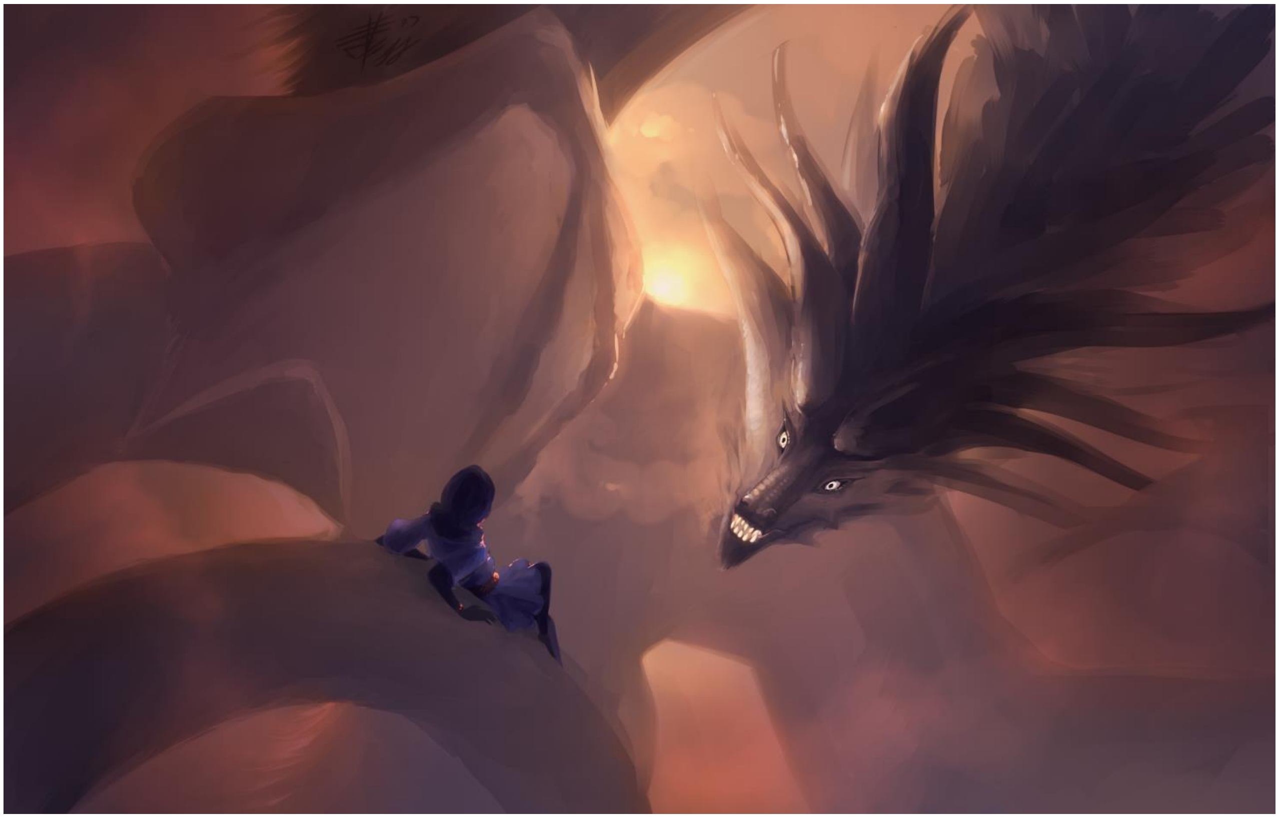
Jennifer Fullerton
CMPM 163, Homework 3
Angus Forbes
March 13, 2018

Final Project: Render Geometry in My Artstyle

For my final project, I will be writing a shader that lights, colors, blurs, and swirls geometry in a painterly-style—specifically, my own art style!

