Vehicle tracking using the k-shortest paths algorithm and dual graphs

Carlos Lima Azevedo[a],*, João Lourenço Cardoso[b], Moshe Ben-Akiva[c]

[a]Singapore MIT Alliance for Research and Technology, Singapore
[b]National Laboratory for Civil Engineering, Portugal
[c]Massachussetts Institue of Technology, United States

Abstract

Vehicle trajectory descriptions are required for the development of driving behavior models and in the calibration of several traffic simulation applications. In recent years, the progress in aerial sensing technologies and image processing algorithms allowed for easier collection of such detailed traffic datasets and multiple-object tracking based on constrained flow optimization has been shown to produce very satisfactory results, even in high density traffic situations. This method uses individual image features collected for each candidate vehicle as criteria in the optimization process. When dealing with poor image quality or low ground sampling distances, feature-based optimization may produce unreal trajectories.

In this paper we extend the application of the k-shortest paths algorithm for multiple-object tracking to the motion-based optimization. A graph of possible connections between successive candidate positions was built using a first level criteria based on speeds. Dual graphs were built to account for acceleration-based and acceleration variation-based criteria. With this framework both longitudinal and lateral motion-based criteria are contemplated in the optimization process. The k-shortest disjoints paths algorithm was then used to determine the optimal set of trajectories (paths) on the constructed graph. The proposed algorithm was successfully applied to a vehicle positions dataset, collected through aerial remote sensing on a Portuguese suburban motorway. Besides the importance of a new trajectory dataset that will allow for the estimation of new behavioral models and the validation of existing ones, the motion-based multiple-vehicle tracking algorithm allowed for a fast and effective processing using a simple optimization formulation.

* Corresponding author. Tel.: +65 6601 1547 ; fax: + 65 6684 2118.
E-mail address: cami@smart.mit.edu

1. Introduction

Most traffic microscopic simulation modelling and analysis heavily depends on detailed data availability, to accurately replicate sub-second vehicle interactions and driving behaviors. In the last couple of decades efforts to collect vehicle trajectory parameters through instrumented vehicles or site-based observations have been deployed. The later makes use of sensoring technologies installed in delimited areas for detailed road traffic trajectories collection. Photo and video cameras have been the main tools used in site-based trajectory extraction studies (Hoogendoorn et al., 2003, Hranac et al, 2004, Laurenshyn, 2010). Cameras are generally placed on poles, cables and high buildings for static observation, or on airborne vehicles such as helicopters, aircrafts, drones and satellites for dynamic observation. Despite the apparent space and time limitations of these data sets, they allowed for several important developments on traffic flow theory, driving behavior analysis and transportation systems impacts modeling (Ossen et al, 2008, Toledo et al, 2009, Knoop et al 2009, Jie et al 2013).

Previous methods struggled with several image processing and computational limitations, along with a non-negligible manual work burden. With the purpose of fine calibration of an advanced microscopic simulation tool for safety assessment, in this paper we describe a new method for automatic vehicle trajectory extraction. The design of the data collection campaign was significantly influenced by time and budget restrictions of the research study, resulting in the development of a method relying on heavy computer vision techniques and graph theory optimization. Although the case-study presented in this article was applied to aerial images collected from an aircraft flying at medium altitude, the proposed method may be adapted to other image-based sensoring frameworks.

2. The Proposed Method

In this section the description of all steps to achieve the trajectory extraction is presented, pointing out the restrictions imposed by our specific case study.

Mention is made to the aerial imagery framework (namely the collection system set up and the image detailed orthorrectification), the method for vehicle detection using background subtraction, which comprises three main tasks (background construction, moving objects detection and vehicle filtering), as well as the use of graphic theory for the vehicle tracking (specifically the graphs construction and the k-shortest path disjoints algorithm).

2.1. The image data set

The network of interest is the A44 road in the region of greater Porto, Portugal. Its geometrics are characterized by a generic dual carriageway two-lane section, less than 5km in total length, and 5 main interchanges. It represents one of the main entrances for the commuters living in the south-west of greater Porto and to heavy vehicles heading to the port of Leixões, one of the main national ports (see Figure 1).
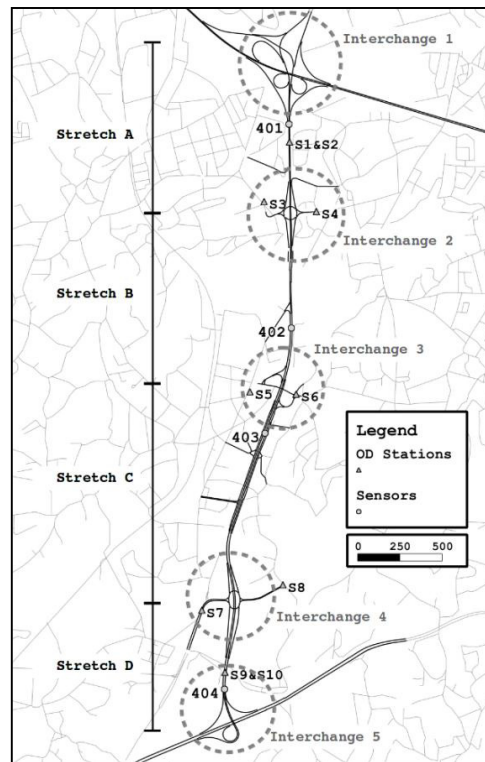
Fig.1 – A44 road layout

2.1.1. Aerial imagery system

Due to budget and time limitations, the data collection flight period was limited to one morning, and the selection of sensoring system relied on the fast image geo-referencing assured by the already calibrated system. Flight images were orthorrectified directly using an existing 3D terrain model of the pilot area, the calibrated camera and lens characteristics, and the precise flight positioning data collected through differential GPS. Furthermore, the choice of such method (instead of static observation or more advanced aircraft platforms) relied also on its ability to collect partial trajectories over the entire length of the pilot study area. The camera had a RGB sensor of 7216x5412 pixels and a fixed focal distance 80mm lens. These high resolution images were collected at an average rate of 0.5Hz, triggered by a fixed maximum image overlapping rate of 90%. Flight characteristics were optimized considering the on-site atmospheric conditions and a desirable ground sample distance of 23cm. The collection campaign resulted in a total of 12 flight runs over the motorway, between 7:45 and 12:00 AM, six per road traffic direction.

2.1.2. Fine image correction

Images were locally rectified to minimize the terrain model and main orthorrectification errors. Each image was divided into grids, scaled and referenced automatically using the SIFT (Scale Invariant Feature Transform) method (Lowe, 2004). The key points in successive images were then matched using the RANSAC (random sample consensus) algorithm (Fischler and Bolles, 1981) and images were rectified accordingly.

2.2. Vehicle Detection

One of the most common techniques for object detection is background subtraction, where each image frame is compared against a background image, similar to the original one but without the moving objects. Reviews of

alternative methods for vehicle tracking may be easily found in the literature (Butch at al. 2011 and Yilmaz et al 2006).

2.2.1. Background construction

The background image may be computed using several methods: frame average method, maximum/minimum intensity value method, median value method, Gaussian and mixture of Gaussian methods and Kalman filtering techniques.

In the present study, for each flight over the A44 a colored background was constructed using the 1-D median filter on all three corrected color channels. For early flights, when congestion is frequently observed, the background computed for later flights was used for smoothing the background pixel values, as slow/stopped vehicles would bias the median value.

2.2.2. Moving objects detection

For each image, the color similarity metric was then used for background subtraction. Foreground pixels in each grid image were then marked considering a uni-modal threshold automatically computed for each image. Shadows were filtered from the foreground using a spectral ratio technique (Tsai, 2006) to minimize the errors during the automatic positioning.

