

Assignment 05: WebGL -- Resubmission

👤 Daniel Morrissey

Differences in resubmission

In order to satisfy the requirement of drawing something other than a cube, I created an obj parser. Because of the security issues with loading files in javascript, I wrote it in python, and did a copy paste of the results into the script. As such, reading the ass05.html file may be slow.

parseobj.py is the parsing script. Run it like:

```
python parseobj.py objfile.obj
```

Outputs a folder of files with the buffer data formatted inside.

```
objfile/  
├─ objfile_norm  
├─ objfile_rgb  
├─ objfile_tc  
└─ objfile_xyz
```

This above not a necessary step, the object I used has already been parsed

The model is a subdivided suzanne from blender. Texture was made by me.

Any object could be drawn, although the monkey is currently hard coded.

No changes were made to the shaders, or any other code besides the buffer values.

To Run

The open_page.sh script runs the following:

```
chromium-browser --allow-file-access-from-files ass05.html
```

If your chromium is called something else, you'll need to adjust for your environment.

OR

Just open ass05.html in Firefox.

Both browsers were tested, FireFox seems to run better, chromium seems to look better (colors are more vivid, may be completely in my mind).

Original README

The purpose of this assignment was to gain familiarity with the WebGL flavor of OpenGL.

I used ex06 as a starting place, and heavily relied on the following sites for code samples:

- <https://developer.mozilla.org/en-US/docs/Web/WebGL>

As such, I make use of the canvasmatrix.js library.

Notes

- Fully lit (no specular) and textured, with normals built as well.
- Rotate the view with the mouse button, or via the Animate Box button
- Cause the light to orbit with the Animate Light button
- The first frame that gets displayed usually wont have textures loaded yet, as they are lazy loaded. Once the textures are loaded, they should be drawn properly.

Other Comments

- Perhaps its a result of the "friendliness" of javascript, but I was very much able to get a better grasp of how the ArrayBuffer system works after doing this assignment. Theres less overhead than with C.
- Leagues easier to debug WebGL, even with shaders. Firefox proved to provide extremely helpful console messages, which was a nice contrast to things just segfaulting in the past.
- Theres no e