Here's a sample of my work:

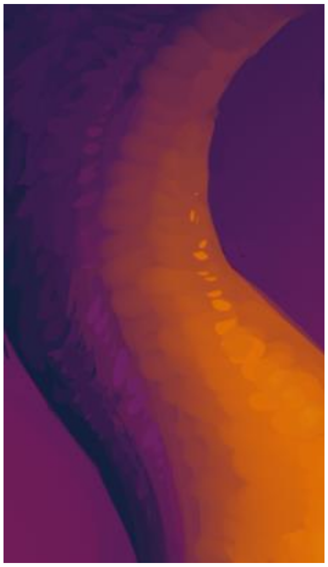
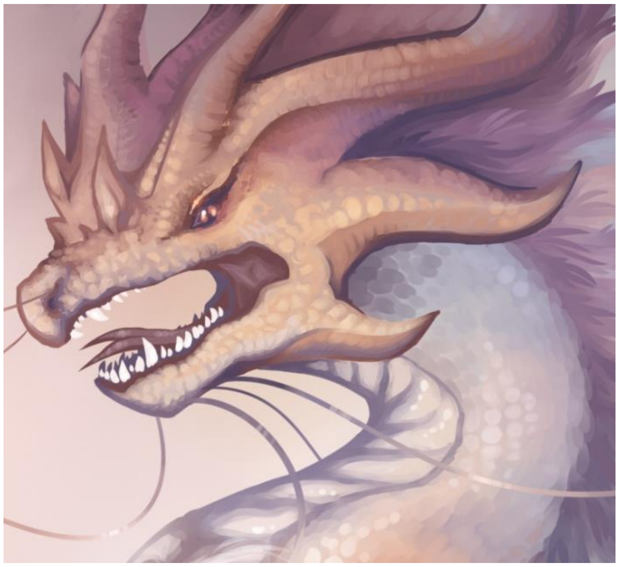
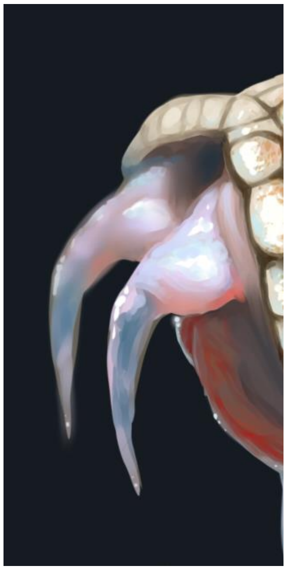


The specific aspects I would like to render will be first and foremost, the little points of light that I use to highlight the brightest areas, followed by a focus-blur, and then (if I can pull it off in time) some swirling and blending of the blur to mimic the paint-style that I use to fill in large areas of color.

My goal is to have all of this be calculated on the first pass of rendering, and primarily in the fragment shader. Some light positions and values may need to be passed in through the vertex shader, though.

Part 1: Specular Circles

In the pictures below, you can see my typical “dot” style of shading, which tends to be most present on highlights.

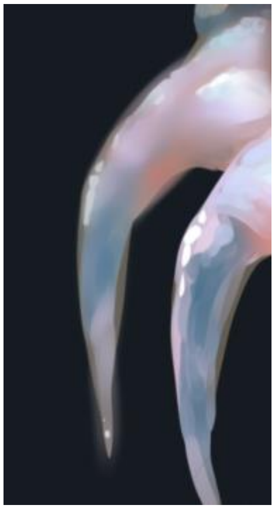




In order to accomplish this effect, I am going to try and use the light specular calculation from the Phong lighting model as a baseline for where the circles appear. I will then draw circles of color that are a mix between the specular color and the diffuse color. I may also have the distance from or intensity of light determine how large the radius is. I may *also* try to find a way to calculate dot “shadows” that appear, as can be seen in the middle image above behind the horns of the dragon.

I am not yet sure how I will draw the circle – I am hoping to accomplish this in first pass rendering, with either a Signed Distance Function combined with geometry information (possibly calculated in the vertext shader), gl_Points, an image kernal, or possibly based on some kind of texture with a second pass. I imagine the circle is going to be the most difficult challenge.

