Project Title: People counting and tracking system

Introduction:

Overview:

In this tutorial you have learnt how to build a “people counting and tracking system” with OpenCV and Python. Using OpenCV, we’ll count the number of people who are heading “in” or “out” of any department store in real-time.

In order to build our people counting and tracking system, we’ll need a number of different Python libraries, including:

* [**NumPy**](http://www.numpy.org/)
* [**OpenCV**](https://pyimagesearch.com/opencv-tutorials-resources-guides/)
* [**dlib**](http://dlib.net/)
* [**imutils**](https://github.com/jrosebr1/imutils)

**Object detection and object tracking**

When we apply object detection we are determining *where* in an image/frame an object is. An object detector is also typically more computationally expensive, and therefore slower, than an object tracking algorithm. Examples of object detection algorithms include Haar cascades, HOG + Linear SVM, and deep learning-based object detectors such as Faster R-CNNs, YOLO, and Single Shot Detectors (SSDs).

An object tracker, on the other hand, will accept the input *(x, y)*-coordinates of where an object is in an image and will:

1. Assign a unique ID to that particular object
2. Track the object as it moves around a video stream, *predicting* the new object location in the next frame based on various attributes of the frame (gradient, optical flow, etc.)

**Combining both Object tracking and object detection**

* **Phase 1 — Detecting:** During the detection phase we are running our computationally more expensive object tracker to (1) detect if new objects have entered our view, and (2) see if we can find objects that were “lost” during the tracking phase. For each detected object we create or update an object tracker with the new bounding box coordinates. Since our object detector is more computationally expensive we only run this phase once every *N* frames.
* **Phase 2 — Tracking:** When we are not in the “detecting” phase we are in the “tracking” phase. For each of our detected objects, we create an object tracker to track the object as it moves around the frame. Our object tracker should be faster and more efficient than the object detector. We’ll continue tracking until we’ve reached the *N*-th frame and then re-run our object detector. The entire process then repeats.

**Purpose:**

**Monitoring of high-traffic areas:** Shopping centers use people counters to measure the number of visitors in a given area. People counters also assist in measuring the areas where people tend to congregate. The areas where people tend to gather are often charged higher rents]

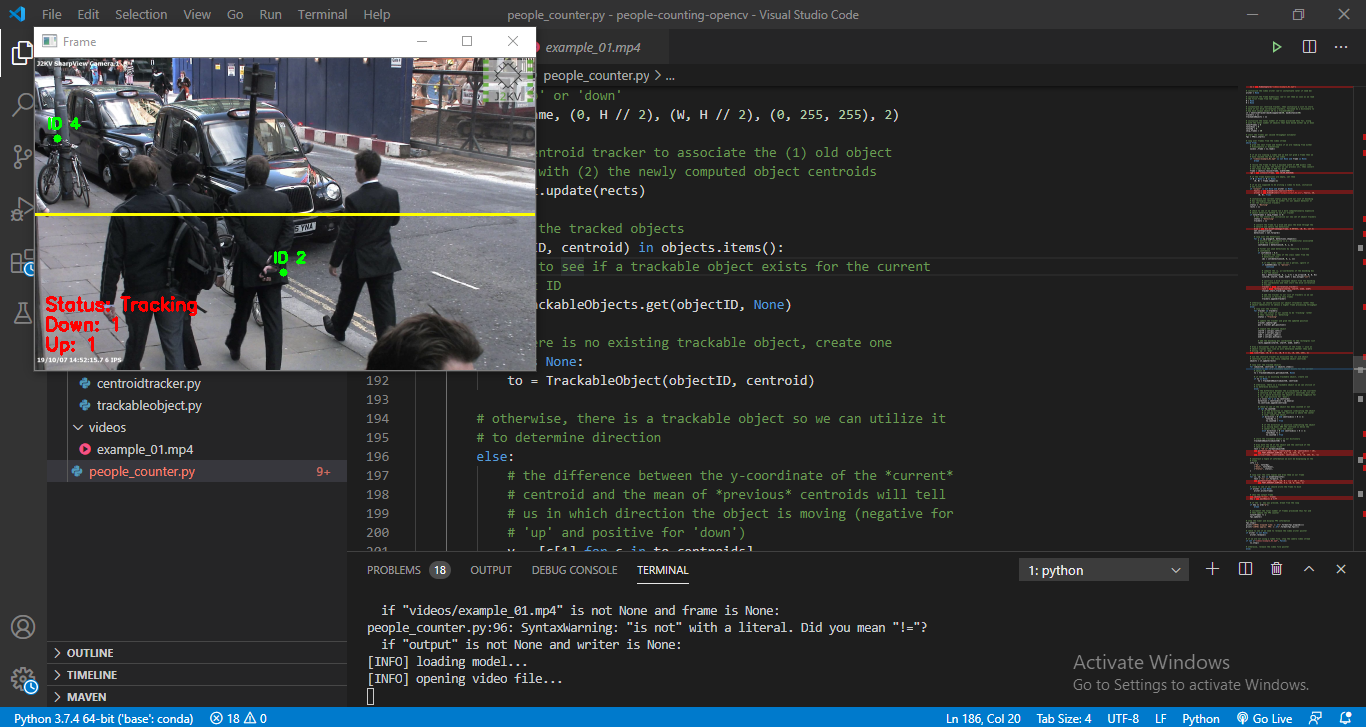
**Determining popularity of particular brands:** Shopping malls prefer to lease space to only the most popular brands. People counters help shopping malls discover popularity by determining footfall patterns and traffic. Shopping mall owners are able to determine the flow of traffic per customer, and which areas and the levels of use of the different mall entrance

