

Advanced Effect: Shadow Mapping

Bojan Cvjetkovic, Johannes Ebster, Martin Hartl

26. Juni 2015

1 Report:

The Advanced Effect we tried to implement is Shadow Mapping. We made use of the hints and explanations from the proseminar lecture about Advanced Effects to do this. The first important thing to consider was that we had to do rendering in 2 different rendering passes.

To get our shadow map texture for the actual scene rendering we had to render the scene not from the camera perspective but from the light source using the depth buffer. First we had to setup the handles for the depth buffer and bind it. After the setup is done we can start with the first rendering pass. We define projection and view matrices for our light and multiply these with our model matrix. We also created different shaders for the first rendering pass. After setting these to the right values we hand the matrices to the vertexshader. Then we render our scene once with the depth buffer and store it in our texture. Now that we have our texture we can start with the second rendering pass.

In the second rendering pass we use the texture we created in the first rendering pass to compute our color values. We create our shadow matrix with which we can compute the vertex coordinates in our texture and hand it to the vertexshader. Then we bind our texture and draw our objects again. Now our fragmentshader should do the following: With the coordinates we computed with the shadow matrix and the texture we got from the first rendering pass a GLSL function will do the tough part. It compares the depth of our current scene and the values in the texture and returns a floating point value. All we have to do now is to multiply our color with that value.

Unfortunately we faced some troubles. It took us a while to find and fix

some bugs and to bind all uniform matrices, as some of those made problems. In the end our program behaved differently on all the team members computers, on one it had problems with the shader and the samplers(it said both samplers tried to access the same texture unit 0) and on another the whole scene was covered in a shadow. We tried to fix those problems for hours but were not able to track the error down.