Assignment 1

Aman Kumar
IMT2018006
IIIT Bangalore
Bangalore, India
aman.kumar@iiitb.org

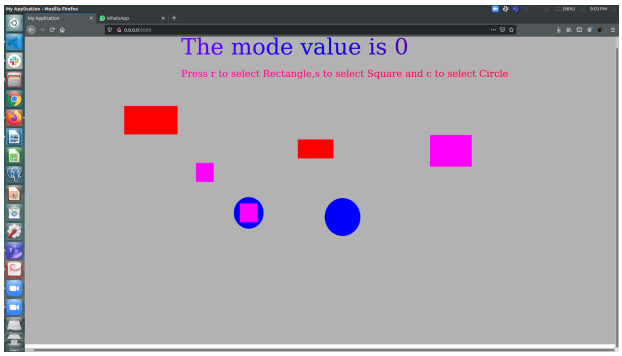
***Abstract*—2D interactive planar rendering with 2D translation, rotation, scaling/zooming.**

I. Introduction

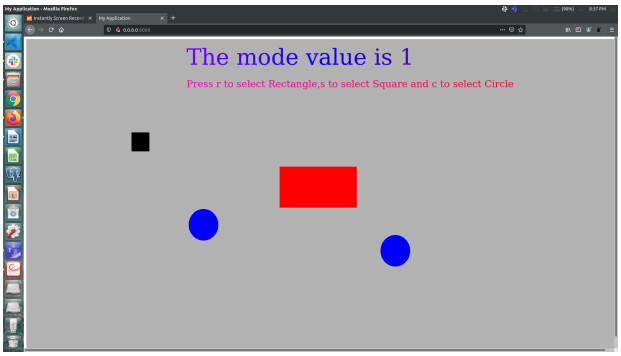
We have 3 modes. In mode==0 we are drawing the primitives.In mode==1 we are translating and scaling the primitives.In mode==2 we are rotating the bounding box and all the contents inside it.

II. Related Work

I have maintained seperate classes for Square and Circle.I am using Square class to make Rectangles.For every primitive shape I am storing the centroid of the object which I determine by the mouse click.I am having this.x and this.y as changing centroids which change when I am translating.The 4 coordinates of the square and rectangle(top-left,top-right,bottom-left and bottom-right) are determined with the help of this.x and this.y.For circle I am generating points as we move this.x and this.y in translation.



I am having this.size for each primitive and which helps in determining the 4 coordinates of squares and rectangles for zooming in and out. Also for circle, I am using this to increase the radius.Here I am not changing the centroid.

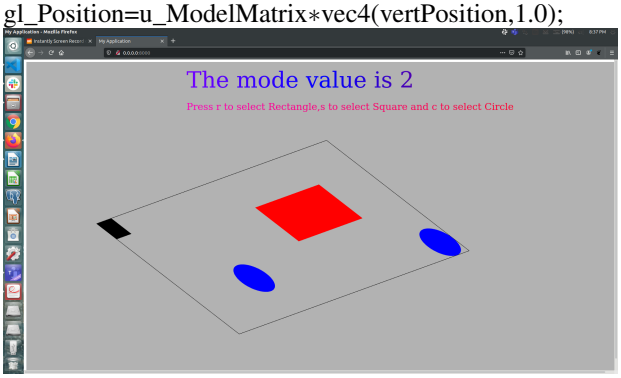


In mode=1 for translation, I am shifting this.x and this.y for translation.I am changing this.size for scaling.

In mode=2 for rotation I am calculating the bounding box by taking the max and min of all 4 box coordinates for squares and rectangles.For circles I am taking the 4 coordinates which are at 12,3,6,9 o clock w.r.t the centroid.

For rotating I am rotating the bounding box and all the shapes in it by using this.modelMatrix.I have a list which stores all objects and I have a drawShape() function for all primitives to draw them on screen.

mat4.identity(this.modelMatrix);
mat4.rotate(this.modelMatrix,this.modelMatrix,this.rotationAngle, this.rotationAxis);
gl_Position=u_ModelMatrix∗vec4(vertPosition,1.0);



I am using the above code for rotation.Here u_modelMatrix is this.modelMatrix.This works for all primitives as gl_Position coordinates change as we multiply by the model matrix.Centroid plays no role here.Even for the bounding box same logic applies.

I am using window.addEventListener() API for detecting clicks and keyboard events.Then I am taking particular action as mentioned in the assignment.For picking I am taking the L2 norm of my mouse click coordinates and object centroid.I am storing the index of the minimum L2 norm object.I am changing it to black and performing suitable actions.

Centroid is important because when we translate or rotate,the coordinates of object change and I am generating the 4 coordinates of square and rectangle based on the centroid.I am calling updateVertex() function whenever drawShape() function is called.Also for rotating since we change the modelMatrix,the centroid is not affected and when we change from mode=2 to mode=1 the centroid gets the original primitive back as rotationAngle is set to 0 and modelMatrix is Identity Matrix.

III. Conclusion

I have learnt a great deal of stuff using the assignment.I learnt the basic functionality of webgl and javascript and some important functions in-built in javascript when doing the assignment.I could apply the concepts taught in class to code them in this assignment.

Acknowledgment

I would like to thank the professors for teaching me a great part of theory which I could implement in this assignment.Thanks to TA's for helping me with issues which I had during the assignment.

References

I used stackoverflow and other sites as provided in the slack channel whenever I was stuck in some part.Most of the part where I was stuck was syntax related which I could figure it out using the console in inspect element and using some youtube videos or links as mentioned below.

Some of the links are

https://www.youtube.com/watch?v=KdyvizaygyY

https://www.youtube.com/watch?v=S0QZJgNTtEw

https://webglfundamentals.org/webgl/lessons/webgl-indexed-vertices.html

https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS