

# Project Rasterizer  
By Leon Thompson

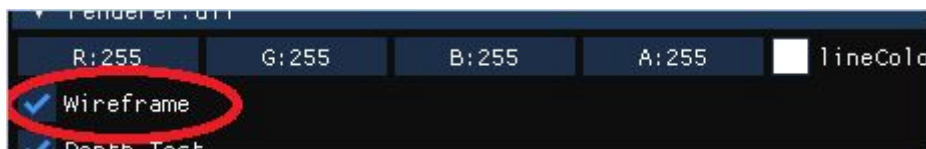
## ## DESCRIPTION

Build 3D software renderer that uses a scene to illustrate the processes in the graphics pipeline(triangle rasterization, clipping, etc.). All math calculations are done without the use of outside library.

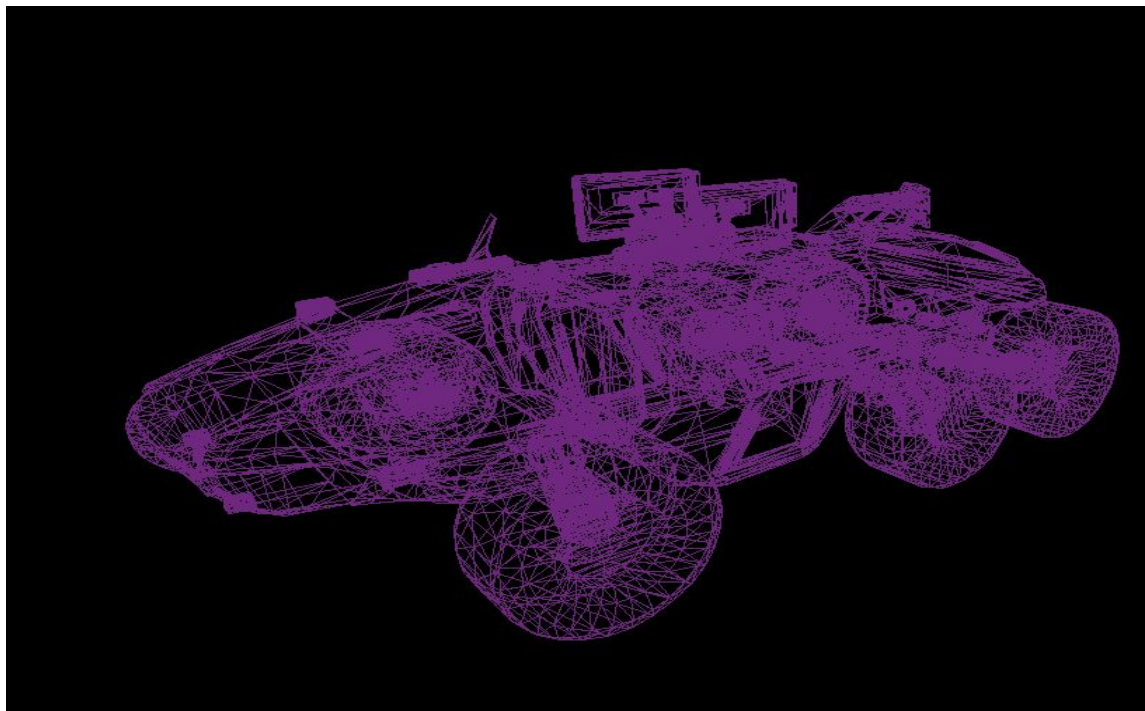
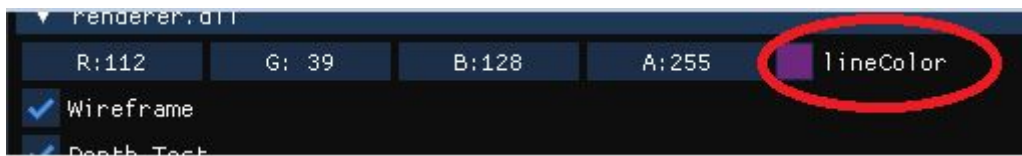
## ### FEATURES:

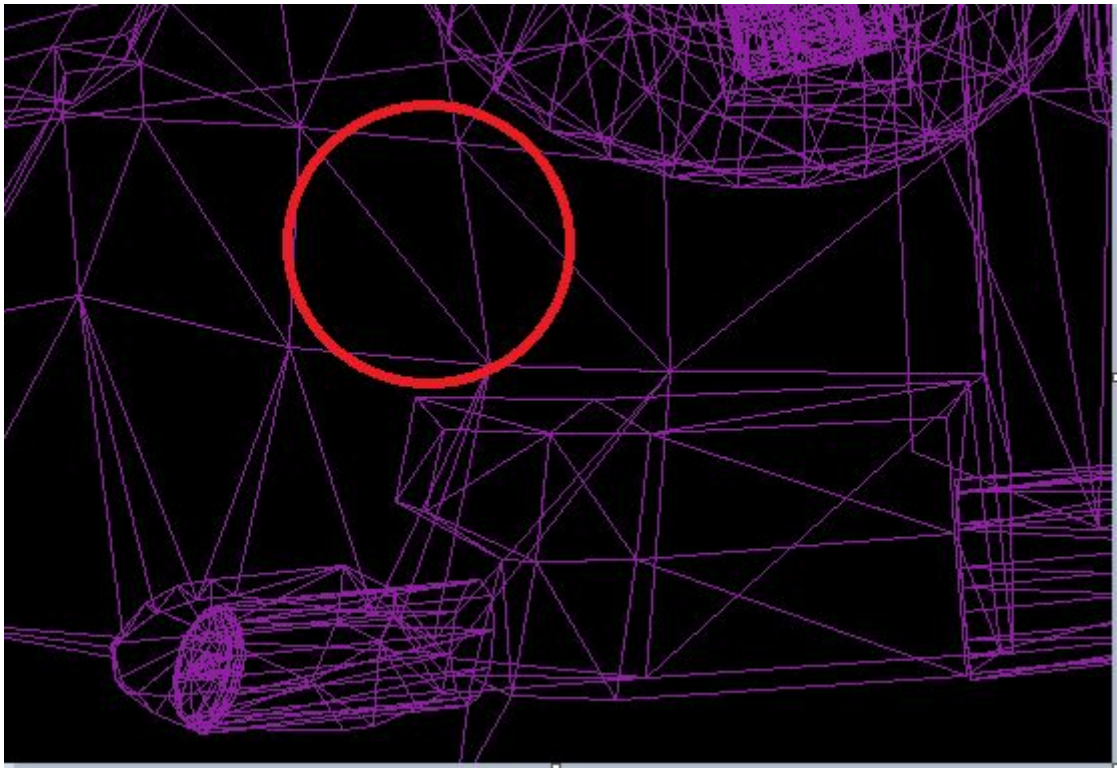
- WIREFRAME

A rendering of each triangle described by the vertices in the .obj file of each model in the scene. To activate the wireframe, click the checkbox beside **"Wireframe"**.



By default the color of the line is white but can be edited by changing the *"lineColor"* option directly above Wireframe.





- Zooming in closer provides a better view of the separate triangles.

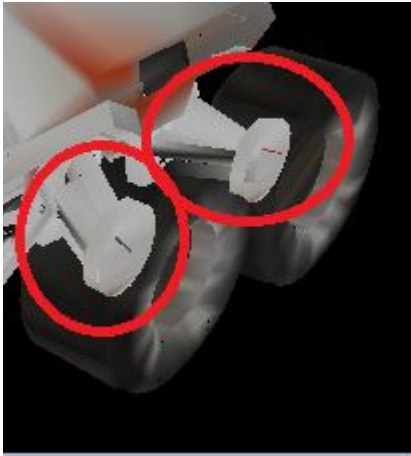
#### - DEPTH TEST



Activating the **depth test** ensures that if multiple colors are to be drawn on the same pixel, only the one that is closer to the screen according to its z coordinate will be displayed.

Please view an example below:

Depth Test Deactivated:



- We can see axles and rims that would normally be hidden by the tire are clearly visible.

Depth Test Activated:



- Looking at the same tow tires we no longer see the axles or rims.

#### - BACKFACE CULLING



**Backface culling** attempts to save on performance by finding all polygons that can not be seen by the camera (based on the normals provided by each vertex) and avoids rendering them.

Backface Culling Deactivated:



- Here we can see the front bumper of the vehicle.

Backface Culling Activated:



- Here we can see that front bumper has receded some and one of the lights below the vehicle

Now my implementation of backface culling is not the best, which I will expand upon more in the BUGS section.

- Toggle Phong & Goraud Shading



**Phong Shading** determines the color of pixels based on lighting calculations on a per pixel basis in the pixel shader. This leads to more natural looking shadows on the model. **Goraud shading** is determined in the vertex shader which often results in shadows that appear to follow the lines of a triangle. Unchecking the “Phong Shading” box automatically enables Goraud shading.

