Video-Based Traffic Analytics at Intersection

Sajjad Pakdamansavoji, James H. Elder

Abstract

Rising traffic challenges due to lack of proper traffic management need to be addressed with intelligent solutions like Traffic Analytics Software with integrated video-analytics which can effectively aid traffic management. The necessity of using computer vision in such a software, stems from the fact that conventional traffic monitoring schemes mostly gather visual data such as videos and images. Therefore, designing this software based on the current traffic monitoring infrastructure rules out the need for additional hardware installation.

Our system leverages existing traffic monitoring cameras and applies computer vision techniques (object detection and tracking) to provide detailed traffic analysis results. It can localize, segment, and track vehicles, pedestrians, traffic signs and other related objects in the intersection. By doing so, it provides informative data regarding traffic volume, movements of interests, rule violations, congestions, and so on.

Method

Our Traffic Analytics Software follows a pipeline consisting of four main blocks: preprocessing, detection and tracking, reprojection, and analysis as depicted in Figure 1.

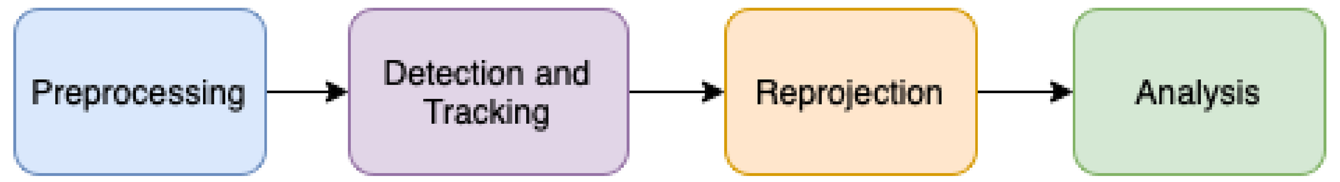

Fig 1. The four main blocks of our traffic analytics pipeline

In the first block, having a calibrated camera, a variety of preprocessing steps such as removing the fish-eye distortion is performed. Once the data is preprocessed, it is fed into a detection and tracking block in which the 2D boundary boxes of vehicles are detected. These boundary boxes are then used to track unique vehicles within a subset of frames. To do so, the common detection by tracking paradigm is used meaning that these two steps are entirely independent in the training phase. Such a decoupled structure is an obstacle preventing the end-to-end fine tuning of the pipeline hence creating the opportunity to integrate these blocks to one unified differentiable unit performing both tasks jointly in the future works.

The third block in the pipeline is responsible for reprojecting points from the image coordinates onto the real-world coordinates. To do that we will use calibrated cameras and pair-matching to estimate the required homography matrix for such a reprojection. This step is performed with human in the loop; as such we developed a GUI in which two views of the intersection(top view and side view) are provided to the user enabling them to select matching points by clicking on them. Our intuition as well as the qualitative results show that reprojecting points and trajectories onto the ground plane will improve the final analysis. A sample of such reprojection is given in Figure 2.

