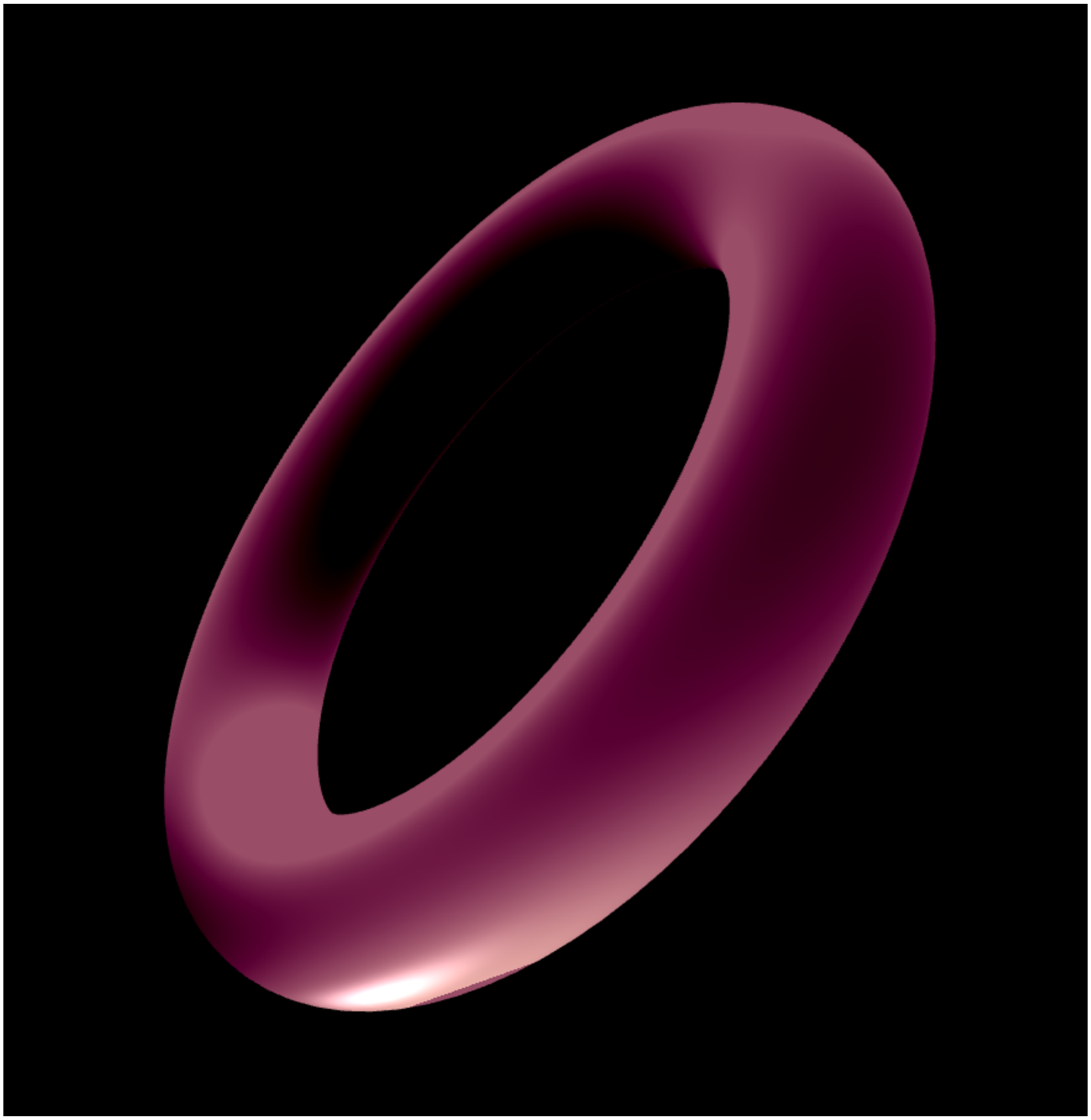
CSE333 – Assignment 3 – Report

Atharv Goel
2021027
atharv21027@iiitd.ac.in
15-Oct-2023

1. Introduction

1.1 Objective

This report concerns the third assignment in the *CSE333: Introduction to Computer Graphics* offered at IIIT Delhi. The primary objective is to render a parameterized object and do a bunch of cool lighting and shading effects.



2. Implementation Details

2.1 – Generating a torus

I wrote a function `createTorusObject()` which generates the vertex data for a torus and writes the data to a VBO, which is then bound to the appropriate VAO. This function uses parametric coordinates to generate the shape:

$$x = (R + r.cos(\phi))\,cos(\theta)$$
$$y = (R + r.cos(\phi))\,sin(\theta)$$
$$z = r.sin(\phi)$$

The normals are computed by calculating the normalized cross-product of the tangent along either axis in the parametric space, resulting in:

$$Nx = cos(\phi)cos(\theta)$$
$$Ny = cos(\phi)sin(\theta)$$
$$Nz = sin(\phi)$$

In the above equations, R represents the outer radius, and r is the inner tube radius. θ covers the outer ring ranging from 0 to 2pi, and likewise ϕ describes the inner tube radius ranging from 0 to 2pi.

