Joonyong Rhee

JJR975

Solar System with Moving Camera and Overhead Shading

I have created here a solar system with a star, two planets, and three moons. I have added two resizing viewports. One of the viewports has perspective projection, and the other has orthographic projection. I have added a ground plane grid, a drawing axis on the ground and one on one of the moons. I have a spacecraft, a cone, a sphere, and a diamond as my 4 multi-color additional shapes to be explored in the world. I have implemented a glass cylinder camera movement approach and an overhead diffuse light.

The user can interact with the model very easily. Using the keyboard, the user can access all the camera movements. The arrow keys move the camera. The left key is a strafe left. The right key is a strafe right. The backward key moves the camera back on a plane parallel to the ground plane. The forward key moves the camera forward, dives if the tilt is pointing downwards, and climbs if the tilt is pointing upwards. The x key rotates the camera view clockwise and the z key rotates the camera view counter clockwise. The shift key tilts the camera view downward and the space bar tilts the camera view upward. The user can also change the speed of rotation of the solar system, spacecraft, and the solar panels by inputting different integers in the input box and pressing submit. Higher values spin the model faster counter clockwise.

I started by designing the two viewports that resize depending on the height and width of the browser, without skewing or distorting the model. I applied a resizing function to draw a new canvas that fits the screen width and then split the viewport by horizontal halves. The viewports can be seen in Figure 1 below.

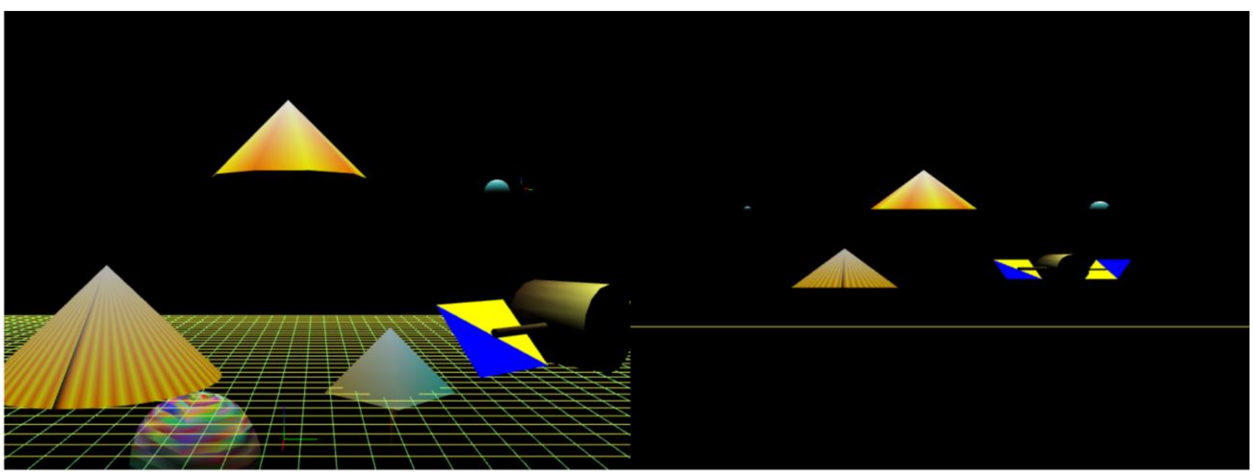


Figure 1: The two viewports split down the middle, fitting the width of the browser screen.

I then worked on setting up one of the views as perspective and the other as orthogonal. I did this using two different implementations of the projection matrix. One I used setPerspective and the other I used setOrtho and then resized the images so they would match the ratio of the canvas width to height. A perspective view shows human perception of the world and the orthographic view does not distort any sizes so shapes are wider and objects do not get smaller as they move farther back. Figure 2 shows how the ground plane and the objects are differing in perspective and orthographic views.

