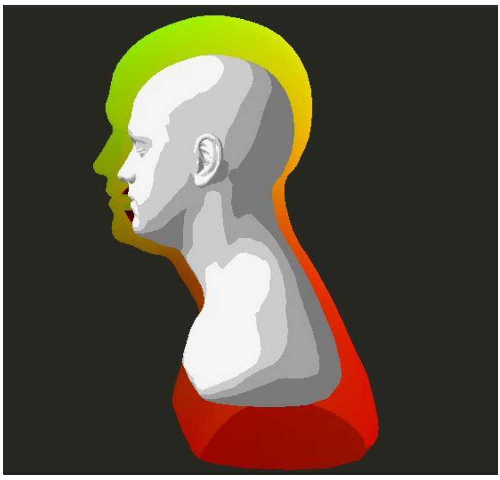
Cel Shading

2013314587
독어독문학과
AnHyeonJang, 안현장

1. Objective
   1) By modifying 'diffuse' in phong shading, implement the cel shading
2. Data Structure & algorithm
   1) Data Structure & algorithm is same as the original shading example
   2) In the fragment shader GLSL, make the two function, *stepdiv and cel*
   3) *stepdiv* is the function to quantize the contiguous diffuse intensity. It makes flooring the specific range of intensity.
   4) *cel* is the modified version of the 'diffuse' in phong shading
      i. $\mathbf{I}rd = \max(\, stepdiv(\mathbf{l} \cdot \mathbf{n}, \text{dividing value})\ \mathbf{k}d\ \mathbf{I}d\ , 0\ )$
   5) The quantization is accept through the keyboard input (+, -), and it is transferred to the argument in *cel* function in by glUniform1ui.
3. Advanced features – Object Outlining
   1) reference - https://learnopengl.com/Advanced-OpenGL/Stencil-testing
   2) Through using the stencil buffer in OpenGL, I made the outline of the object.
   3) It needs two render sequence – object, outline.
   4) In function *user_init()*, through calling glEnable(GL_STENCIL_TEST), enable to use the stencil buffer and assign stencil operation to use glStencilOp(GL_KEEP, GL_KEEP, GL_REPLACE)
   5) In function *render(),* through calling glClear(GL_STENCIL_BUFFER_BIT), glStencilbuffer(GL_ALWAYS, 0xFF), enable stencil writing and writes the fragments into stencil buffer (written stencil value is one)
   6) In new render function *stencil_render()*, disable depth testing and stencil writing first. Then through using the existing vertex shader and the new fragment shader *stencil.frag*, draw the bigger object scaled by a little amount
   7) The new object is drawn where the stencil value is not equal to 1
   8) The bigger scale model seems like outline.
4. Discussion



   1) As we can see the picture, the outline is not perfectly correct(To see this result, I oversized the outlining scale). Some part of the outline is thinner than others, especially the shoulder of the model. This is because the scaling is adapted not equal to every part of the model.
   2) I think this methods becomes worse when the model has more curves, the outline must be thicker, and the model becomes more complicate.
5. More Advanced feature – thicker wireframe(linewidth)
   1) reference - https://www.flipcode.com/archives/Object_Outlining.shtml
   2) Because scaling model method makes the distortion in outlining, I search other methods.

3) It also uses stencil buffer and writing, almost same as the scaling model method

4) However it does not draw the bigger scale object, just draw thicker line through using glLineWidth(10) with glPolygon(GL_FRONT_AND_BACK, GL_LINE).

5) But It also has problem that we can see more crack and glLineWidth affects to the line width of the original object's wireframe.