

Wyatt Colburn
December 3, 2024
EE 428-Computer Vision
Homework 4

Explanation:

A lot of code was already given to us, and I looked at the structure of the code. The main loop handles the argument passing and retrieves the intrinsic values of the camera, as well as the ball's radius. A MagicWand object is created, and then for every frame in the video the `wand.process_frame` is ran on that frame. The `process_frame` method showed me which functions I needed to fill out.

First, I turned the frame into a gray scale and then used `canny` to get the edges. Then using a Hough transform I was able to detect the circle, I used the range of `(40,55,2)` to get a pretty similar radius to that of the ball in the red video. This function returns the center `x`, center `y` and the radius. I used this source to help:

[Circular and Elliptical Hough Transforms — skimage 0.24.0 documentation](#)

Then using the camera project equations to get the real world coordinates. Then I had all the values to draw the ball, most of the function (`draw_ball`) was provided. I add one line to display the subplot on the image. Then again used some projection equations to return the 2D location of the projection in the image. Then used the `project` method to get `x1,x2,y1,y2` to have coordinates to draw the line. Then the function provided `draw_bounding_cube` was provided and did most of the heavy lifting.

Discussion Questions:

1. Does the ball detector seem accurate? Explain when it fails to detect the ball and hypothesize why it fails.

The ball detector is accurate. I had to tune my hough transformation a couple times, and I found adding more "`total_num_peaks`" allowed me to see more of what the system was doing when the box was not just only on the ball. It seems like the box will sometimes detect the ball holder as it is also circular. I assume there would be even better performance if the ball was on top of something rectangular.

2. Would it be possible to correctly rotate the 3D box according to the ball's orientation? Why or why not? If not, how could we physically modify the magic wand so that we could calculate the correct rotation of the box

I do not think it would be possible. I don't think you are getting enough data from a uniform ball. One approach would be to add markers on each orientation, to provide ourselves with more information. You would be able to detect the orientation if there were degree marking, based on

the values to the left and right of the center, it should be novel to calculate your orientation.
Perhaps you would need also coordinates not just on one plane but both x,y. So the ball would have coordinates like this:

 y1
 y2
 y3
 y4
X1 x2 x3 x4 x5,y5 x6 x7 x8 x9 x10
 y6
 y7
 y8
 y9
 y10