Estimating OD matrices at intersections in airborne video - a pilot study

Viktor Braut, Mateja Čuljak, Vedran Vukotić, Siniša Šegvić
Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb
e-mail: name.surname@fer.hr

Marko Ševrović, Hrvoje Gold
Faculty of Transport and Traffic Sciences
Vukelićeva 4, 10000 Zagreb
e-mail: name.surname@fpz.hr

***Abstract*—This paper presents a pilot study towards estimating complex traffic flow parameters in airborne video. The study presents two prototype software systems attempting to solve intermediate tasks in recovering microscopic OD (origin-destination) matrices at complex road intersections. The first system employs background modelling in order to estimate the OD matrix of an intersection imaged by a fixed camera. The second system explores the feasibility of applying such approach to input video acquired from a hovering aircraft by pre-warping the whole video towards the coordinates of the first frame. The experimental part presents performance evaluation of the two prototype systems on real traffic videos acquired from a tall building and a non-rigid airship. The paper is concluded by discussing the achieved baseline performance and proposing suitable directions for future research.**

I. Introduction

Modern transportation systems are designed by optimizing traffic flow models which are parameterized by actual demands estimated from empirical measurements. Typical traffic flow parameters include frequency, density, headway, etc. of the vehicles at relevant sections of the transportation network. There are many suitable commercial technologies for estimating these parameters at straight sections of the network. However, none of them provides satisfactory performance at intersections where one needs to establish temporal correspondence between the detected vehicles in order to estimate microscopic OD (origin-destination) matrices [1], [2]. The most general approach for estimating intersection parameters corresponds to manual vehicle counting. However, that approach is hampered by organizational and financial difficulties, since measuring complex intersections usually requires many trained human operators. Encouraging results have been achieved by computer vision based approaches [3], [4] in video acquired from viaducts or telescopic cranes, but this is not applicable at many intersections. Airplane imagery [5] has a broader scope but is often infeasible due to the high costs involved. Acquisition from a helicopter [6] is more accessible, but the costs are still significant. Additionally, if affordable standard resolution cameras are employed, one has to address the problem of small-sized vehicles in images which makes it difficult to count them automatically.

This paper presents introductory research towards estimating intersection parameters in video acquired from an unmanned aerial vehicle (UAV) hovering at the heights of about 50 m. The main commercial advantages of such approach include lower acquisition costs and better performance with standard resolution images. In particular, the paper evaluates performance of the following two intermediate tasks. In the first task we assess the accuracy of OD matrices estimated in video acquired by a fixed camera from a tall building, based on a straight forward background modelling technique. The second task involves stabilizing video from a hovering airship, in the ambition to extend the applicability of fixed-camera approaches to UAV-based applications.

In the rest of the paper, we first present a short review of the related work in Section II. Subsequently, in Section III we explain the broad significance of airborne video for estimating parameters of a transportation system. We detail the theoretical backgrounds for the considered intermediate tasks in Sections IV and V. Subsequently we present experimental results obtained on videos of real intersections acquired from a tall building and a hovering airship in VI. Finally, in Section VII we present conclusions and directions for future work.

II. Related work

Extraction of complex intersection features such as OD matrices and headway from airborne imagery has many advantages over ground-based approaches [5], [6], however the involved costs have been very high. Recent development of low-cost UAV technology encourages new research on airborne traffic flow feature collection [4].

A large body of previous work addresses videos acquired with fixed cameras. These approaches have been mostly based on image motion detection [7] and background modelling [3], [8]. In both cases, the transition towards a moving airborne camera [6], [9] is not straightforward and requires further research. More versatile approaches build on generic object detection [10], however most of them address images in which the street scene is viewed from an orthogonal direction. This would not be the case in images acquired from low-height howering UAVs where we observe both appearance variety (due to different relative poses of the object with respect to the camera), and the variety of object categories (cars, trams, buses, pedestrians, cyclists, etc). This variability would have to be addressed by a multi-class detection approach based on feature sharing such as [11], however the performance of such approaches still appears insufficient for our purpose.

Stabilization of the acquired video is an important issue in airborne surveillance. Camera orientation can often be stabilized within some limits by suitable gyroscope platforms. Medium to high frequency vibrations generated from UAV's motors and wind can also be problematic and can be addressed by suitable absorbing materials. Unfortunately, many of the UAVs considered suitable for traffic surveillance have significant payload weight limitations,

which greatly reduce the possibility of utilising hardware stabilization setups. This accentuates the importance of software based stabilization [6], such as the approach presented in V and VI-C.

