**Project 8 Part 3 Rotating a platonic solid with OpenCV (perspective)**

Name: Ramya Reddy Period: 4 Date: 5/25/22

Did you name your file l083.cpp (Lower case L, then 083)? Yes

Does your file compile & run on terminals? Yes

Did you use a rotation matrix? Yes

Did you start from the coordinates I provided for the cube?? Yes

Describe here in words all the transformations you applied to vertices, for each describe how you implemented it in your code (by multiplying with a matrix, what was the matrix, or by adding a matrix, what was that matrix… be specific):

I initially scaled the vertices by a factor. I then rotated the vertices by multiplying them with a rotation matrix. I applied the perspective transformation using the vector projections and systems of equations we talked about in class, and then translated the vertices so the shape would be centered in the middle of the screen by multiplying it with a translation matrix.

Describe in words the rotation you did:

I iteratively rotated the vertices by one degree in the X and Y axes and did nothing to the coordinates of the Z axis.

Did you use homogenous coordinates? Yes

(that allows you to combine all transformations into one matrix)

Did you combine all those transformations into one single matrix? No

Did you do a perspective rendering? Yes

What is the position of the eye you used? e = (800, 50, 123)

What is the plane of the screen you projected on? a = (500, 300, 200), n = (1, 2, 3)

Did you name your video rotation.avi? Yes

What functions/methods from OpenCV did you use?

VideoWriter: creates and write to AVI file

VideoWriter.write(): writes instance of Mat object to VideoWriter object

Mat: creates matrix object to make frames for video

Mat.t(): transposes matrix object

line(): draws line on Mat object

Scalar: creates tuple of 3 numbers

Vec3f: creates a vector of 3 floats

Vecf.dot(): does dot product between two Vec3f objects

What functions/methods from OpenCV did you experiment with but ended not using?

I ended up using all the OpenCV functions/methods that I experimented with.

Obs.: feel free to rotate any platonic solid, around any line, and you may put the position of the screen/viewing window in any place as long as the rotating platonic solid can be seen reasonably.