**Public transportation**

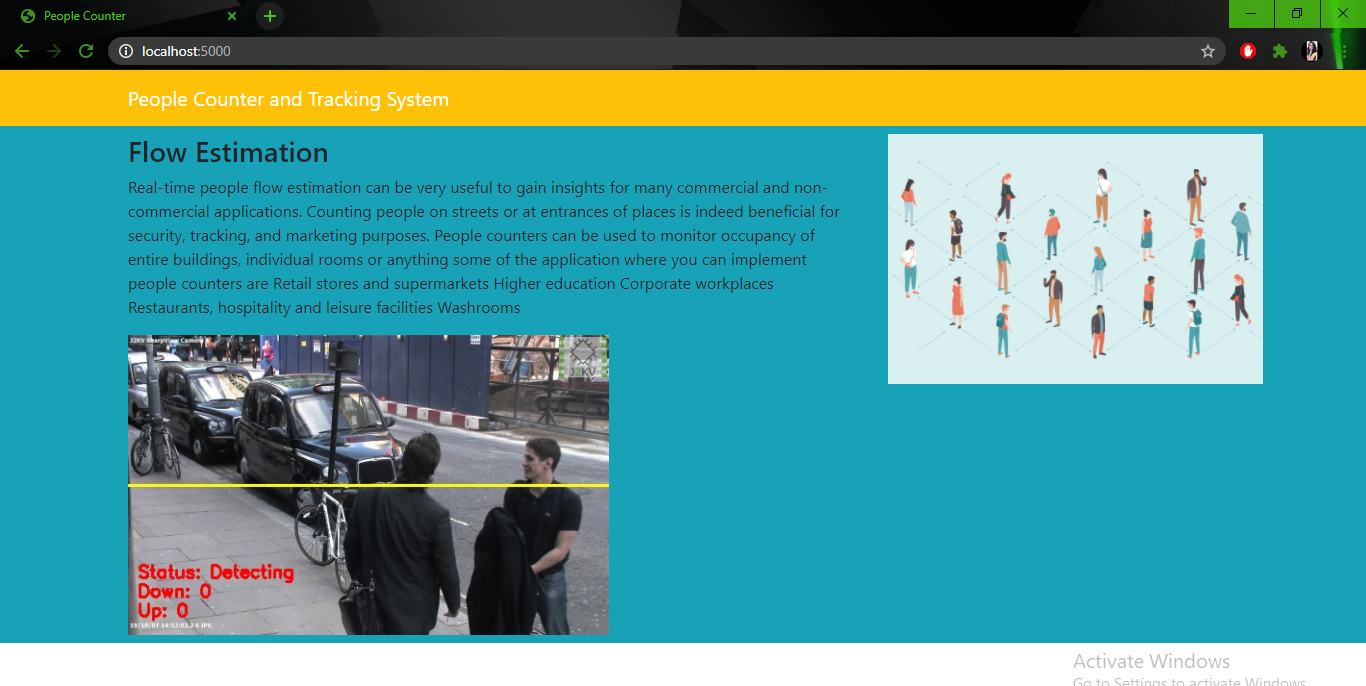
**Occupancy throughout the route** Combining with GPS positioning, people counting system inside a bus or train can help to measure the actual occupancy buses throughout their journey. This would help bus operator to allocate funding and resources to routes.

**Result:**

****

****

****

****

****

**Applications:**

I want to implement in my resort for tracking people in the entrance.

It can also be used for survellience purpose in the malls

It can be used in restaurants,industries and on many commercial and non commercial places where we require to keep the watch on people and analyse there count.

**Conclusion:**

I have successfully implemented and understood the basic understanding of the project with all the packages along with its implementation structure in effective way.

**Future Scope:**

As I have said earlier I will try to implement this project in real time on my resort where we can keep watch of the people in the entrance which will help us a lot in long run.