An additional concern specific to UAV operation is automatic control. Autonomous UAV capabilities would be interesting in order to i) relax the human operator involvement, ii) ensure optimal acquisition in bad weather through UAV stabilization, and iii) minimize the risks of damage during take-off and landing. A recent study [12] confirmed the potential of visual control, even when sophisticated alternatives are present such as inertial sensors and GNSS (GPS, Galileo).

III. Measuring traffic flow parameters at the microscopic scale

The most critical part of every transport system are its nodes or intersections. Design, throughput and capacity of a single intersection can affect even very distant parts of the transport network. Therefore studies leading to appropriate intersection design and traffic control are the most important factors in transport network optimisation. An important category of these studies is performed at the microscopic scale, where traffic flow parameters are measured by considering each vehicle as an individual. In this paper we are especially interested in measuring OD matrices and time headway at complex intersections.

OD matrix is also know as a trip table. It contains the number of vehicles going from each intersection entry to each intersection exit during the considered time interval. OD matrices are especially hard to recover at complex intersections and interrelated junctions where comprehending and accurately counting the entries is highly impractical and expensive both for human operators and commercial sensor technologies.

Time headway [13] corresponds to the elapsed time between the front of the first vehicle passing a road cross-section and the front of the subsequent vehicle. Headway is the most essential microscopic traffic flow parameter. Interaction between headway and vehicle speed is responsible for the overall traffic flow performance. Precise measurements of headway in relation to speed provide a detailed insight in the capacity limitations and design flaws at observed intersections. Unfortunately, most existing studies focus on average headway within a limited period of time, mostly due to imperfections of various sensors used for vehicle counting and traffic parameter measurement. Headway-to-speed ratio is particularly difficult to recover at intersections since it is not constant during intersection passing [14]. Therefore, there is a strong need for future sensor technologies which would be able to map this parameter over the intersection area.

IV. Vehicle detection by fixed cameras

When viewing a rigid scene from a fixed viewpoint, each pixel is always projected from the same part of the scene. By analyzing the changes of a pixel in time, a model can be constructed to determine whether a current realization of that pixel is more likely to belong to a moving foreground object or to the background scenery [8]. An object detection system based on such model would need to include the following processing steps.

- Creation of a per-pixel background model for the desired scene.
- Comparing the current video frame against the background model to identify foreground pixels.
- Grouping the foreground pixels into high-level objects.

The background model is often built and updated concurrently with the object detection process. The main advantage of such approach is the ability to make an adaptive model which would be able to tollerate small changes due to different time of day. However, we do not need that feature, since the traffic parameters are estimated in limited time intervals (less than one hour), usually at the time of peak traffic. Therefore we build the background model in an off-line fashion, by employing all available frames of the input video. The object detection is consequently performed on the same video, after having rewound it back to the start, as described in Figure 1.

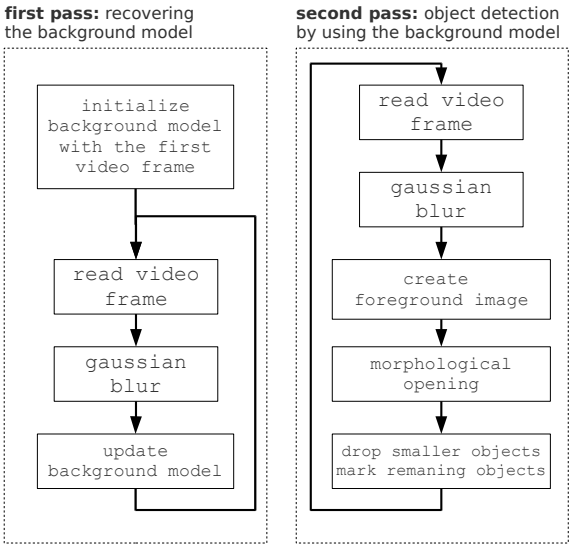


Fig. 1. Flow chart of the proposed procedure

The first step in both processing phases is to apply Gaussian filtering to the current frame. Doing so greatly reduces the noise and the undesired effects of minute camera vibrations. This technique therefore improves both the accuracy of the background model and the reliability of foreground object detection.

