Object Tracking Using Python

In this project using python and OPENCV module we are detecting objects from videos and webcam. This application consists of two modules such as ‘Browse System Videos’ and ‘Start Webcam Video Tracking’.

Browse System Videos: Using this module application allow user to upload any video from his system and application will connect to that video and start playing it, while playing if application detect any object then it will mark that object with bounding boxes, while playing video if user wants to stop tracking then he need to press ‘q’ key from keyboard to stop video playing.

Start Webcam Video Tracking: Using this module application connect itself with inbuilt system webcam and start video streaming, while streaming if application detect any object then it will surround that object with bounding boxes, while playing press ‘q’ to stop web cam streaming.

To implement this project we are using object tracking algorithms from OPENCV python API.

**What is Object Tracking?**

Simply put, locating an object in successive frames of a video is called tracking.

The definition sounds straight forward but in computer vision and machine learning, tracking is a very broad term that encompasses conceptually similar but technically different ideas. For example, all the following different but related ideas are generally studied under Object Tracking.

**Opencv will use following algorithms to track object in videos**

Dense Optical flow: These algorithms help estimate the motion vector of every pixel in a video frame.

Sparse optical flow: These algorithms, like the Kanade-Lucas-Tomashi (KLT) feature tracker, track the location of a few feature points in an image.

Kalman Filtering: A very popular signal processing algorithm used to predict the location of a moving object based on prior motion information. One of the early applications of this algorithm was missile guidance! Also as mentioned here, “the on-board computer that guided the descent of the Apollo 11 lunar module to the moon had a Kalman filter”.

Meanshift and Camshift: These are algorithms for locating the maxima of a density function. They are also used for tracking.

Single object trackers: In this class of trackers, the first frame is marked using a rectangle to indicate the location of the object we want to track. The object is then tracked in subsequent frames using the tracking algorithm. In most real life applications, these trackers are used in conjunction with an object detector.

Multiple object track finding algorithms: In cases when we have a fast object detector, it makes sense to detect multiple objects in each frame and then run a track finding algorithm that identifies which rectangle in one frame corresponds to a rectangle in the next frame.

Tracking vs Detection

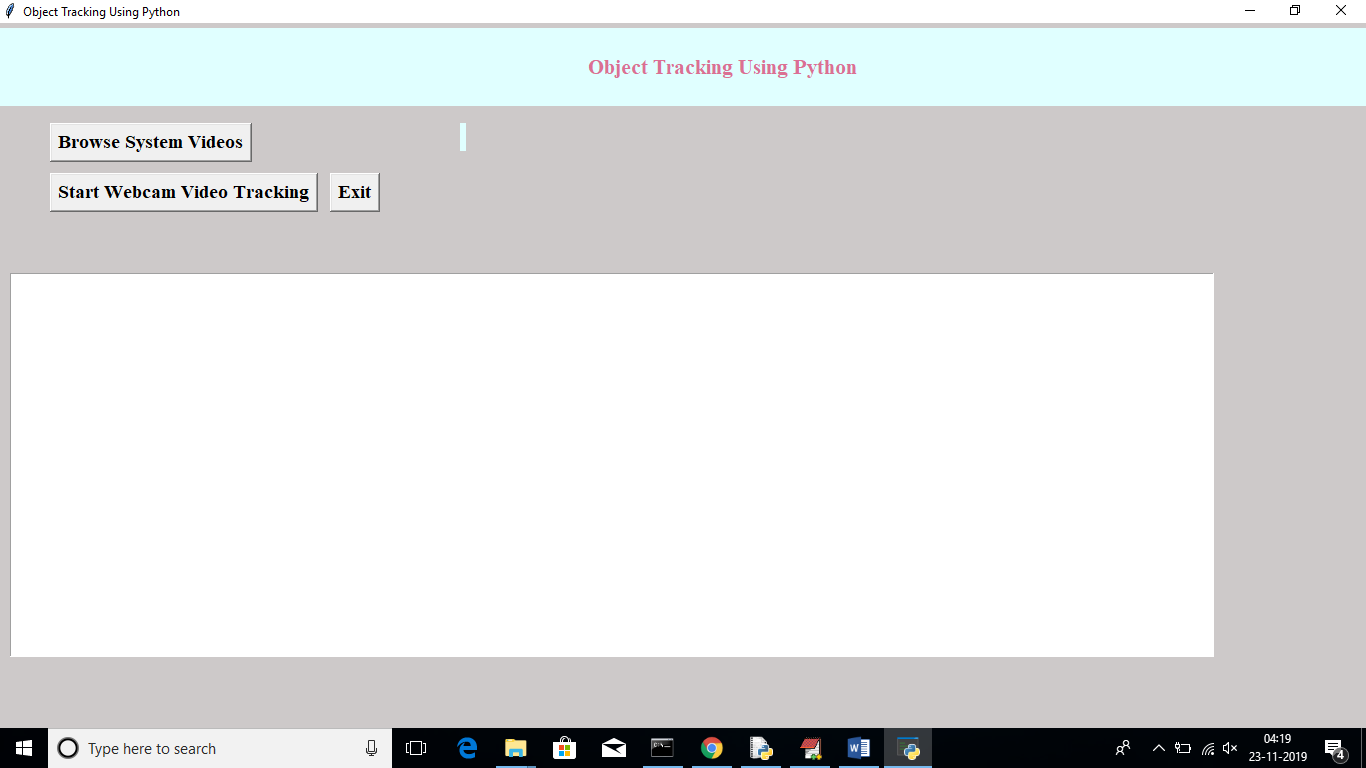
If you have ever played with OpenCV face detection, you know that it works in real time and you can easily detect the face in every frame. So, why do you need tracking in the first place? Let’s explore the different reasons you may want to track objects in a video and not just do repeated detections.