2.2.3. Vehicle Filtering

Non-shadow moving pixels were used in a region-based analysis or segmentation analysis (Veeraraghavan et al, 2003) to extract blobs (or contours) out of connected pixels and the vehicle candidate positions. These blobs are filtered based on their features: minimum and maximum projected area, minimum and maximum projected width and length and specific shape based relationships.
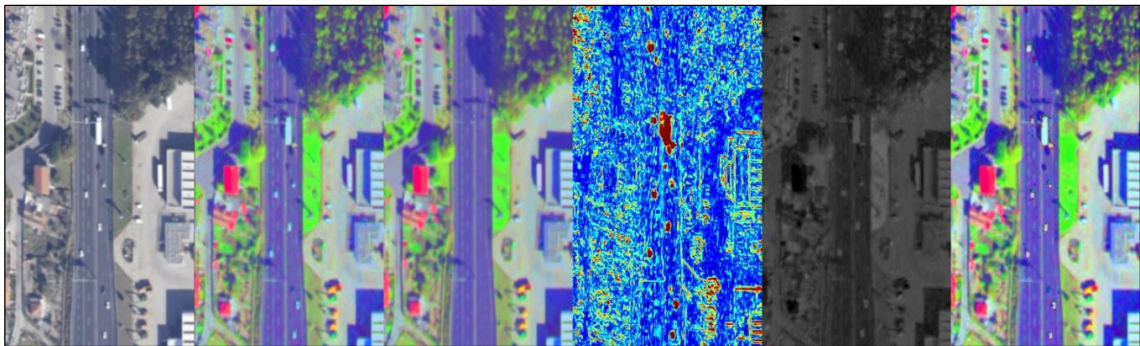

Fig.2 – Image processing steps

All image processing tasks were carried out with a computer server, holding 48Gb of RAM memory and quad-core processors at 2.4Ghz, allowing for a faster computation during the heavy image processing. The code was built in MATLAB using its Image Processing Toolbox.

2.3. Vehicle tracking

One of the most common approaches for vehicle tracking relies on Kalman filters applied to blobs or contours identified in each frame (Cheung and Kamath, 2004, and Yilmaz et al. 2006). However, under congested traffic conditions, vehicles may partially occlude one another, making individual blob identification much more difficult. Feature-based tracking is another common approach, which is based on tracking points that are invariant to changes in illumination and camera viewpoint. These points (features) are then grouped considering spatial proximity or similar motion patterns along multiple image frames (Saunier and Sayed, 2008, and Ismail, 2010). Finally, other

methods use a-priori knowledge about the objects to be identified or motion patterns detected by optical flow (Haag, M. and Nagel, 1999). Each of these methods has, however, their own weaknesses, such as frequent identity switches or non-simple tuning of its model parameters (Buch et a.l 2011).

Optimizing algorithms have also been used to track detected objects, formulated as a graph theory problem (Berclaz et al., 2011). These have mainly been solved using Linear Programming for min-cost flow optimization (Song and Nevatia, 2007). However, the computational complexity of the dynamic programming approach can be prohibitive when the frame or/and vehicle numbers are high. Recently Berclaz et al. (2011) used the k-shortest disjoint paths algorithm on a directed acyclic graph as optimizing algorithm instead of the typical Linear Programming formulation with a significant increase in computational efficiency.

In Berclaz et al. (2011) every region in a frame was represented by a node in the graph. A link between each region in two consecutive frames is generated and labeled with a discrete variable representing the number of objects moving from linked nodes, resulting in a directed acyclic graph. Two additional nodes (source and sink) were added to account for a consistent flow of vehicles in the data set. Any path between the source and the sink nodes represent the flow of a single object in the original problem along the edges of the path, hence a trajectory. Finally, in Berclaz et al. (2011), the k-shortest disjoint path algorithm proposed by Suurballe (1974) was adapted for the computation of the total trajectory set. In our current application the information obtained from the segmentation analysis in the vehicle detection task, such as blob area or average blob color, is error prone due to the small ground sample distance and the lighting conditions. Thus, vehicle dynamic parameters were used in the graph cost flow computation by constructing dual graphs.