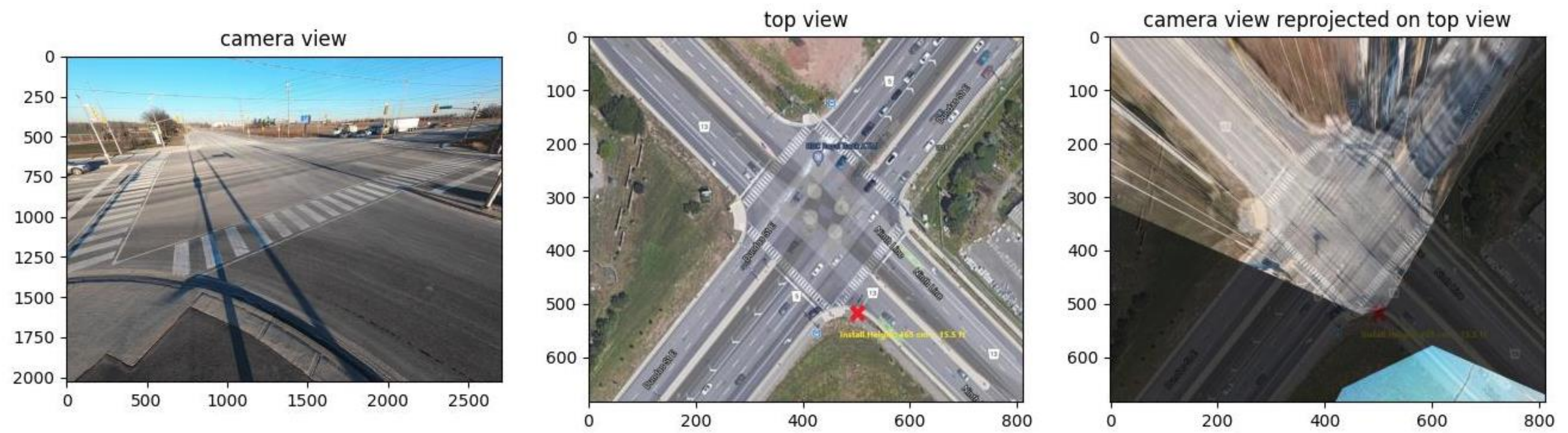

Fig 2. Reprojection of points from camera view to top view

Once such a homography is in hand, each tracklet can be mapped to its corresponding trajectory on the ground plane. To do so, the bottom center of each 2D boundary box is assumed as the contact point to the ground plane. While this assumption will introduce a bias and hence error in the resulting analysis, our qualitative results suggest that this approach is far from oversimplification and produces acceptable reprojection results as shown in the Figure 3.

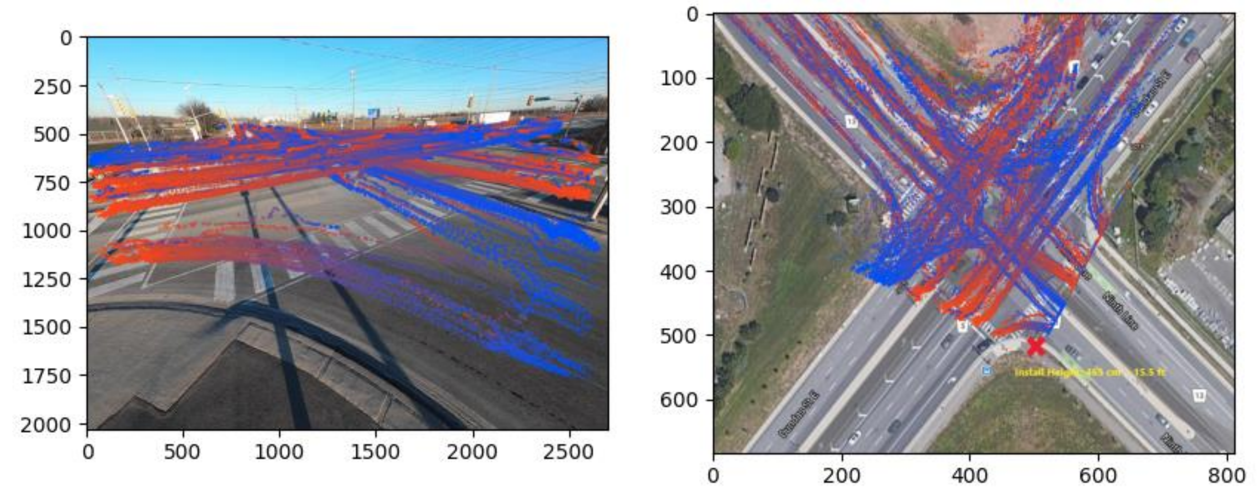

Fig 3. Visualization of tracklets of camera view(left) and their corresponding trajectories on the ground plane(right)

As for the last block, analysis, depending on the downstream task several algorithms are used. For instance, in the case of estimating traffic flow and recognizing movements of interest(MoI) Contour Mapping (CMM) is used to match a trajectory into a commonly traversed path. CMM, in short, is a dynamic programming algorithms in which points from two sets are matched so that the overall cost(distance) of the matching is minimized. CMM also considers order of appearance in its calculations making it consistent with the sequential nature of trajectories.[1]

This algorithm clearly is in need for the selection of commonly traversed paths; to address this requirements we suggest two approaches. The first one is to provide users with another GUI to annotate a few trajectories. In that case, each query trajectory will be classified to its nearest commonly traversed path in a 1NN approach. The second approach, on the other hand, is to eliminate further human annotations by using unsupervised clustering algorithms. For instance, Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is used to cluster trajectories into several clusters in Figure 4. Once trajectories are clustered into groups, for each cluster a number of representative trajectories can be selected for our use case and according to the specific needs of the CMM algorithm choosing longest trajectory from each cluster could be an appropriate choice. It is worth mentioning that using clustering does not fully remove the need for human interactions as the cluster representatives have to be labelled at some point; however, this approaches reduces the annotation complexity and quantity significantly.