A background model of an intersection tries to mimic how an image of the intersection would look like without any moving objects (vehicles, pedestrians, etc.) on it. Two different approaches were considered for creating a background model of the intersection. The first approach was to calculate the moving average for each pixel on the picture. Due to the fact that vehicles often differ in color and keep a safety distance from other vehicles the average of each pixel converges to the color of the underlying intersection in that position. The second approach was to keep a histogram for each pixel and choose the most frequent color for the background model. An another option for creating a background model would

be based on Mixtures of Gaussians [15], however preliminary results did not show dramatic improvement over unimodal approaches and so we preferred to invest our time elsewhere.

After the background model converges, the current frame is compared against it in order to form a binary (1 bit black and white) image showing whether a pixel differs from the background or not. A white pixel indicates that the corresponding pixel from the current frame differs significantly from the background model. A black pixel indicates that the current value is very similar or identical to the background model. Vehicles are detected by growing a region around each unvisited white pixel. Coordinates of each pixel of the current region are stored in a vector. When there are no more unvisited white pixel neighbors, vertical and horizontal extremes are found in the vector and a rectangle is drawn so that it encloses the detected group.

Sometimes a vehicle has parts that do not differ significantly from the background. The most common case is the gray reflection of the sky on the windshield that can be very similar to the gray background of the asphalt. This can cause a single vehicle to be detected as two separate groups of foreground pixels. To address this problem, morphological opening [16] is applied to the binary image, by successively applying dilation and erosion. Dilation widens the groups of white pixels causing their melting with neighbouring groups. By applying erosion the groups regain their initial size, but remain connected to their neigbours. In the last step, objects greater than a specific size are detected as vehicles while smaller objects are ignored since they're likely due to pedestrians or noise.

V. Stabilization of Video Acquired from Hovering Aircraft

In the case of acquisition from hovering aircraft, the acquired video is often unsuitable for background modelling due to 6 DOF motion of the attached camera. We wish to stabilize the video in a way that the image plane remains in consistent relation with the starting video frame (or any other reference frame).

The developed system consists of two main parts: feature tracker and homography estimator. In the first part, a large number of features are tracked throughout the video sequence from the reference frame to the current frame, by using Kanade-Lucas-Tomasi (KLT) feature tracking algorithm. The KLT tracker selects only those features that can be reliably tracked [17].

Before estimating the transformation between the reference frame and the current frame, it is neccessary to remove features whose movement deviates from other features, i.e. outliers. In our case, most of outlying features are projected from moving vehicles, which deviate from the desired transformation of the ground plane in two images. The removal of outliers is done by a Random Sample Consensus (RANSAC) algorithm [18].

Coordinates of every tracked feature are then used to estimate the desired transformation. We assume that aircraft is high above the ground so that the transformation can be fairly well-approximated by a homography. To estimate homography, we need to solve equation of the form:

$$\mathbf{A}_{2n\times 9} \cdot \begin{bmatrix} \mathbf{H}_1^\top \\ \mathbf{H}_2^\top \\ \mathbf{H}_3^\top \end{bmatrix} = \mathbf{0}_{2n} \ . \qquad (1)$$

Every corresponding pair of coordinates participates with two linearly independent equations to the overconstrained linear system (1). Each independent equation contributes one row of the matrix $\mathbf{A}_{2n\times 9}$, where n is the number of such corresponding pairs. Matrix $\mathbf{H}_{3\times 3}$ is a solution which we are searching for, and it contains the coefficients of the desired homography (note: $\mathbf{H}_i$ is i-th row of matrix $\mathbf{H}$). In our system, the linear system (1) is estimated by SVD matrix decomposition. Since this estimation does not maximize likelihood, the solution is additionally improved by using gradient optimization – Levenberg-Marquardt algorithm.

The estimated transformation is applied to current video frame as follows:

$$\begin{bmatrix} x' & y' & 1 \end{bmatrix}^\top = \mathbf{H}_{3\times 3} \cdot \begin{bmatrix} x & y & 1 \end{bmatrix}^\top \ . \qquad (2)$$

In the above equation, (x, y) is point in current frame, (x', y') is (x, y) put into the reference image plane and $\mathbf{H}_{3\times 3}$ is a homography matrix. An example of video stabilization is shown in Figure 2.

