Computer Graphics

by Ruen-Rone Lee

ICL/ITRI

Assignment #2

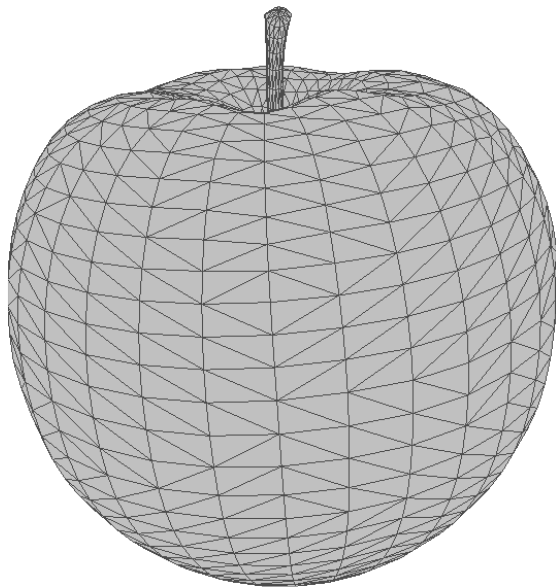Per Vertex Lighting

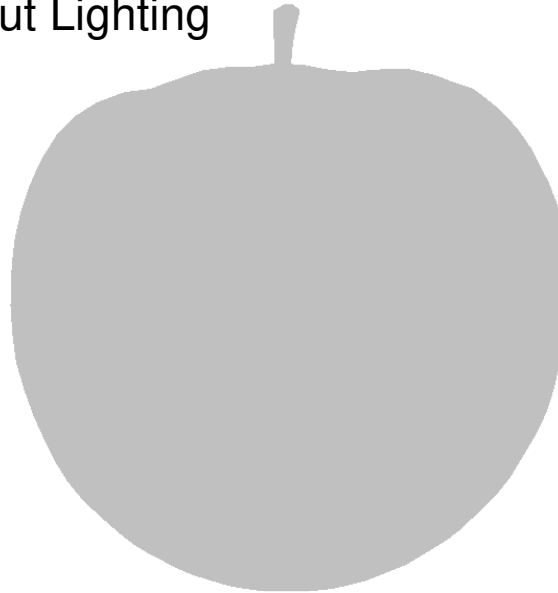Per Pixel Lighting

Lighting
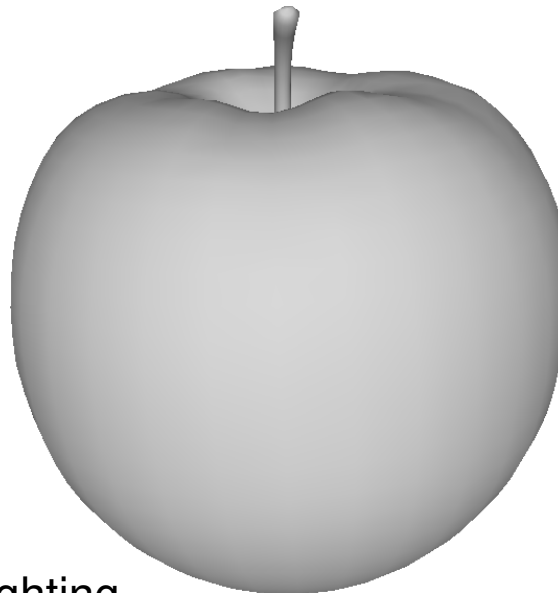
Lighting 3D Models

Without Lighting

Original 3D Model
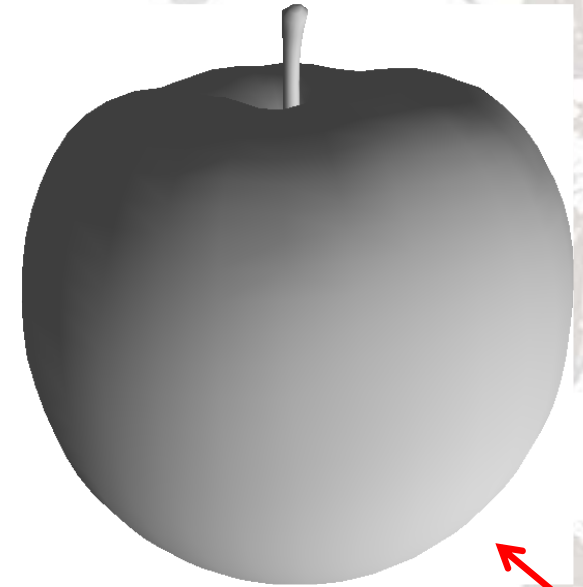
light source

light source

With Lighting

3

Requirement

◆ **You are required to write a program that can accept 3D test models as in assignment #1 and render the 3D models with smooth shading (without ground floor polygon)**

