Part A) Originally, I had misunderstood the usage of basic materials provided in three.js and used the phong shader and basic mesh material to render two of the three objects. As such, this was rewritten and the final objects are as follows. A dingo by Google Poly that is textured by the provided texture and uses lambertian lighting through threejs' point lights as following this blog, pointed to me by a fellow classmate and was useful for understanding how to pass point lights as uniforms. https://csantosbh.wordpress.com/2014/01/09/custom-shaders-with-three-js-uniforms-textures-and-lighting/ . Next, a cone that has a scan line shader overlaid on it. I found an interesting site that had examples of shaders called ShaderFrog and I looked around for something simple that would help me understand passing variables through the uniforms and this seemed like a simple example. https://shaderfrog.com/app/view/143 . Last, I used the example of vertex displacement that was in the class code and applied it to a torus knot
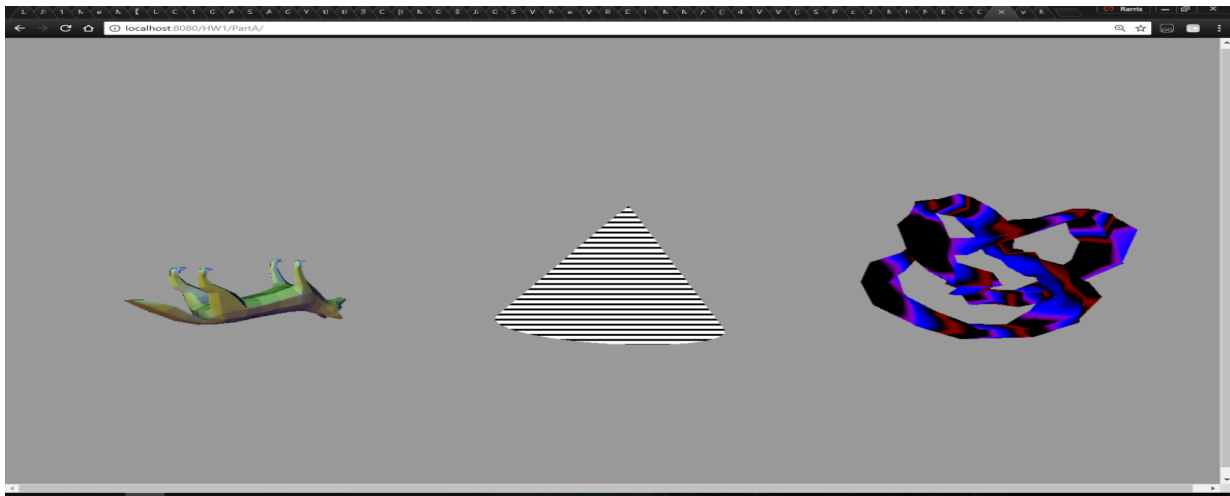
Part B) This part is based off the edge detection code that was shown in class. I modified it to use a sharpen filter provided by wikipedia which has the matrix [0,-1,0,-1,5,-1,0,-1,0]

Part C) Building off the pingpong example that was provided in class, I followed the rules posted on the linked blog post and implemented the 313 pattern by David Griffeath which has a threshold of 3, 3 states, and a range of 1. The 3 states I have are pure R, G, and B and it can be quite painful to view with the eyes but at the same time somewhat psychedelic.
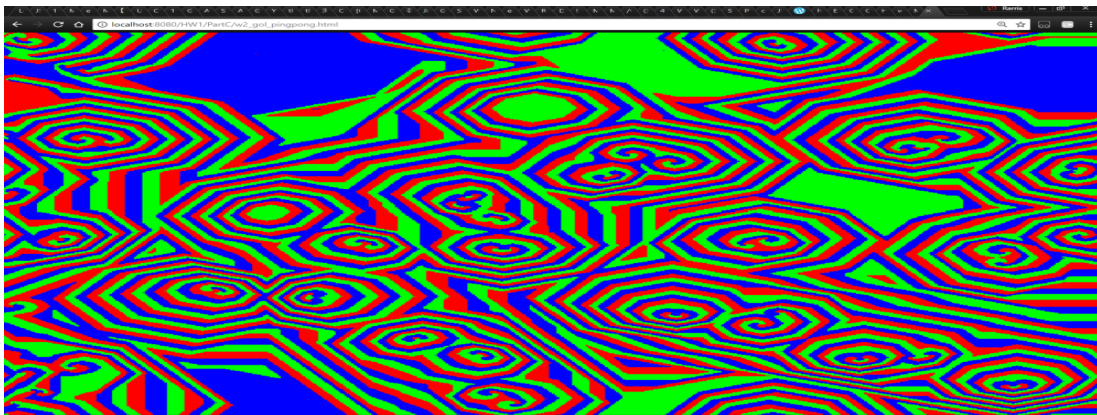
Part D) The visuals I will be discussing are from Antichamber by Alexander Bruce.