Project B: Flying through World with Bowing Figure, Crane, and More

Jacob Emmons

NetID: JME9166

User's Guide:

Goals

My main goal in this project was to create a 3D world populated with multiple different animated assemblies, which users could move around freely using keyboard controls. With the assemblies, I specifically wanted to include at least one with 3 separate joints, all of which would have adjustable angles via sliders underneath the canvas, and to include at least one which would integrate quaternion-based mouse rotations, depending on the camera's position and aim. I achieved the former with the crane-shaped assembly with the circular-spinning diamonds on each end, and the latter with the bowing figure made of an isosphere and torus. I further showcased the rotation of the 3-jointed crane with the inclusion of axes on one of the diamonds, making clear to viewers how each joint's rotation affects the drawing axes. I also wanted to showcase color interpolation on all figures in the explorable world. To aid in camera positioning and aiming, I added a gray and white ground grid, as well as a set of world coordinate system axes in the center of the grid. Finally, I sought to split the canvas into two separate viewports, one showcasing a perspective view and the other showcasing an orthographic view, and to make the canvas resize itself to match the size of the browser window it is being run on. It is important to note that the two separate viewports still share the same camera location and aim point, so that users can always clearly see the differences between the two types of views.

Overall, my goals can be summed up as follows:

- To design and draw multiple 3D jointed assemblies and with them:
    - Showcase at least one assembly with 3 sequential rotating joints, the angles of which are adjustable via sliders
        - And showcase the changes in drawing axes at the end of the assembly with a set of axes
    - Showcase at least one assembly that can be rotated with quaternions, triggered with mouse drag movements
    - Have each part of each assembly showcase color interpolation between 3 different colors
- To allow users to freely move the camera around a 3D world in which the assemblies are placed, aided by a ground grid and 3D world coordinate axes
- To split the canvas into two separate viewports, one of which showing a perspective view and the other an orthographic view, and to have the canvas automatically resize to fill the browser window

User Interaction

The keyboard can be used to interact with the 3D world in the following ways

- W and S keys move the camera directly forward in the direction it is looking and directly backward from the direction it is looking, respectively
- A and D keys move the camera directly left from the direction it is looking and directly right from the direction it is looking, respectively
- Up and Down arrows tilt the camera looking direction up and down, respectively
- Right and Left arrows move where the camera is looking directly right and left, respectively

Additionally, the 3 sliders directly underneath the canvas affect the angles of the assembly with the diamonds on its ends in the following ways. Please note that the joints are listed in their order from the ground grid up the positive z-axis

- The top slider interacts with the first joint's (the one between the completely vertical arm and the arm directly attached to it) angle by changing its maximum value from 0 degrees on the left to 180 degrees on the right, affecting how far the arm directly attached to the vertical joint can rotate.
- The middle slider interacts with the second joint's (the one between the two joints in between the vertical arm and the arm with the diamonds) angle by changing its maximum value to a desired one between 0 degrees on the left and 180 degrees on the right, affecting how far the arm connected to the one with the diamonds can rotate.
- The bottom slider interacts with the third joint's (the one between the arm with the diamonds and the arm directly attached to it) angle by changing its maximum value to a desired one between 0 degrees on the left and 180 degrees on the right, affecting how much the arm with the diamonds can rotate.

The drawing is also affected by mouse drags on the canvas

- Dragging the mouse in the x or -x direction (horizontally across the canvas) rotates the bowing torus on top of the isosphere around the z axis, the direction of the rotation being determined by whether the drag is left-to-right or right-to-left
- Dragging the mouse in the y or -y direction (vertically across the canvas) rotates the bowing torus about the axis perpendicular to the camera's viewing direction and the world z axis, the direction of the rotation being determined by whether the drag is top-to-bottom or bottom-to-top

Results:

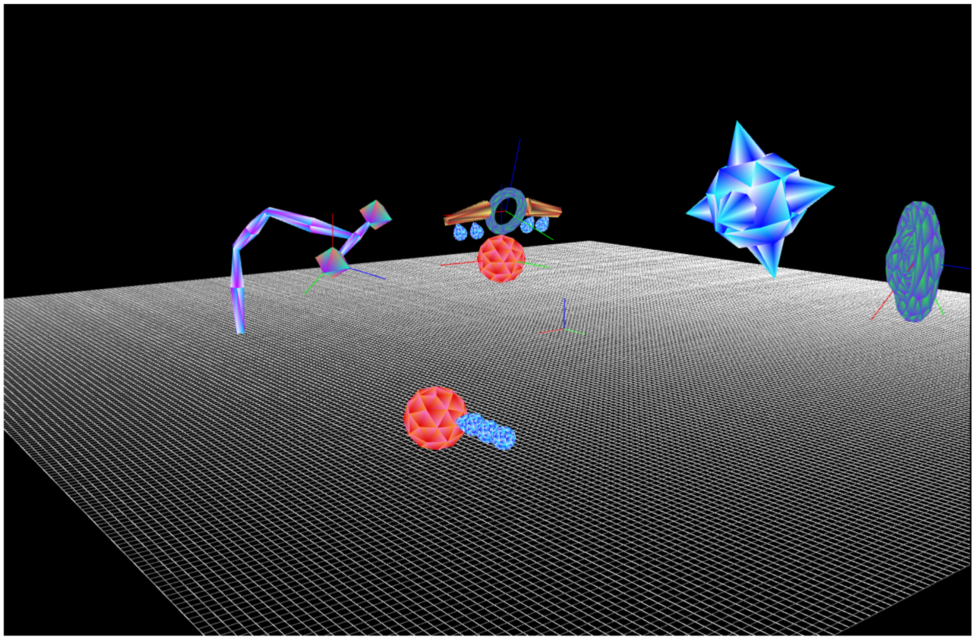

Figure 1: Perspective Viewport

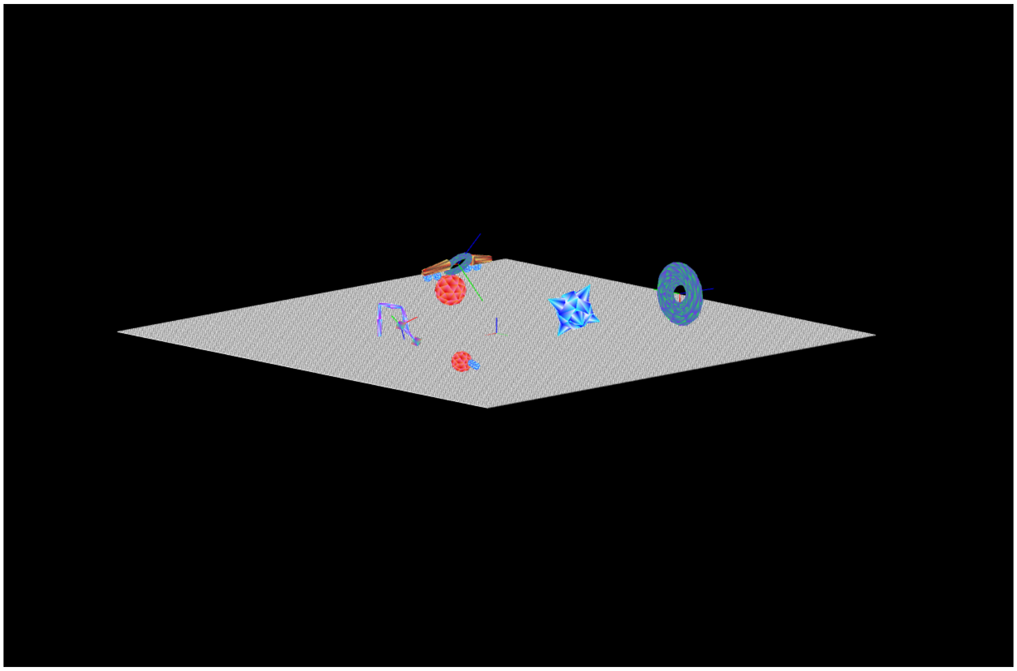

Figure 2: Orthographic Viewport

Figures 1 and 2 show the perspective viewport and the orthographic viewport respectively, which appear side by side when running the program in a web browser. As can be seen in the figures, the two share a camera position despite being different types of views, and the orthographic viewport is defined as the -z/3 plane of the frustum from the perspective view. In these figures, the camera is slightly further back than its default value, but still at the same place, as camera movement is shared between the two viewports