Goraud Shading:



Phong Shading:



- We can see that with Phong shading the direction of the light (top left -> bottom right) is more easily depicted.

#### - ALPHA BLENDING

The higher the alpha on an object the more opaque it becomes. Alpha blending allows the object to blend or mix with the color of the background as its alpha value is lowered from 1 to 0.



Above "clearColor" indicates the background. Alpha blending is enabled/disabled by clicking its corresponding checkbox and "Alpha Value" indicates the alpha of the model in the scene.

Alpha Blending: Enabled  
Alpha Value: 0.1



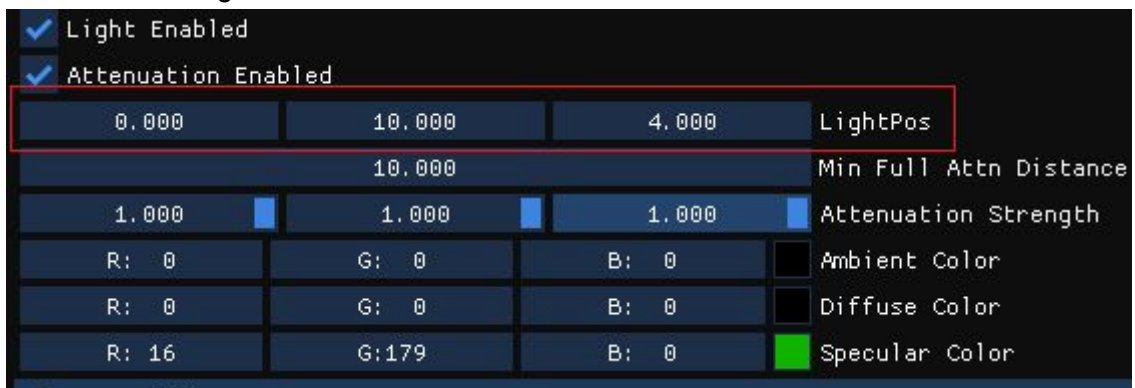
Alpha Blending: Disabled

Alpha Value: 0.1

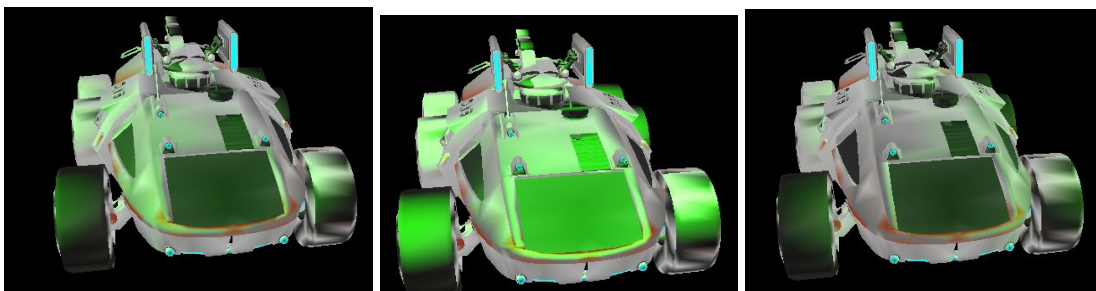


## - LIGHTING

### - Light Position



Below the x position of the light is being manipulated from negative to positive:



### - Attenuation

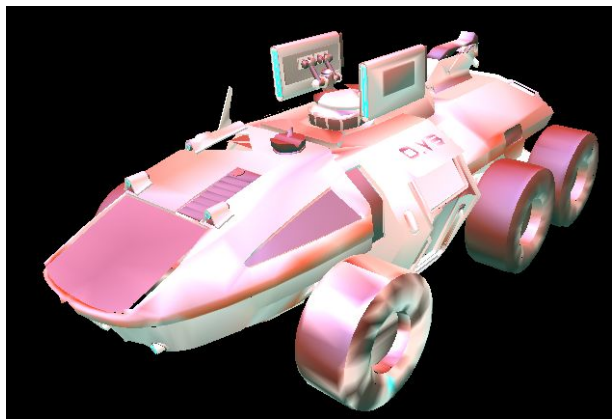


Attenuation determines the strength of lights being applied to the model. By modifying “Attenuation Strength”, you change the percentage of light (red, green & blue).

