

Assignment 6: Rendering to a Texture

In this assignment, you will use Framebuffer Object (FBO) to redirect the rendering output to a texture and later, apply the texture generated on the fly onto the surface of a cube. When attaching a texture to a framebuffer, all rendering commands will be written to the texture as if it was a normal color/depth or stencil buffer. The advantage of using textures is that the result of all rendering operations will be stored as a texture image that we can then easily use in our shaders. Any change in the output because of transformation will also be readily reflected on to the texture, thus making the rendering more dynamic and flexible.

As you can see from the figure below, the cube has been generated by rendering the bunny to an internal texture and later, the texture has been applied to the faces of the cube using as usual texture mapping. As the transformation (here rotation) being applied to the bunny, that's also readily reflected onto the rendering output as shown to the right of Figure 1.

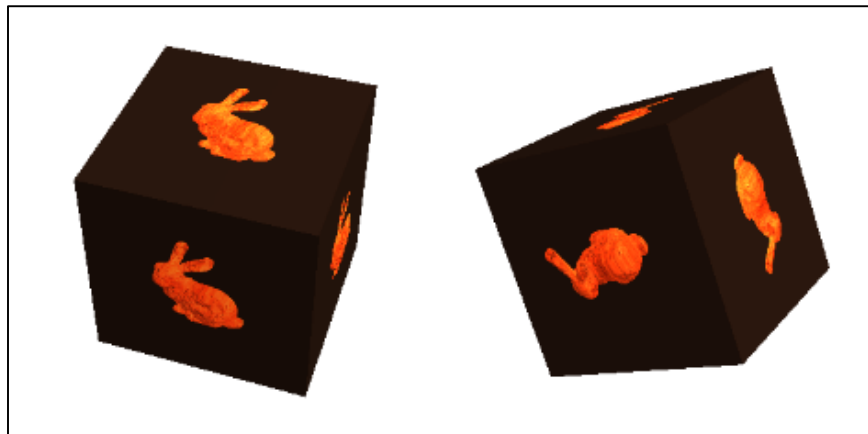


Figure 1: Rendering to a texture.

You need to create a separate rendering target buffer, i.e. the Frame Buffer Object, attach texture to the FBO, and next render to the FBO in exactly the same way as we

render to the default frame buffer. You need to swap in to the FBO when rendering to it and swap out when rendering to default frame buffer.

You need to be careful to set up the viewport and the view and projection matrices appropriately for FBO. As done with default frame buffer, similar changes should be made to the view and projection matrices to match the aspect ratio of the viewport and set up the scene to be rendered to the FBO.

Submission:

Submit the assignment in a zipped file via canvas. Name the file as Firstname_Lastname_6_CSCD471.zip. Deadline is **Wednesday, March 2, 11:59 pm**.

This assignment carries a weightage of **12.5%** of this course.