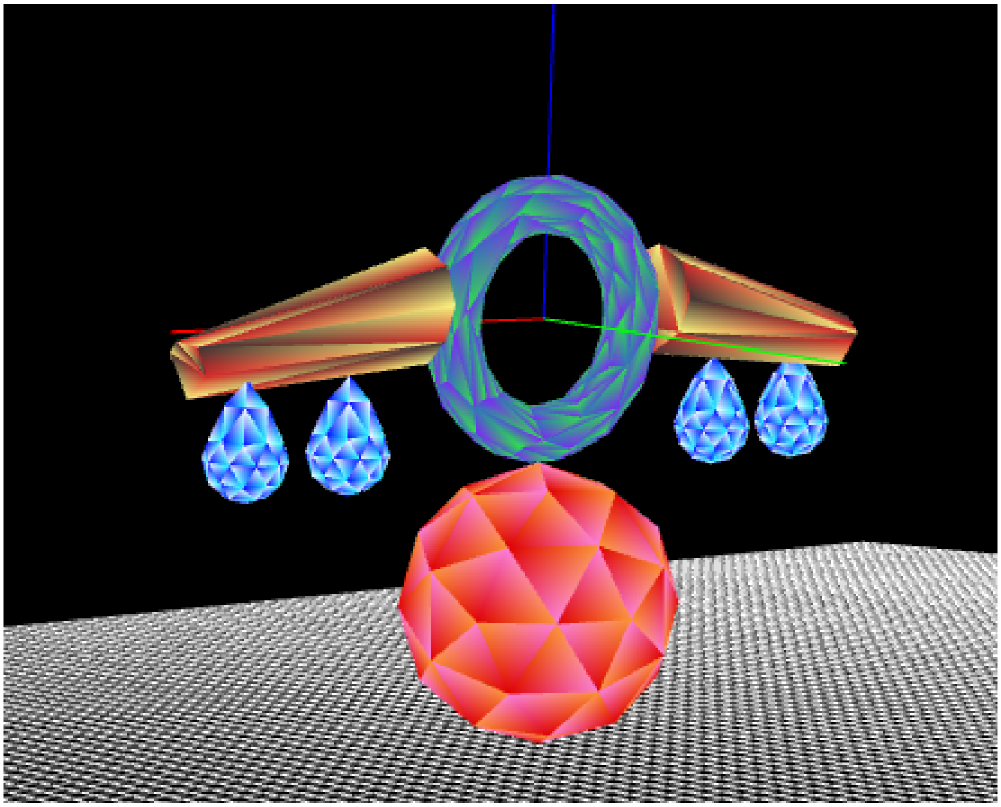

Figure 3: Bowing Figure without any rotation

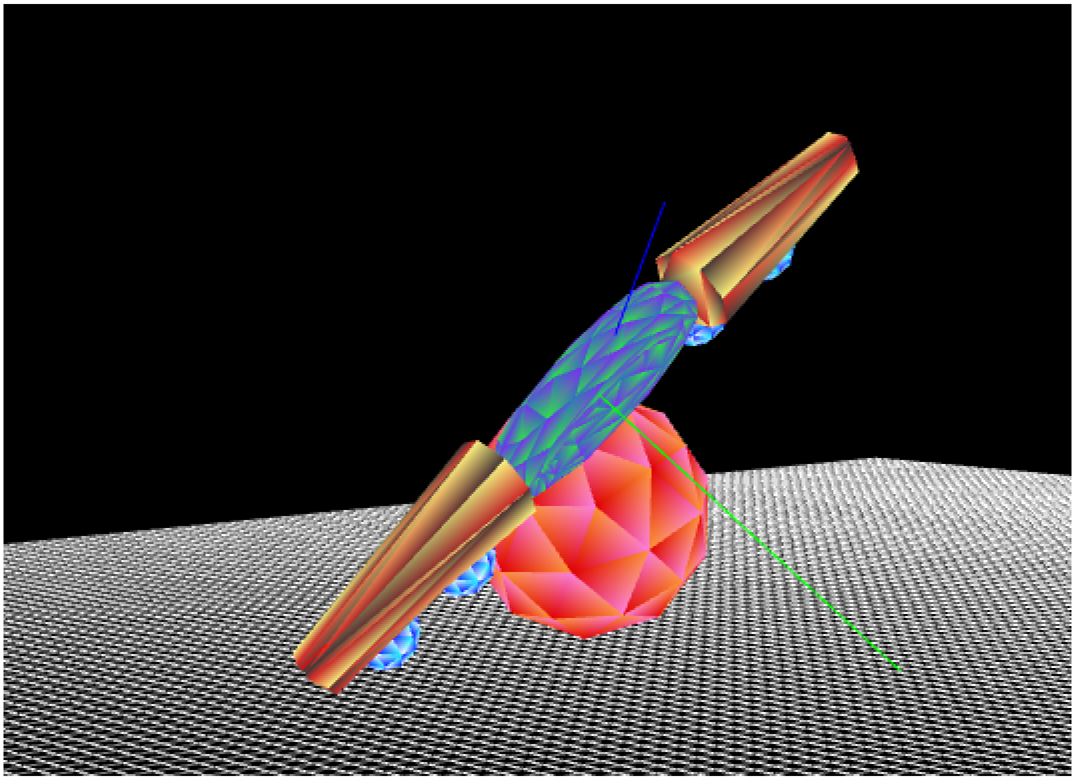

Figure 4: Bowing Figure after dragging in negative y direction

Figures 3 and 4 show the bowing figure before and after rotation, respectively. Figure 3 shows the figure as it is originally drawn, i.e. no drags have taken place. As can be seen, it is standing straight up, reflected by the completely vertical blue axis, which represents the world z axis. Figure 4 shows the figure after a drag in the negative y direction. After the drag, the figure has rotated toward the camera about the axis perpendicular to the z axis and the camera eye-to-aim vector.