Tracking is faster than Detection: Usually tracking algorithms are faster than detection algorithms. The reason is simple. When you are tracking an object that was detected in the previous frame, you know a lot about the appearance of the object. You also know the location in the previous frame and the direction and speed of its motion. So in the next frame, you can use all this information to predict the location of the object in the next frame and do a small search around the expected location of the object to accurately locate the object. A good tracking algorithm will use all information it has about the object up to that point while a detection algorithm always starts from scratch. Therefore, while designing an efficient system usually an object detection is run on every nth frame while the tracking algorithm is employed in the n-1 frames in between. Why don’t we simply detect the object in the first frame and track subsequently? It is true that tracking benefits from the extra information it has, but you can also lose track of an object when they go behind an obstacle for an extended period of time or if they move so fast that the tracking algorithm cannot catch up. It is also common for tracking algorithms to accumulate errors and the bounding box tracking the object slowly drifts away from the object it is tracking. To fix these problems with tracking algorithms, a detection algorithm is run every so often. Detection algorithms are trained on a large number of examples of the object. They, therefore, have more knowledge about the general class of the object. On the other hand, tracking algorithms know more about the specific instance of the class they are tracking.

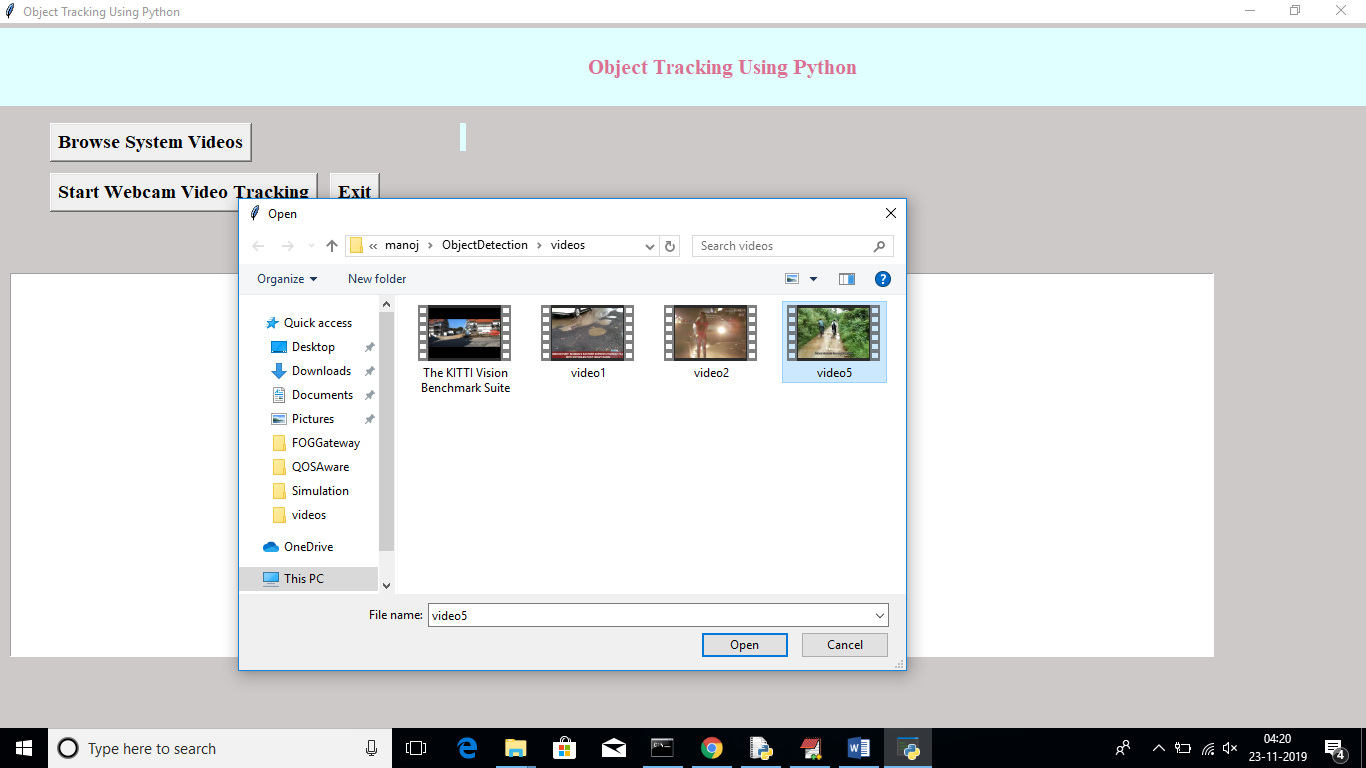
Tracking can help when detection fails: If you are running a face detector on a video and the person’s face get’s occluded by an object, the face detector will most likely fail. A good tracking algorithm, on the other hand, will handle some level of occlusion.

Screen shots

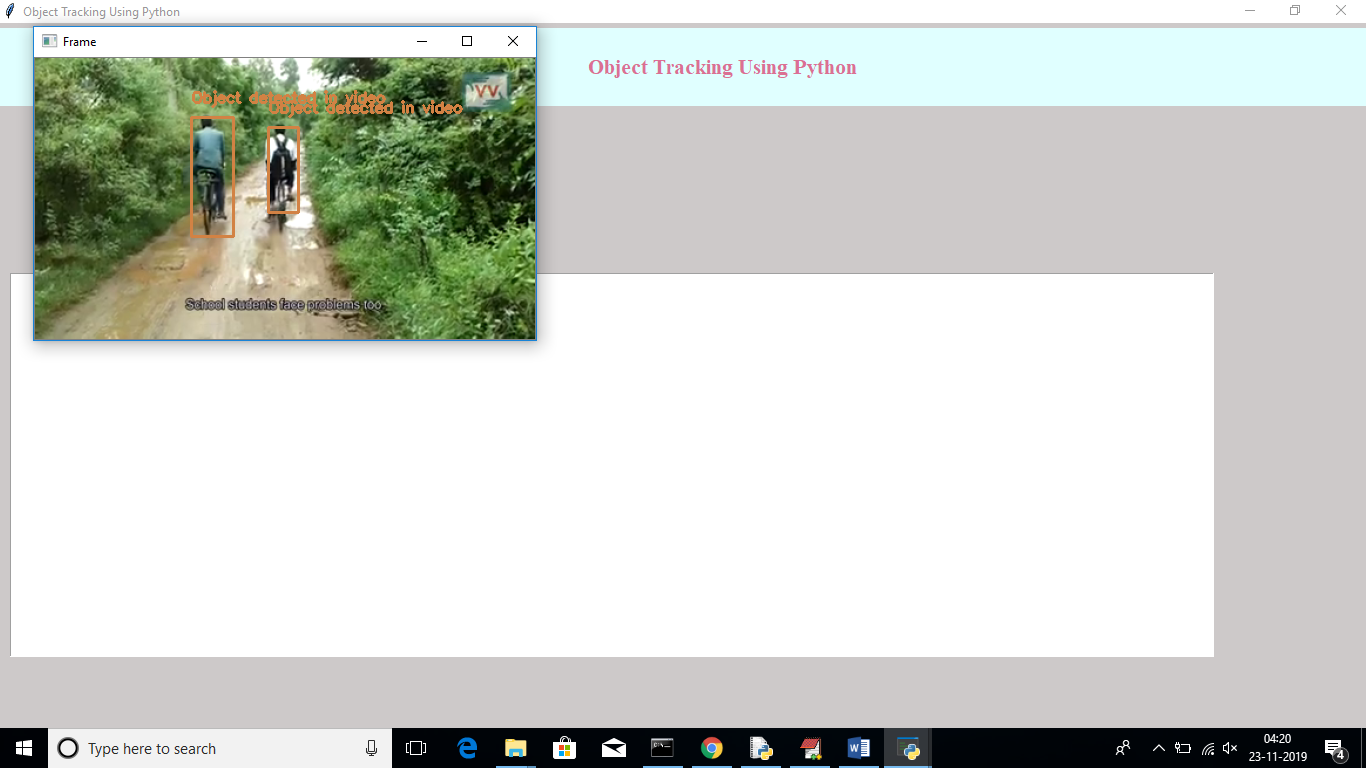
Double click on ‘run.bat’ file to get below screen