<input checked="" type="checkbox"/> Light Enabled			
<input checked="" type="checkbox"/> Attenuation Enabled			
0.000	10.000	4.000	LightPos
10.000			Min Full Attn Distance
1.000	1.000	1.000	Attenuation Strength
R: 0	G: 0	B: 0	Ambient Color
R: 0	G: 0	B: 0	Diffuse Color
R: 16	G:179	B: 0	Specular Color

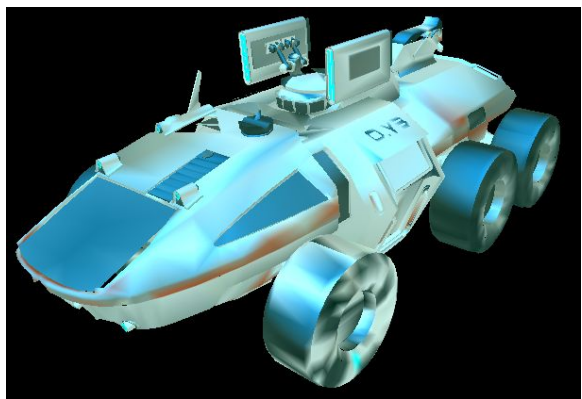
Example with all attenuation strength values at 1.

1.000	1.000	1.000	Attenuation Strength
R: 24	G:136	B:100	Ambient Color
R:213	G: 0	B: 0	Diffuse Color
R: 0	G:104	B:186	Specular Color

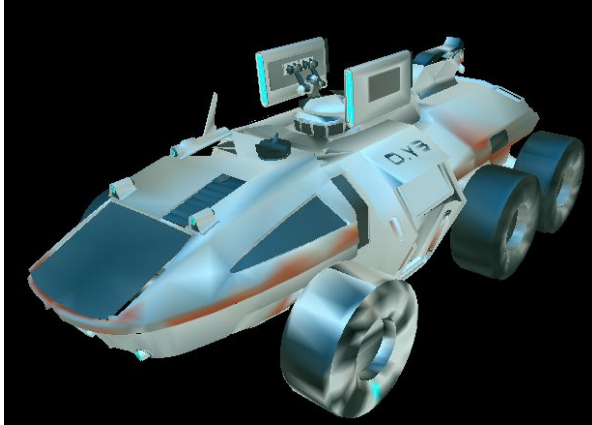


Same example with the red attenuation strength value at 0.

0.000	1.000	1.000	Attenuation Strength
R: 24	G:136	B:100	Ambient Color
R:213	G: 0	B: 0	Diffuse Color
R: 0	G:104	B:186	Specular Color

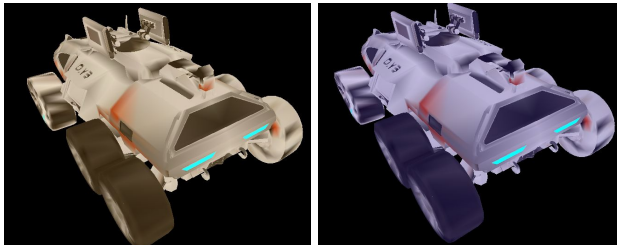


Also note the option for **Min Full Attn Distance**. This option determines the minimum distance at which a light will be at 100% of its attenuation value. In the above two examples of attenuation this value is at 10. Below it is set to 5 and the light is in the same position:

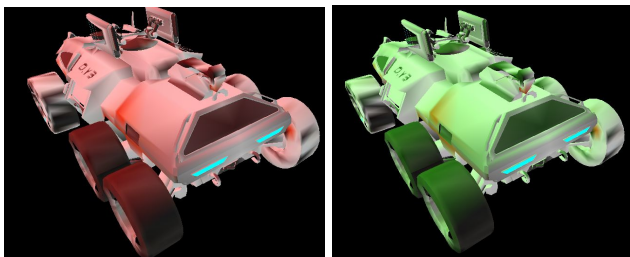


- Ambient light

The **Ambient** light adds a flat color equally across the entire mesh:



