**Camera Used :** Logitech BRIO 4k Pro WebCam

**Python Version : 3.6**

**Libraries for running the codes:**

scipy

getopt

sys

math

numpy

cv2

glob

datetime

csv

**Switching Camera:**

cap = cv2.VideoCapture(0)

* Search for this line in the code aruco\_tracker.py and aruco\_tracker\_single.py
* 0 here refers to the primary camera of your laptop , if you attach another camera to the laptop for example the Logitech BRIO 4k Pro WebCam it would be detected as a secondary camera.
* So, either you could make Logitech BRIO 4k Pro WebCam as the primary camera through your laptop settings or change

cap = cv2.VideoCapture(0)

to

cap = cv2.VideoCapture(1)

in code .

**Calibration:**

* I have calibrated the code for the “Logitech BRIO 4k Pro WebCam” camera and the images I took were in the resolution 1920 x 1080.
* So, the code also opens the camera in resolution 1920 x 1080 and not 4k.
* I did the calibration and have hardcoded the value in the code, if you are planning to use other camera you will have to do the calibration again. But I would suggest sticking with this camera.

**Converting the python code to a .exe file:**

* For converting the File to .exe, I made use of the python package called pyinstaller.
* You will have to run this in a windows machine to get the exe
* <https://www.geeksforgeeks.org/convert-python-script-to-exe-file/> you can follow this tutorial to convert the code into .exe.

**Running the code or .exe (Tracking code):**

**Note:** This code does correction between the aruco markers center point and tags location on fly.

**Single shot: (aruco\_tracker\_single.py)**

1. Run the .exe/.py file
2. Enter the following parameters

Enter the tag Size: <Tag size in cm>

Enter the file Name<path to file without any extensions> (for eg :/Users/chitti/Desktop/RA/ArucoTrackerPython/pythonCodes/testing)

Input the x-coordinate in cm: < x in cm>

Input the y-coordinate in cm: < y in cm>

Input the z-coordinate in cm: < y in cm>

1. As soon as it detects the tag it will close and save file

**Normal : (aruco\_tracker.py)**

1. Same as steps 1 and 2 in single shot
2. The code runs continuously and takes reading when you want to stop reading press “command + q” it will save the file

**Correction code:**

* The correction.py code is used for doing correction after collecting the data, was using this for the use case were the aruco\_tracker.py was not doing the correction on fly. The current code does it on the fly so you won’t need to run this correction.py code .

Steps to run the code:

1. Run the .py file
2. Enter the following parameters

Enter the tag Size: <Tag size in cm>

Enter the file Name<path to file without any extensions> (for eg :/Users/chitti/Desktop/RA/ArucoTrackerPython/pythonCodes/testing)

Input the x-coordinate in cm: < x in cm>

Input the y-coordinate in cm: < y in cm>

Input the z-coordinate in cm: < y in cm>