2.3.1. Graphs construction

The original graph was constructed using the position of the detected vehicles in each image as nodes and linking possible connected position candidates by limiting a set of possible motion parameters characterizing: longitudinal speed and lane connectivity.
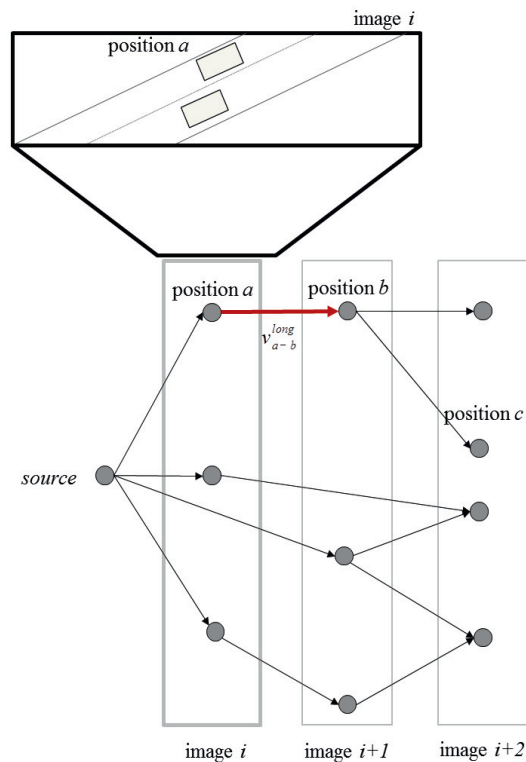

Fig.3 – Original graph construction

From this graph, both acceleration dual graph and a second order acceleration variation dual graph may be computed (Winter and Grünbacher, 2002). Typically used to account for turn costs in network graphs, the dual graph allows the computation of edge-to-edge cost. Accelerations from the speed edge pairs from the original graph can be calculated (see Figure 4), and acceleration variations can be computed in a second order transformation.

The use of dual graphs relied on the assumption that any driver has a motion-based optimizing function, i.e., that any trajectory relies on a set of motion-based objectives of the driver. Ideally, complex microscopic driver behavior models and Kalman-filter dynamics model may be used in this process using large number of motion variables and parameters (gaps, headways, accelerations, etc) to reconstruct trajectories along with the k-shortest disjoint path algorithm. Due to specific nature of the current application a simpler approach was considered. In free-flow conditions, it is known that a driver tends to reach and maintain their target speed. When relaxing the free-flow constrain, one may assume that the driver tends to minimize changes in acceleration. These changes are even smaller if observations are more frequent, due to vehicle dynamics characteristics. Regarding lateral movement, a similar approach can be formulated with the inclusion of lane change tags: when the lateral acceleration is different from zero for a certain period of time, a lane change might be tagged for that trajectory.
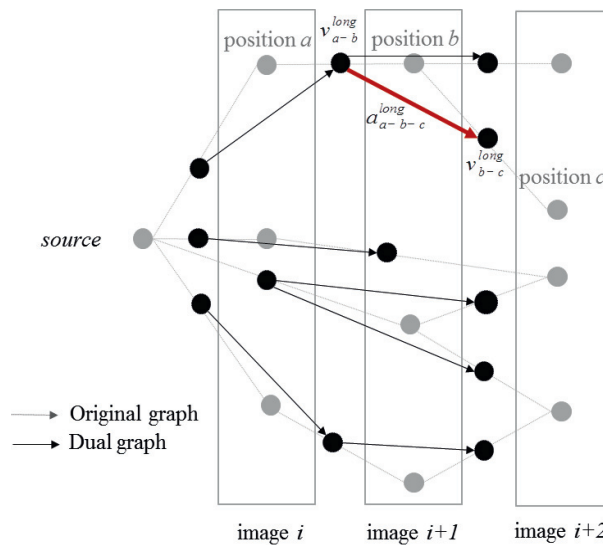

Fig.4 – Acceleration dual graph construction

2.3.2. k-shortest disjoints path algorithm

The Suurballe's algorithm relies on the iterative augmentation of signed paths and on generic shortest path algorithm on a modified costs graph. In this section we briefly present the main concept, but the reader should refer to both Suurballe (1974) and Berclaz et al. (2011) for the full algorithm formulation. The algorithm makes use of signed paths, which are a sequence of sign-labelled edges connecting them in order to form a path in a directed graph G, and path operations, such as interlacing, node splitting and augmentation.