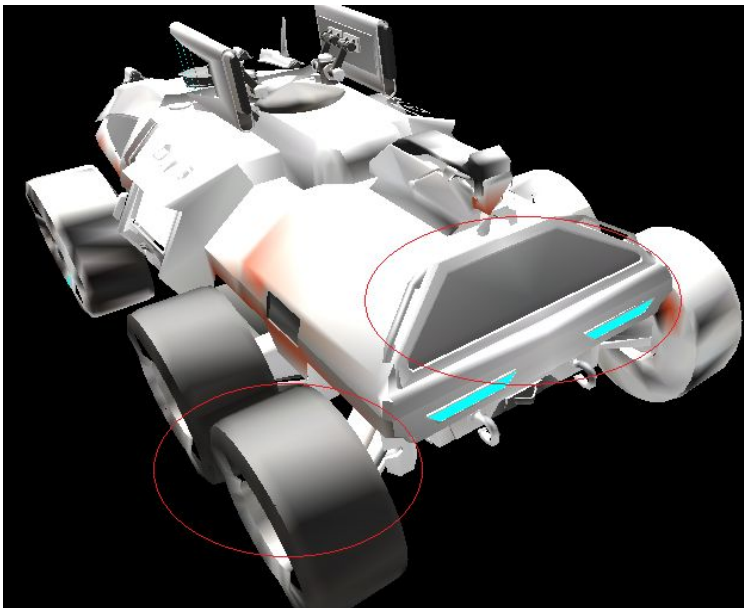
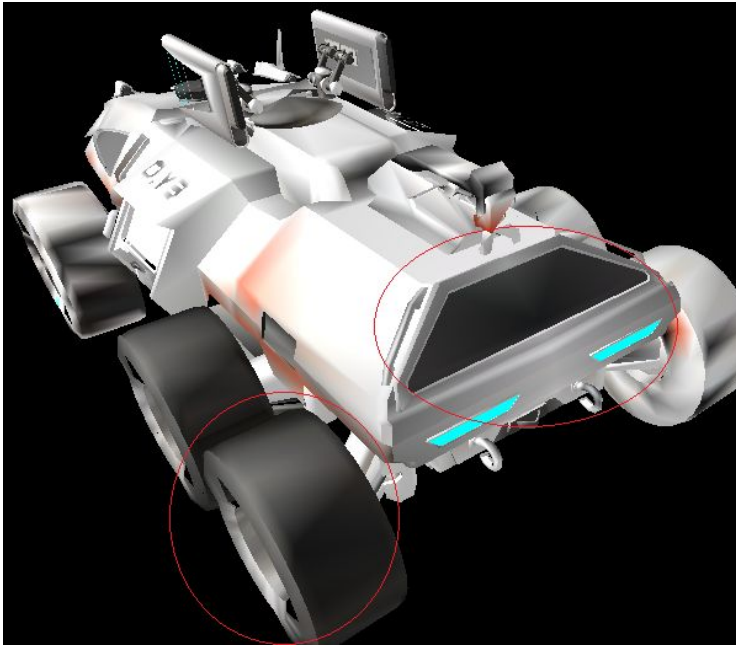
- Diffuse Light



- Specular Light

Specular light is light reflected towards the camera, thus the render is dependent upon the camera's position.





### BUGS:

### RESOURCES:

- Phong Reflection Model
  - [https://en.wikipedia.org/wiki/Phong\\_reflection\\_model](https://en.wikipedia.org/wiki/Phong_reflection_model)
  - Formulas for lighting and reflection:
- Loading Images by Pixel
  - <https://www.cplusplus.com/forum/beginner/267364/>
- Determining if a given pixel is inside a given triangle.
  - <https://stackoverflow.com/questions/2049582/how-to-determine-if-a-point-is-in-a-2d-triangle>
  - Answer by: Glenn Slayden
- Depth Test & Depth Buffer

- <https://www.scratchapixel.com/lessons/3d-basic-rendering/rasterization-practical-implementation/visibility-problem-depth-buffer-depth-interpolation>
- Backface Culling
  - [https://en.wikipedia.org/wiki/Back-face\\_culling](https://en.wikipedia.org/wiki/Back-face_culling)
- Barycentric Coordinates & Color Interpolation
  - [https://en.wikipedia.org/wiki/Barycentric\\_coordinate\\_system](https://en.wikipedia.org/wiki/Barycentric_coordinate_system)
  - <https://codeplea.com/triangular-interpolation>
- Alpha Blending
  - [https://en.wikipedia.org/wiki/Alpha\\_compositing#Alpha\\_blending](https://en.wikipedia.org/wiki/Alpha_compositing#Alpha_blending)
- Obj Model
  - <https://www.renderhub.com/rip-van-winkle/landing-vehicle-from-mass-effect-andromeda>

