1. Read the aruco marker in the first frame and make sure all 4 markers are read.
2. Use the aruco markers to find position of the table on the image.
3. Use the corners of the aruco marker as points to find homograph and perform warp perspective to orient the image to the user.
4. Perform Gausian blur.
5. Convert image from BGR to HSV.
6. Create a HSV mask using the lower and upper threshold for the table color.
7. Create an ellipse kernal and perform opening and closing on the mask with kernal.
8. Perform thresholding using the mask on the image and type as THRESH\_BINARY\_INV and create a binary\_img.
9. Find contours using the binary\_img.
10. Find the radius and center of the contours using minEnclosingcircle and pick only the contours that have radius in the range of 60-100 and area in the range 8000 – 25000.
11. Find the average color in area of the contour and apply the average color in a copied image.
12. Take the color at the center of the averaged image and take the predefined hsv value of each color and find the normalized distance between then when the distance is below 40, we have found the ball with that color.
13. At first read note down the positions of each ball.
14. After that calculate the distance between noted location and current position of the ball and if the distance is above 100 add the x,y coordinates to the corresponding list.
15. When the length of the list of each colored ball is greater than 1 draw lines between the points in the list we build before with the balls averaged color.
16. Display the updated frame and write it to the output video file.