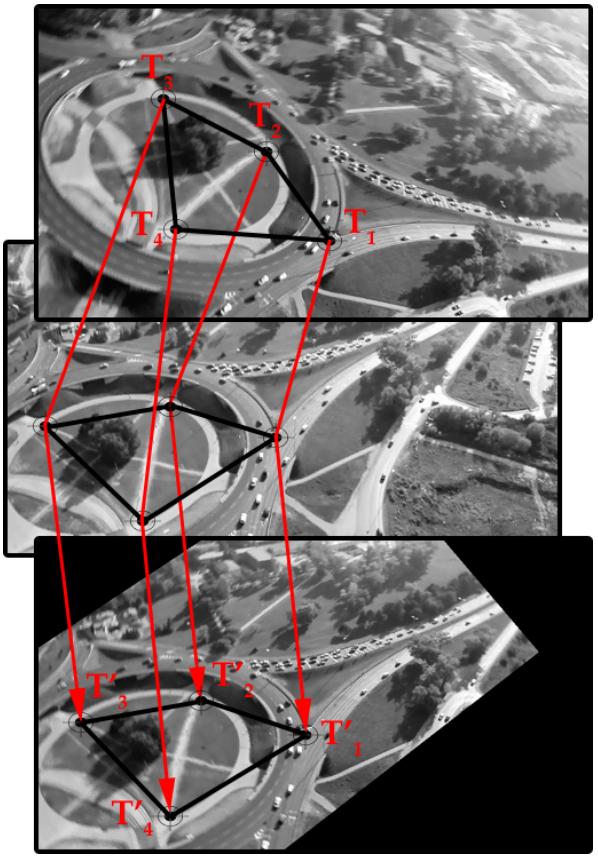


Fig. 2. Stabilization of video acquired from hovering aircraft. The figure shows the current frame (top), the reference frame (middle), and the transformed current frame (bottom). Points T_i in the current frame are transformed to the points T_i' by the homography between the current frame and the reference frame.

During the tracking of point features through image sequence, error is accumulated. To improve results, as

a final step we check consistency of features. For each feature, the similarity between the intensity of tracking window in current frame and the intensity of tracking window in reference frame is calculated. If the two tracking windows are not similar, the feature is inconsistent and can be rejected as a bad feature.

VI. Experimental results

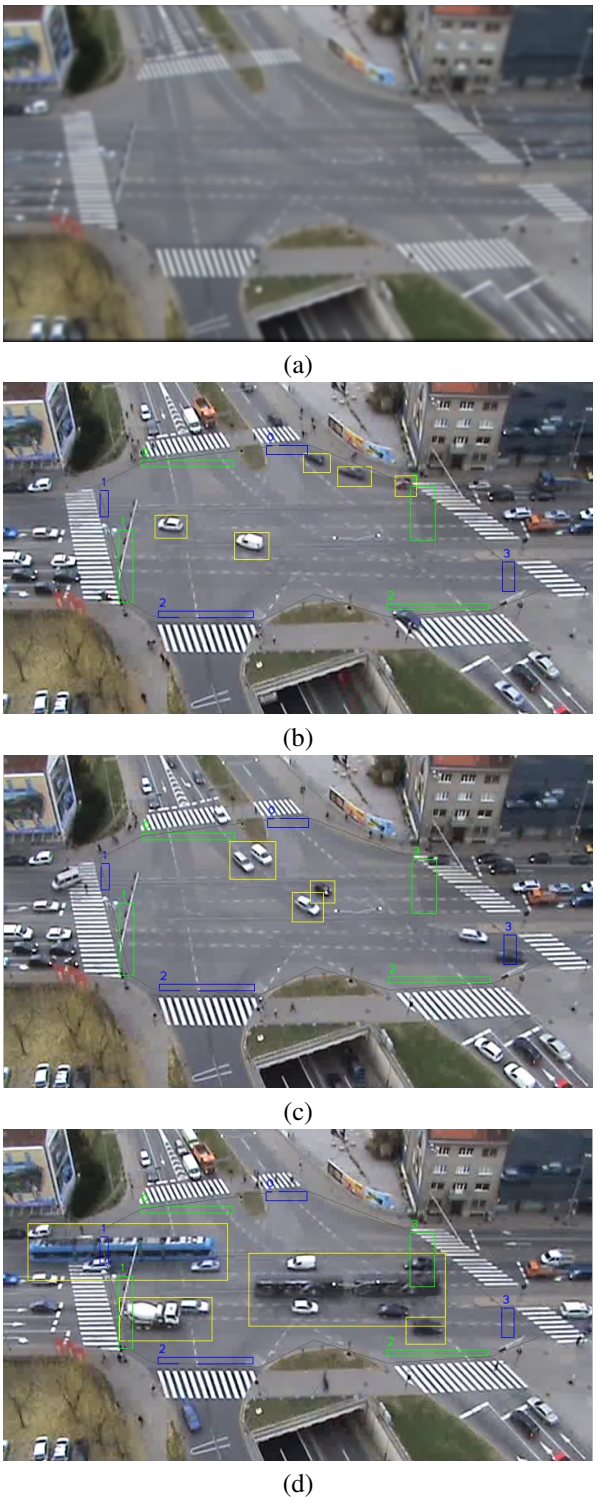
A. Vehicle detection based on background modelling

