**Reflection Essay**

**1. Introducing AR**

**Algorithm:**

1. Read images and then calibrate camera to get instrinsics and extrinsics of the camera matrix.
2. Next get the world coordinates of the 8 corners of the book.

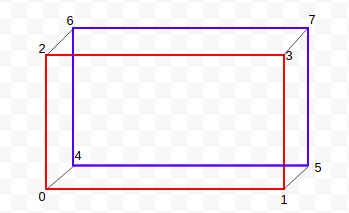
a. Fixed position of the book:

With user input length, breadth and width of the book, fix the position of the book such that book binding is parallel to the horizontal.

b. Position of the book with given user input data:

From user we would get the relative position of the center of the book (x,y) and orientation (theta); angle made by book binding with horizontal axis. By taking the co-ordinates axis of size 8\*8, we would get the co-ordinates of the center of the book as x = 8\*x/100, y = 8\*y/100. Considering the initial coordinates (C=(x,y,z)) of center of the book be (0,0,0) and the co-ordinates of the corner according to length, breadth, and width of the book (which are also input from the user). The transformed co-ordinates of the book would be C’ = R\*(C,1) = R\*(x,y,z,1) where R =.

1. Calculate the respective pixel co-ordinates of the book using the camera matrix calculated.
2. Using these pixel co-ordinates draw outline of the book on a given image (view).
3. ***Realism:*** To make it more real we stick texture of the front side with given images and left, right, top with white color. In order to do this we need to know two things. First, for a given view/orientation which side will be visible and which side isn’t. Second we need to apply transformation on the texture images to place them, such that they match position and orientation.

Let us consider cuboid shown below. Here 0123 represents front face and 4567 represents back face, which is sticked to the wall. The front face will be visible until either of the parallel lines doesn’t coincide. A particular face (let say 2367) other than front face will be visible when the line which is common to the back face (67) and the line (01) parallel to the common edge (23) to the front face lie on the opposite side of the common edge and vice-versa. In the given diagram, side 0246 will not be visible as the lines 46 and 13 lies on the same side of the common edge 02. Whereas, side 1357 will be visible as the lines 57 and 02 lies on the opposite side of the common edge 13.

Once we know which side will be visible and which is not, we need to transform texture images such that they fit in the border of the side correctly.

**Q1.1 Have the choices of views been made so that all the surfaces (except the back-cover) visible in at least one view?**

*Solution:* Yes, we have taken four views of the wall top, bottom, front, and left. The right side of the book would be visible either in top or front view.

**Q1.2 How did you texture the book in the various views? If we were to provide you with a new**

**view, or rotate the book by 90 degrees will your code be able to render the final result correctly?**

*Solution:* Yes we would get the final result correctly for all different orientation of the book. The procedure we have used to obtained this have been explained in step 5 of the Algorithm.