UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

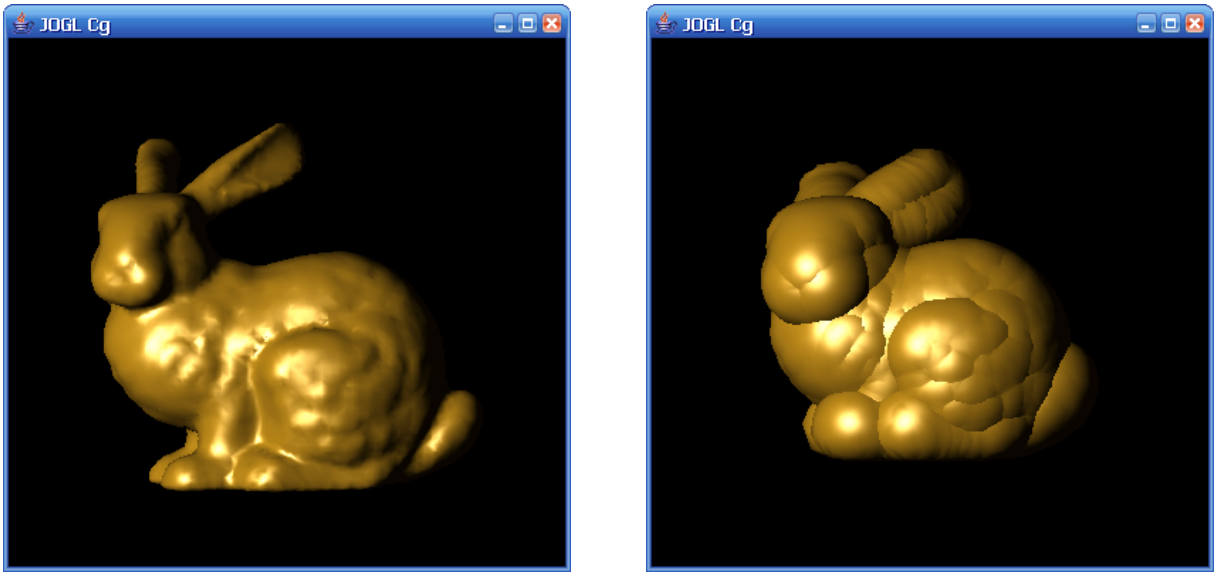
Advanced Rendering Assignment sheet 2 – ST 2013
Prof. Dr. Gitta Domik, Stephan Arens
In Lab on April 22 to 26 – Homework due to May 1, 9:15
(Send homework to stephan.arens@upb.de)

GPU Programming

Assignment 6

Download the Shading_Template.zip. Rewrite the vertex shader program of the Phong per pixel example `vp_phongPerPixel.cg` to change the size of the objects in your OpenGL scene. Therefore, move the vertices along their normals. Use a uniform parameter to change the size over the time or interactively.

- **Homework:** Send 1 screenshot similar to the right image underneath.



Assignment 7

Now change the position of the light source in the fragment program to let the scene always been illuminated from the actual camera position.

Assignment 8

1. Add texturing to the Phong per pixel shader program and test it on any textured model.
2. Extend the Phong per pixel lighting for two light sources.

- **Homework:** Send 1 screenshot containing textures and two light sources.

Assignment 9 (team work for 2 weeks)

Use depth sorting and alpha blending to make objects in your project transparent. First, create bounding boxes for all of the transparent objects in your scenery. This can be done by computing the minimum and maximum x,y and z positions of all vertices while the mesh is loaded. Store all *transparent* objects **in a separate data structure** that can easily be sorted (e.g. use a `Vector` and let each object implement `Comparable`, then use `java.util.Collections.sort`).

Alpha blending with depth sorting works as follows:

1. Draw all **opaque** objects of the scene.
2. Disable backface culling.
3. Enable alpha blending using `gl.glBlendFunc(GL.GL_SRC_ALPHA, GL.GL_ONE_MINUS_SRC_ALPHA)` as blend function.
4. Disable **writing** to z-buffer (`gl.glDepthMask(false)`).
5. **Sort** all transparent objects in descending order by their distance to the camera. Therefore, calculate the distance between the midpoint of the bounding boxes transformed by the modelview matrix and the camera's position.
6. Draw all **transparent** objects in order from back to front.
7. Enable writing to z-buffer and backface culling, disable blending.