2.2 – Spotlights!

I created a function called `initSpotLight()` which initializes a spotlight light source featuring a source position, color, and direction. The spotlight shoots a beam of light in a conical frame, characterized by the two variables `innerConeAngle` and `outerConeAngle`. The values for these five parameters are initialized and passed on to the fragment shader via uniform variables.

In the fragment shader, lighting is computed rather intricately.
The program first evaluates whether the fragment falls within the conical frame of the spotlight beam. This is done through a simple dot product comparison – if the cosine of the angle the fragment makes with the direction of the light source is greater than the cosine of the outer cutoff angle, that implies the fragment falls within the target area. In this case, the ambient, specular, and diffuse components of the light are added up as per the Phong model. This result is finally scaled with an attenuation factor to provide soft shadows that gradually get dimmer as the object is further away from the light source.

However, if the fragment does not fall within the cutoff region, only the ambient component of the light is taken into consideration because there is no diffuse or specular component in this case.

3. Important Correction

The problem statement mentions the following scaling factor for modeling the gradual decay in light intensity from the inner to the outer end of the spotlight cone.

$f = (\phi - \theta in)/(\theta out - \theta in)$

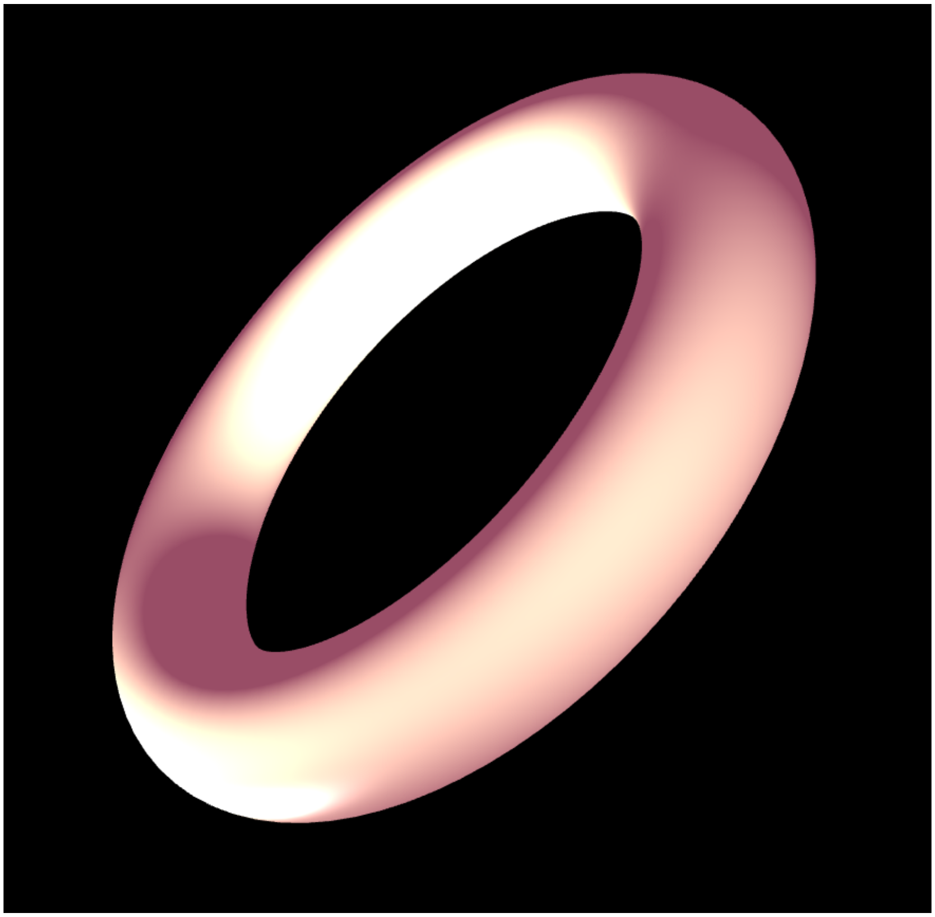
This gives the render displayed at the beginning of this report.

However, I believe it is incorrect. This expression evaluates to 1 at $\phi = \theta_{out}$ and 0 at $\phi = \theta_{in}$, whereas it should be the other way around.

The correct scaling factor is:

$f = (\theta out - \phi)/(\theta out - \theta in)$

This yields the following render



This is a much more accurate render than the previous one.

4. How to Run:

Run the following commands.

```
cd code
./run.sh
```