Consider a simple graph G with just 4 nodes $\{i, j, k, l\}$ and 5 edges $\{ij, ik, jk, jl, kl\}$ with costs $\{1,2,1,2,1\}$ (see Figure 5). In the first iteration of the Suurballe's algorithm the shortest path $p_1 = ijkl$ is computed using any generic shortest path algorithm. Nodes in p_1 are then split (using auxiliary links of cost zero) and a signed path corresponding to a labelled p_1 is inverted in G. A second path is computed using a generic shortest path algorithm in the transformed graph G', $p_s = ikjl$. Finally, the signed path p_s is used in the augmentation of the existing shortest path set at iteration n, p_1 in this first iteration: $p_1 \oplus p_s = \{ijl, ikl\}$.
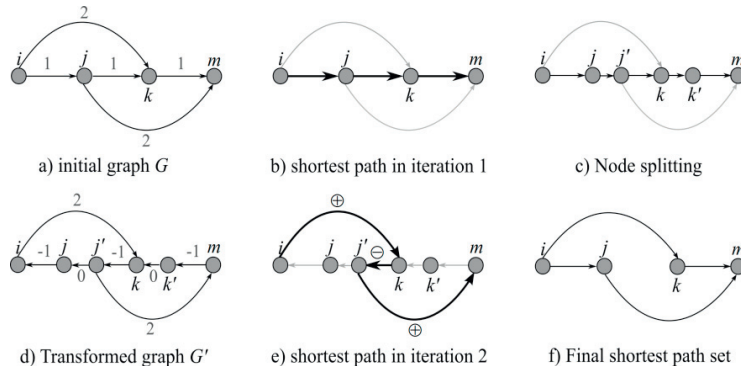
Fig.5 – Suurballe's algorithm general framework

Berclaz et al. (2011) extended this algorithm and developed a framework to also optimize the total number of paths. Using this algorithm, both edge and node-disjointness are achieved. For further details on the implementation of the k-shortest disjoint paths algorithm to this specific case study, the reader is referred to Lima Azevedo et al (2014).

3. Results and Conclusions

A total of 1855 trajectories for all twelve flights were successfully collected with the described framework. During the first three flight runs, congestion was observed in the South-North direction. Levels of service E and F were observed for this subset, during the 7:45-8:30 AM period (see Figure 6).
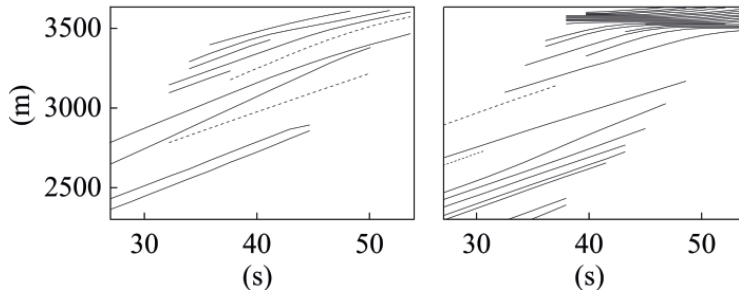

Fig. 6 – Partial trajectories on the left lane (left) and right lane (right), near a weaving area

Typical space-time trajectory datasets were collected for the entire length of the A44 and its entry and exit links. This dataset constitutes the first of its kind in Portugal and will allow the test of existing driving behavior models for this specific case study scenario by analyzing detailed traffic variables such as the distribution of speeds, accelerations, headways, etc. in different sections of the network and time periods (see Figure 7).

