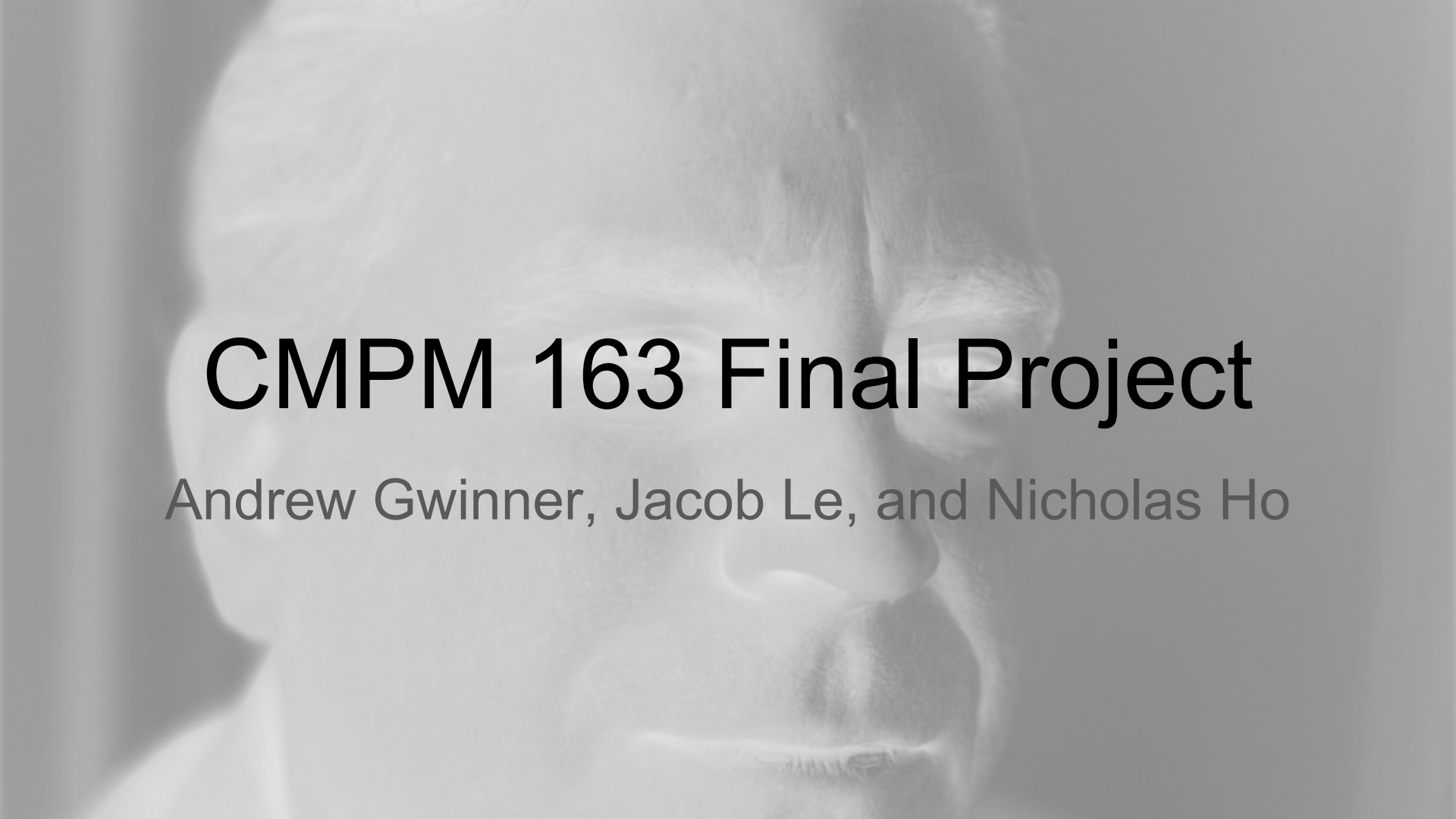
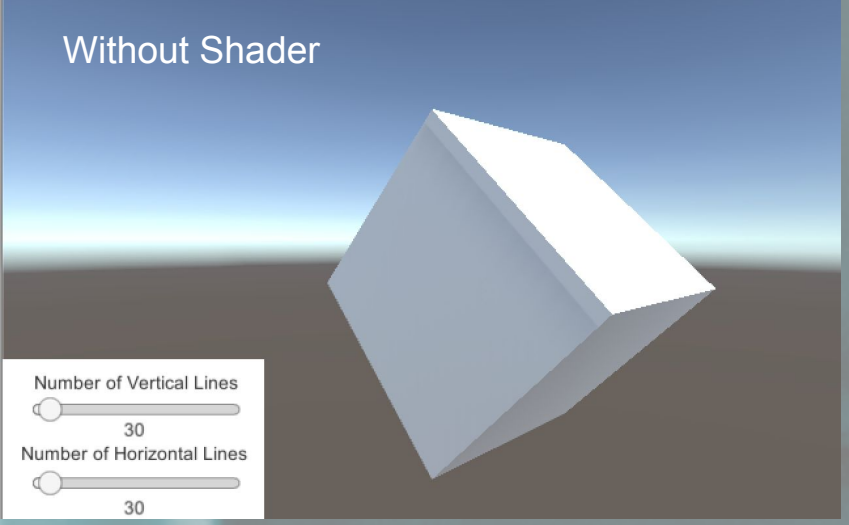