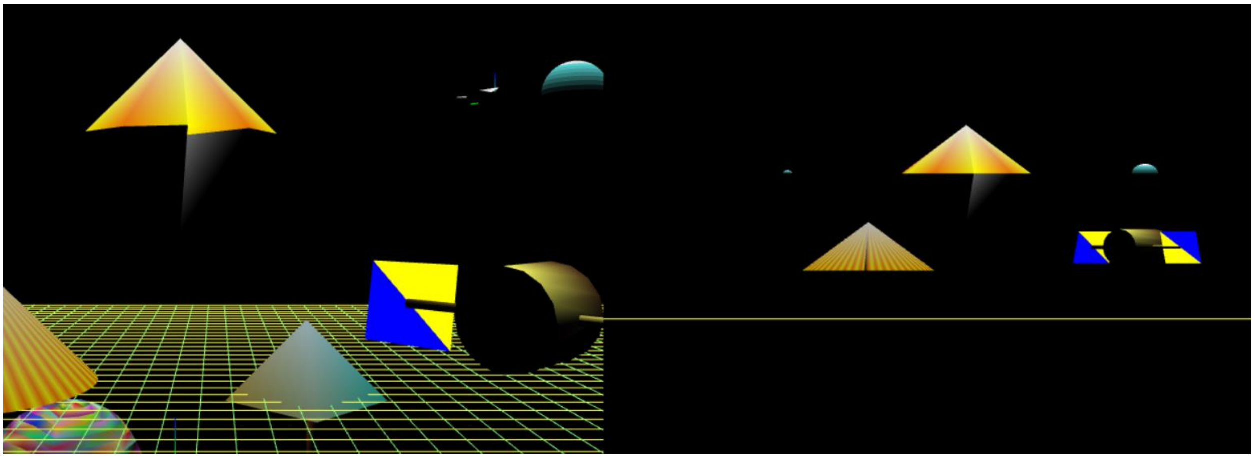


Figure 2: The perspective view on the left shows objects getting larger as they are closer and objects are not as wide as objects in the orthographic projection on the right.

I decided to work on the camera movements next. I used the glass cylinder method to get camera movements to work. I applied distance changes on every position and 'at' coordinates in order to get smooth strafing and translating with the camera. I added cos(theta) and sine(theta) to the horizontal plane vectors to calculate rotation. I kept position the same and changed the 'at' coordinate by a certain distance to calculate tilt. I input in all of these calculations into the lookAt of viewMatrix. See Figure 3 for a different view using camera movement.

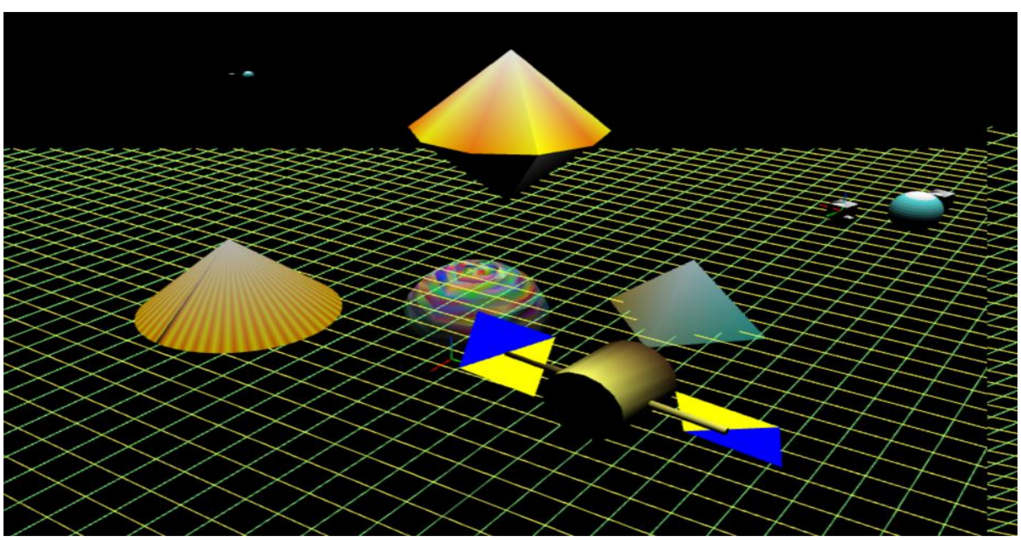


Figure 3: A different view of the model obtained through translating, rotating, and tilting.