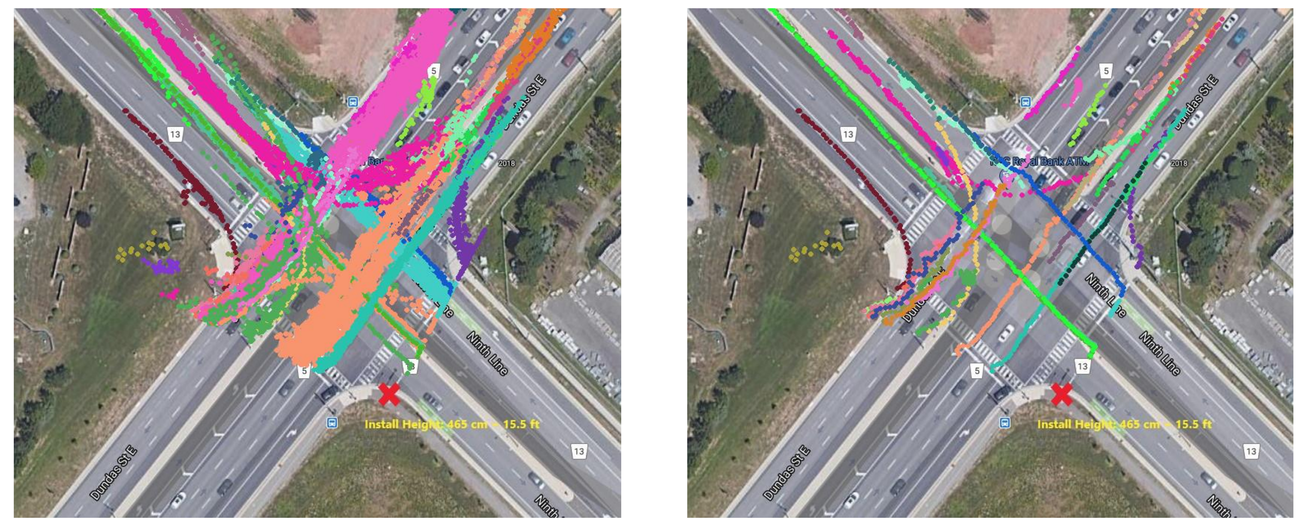

Fig 4. Trajectories clustered into 18 groups (on the left) and their corresponding representatives (on the right) based on longest trajectory length

Results

Deriving accurate numerical results for the presented pipeline requires a vast dataset with known calibrated camera. Since these requirements are not commonly satisfied for available datasets, we are capturing a novel dataset that includes two distinct traffic surveillance scenarios in the GTA. The first dataset, created in collaboration with our industry partner TransPlan, was captured by a GoPro camera temporarily mounted on a light pole at the corner of a number of intersections. The second dataset, created in collaboration with the University of Toronto and the Region of York, features synchronized video from four cameras mounted on each corner of a major intersection in Richmond Hill. As such here we only present the numerical result for one intersection located at Dandas St. and 9th Line.

MoI	GT	DT	Diff	Err%
1	40	39	-1	2.50
2	14	50	36	257
3	29	22	-7	24.1
4	20	7	-13	65.0
5	20	18	-2	10.0
6	18	11	-7	38.8
7	176	176	0	00.0
8	28	26	-2	7.14
9	18	75	57	316
10	127	98	-29	22.8
11	9	11	-2	22.2
12	10	12	2	20.0
All	509	545	36	7.07

Table 1. Numerical results of the pipeline. MoIs numbered counter clock-wise

Conclusion

The proposed pipeline provides a way of analyzing traffic flow at intersections which is independent of the camera angle and other physical constrains. This works has yet to be improved in terms of numerical results after being provided with our novel dataset.

References

[1]: Design and Perceptual Validation of Performance Measures for Salient Object Segmentation Movahedi, V. & Elder, J.H.