

**CPE 428: Computer Vision**  
**Homework 2: Magic Wand**  
**Instructor: Jonathan Ventura**

In this assignment you will implement a script to detect a colored ball, estimate its 3D position, and draw a 3D cube over the ball. You could imagine using this technique to implement a 3D mouse or a VR game controller.

The intrinsic parameters of the camera are given in the file `iphone_calib.txt` and the radius of the ball is 3 cm. You need to figure out how to calculate the (X,Y,Z) coordinates of the ball using your knowledge of pinhole projection.

Your code must fill in the missing functions in the MagicWand class in `magicwand.py`. These functions will be tested in an automated script during grading, so be sure not to change their names, arguments, or return types. The provided script `tracker.py` is a command-line app to test your MagicWand implementation.

To detect a ball (`detect_ball` function), you should convert the image to grayscale and apply a Gaussian blur of size 9x9 with  $\sigma=2$ . Then run `HoughCircles` with `method=HOUGH_GRADIENT`, `dp=2`, and `minDist=10`.

Modify the operation of `tracker.py` so that it will make a 3D plot of the ball's trajectory at the end of the script. Only record the position of the ball when exactly one ball is detected. Then make a 3D plot of the ball's trajectory using [matplotlib](#).

Once you have the ball tracker implemented and working, inspect the results and answer the following qualitative evaluation questions:

1. Does the ball detector seem accurate? When does it fail to detect the ball and why?
2. Is it 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?

The `wand.mov` video shows a wand with a ball on each end. Use this video to quantitatively measure the accuracy of the system. I measured the distance between the ball centers to be 36 cm. When both circles are detected, calculate the distance between the 3D centers of the two balls. Once all the video frames have been processed, calculate the mean of all the distances calculated and compare to the ground truth measurement given above.

Required libraries:

- `cv2` (OpenCV)
- `numpy`
- `matplotlib`

Links to video files and example results:

- [iphone\\_calib.txt](#)
- [red.mov](#)
- [red.mov output](#)

- [blue.mov](#)
- [blue.mov output](#)
- [redblue.mov](#)
- [wand.mov](#)

Deliverables:

- magicwand.py
- tracker.py
- Text document (plain text, Word doc, or PDF) explaining your solution

Please upload these files individually, not as a ZIP – this makes grading easier.