happen to your scene, what is that? What is the way around it?
2. Recall from tutorial 9, I briefly talk about directional and point light sources. Well it is time to know more. In this program, I have used point light, that means that the light source is close to objects. In order to visualize the difference go to the line 44 and replace the 1 by 0. You will now notice that the shape of the shadow **has changed**.

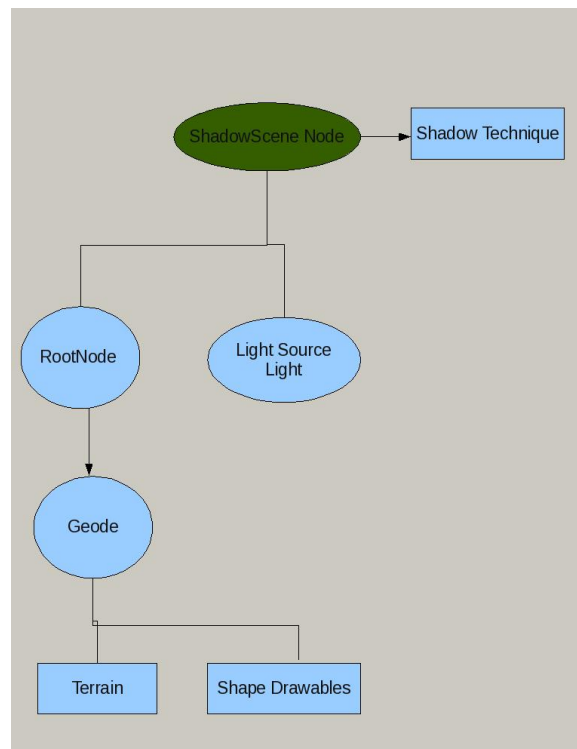


Figure 1: The scene graph

The shadow is now located at the bottom of the object. It acts like the light is miles away.

3. The shadow technique used here was a simple shadow mapping. OSG comes with some examples where different techniques are implemented, for instance shadow textures. Study them and implement them to this code. You should look at the following examples from OSG: **osgdepthshadow**, **osgshadow**, **osgshadowtexture** all provide with the latest OSG release.

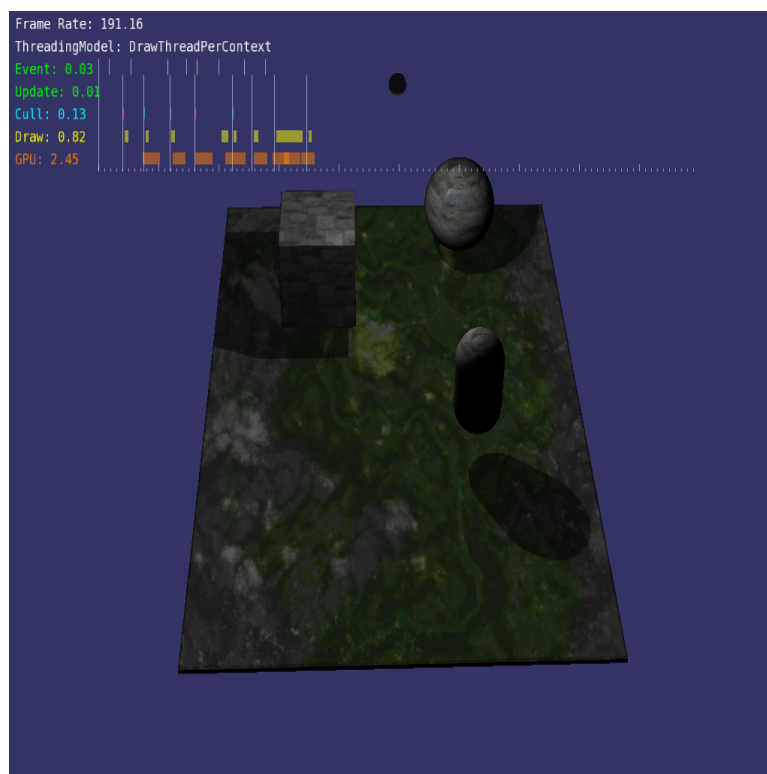


Figure 2: A scene enhanced with shadows