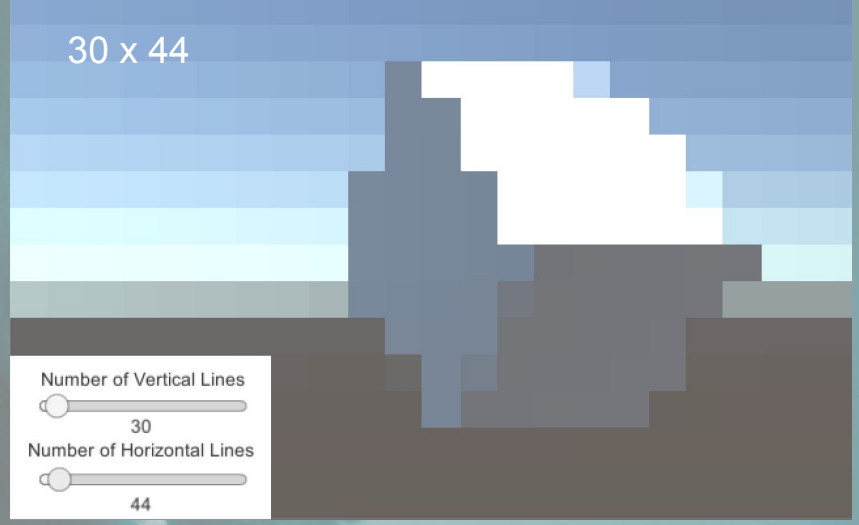
CMPM 163 Final Project

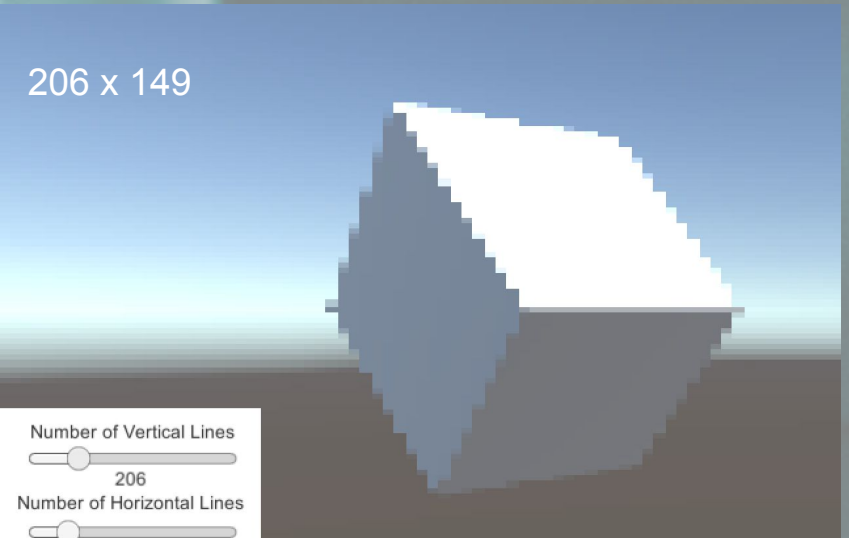
Andrew Gwinner, Jacob Le, and Nicholas Ho