FRONT



BACK

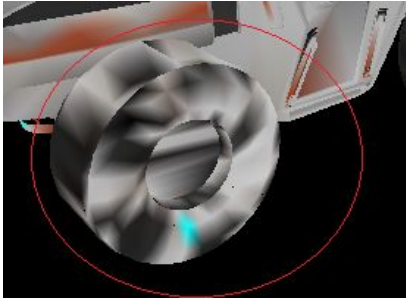


### BUGS:

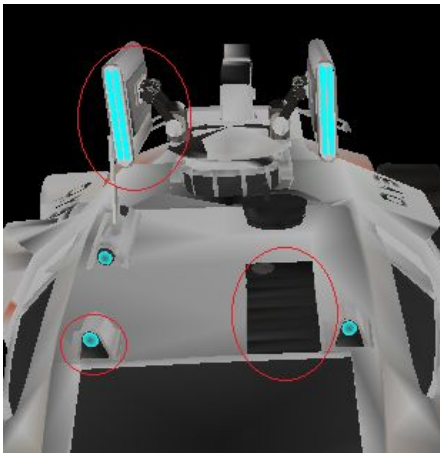
1

My code is capable of accepting an obj with multiple textures. However, there are several inaccuracies with models using multiple textures vs 1 texture:

We can see here that some of the tiles are shaded completely incorrectly:



However the shading on other portions of the vehicle seems to be more accurate:



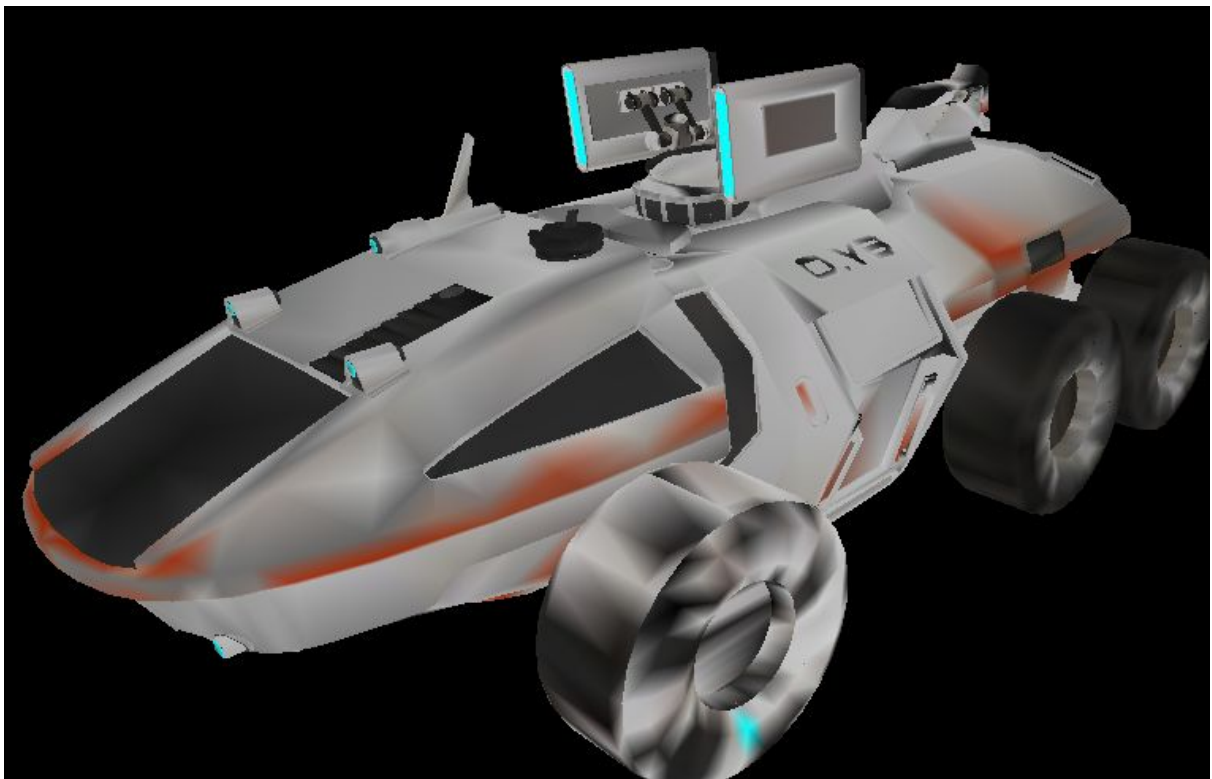
Now we see the render of a cat where only 1 texture is used and does not seem to be suffering from the inconsistency issues that the vehicle above is:



2

My implementation of backface culling appears to be too aggressive as there are often holes or even small sections of the vehicle that are missing when it is enabled.

Backface Culling disabled:



Backface Culling enabled:

