



# COS344 HA Report

Group: Batman & Robin

# Report

## North and South walls

Since the walls have repetitive patterns they will be rendered using a mesh that mimics the contours using vertices and indices. We can use cutouts for the windows on flat concrete walls. For the data structures used, they will be VAO's and VBO's for the vertices values. For window parts of the mesh we would not use the default wall texture map. It will be a transparent panel. We could just store a Boolean value if the part of the mesh is a window or not. There will be a tile, flat and window texture used for the wall. The pattern will also be able to repeat.

For some optimization we would use view culling to not render the walls not in the viewing angle. Also we can use instancing for repetitive elements like the wooden panels which will be done as a bonus mark as well.

Another bonus mark will also be the offices behind the windows or shaders that are just closed. They can be very low level of detail because they aren't really visible.

## East and West walls

These glass panel walls are a much simpler geometry so basic square windows will be sufficient to fill the wall. The windows will have almost full transparency with the background rendered in very low detail to save computing power. The background will just be very basic polygons. The reflections can be handled using environment maps. Cube maps will simulate reflections based on the angle the drone camera is from the actual glass panel.

The black part of the wall is just a simple set of vertices for rectangular wall and texture mapping for the tile look.

View culling can be used again for efficiency and also instancing since the windows are basically the same object.

The same data structures for the north and south walls can be used again.

## Roof

A triangular mesh can be used for the curvature of the roof. We can have a lower density of triangles for efficiency but would degrade the curvature a bit. To generate the mesh vertices we could mathematically calculate it using the equation for a semi cylinder.

There will be rectangular panels to construct the roof rib by fitting them to the mesh. Each triangle will then either be a blue or yellow colour to represent the different colours. It will also have different transparency values. The whole roof will have a repeating texture map to the mesh. The panels will influence the colour of the sun that passes through each panel.

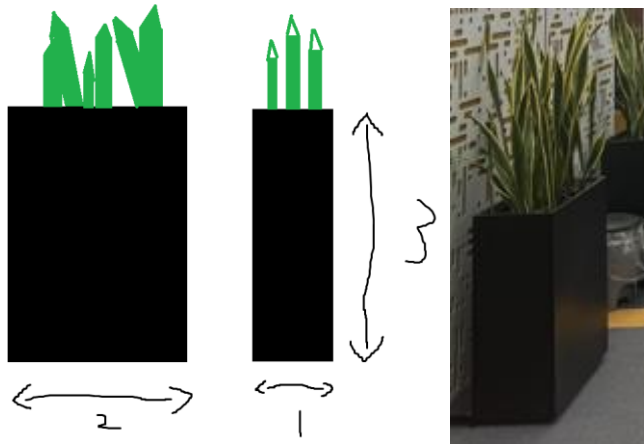
We can use view culling again for efficiency sake.

## Floor plan

Unfortunately the floor was not measured exactly because that would have been a time consuming activity. We do however have the amount of feet (length of our feet) per object.

Our objects are:

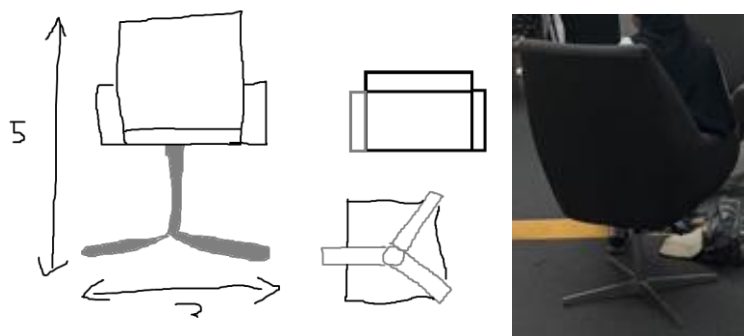
### 1. Plant pot

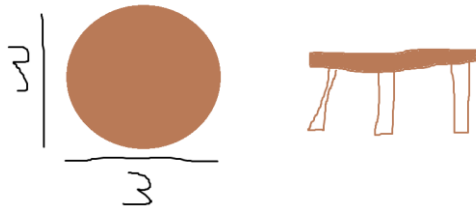
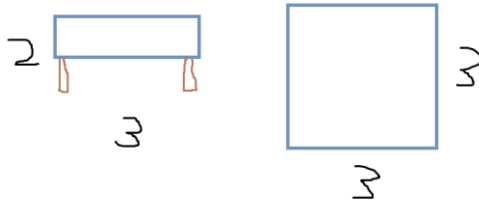
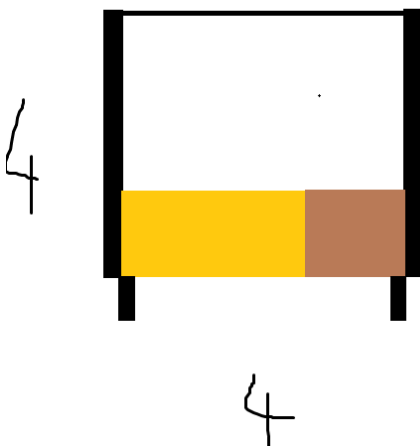


