

CS 557 Computer Graphics Shaders

Project #1

Step- and Blended-edged Elliptical Dots

Name: Rachel Xing (#934229189)

Email Address: xingru@oregonstate.edu

Video Link: <https://youtu.be/qTY5XmIstaI>

Description:

This project shows the use of GLSL vertex and fragment shaders to create a repeating pattern (elliptical dots) on the surface of a 3D model. The selected model is a sphere generated using the `OsuSphere` function provided by “`osusphere.cpp`”. The pattern of elliptical dots is achieved by constructing a rectangular grid on the model's texture coordinates (s, t) and plotting ellipses within each grid cell using the ellipse equation. Two variables, `uAd` (ellipse diameter in s) and `uBd` (ellipse diameter in t), define the ellipse dimensions. To create smooth edges for these elliptical dots, the `uTol` variable defines the blend width between the ellipse and the non-ellipse region. The smoothstep function generates a gradient from 0 to 1 for pixels near the boundary within the `uTol` range. The mix function is then used to blend colors, thus creating a smooth color transition from the ellipse interior to its exterior (smooth edges).

Per-fragment Lighting is implemented with associated variables (`uKa`, `uKd`, `uKs`, and `uShininess`) using the code from the Project #1 Handout. These lighting variables can be modified through `glman`. Additional color variables, `uEllipseColor` and `uObjectColor`, control the pattern and sphere colors, which can also be adjusted using `glman`'s color palette.

This project can be run using either `Glman` with “`oval.glib`” or the GLSL API. An animation was created using a sine function to vary `uAd`, `uBd`, and `uTol` between smaller and larger values, with keyboard controls to toggle the animation for each variable (if using GLSL API):

- Key “a” or “A” – toggle animation for `uAd`
- Key “b” or “B” – toggle animation for `uBd`
- Key “t” or “T” – toggle animation for `uTol`
- Key “f” or “F” – freeze or resume the overall animation

Screenshots:

1. *Hard-Edged and Smooth-Edged Elliptical Dots by changing `uTol`*

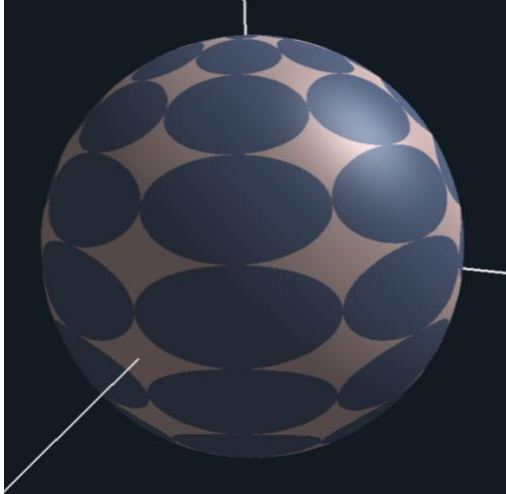


Figure 1. Hard-edged Elliptical Dots with $uTol = 0$ and default uAd , uBd

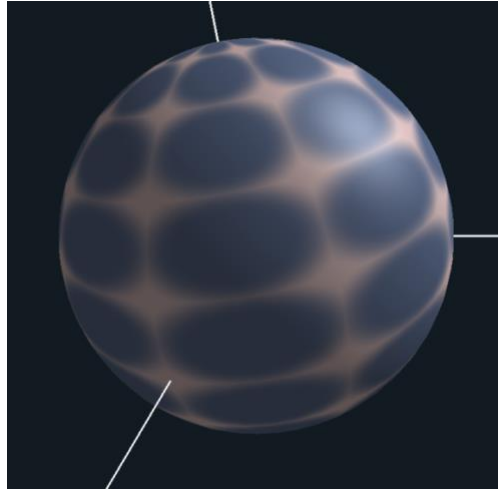


Figure 2. Smooth-edged Elliptical Dots with $uTol = 0.5$ and default uAd , uBd

2. Elongated Elliptical Dots by changing uAd and uBd

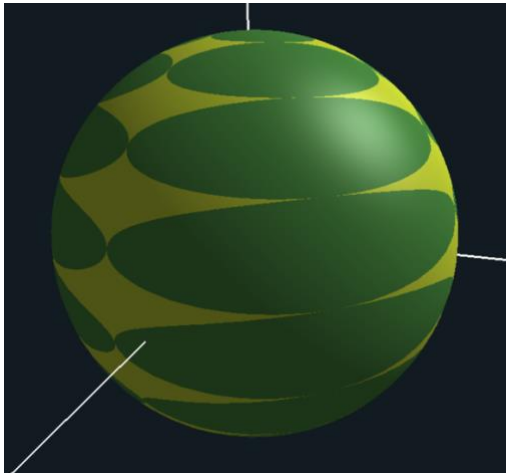


Figure 3. Elliptical Dots with **large uAd** and default uBd

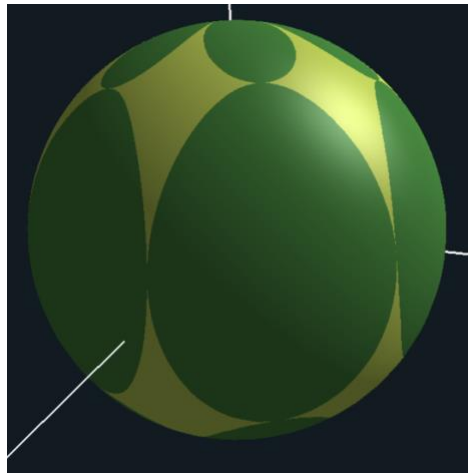


Figure 4. Elliptical Dots with default uAd and **large uBd**

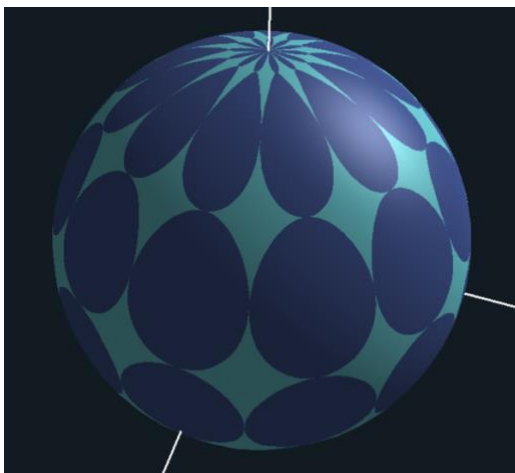


Figure 5. Elliptical Dots with **small uAd** and default uBd

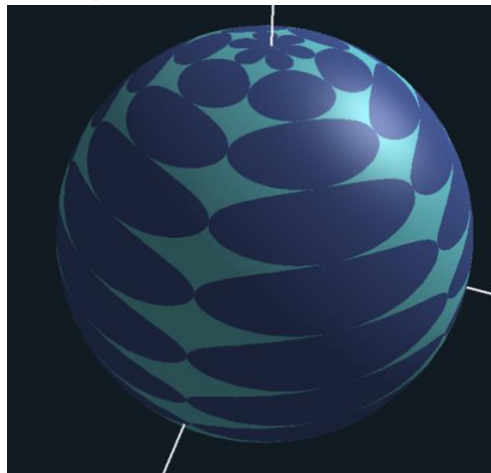


Figure 6. Elliptical Dots with default uAd and **small uBd**

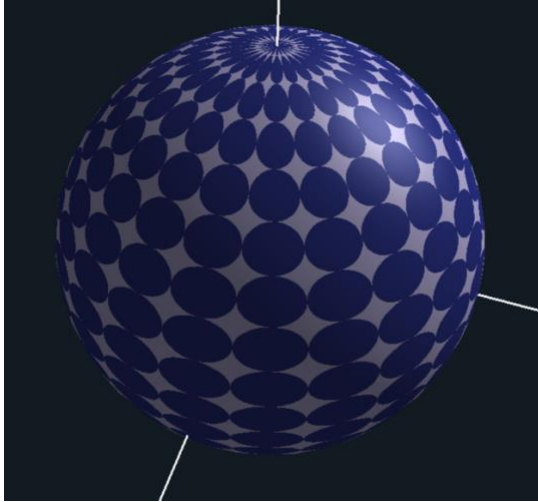


Figure 7. Elliptical Dots with **small uAd** and **small uBd**

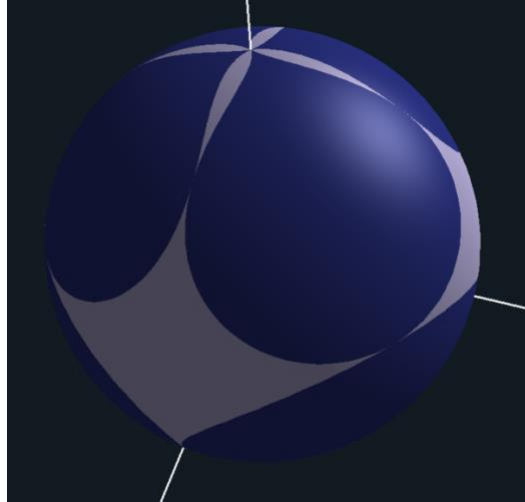


Figure 8. Elliptical Dots with **large uAd** and **large uBd**