

Assignment 2: Skinny Chubby Bunny

Objective:

In this assignment, you will draw a complex 3D model onto the screen; then apply lighting and shading. Next, the model will be deformed to give different looks.

Specification:

You will implement a sine wave update of a bunny model with time using **glBufferSubData** as discussed in the class. Basic sine wave as a function of time can be described as follows:

$$F(t) = A * \sin(2\pi f t + \phi) + D ;$$

where

‘**A**’ is the amplitude of the wave, which represents the highest and lowest points of the wave (consider $A = 0.06$ in this assignment);

‘**f**’ is the frequency, which controls the number of times the wave repeats in unit time (consider $f = 1.0$);

‘**t**’ represents time (that is updated by ‘**updateTime()**’function’) and

‘**φ**’ represents phase *i.e.*, the initial angle of a sinusoidal function at its origin (consider $\phi = 0.0$);

‘**D**’ represents the vertical shift in the sine wave function (0.05).

From the sine wave function definition, it is clear that the vertices of the bunny model will be periodically updated. The update will be performed along the vertex normal with a magnitude of $F(t)$ (i.e, $\mathbf{v}_i(t) = \mathbf{v}_i + F(t) * \mathbf{n}_i$). When ‘ $F(t)$ ’ is positive, the bunny will gradually grow fatter whereas with negative $F(t)$, the bunny will shrink.

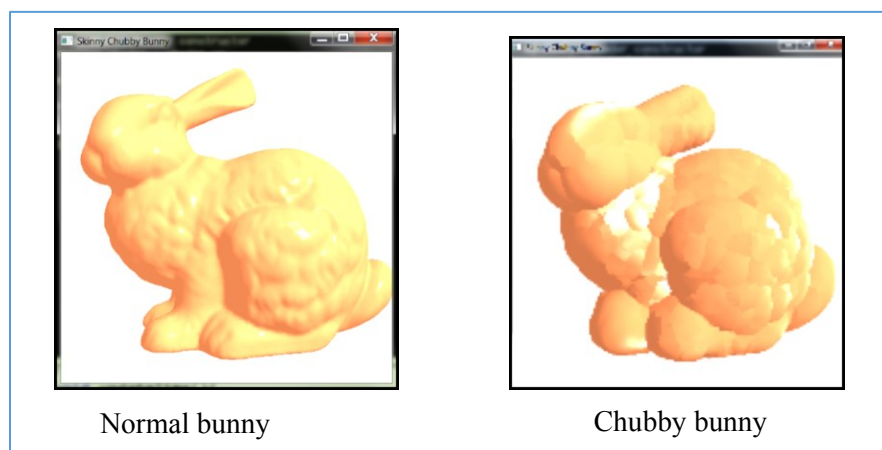


Figure 1: Bunny model as a function of sine wave.

Implementation:

1. Download the zipped file (**Assignment2.zip**). When you compile and run the uploaded file, it shows a bunny model not yet lit or shaded as shown to the left of Figure 2. The bunny model has been uploaded from an obj file named “**bunny.obj**”. You need to complete the following function in **objloader.cpp** file that will compute the vertex normals. Once you complete **computeNormals** function, the bunny will look shaded as demonstrated to the right of Figure 2.

```
void computeNormals(std::vector<glm::vec3> const &vertices, .....);
```

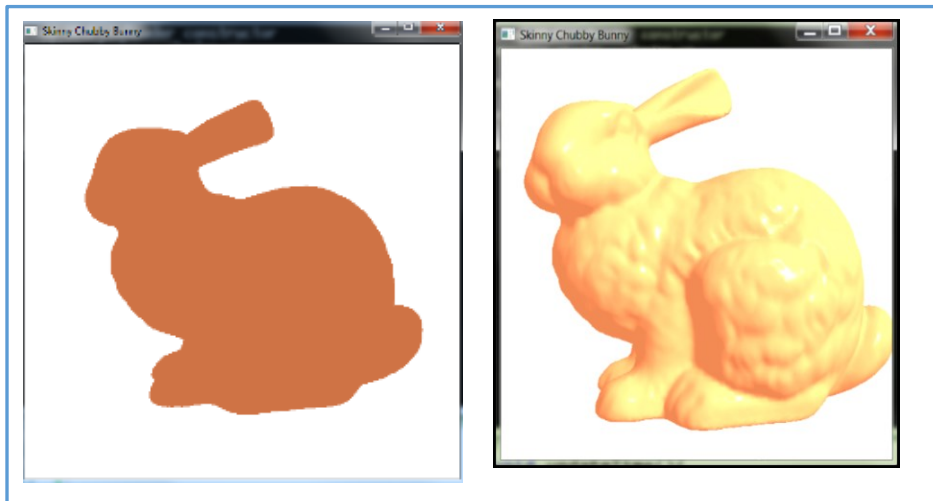


Figure 2: A bunny model (a) not yet lit or shaded and (b) lit and shaded.

2. In the commented part in ‘**Display()**’ function, you need to use either **glBufferSubData** so that vertex update is reflected at each time interval.
3. You need to complete the following two functions in **main.cpp**:
 - a. `void updateBunnyVertex(vector<vec3> &ptr, vector<vec3> verts, vector<vec3> &norms, int count){}`
 - b. `void updateBunnyNormal(vector<vec3> &vNormals, vector<vec3> vertices, vector<int> indices){}`
4. You need to use **glutTimerFunc()** to update the bunny model at a regular interval of 50 milliseconds;.
5. You do not need to do any modification in the shader files.
6. Though some part of the code has been written in C++, you do not need to worry about that for this assignment;

Submission:

Submit the assignment in a zipped file via canvas. Name the file as

Firstname_Lastname_2_CSCD470.zip. Submission deadline is **Thursday, May 3, 11:59 pm**.

This assignment carries a weightage of 10% of this course.