### 2. White wall



### 3. Armchair



4. **Wooden table**5. **Blue seat**6. **Black box seat**

For the floor plan we have the layout picture. Its pretty self explanatory. To the west is the coffee shop where there is stairs at the top and the coffee shop right below it with the coffee machine on the table and then to the right of that is the dustbins and table for cup lids. Then the only other unclear thing is the grey squares to the left which is the gaps in the floor which shows the informatorium halls. There are some other objects not shown but they are really simple objects like bins which are just cylinders or different types of chairs but all the chairs will be the same.

The height of this picture is about 60 feet and a width of 25 feet. The height is estimated at 30 feet.





## Materials

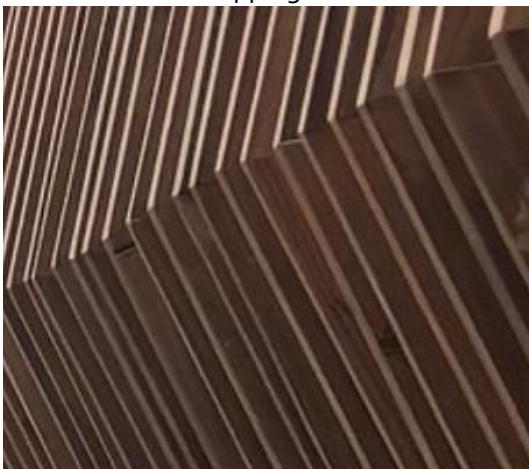
1. **Concrete walls**, the colour will just be a shade of grey. The roughness will be mapped with a texture map. The shading will be diffuse since walls don't reflect much light. The tile effect can also be part of the normal texture map. This will be implemented in the wall shaders.



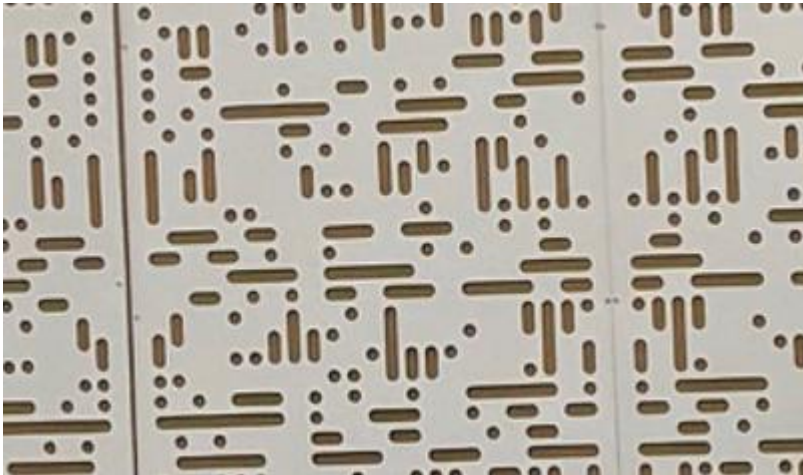
2. **Carpet**, there are dark grey, light grey and yellow carpets on the floor. It will also be used for the armchairs seating with their various colours. Diffuse shading again with a carpet normal texture map.



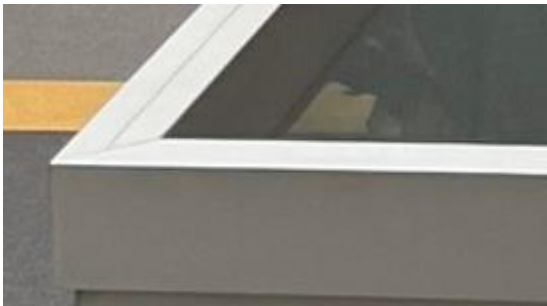
3. **Wooden panels**, brown colour with visible grains. The reflection is moderate specular. Normal texture mapping with the wood look for the grain.



4. **Decorative walls,** brown walls with a normal texture map that repeats the pattern. The texture map will be white with cutouts.



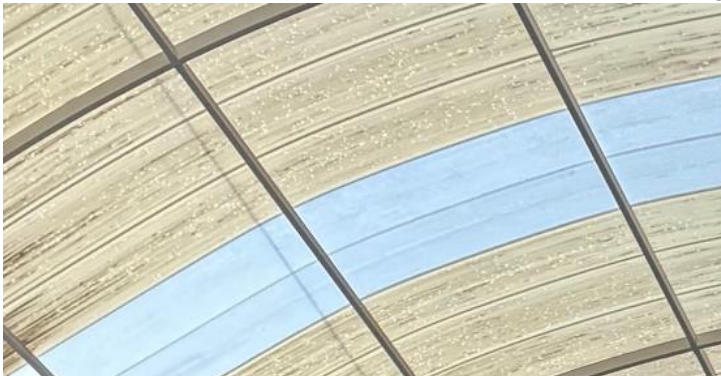
5. **Metal railings,** smooth grey surface that has shiny reflections. High specular coefficient in the shader.