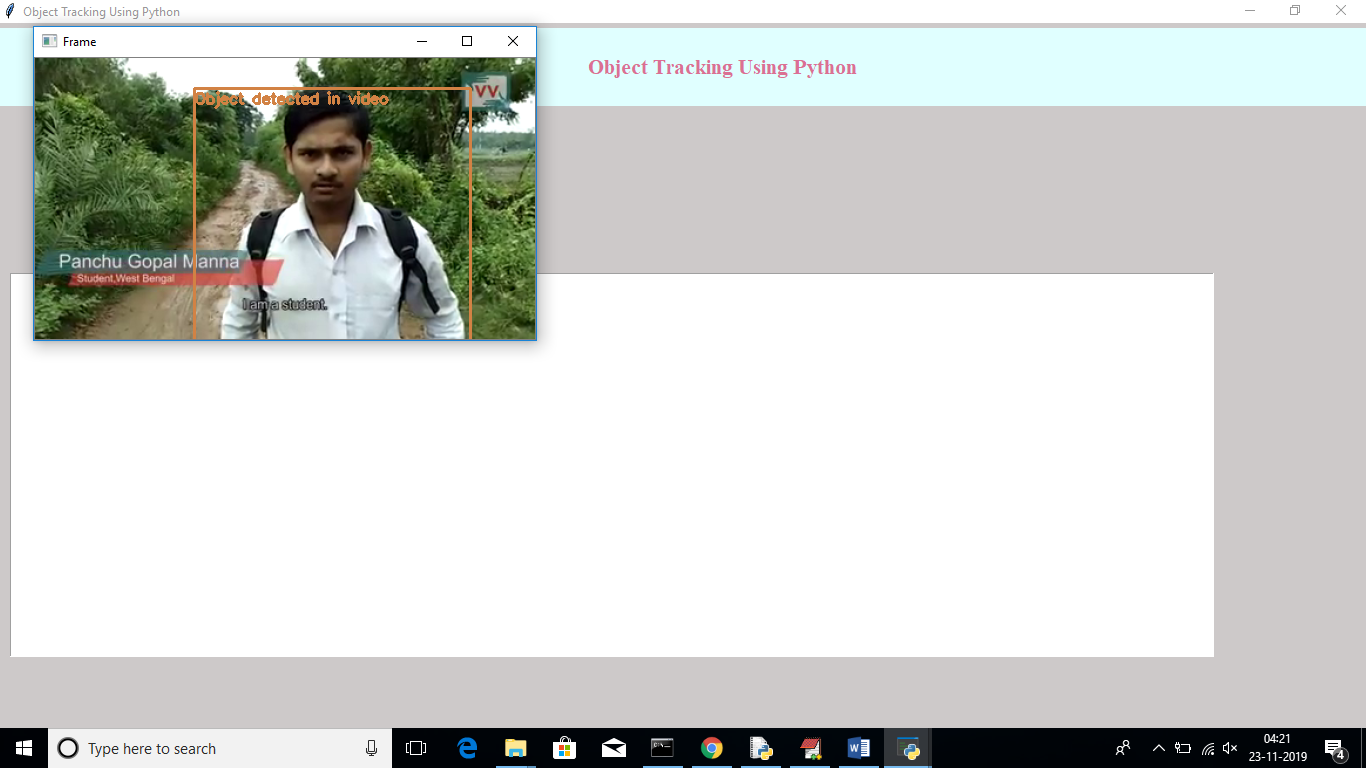
Now click on ‘Browse System Videos’ button to upload videos from system