◆ **The models should be rendered with given light sources**

◆ **Three different light sources, directional light, positional light, and spot light, should be implemented.**

Requirement
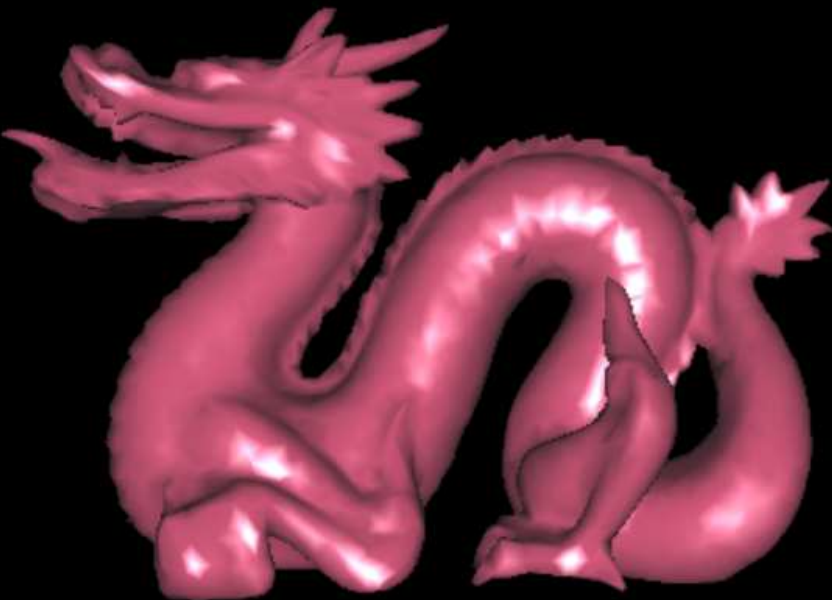
- **Light source position should be able to change manually**
  - Please check TA's instruction in how to control the light source
- **Vertex lighting is required**
  - All the lighting calculations are performed in vertex shader.
- **Per pixel lighting is also required**
  - All the lighting calculations are preformed in fragment shader.

Requirement

◆ **Display different lighting model (per-vertex and per-pixel) with different light source types (directional, positional, spot) side-by-side for comparing the rendering effects**

Per-Vertex Lighting / Point Light   Per-Pixel Lighting / Point Light

Requirement

- **Use keyboard and mouse to control the objects transformation as implemented in assignment #1**

- **Use a key to switch between different lights source types for both per-vertex lighting and per-pixel lighting simultaneously**
  - **Show the current light source type on console window**

- **Display help file for how to control the actions of your program as in assignment #1**

Input Model Format

- ◆ **Wavefront 3D Graphics model description file with extension .obj and material file .mtl**
  - ■ **Material file will have the material name follow by newmtl and the ambient, diffuse, and specular coefficients of Ka, Kd, and Ks, respectively**
  - ■ **Please ignore the other material values and set the specular exponent properly by yourself**
- ◆ **Model file contains not only the vertex positions but also the vertex normals**

Hints

◆ **Normal transformation is necessary to derive correct lighting result**

◆ **Normalization to the transformed normals is also necessary to obtain correct lighting result**

◆ **Per pixel lighting can be achieved by passing the transformed vertex normals to rasterizer for generating per-pixel normals and then do the lighting calculations in fragment shader**

   ▪ **E.g., replace the vertex colors in assignment #1 by vertex normals**

Hints

- ## **Draw two models side-by-side**
  - ### ■ **Set the viewport before the draw command**

w

glViewport(0, 0, w/2, h)   glViewport(w/2, 0, w/2, h)