I then decided to complete the project with adding a simple diffuse overhead shader. I completed this by first adding the surface normal attributes, light direction vector, and normal matrices. I dotted the normalized normal vector and normalized light direction vector and then clamped it. I then multiplied it with the color attribute in the vertex shader. I increased my floatsPerVertex to 10 to make room for the normal vectors and then proceeded to calculate the surface normal vectors for all of my shapes. See Figure 4 to see how the diffuse overhead light changes the coloration of the object and shades the bottoms.

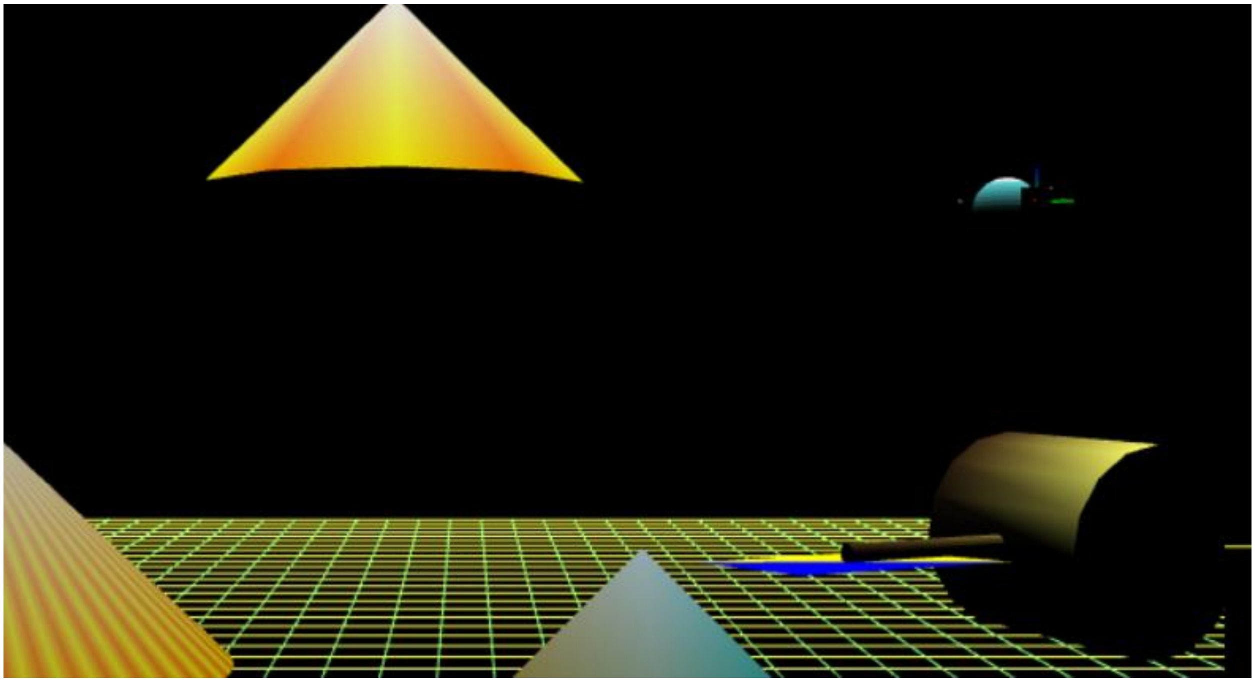
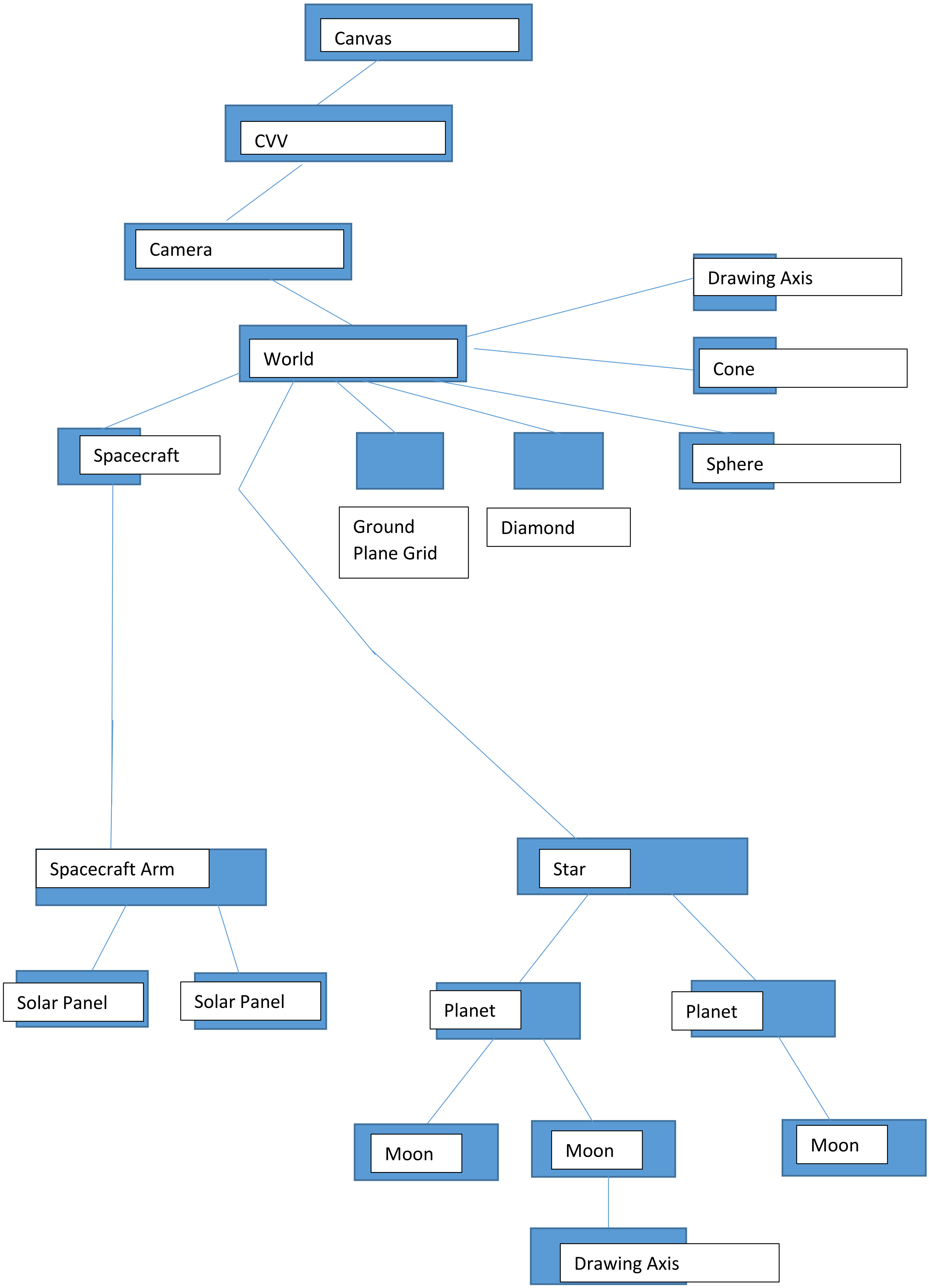


Figure 4: The star, planet, moon, spacecraft, diamond, and part of a cone lit from the top by an overhead diffuse light.

All the shapes I have three different colors and there are drawing axis on the ground and one on one of the moons. Below I have a scene graph with all the nodes as different shapes that come together to form jointed objects through transformations depicted by the lines below the world. At the top is the canvas which leads to the CVV. The CVV is adjusted by the perspective or orthographic projections. The next node is the camera which can be adjusted by the view Matrix. The next node is the world with the ground plane, drawing axis, and shapes.

See Scene Graph 1 for a visual.

Scene Graph 1(below): The scene graph for the model.



Canvas
CVV
Camera
Drawing Axis
World
Cone
Spacecraft
Sphere
Ground
Plane Grid
Diamond
Spacecraft Arm
Star
Solar Panel
Solar Panel
Planet
Planet
Moon
Moon
Moon
Drawing Axis