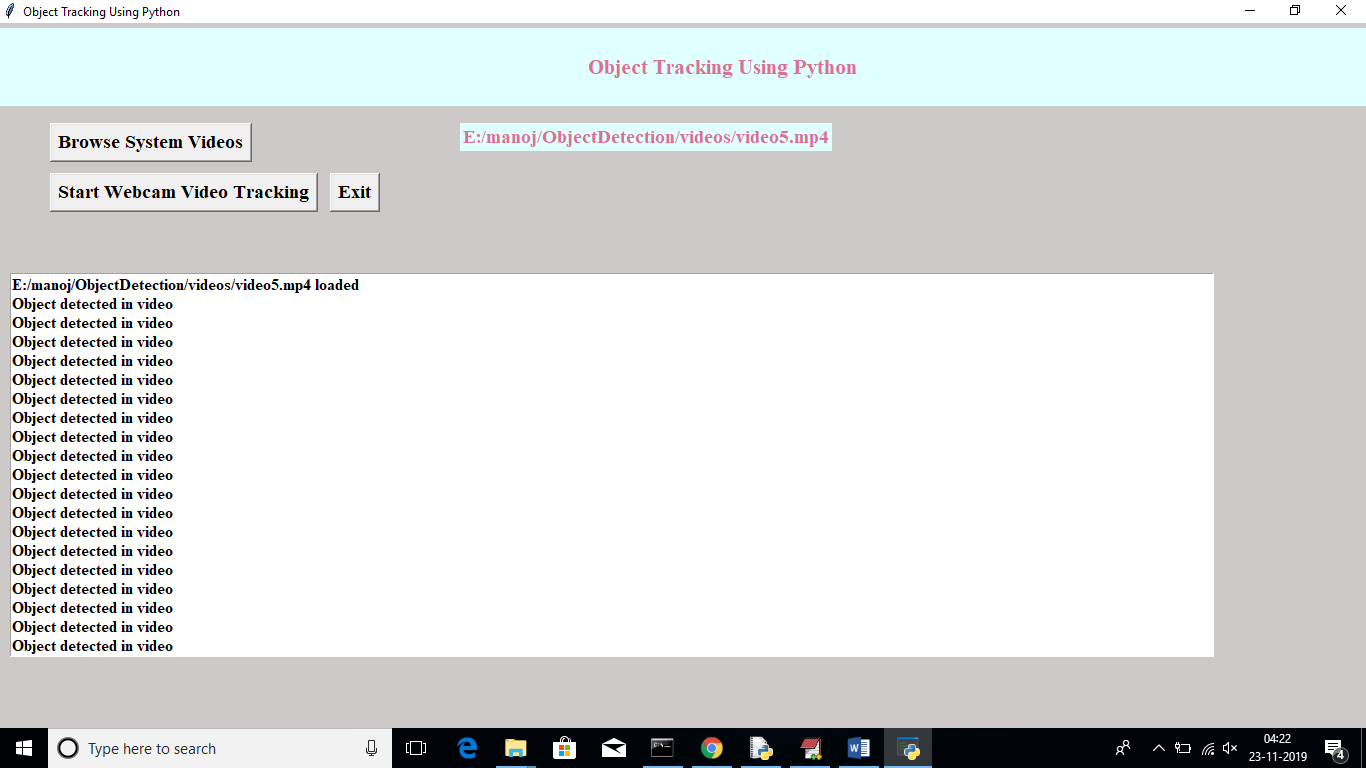


In above screen I am uploading one video, after upload will get below screen

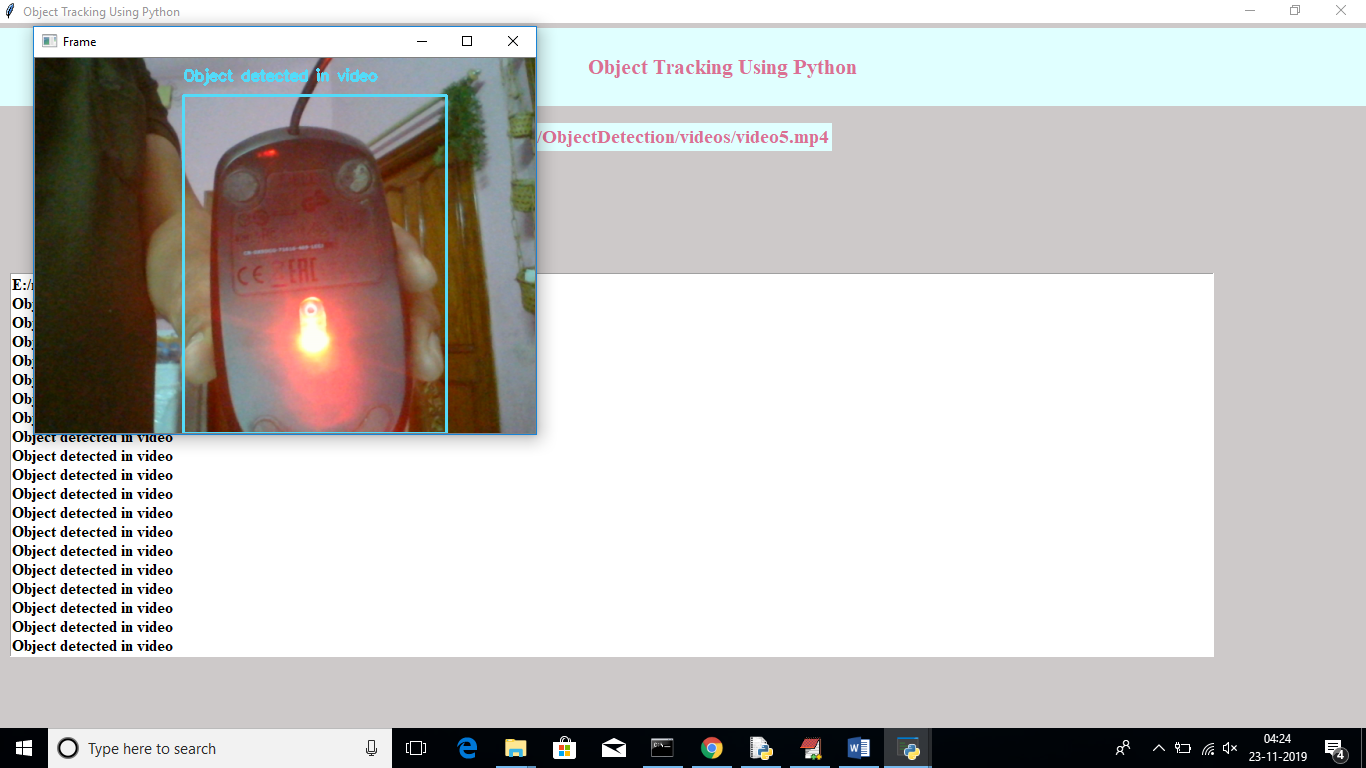


In above video we can see application start tracking objects from video and mark them with bounding boxes. Similarly we can upload any video and track objects from video





In above screen now click on another button called ‘Start Webcam Video Tracking’ to connect application to web cam and start streaming. After connecting to webcam will get below screen



In above screen we can see objects is getting tracked from webcam also. In above screen it track computer mouse from web cam video