h
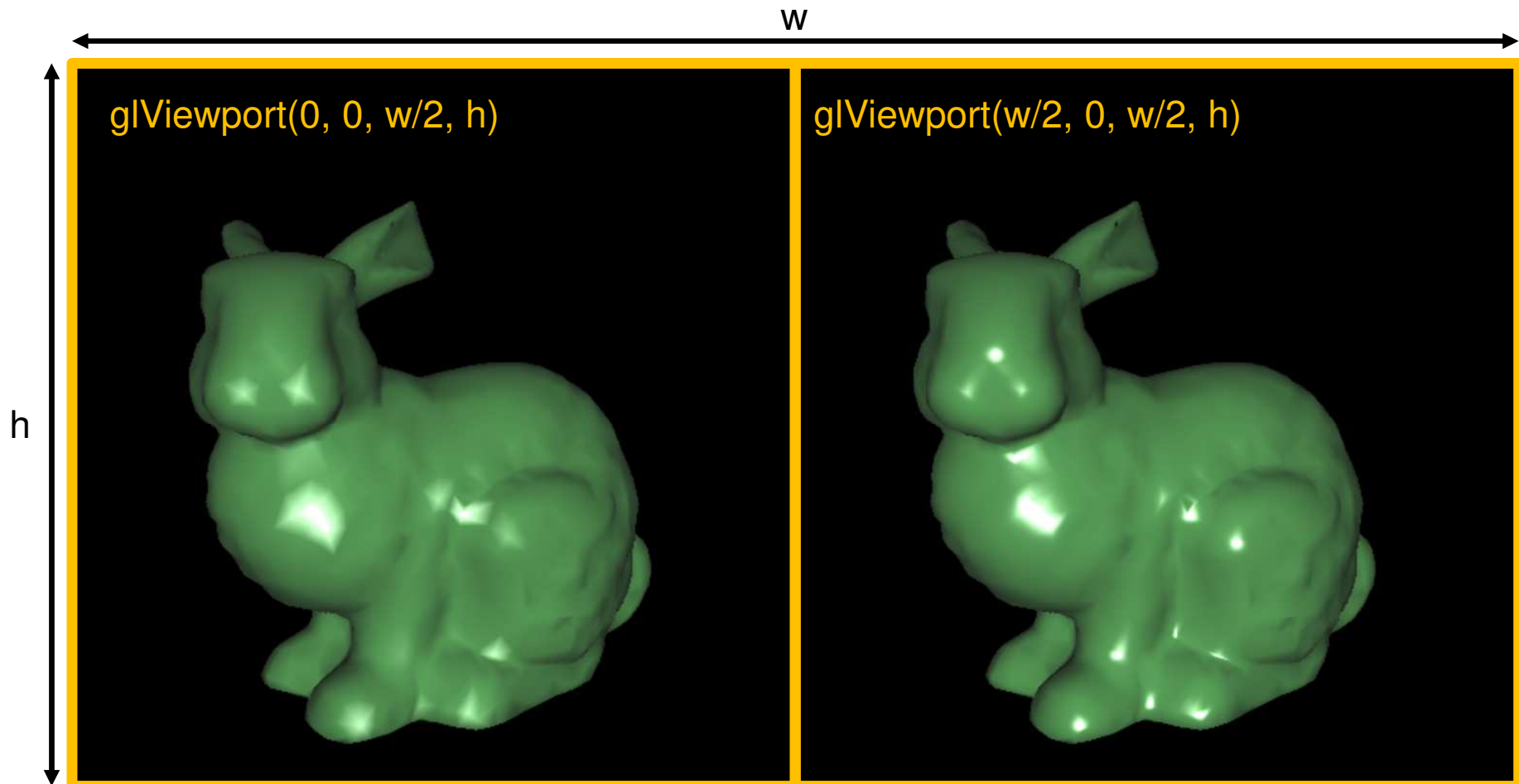
Hints

- ◆ **Some 3D models with vertex normals are provided for verifying your design during program development.**

- ◆ **Please download the test models from iLMS**

Due Date

- **2.5 weeks after the due date of first assignment (should be 5/23)**
- **Late submission is allowed with less score**
- **No score if you did not submit you assignment**
- **Plagiary is strictly forbidden**
  - **If you copy from others, your score will become zero**
  - **The score to the one who provide the original copy will also be downgraded**

Submission Guide

- **Please submit your works per TA's instructions**
  - *Notice: E-mail submission will not be accepted*

- **Submission should include**
  - **Source codes (including solution and project files)**
  - **Executable binary (can be run on PC/windows)**
  - **Documentation (explain how you did it and how to operate it)**
  - *Notice: please do not submit any 3D models to save the disk space (i.e., you have to use relative path instead of absolute path)*

- **Contact with TAs if you have problem in submission**

Q&A