6. **Glass panels,** very high transparency with a slight blue tint. Cube maps to handle the reflections. Shaders will handle the light pass through.



7. **Roof panels,** normal texture mapping with some roughness and dirt scattered. Colours will be in the mesh but entire mesh will have the same repeating texture map.



## Lighting

We will have 2 sources of lighting. The sun and the small LED lights.

For the **sun** it would be regarded as a directional light with a width of the entire kiosk. The directional light will not move in the 'sky'. The colour of the light will depend on the time of the day as described in the bonus marks. But the colour will range from yellow to red in the day. At night there will be no directional light or maybe minimal white lighting with low intensity. Since the directional light will not move we can have shadow maps for each object from the sun which would be computationally efficient. There will be reflections off surfaces which would diffuse the light a bit making it more realistic. This reflection will depend on the material of the object and its colour.

For the **LED lights** they will be regarded as white point lights with their own respective shadow maps which would save computations again. They will have the same type of reflections from surfaces as the sun. They will be predominantly present at night time to illuminate the room. There will be 9 of these point lights on each of the North and South walls. There will also be 2 more point lights at the West wall close to the shop.

## Code design

### Data structures:

- Vectors will be used to store any geometric data like vertices, normals and directions. We can use the built in glm vec functions for better integration.
- Maps will be used for shaders and textures because of the quick lookup and efficiency.
- Matrices will handle all transformations to objects using the built in mat4 matrices.

### Design patterns:

- The singleton pattern would be really useful in handling the resources that will get used multiple times. This is really important for efficiency when it comes to all the maps used in the textures.
- The observer pattern for the windows since their reflections rely on the drone camera position and the camera will be constantly moving.

## Drone

The drone will be movable with W-A-S-D keys for position and direction will be controlled by the mouse. The mouse will influence the direction vectors and W-A-S-D will translate the drone in those directions. Q and E buttons will be used to influence the altitude of the drone easily without the need for mouse functionality.

The camera will be placed at the nose of the drone and will be a perspective projection to be as realistic as possible. The camera will move with the drone. The camera viewing direction will always be in the W direction vector.

The drone will not have shadows since they will not really be visible during the tour.

The drone will have a 3<sup>rd</sup> person toggle which will be TAB. The rest of the 3<sup>rd</sup> person will be explained in the bonus marks as well as the drone design.



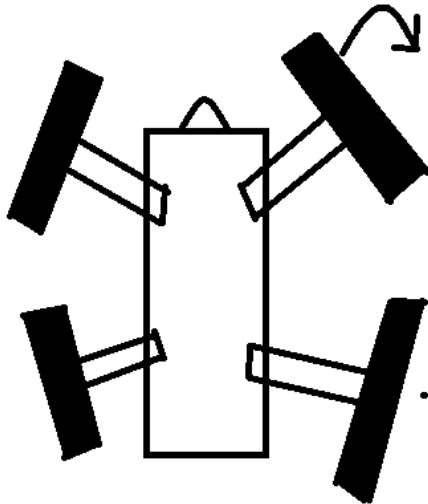
## Bonus marks

### Drone HUD

When in first person view the drone will display a HUD that has key statistics like current position, speed, compass direction and some overlay graphics that make it a more realistic HUD.

### 3<sup>rd</sup> Person view of the drone

Render the drone when switching to 3<sup>rd</sup> person view and also move the camera axis to track the drone from the top back of the drone. Also disable the HUD. The drone will be formed out of simple polygons with basic colours. Will try to have the propellers to spin.



### Adding wood siding on walls

Add some basic rectangles with a wood texture mapping with no bumping since the wood is smooth. Can be mathematically calculated with a sin wave for their positions since the pattern repeats.

### Change the time of the day

Have buttons to increment/decrement the time of day. Have a mathematical function influencing the colour of the sun (directional light). The colour of the sun is more red when it gets closer to the horizon. There should be no/little directional light at night.

### Adding extra details behind the glass on each wall such as offices

Render basic offices using basic polygons behind the windows to add a more realistic look to the kiosk. For some offices the shaders will be closed which will just be white rectangles behind the transparent windows.

### Adding the coffee shop, including snacks and a coffee machine

Render the counter as a rectangular prism with some cylinders as cans and a coffee machine which would a complex set of polygons fit together. Could also add snakes like chocolate bars which are the right colour and rectangular prisms. Add the blackboard behind the counter for the menu. Not necessary to add text but would if its easy.