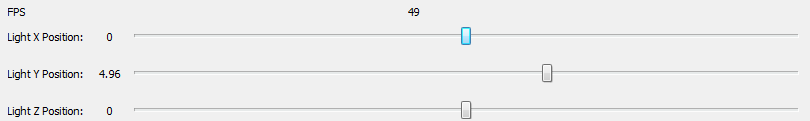
**Computer Graphics – Assignment 1 – Ryan MacDonnell**



Camera start location for application



A closer view of the scene shows a light source in front of the leaf with a plane in the background. When the light source is moved (via debug menu) the light will reflect onto the leaf appropriately. The plane is the back does not take light into account and just displays the texture completely.

   
Debug Menu located above Scene with some preset values as well as minimum and maximums of the sliders. The FPS is a calculation of the time that the application is taking to update.

**Requirements:**

1. Place the camera startup position where it makes sense.
2. Apply a binary alpha map to a plane with an appropriate matching texture.
3. Display some scenery behind the binary alpha-mapped plane.

**New Code to implement Requirements:**

1. Modify the TextureInfo class to contain more than 1 texture for the multi-texture feature of the lab.
2. Update the Renderer class to accommodate multiple textures in a Renderable object.
3. Update the MeGame manager class to support multiple textures.
4. Update the MultiTextureVertex/FragmentShaderCode files to allow the multiple textures to be passed through as well as lighting to work on the new shader code.
5. Add textures that can easily tell that parts of the original texture is being discarded.

**Fulfillment:**

1. The objects contained in the TextureInfo class now contain an array of size 2 for the TextureID and TextureIndex to allow multiple textures to be applied to the same object. I also applied a Boolean object for myself to let other sections of code know whether there are multiple textures for an object or not.
2. The renderer class has been updated to support multiple textures in the following methods
   1. addTexture now has an overload method to have contain the following
      1. char\* const file1 – original texture
      2. char\* const file2 – second texture (currently used as a mask)
      3. rend\_uint index – index for first texture
      4. rend\_uint index2 – index for second texture
   2. doPass now contains a check for whether or not to setup Uniforms based on if the renderable passed in has multiple textures or not
   3. setupUniforms was created to setup the uniforms for a Renderable that contains multiple textures
   4. TextureInfo2 has been deleted as it is no longer needed.
3. In the meGame class the doRendererStuff method has been updated to support multiple textures with the following
   1. ShaderInfo\* multiTextureShader - adds a shader to the renderer for the multiTextureShaderCode.glsl files
   2. TextureInfo\* multiTexture1 – adds the textures for a single object the Leaf and whiteLeaf files.
4. The MultiTextureVertex/FragmentShaderCode.glsl files have been updated to be able to use multiple textures
   1. MultiTextureFragmentShader.glsl now contains a method to calculate light for the object and applies light determined if the object is facing forward towards the light or not.
5. The following textures have been added to display the discardment of pixels is applied correctly
   1. Leaf.png
   2. whiteLeaf.png

**Additional:**

1. Pressing ‘w’ and ‘s’ moves the camera on its forward vector in the negative and positive direction respectively
2. Pressing ‘a’ and ‘d’ moves the camera on its right vector in the positive and negative direction respectively
3. Pressing ‘r’ and ‘f’ makes the camera rise or fall on the up vector of the camera in the positive and negative direction respectively
4. Holding down left click button on the scene and moving the mouse rotates the camera based on the direction you move the mouse.
5. The sliders in the debug menu can be moved by left clicking on the mouse and dragging the mouse in the direction you wish to move the slider. Sliders are scaled from negative (left) to positive (right)

**To Play**

1. Compile and run ‘MeOpenGLScratchPad2’ for debug to allow the sliders to exist and release to not worry about the sliders.
2. Move the camera around to an appropriate position for yourself to see what you want to in the scene. I have the camera farther away since I was previously working with larger objects and scenes.
3. Move sliders to manipulate how the scene is currently behaving as well as any checkboxes that may be incorporated.

**Note:** If you are having problems with linkers or a related compile error, then there are some configurations that need to be used in the properties manager of the project itself. I can show how to properly have the configuration if you need to meet with me.