It is worth noting that for the earliest flight run (flight run number 1), low speed values were still collected in some sections of the A44, resulting in a bimodal nature of its distribution (see Figure 7.a). Acceleration and deceleration follow a half-normal distribution with the typical low upper and lower range values for non-aggressive maneuvers.
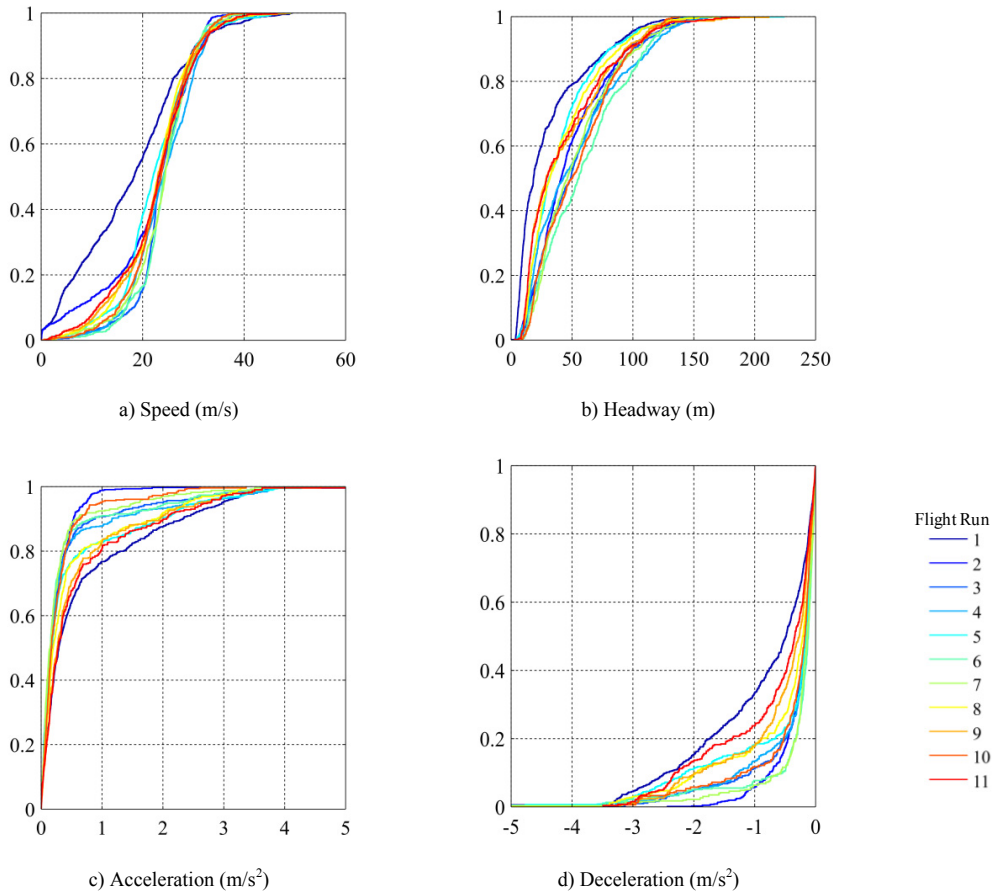
Fig.7 – CDF of motion variables for the different flights

Regarding the algorithm efficiency, approximately 95% (reaching up to 99.4% in free-flow conditions) of the trajectories were successfully extracted even with high temporal and spatial resolutions of 2s and 0.25m respectively. The impact of negative lighting conditions on tracking reliability was minimized using colored imagery and advanced spectral filtering while computer vision techniques helped to enhance the image referencing process and reduce its manual work load. However, if such resources are not available, Gaussian mixture models should be considered in the background construction task. Shadows are always a serious problem during the analysis of many outdoor image sets. Although the advanced spectral filter using colored images limited errors in the position extraction, it still originated false negative, due to the very low ground sampling distance and shooting frequency. Dynamic shadow and 3D vehicle models may be collected in the literature and used to partially minimize these issues; also, the use of stereo imagery would contribute to avoid these modelling burdens, albeit currently at a higher cost. The tracking efficiency drop under dense traffic conditions is also due to the nature of the Suurballe algorithm when applied to dual graphs. If fact, it was detected that the algorithm may not converge to the true optimal solution leading to node-joint paths in the final trajectory set. Future work on this topic is expected to bring additional enhancements to the proposed method.

