CS461 PROJECT REPORT

Topic : 3D Model of a City

Submitted By : **Partha Pratim Malakar (170101043)**
Pranshu Srivas (170101048)

❑ Introduction

We made a 3d model of a city using WebGL.

WebGL is a graphics library to display 2d and 3d graphics object using javascript and html only.The reason we used webgl is that ,html file can be opened in any browser.So our project will also run in all browser and in android devices also.

In our project ,we draw various objects in the canvas to look like a 3d model of a city and we also implement many others functionality .

❑ Components

- ❖ House
- ❖ Apartment
- ❖ Castle
- ❖ Trees
- ❖ Ground
- ❖ Signboard
- ❖ Helicopter
- ❖ Train
- ❖ Aeroplane
- ❖ Person
- ❖ Skybox

❑ Other functionalities

- ❖ Drag and Rotate by mouse or touch screen
- ❖ Zoom in,Zoom out
- ❖ play/ pause button
- ❖ Speed
- ❖ Time of day
- ❖ Translate along X,Y and Z axis
- ❖ Scale along X,Y and Z axis
- ❖ Reset button
- ❖ Examine button
- ❖ Animation

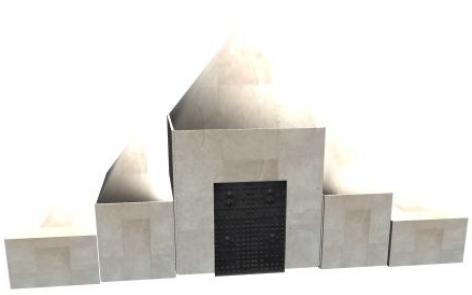
❑ Implementation

Creation of components

We used Hierarchical modeling.We used basic shapes such as sphere, cube, cylinder, plane, cone.They were provided by twgl, a webgl helper library.We also created a pyramid shape using vertex,triangles and normal.The we stored all the shapes in buffers.Then we created complex shapes by combining the basic shape together with rotation and translation.We used texture on each component .For texture we use images .We resized them and then converted to base64 string.After creating the components we placed them in appropriate position in canvas .

Here we explained some components and how they are created

- ❖ House,Apartment ,Castle



We used cubes,pyramids to create these shapes. For the smoke we use spheres and apply translation to look them moving.

- Hut.

It's a combination of a cylinder and a cone.

- Trees:

They are a combination of cones and cylinders.

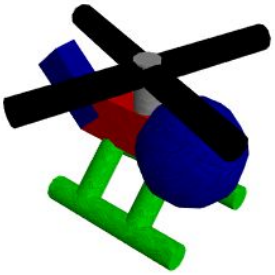
- Person:

We used one sphere for head,one cube for the body and the hands and legs are also cube.To move hands and legs, we use rotation in range 70 degree to -70 degree.

- Road:

It is a cube.

- Helicopter and Plane.

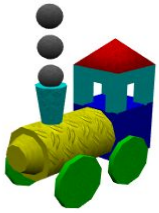


We used cylinders,spheres,cones and cubes to create the shape.The fans are at a fixed distance from each other and they are rotated in the same amount for each frame.

The helicopter will move in a spiral trajectory. For creating this trajectory we used circular movement about y axis and translation in both upward and downward direction along y axis.

The plane will move in a trajectory that looks like the infinity symbol ∞ .We created this trajectory using 2 hermite curves.

- Train



It is created using cubes,cylinders,truncated cones.For the smoke, we used the same technique as in the house. The train will rotate in a circular path about y axis.

- Skybox:

It's the whole Universe.Its is created using 6 planes,which was provided by twgl.

❑ Explanation of other functionalities

- ❖ Drag and Rotate by mouse or touch screen

First we calculate degree per pixel along X and along Y axis.Then we store the position at which mouse button is pressed (or the dragging start).We considered this point as some initial point.When mouse is moved ,we calculate distance between the initial position and new position along x and Y axis.We multiplied these values with degrees per pixel values to find degree of rotation along x and y axis.Then we create the rotational matrix and multiply with view matrix to apply the rotation.Then we consider the new point as initial point.

- ❖ Zoom in /Zoom out:

When the mouse wheel is down we increase the z component in view matrix and when wheel is up decrease z component in view matrix.

- ❖ Play/ Pause button

We have a boolean global variable which when true rotation is applied, else rotation will not apply. This button set the value of this variable to true and false.

- ❖ Speed:

If we increase speed, speed of animation will increase,if we decrease it,animation speed decreases.

- ❖ Time of day:

We implemented a light source in our project.Changing time of day will move the light source along the x axis.

- ❖ Translate along x,y, z axis:

We implemented it by translating our view matrix along the x,y,z.

- ❖ Scaling along x,y,z axis:

We implemented it by scaling our view matrix along the x,y,z.

- ❖ Reset button:

It will set translation and scaling along x,y,z to an initial predefined value.

- ❖ Examine button

If the button is checked then we can examine each individual component separately.To implement this we used a boolean value,if value is true ,the draw function will draw the selected component from the drop down list,else the city will be drawn.

- ❖ Animation:

We used window.requestAnimationFrame(draw) function for the animation.

We created 10 javascript files and one HTML file.We defined the shaders,canvas in and the components outside the canvas in the html file.And in the javascript file we wrote code for creating components ,rendering and all other features.

❑ Difficulties and challenges

- ❖ The first challenge for us was to create and write code for each complex shapes.We overcome this by using hierarchical modelling.It allowed us to create complex shapes by combining basic shapes.
- ❖ Another difficulty was perfectly positioning and rotating (not for all shapes) the basic shapes to create complex shape.We spend a lot of time on creating these shapes.
- ❖ Positioning components on the canvas was also time consuming.

❑ Further improvements

- ❖ Adding more components.
- ❖ Making the components more realistic.
- ❖ Adding day-night and weather simulation.