Michael D. Saneke

Object Tracking with Computer Vision and EV3 Robot

Introduction

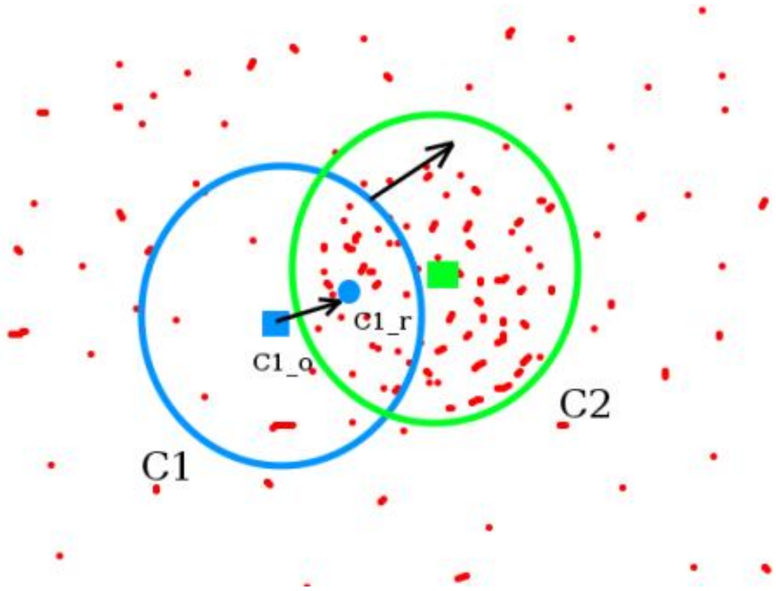
Computer vision is one of the subfields of Robotics that is concerned with giving a robotic construct the ability to sense its environment visually. This sensing is achieved through the use of visual sensors such as a camera. The camera takes visual input from the physical environment which is sent to the robot. Computer vision algorithms are applied to the input data to enable the robot make sense of what is in the physical environment. A robot that is equipped with vision capabilities can perform functions such as navigate a physical environment using sight, detecting and tracking objects in the environment etc.

In this project, the aim is to track an object in a live stream video using the computer vision library OpenCV and a computer vision algorithm known as the **Meanshift tracking algorithm.** As the algorithm tracks the object in a live stream, commands based on the direction of the object will be sent to an EV3 robot that will be made to move in the same direction as the tracked object is moving in real time.

Michael D. Saneke

Theoretical Background

Mean shift tracking in computer vision is derived from a statistics technique known as Mean-Shift clustering. This works on a principle of centroids where the mean of points within a given region known as a centroid is constantly calculated so as to shift the center of the centroid to the mean of the points within that region at a given instance. From the figure below, centroid C1's original center is C1_o, however the mean of the points within C1 are calculated to be C1_r. The center then shifts from C1_o to C1_r and the new centroid becomes C2.



(img source: https://docs.opencv.org/trunk/db/df8/tutorial_py_meanshift.html)

In a paper by Gary R. Bradski about implementing mean-shift algorithm for tracking faces in a perceptual user interface, a histogram of colours in a region of a video frame is used for finding the mean shift of that region. Because the operation of mean shift is based on probability distributions, tracking objects in sequential video frames involves representing the colour data of the region of interest for each individual frame in the video as probability distributions using colour histograms. From this, the mean of the colour histogram constantly calculates and shifts to the new mean per each frame in the video. Steps for calculating the mean shift involves:

1. Choosing a size for the region of interest (where mean shift will be applied to track whatever that is within this region)
2. Choosing the initial position for region of interest
3. Computing the mean location of the region of interest
4. Center the region of interest at the mean location from step 3.
5. Repeat steps 3 and 4 to continuously track the colours within the region of interest

(Bradski, 1998)

Approach

The approach in undertaking this project involved 3 steps:

1. <u>Set-up OpenCV library and run mean shift algorithm</u>

   The first thing I did was to download OpenCV library and set it up on my computer. I then retrieved an algorithm for mean shift tracking from the OpenCV documentation. I then run the algorithm to ensure that it was working.

2. <u>Build on the mean shift algorithm to send positional data of the tracked object to the EV3 robot</u>

   I built on the algorithm to send real-time commands to the robot based on the current position of the object so that the robot would move in the same direction as the object being tracked. This data was sent to the robot using python web sockets.

3. <u>Program EV3 to move in the same direction as the object being tracked in OpenCV</u>

   Wrote a program to be run on the EV3 robot that would be constantly listening on a port for commands from the OpenCV program. These commands would tell the robot which direction to move relative to the object being tracked in OpenCV.

Results

The algorithm was successful in tracking an object in a live stream. The EV3 was also successful in following the direction of the object being tracked in the live stream however the robot was not able to follow the object in real-time as was intended. This is because using web sockets to communicate between OpenCV and the EV3 robot caused “lag” in the process hence there was a delay between the OpenCV program sending the commands and the EV3 robot receiving the commands to act on it.

Challenges

I encountered a few challenges during the undertaking of this project. These challenges were:

- OpenCV library was not working efficiently.
- The mean shift algorithm from the OpenCV documentation did not accurately track objects in a live stream.

Conclusion

I have learnt a lot about computer vision through undertaking this project, I was exposed to computer vision algorithms such as haar cascades for face detection, shi-tomasi for corner detection and mean shift for object tracking. I experienced first-hand the importance of computer vision within the field of robotics. I also learnt how to use python web sockets to communicate between two clients.

References

Bradski, G. R. (1998). Computer vision face tracking for use in a perceptual user interface.

sklearn.cluster.MeanShift — scikit-learn 0.19.1 documentation. (2017). *Scikit-learn.org*. Retrieved 15 December 2017, from http://scikit-learn.org/stable/modules/generated/sklearn.cluster.MeanShift.html

Saneke M. (2017). Object Tracking with Computer Vision and EV3 Robot, Poster