Acknowledgements

The authors would like to thank the support of Prof. João Paulo Costeira, Dr. Manuel Marques and from the computational resources of their home institution: the Institute for Systems and Robotics of the Instituto Superior Técnico, University of Lisbon, Portugal. The authors would like also to thank InfoPortugal S.A. for the collection of the raw aerial images and the orthorectificaiton using the 3D terrain model.

References

Berclaz, J., F. Fleuret, E. Türetken, and Fua P. (2011) Multiple Object Tracking using K-Shortest Paths Optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33 (9), 1806-1819.

Buch, N., Velastin, S. and Orwell, J. (2011) A review of computer vision techniques for the analysis of urban traffic, IEEE Transactions on Intelligent Transportation Systems, Vol 12 (3,\) 920-939.

Cheung, S. and Kamath, R. (2004) Robust background subtraction with foreground validation for urban traffic video. IS&T/SPIE's Symposium on Electronic Imaging. San Jose, CA, USA.

Fischler, M. A. and Bolles, R. C. (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24, Issure 6, 381-395.

Hoogendoorn, S. P., Zuylen, H. J. Van, Schreuder, M., Gorte, B., Vosselman, G. and Van Zuylen, H. J. (2003) Microscopic traffic data collection by remote sensing. Transportation Research Record, (1855), 121–128.

Hranac, R., Margiotta, R. and Alexiadis, V. (2004) NGSIM Task E.3: High-Level Data Plan. Publication No. FHWA-HOP-06-011. Tech. Rep. July, Cambridge Systematic, Inc., Cambridge MA, USA.

Ismail, K. A. (2010) Application of computer vision techniques for automated road safety analysis and traffic data collection. Ph.D., UBC.

Jie, L., Van Zuylen H., Chen Y., Viti F., and Wilmink, I. (2013) Calibration of a microscopic simulation model for emission calculation. Transportation Research Part C: Emerging Technologies, Volume 31, 172–184.

Knoop, V. L., Van Zuylen, H. J. and Hoogendoorn, S. P. (2009) Microscopic traffic behaviour near incidents. In: Proceedings of Second Sino-Dutch Joint Workshop in Transportation and Traffic Study. 2007. Shanghai, China.

Lima Azevedo, C., Cardoso, J. L. and Ben-Akiva, M. E. (2014) Applying graph theory to automatic vehicle tracking by remote sensing. In Proceedings of the 93rd Annual Meeting of the Transportation Research Board, Washington D.C., USA.

Laureshyn, A. (2010) Application of automated video analysis to road user behaviour. Ph.D. thesis, Lund University.

Lowe, D. (2004) Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60 (2), 91–110

Haag, M. and Nagel, H. H. (1999) Combination of edge element and optical flow estimates for 3D-model-based vehicle tracking in traffic image sequences. International Journal of Computer Vision 35 (3), 295–319.

Ossen, S. and Hoogendoorn, S. P. (2008) Validity of Trajectory-Based Calibration Approach of Car-Following Models in Presence of Measurement Errors, Transportation Research Record: Journal of the Transportation Research Board, 117-125.

Saunier, N. and Sayed. T. (2006) A feature-based tracking algorithm for vehicles in intersections. The 3rd Canadian Conference on Computer and Robot Vision. IEEE, 2006, Quebec, Canada.

Song, X. and Nevatia R. (2007) Detection and Tracking of Moving Vehicles in Crowded Scenes. Proceedings of the IEEE Workshop on Motion and Video Computing, 4.

Suurballe, J. W. (1974) Disjoint Paths in a Network. Networks, Volume 4, 125–145.

Toledo, T., Koutopoulos, H. and Ben-Akiva M. (2007) Integrated driving behavior modeling, Transportation Research Part C 15(2), 96-112.

Veeraraghavan, H., Masoud, O. and Papanikolopoulos, N. (2003) Computer vision algorithms for intersection monitoring. IEEE Transactions on Intelligent Transportation Systems 4 (2), 78–89.

Winter, S. and Grünbacher, A. (2002) Modeling Costs of Turns in Route Planning, Geoinformatica, Volume 6, Issue 4, 345-361

Yilmaz, A., Javed, O. and Shah, M. (2006) Object tracking. ACM Computing Surveys 38 (4), 13.