COSMOS Smart Intersection: Edge Compute and Communications for Bird's Eye Object Tracking

INTRODUCTION

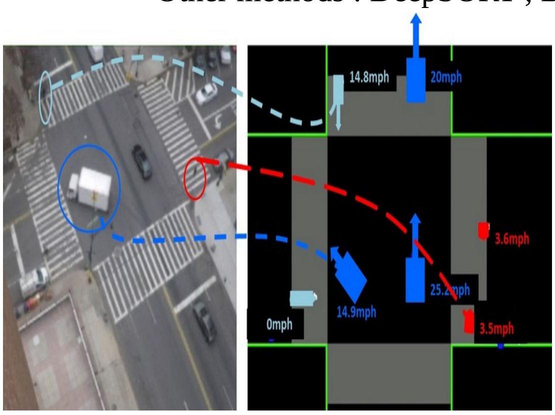
Smart-city intersections are the key infrastructural nodes of the emerging technologically-enabled smart cities. This is an ideal location for positioning the communications equipment and edge computing nodes used for collection and processing of data, and for interaction with traffic participants.
It constitutes a smart intersection: bird's eye videos, edge computing, and contemporary deep-learning-based detection and tracking of objects in an intersection.

ALGORITHM

1. **DATA ACQUISITION :** First of all collect the data. Video is the most frequently used data source for detection and tracking of moving objects. We can collect some of the videos that are captured by the bird eye's cameras placed on the crowded places.
   (Data used by the paper : Videos acquired by COSMOS cameras Hikvision (DS-2CD5585G0-IZHS), and some videos acquired by GOPRO (Hero-6))

2. **DATA PROCESSING :** Then we need to process the collected data. Annotate all the calibrated datasets with location and identity information of objects.
   (Due to diff. camera angles scale may vary and hence the results, so apply video calibration to the raw videos and also black out the irrelevance data.)

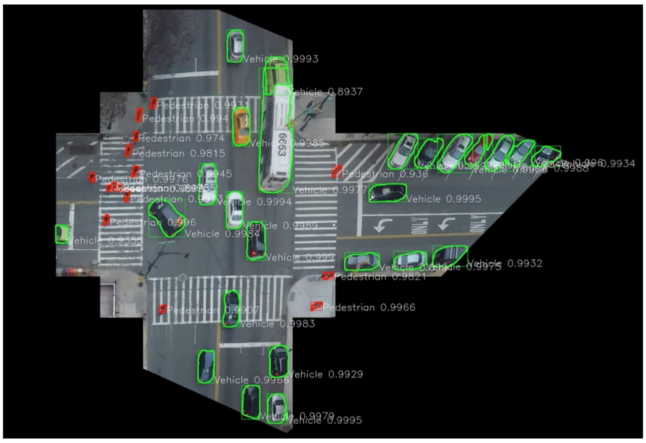
3. **AI FOR DETECTION, TRACKING AND FEEDBACK :**
   1. Object Detection: It requires models to automatically generate the location and class information of objects inside the scenes.
   2. Object Tracking: Identify the presence and know the location of the objects, measure the movement and speed of the objects at each time instance. Multiple Object Tracking (MOT) system is needed for tracking the speed and direction. Other methods : DeepSORT , Deep Affinity Network (DAN)



ADVANTAGES

1. Improved Latency : The latency is improved by a significant amount.
2. Provide intelligent feedback and input to control systems.
3. Processing of the data by taking advantage of low-latency high-bandwidth communications.
4. Higher Accuracy in determining the objects at the crowded places.

VISUALISATION OF THE RESULTS FOR THE PERFORMED EXPERIMENT



In choosing the metrics, we paid particular attention to those used by several widely-known challenges: for detection, the Pascal Visual Object Classes (VOC) Challenge and the Common Objects in Context (COCO) Challenge; for tracking, the MOT Challenge.
$IOU = |A \cap B| / |A \cup B| = |A \cap B| / (|A|+|B|-|A \cap B|)$
Used the detection and the tracking models. An example of the segmentation-based detection using Mask-RCNN. Red contours represent pedestrians while green represent vehicles. This also illustrates the difference in scale between vehicles and pedestrians.Every object in the figure is marked by a colored bounding box which contains the identifier assigned to the object. Many objects of both classes in frame 825 (left) maintain the same color and ID number in frame 800 (right), indicating successful tracking.

CONCLUSION

It results in significantly different accuracies in detection and tracking of vehicles vs. pedestrians, where pedestrians are much harder to reliably detect. A latency target of 33.3 ms is defined for detection/tracking of vehicles. Observed that pedestrians in intersections may move 3-10 times slower than vehicles, therefore latencies for pedestrian detection may be allowed to be 3-10 times larger that latencies for vehicles. The evaluation of the timing of contemporary deep-learning based methods indicates that detection of vehicles needs to be sped up at least three times to meet the execution time/latency of 33.3 ms.