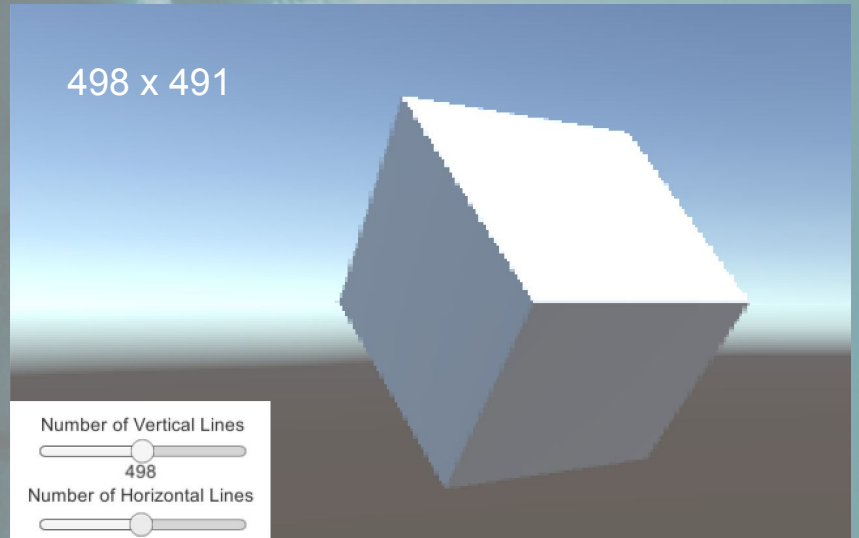
Pixelizer

- I store the number of vertical and horizontal slices in Ranges in Properties so the number of each can be variable within the shader and the scene
- I continually divide the screen width and height by the number of each type of slice to determine where each slice is
- I keep generating slices until finding the ones that surround the current pixel and save those values
- The midpoint of the vertical slices that bound a pixel and the midpoint of the horizontal slices that bound a pixel are determined.
- After dividing these values by the screen width and height appropriately the (x,y) coordinate gets converted into (u,v)
- The color value from that (u,v) coordinate is used for all pixels bounded by these slices


Without Shader
Number of Vertical Lines
30
Number of Horizontal Lines
30

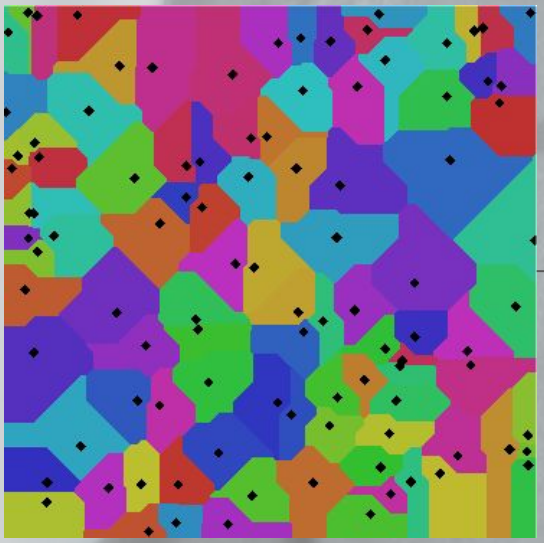

30 x 44
Number of Vertical Lines
30
Number of Horizontal Lines
44

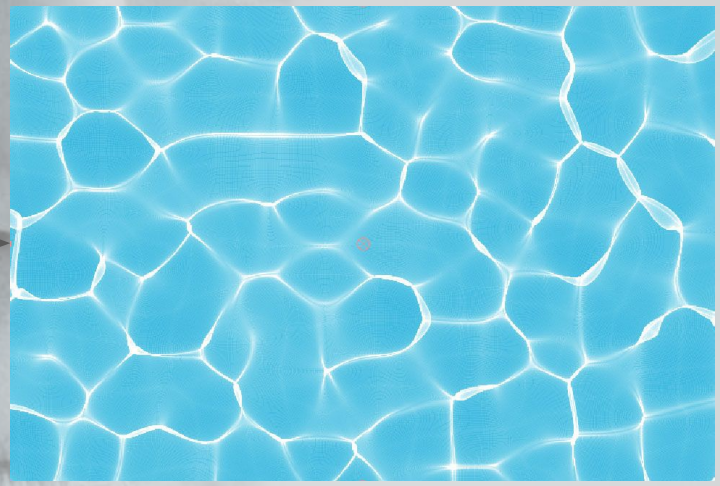

206 x 149
Number of Vertical Lines
206
Number of Horizontal Lines


498 x 491
Number of Vertical Lines
498
Number of Horizontal Lines

Animated Voronoi Caustics

- Use animated Voronoi Diagram to create caustic effects
  - Blue/white color
  - Move points over time
  - Stretch goal: Use signed distance functions to create a softer effect



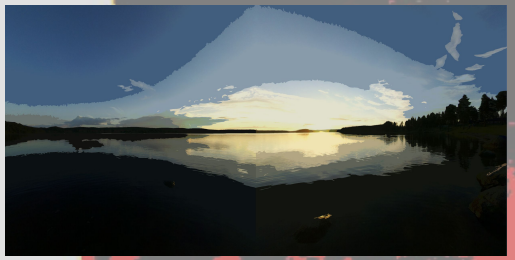


Voronoi Diagram

- Pass in point coordinates through uniforms
- Take current position and use it to find the manhattan distance between the two
- Loop and find the shortest distance, color appropriately
- Eventually points will move (x,y) every tick according to a noise function

God Rays

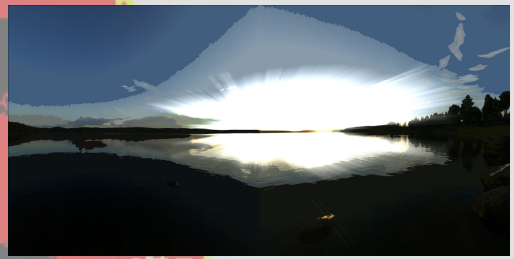
1. Start with our scene with a light source



2. Find what is light and what is blocking the light

3. Blur it outwards

4. Mix it with the scene



WIP

Rays are more round than shafts at the moment