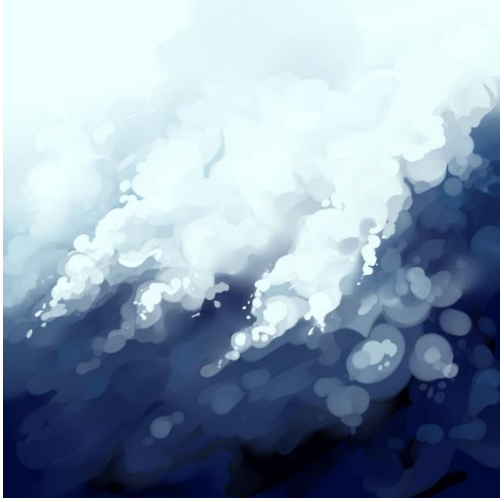
Part 2: Blurring

As seen on the pictures on the next page, another key part of my art style is to using blurring to add some kind of focus to the image. My plan is to use the z coordinates on the geometry to make a depth-of-field blur. The basic blur will probably follow a simple box-blur model (based on the average of surrounding pixels), and the intensity of the blur will be based on depth to mix the normal rendered pixel value with the blurred value. Again, my goal is to do this on first pass, but it may end up becoming a second-pass effect, where the depth field is preserved and saved to a texture to be used as input in the next shader.

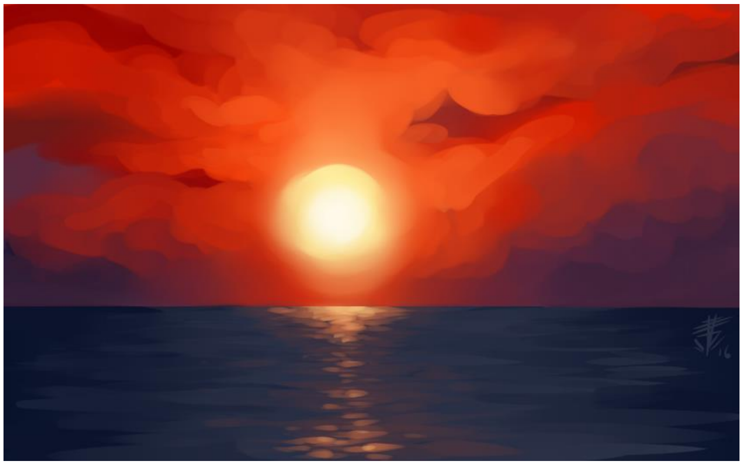


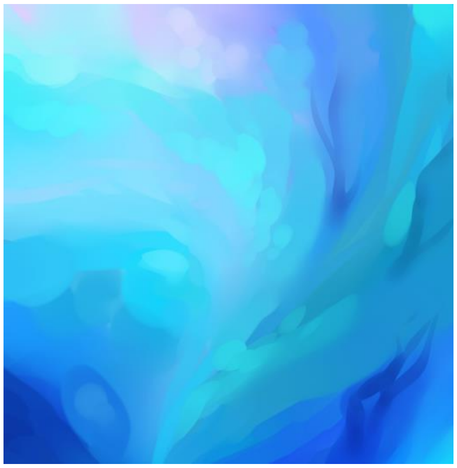
Left: Snake fang being blurred to bring focus to the first fang.

Right: some foam in the background of the "wave" and some of the water in the lower left is a bit more blurred and smoothed out to bring focus the foreground-wave foam and lighting.



Part 3: Swirling Colors in a "Painting" Style





Finally, I hope to augment the blur with either noise or some kind of kernal that I develop that looks at the gradation between colors so that the blur is not even. In some areas, there should be clear edges between colors, and in others a perfect blend. This helps add depth and motion to the image.

Project Partners:

- Joshua Navarro: post-processing image styling effects
- Lisa Durand: animation keyframe interpolation