Gaussian filtering was performed by convolving an image with a Gaussian filter kernel. A good quality/performance ratio was obtained by applying a 7×7 Gaussian kernel with a standard deviation σ=3.5. Due to its symmetry the Gaussian kernel can be applied separately to each dimension. Separating the filtering process in vertical and horizontal directions and using OpenMP for parallelization resulted in a faster implementation of the filtering algorithm.

Creating the background model by computing a moving average uses less memory but requires more time to converge. Histogram-based background models converge faster but, on the downside, require considerable memory for storing histograms in each image pixel. In order to save space, our approach stores only marginal RGB histograms (we assume unimodal distributions) and therefore requires a total of $3\times256\times$width$\times$height bytes for an 8 bit RGB image. Besides the differences in memory usage and convergence speed no other relevant performance differences were noticed between these two approaches. Both approaches can be precomputed or used in real time. We chose the precomputed approach in order to i) avoid having to wait for model convergence, and ii) being able to focus on object detection instead of on background modeling. The main drawback of precomputed background models is their inability to adapt to changes, however this is not a problem in our context since traffic analyses are typically carried out in limited time intervals. The foreground image is determined by comparing the current image to the previously recovered background model. For simplicity, the comparison is implemented by comparing absolute RGB differences with a fixed threshold. In all experiments the threshold was set to 20.

Before further processing, the binary foreground image is subjected to morphological opening implemented as three dilations followed by three erosions, by using the structuring element 1×1. This procedure sometimes causes neighbouring vehicles to be detected as one. However, empirical analysis has shown that these losses are outweighed by the gains from having fewer fragmented objects. The obtained results have been illustrated in Figure 3.

Experimental evaluation revealed two major problems. The biggest problem is the tram entrance to the scene during which our system detects a large object which predates all vehicles moving near the tram. Therefore, all experimental results which we show here have been obtained in parts of the video in which trams are not present at the intersection. The most common problem is detecting multiple neighbouring vehicles as only one object. Additionally, some vehicles are detected as two objecs. This may occur either when a car is similar to the background or due to the reflections of the gray sky on the windshield.



(a) (b) (c) (d)

Fig. 3. Results of object detection: background model (a), successful detection of multiple objects (b), two vehicles detected as one (c), problems with big vehicles (d).

B. Estimating the intersection OD matrix

The developed prototype system allows the user to define regions that represent origins and destinations in the intersection and to count the number of objects that pass from each origin to each destination. The OD matrix is displayed as an overlay in the video and saved to a file

for further analysis.

Figure 4 shows the intersection with marked origins (green) and destinations (blue). All objects detected outside of the red polygon are discarded. This had to be done due to reflections on the building windows and due to groups of pedestrians which may be large enough to be detected as vehicles.

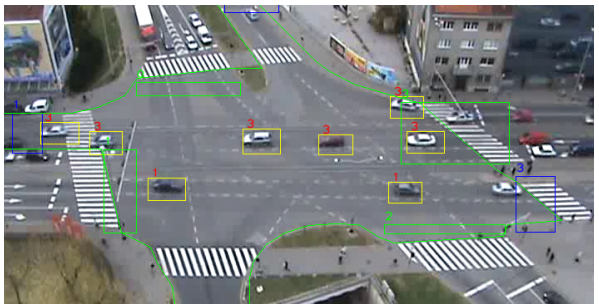


Fig. 4. The regions employed in the process of estimating the OD matrix are defined manually as green (source) and blue (destination) rectangles. The employed background model is shown in the background. A demonstration video can be viewed at http://www.zemris.fer.hr/~ssegvic/pubs/braut12ok.avi.

The recovered OD matrices have been evaluated with respect to the groundtruth. In the evaluation we have omitted frames in which the tramways are passing through the intersection. The obtained results have been summarized in Table I. The table entries show the number of estimated vehicles compared to the groundtruth.

TABLE I
THE ESTIMATED OD MATRIX COMPARED TO THE GROUNDTRUTH.

	destinations			
origins	0	1	2	3
0	0:0	9:9	15:17	42:44
1	14:14	0:0	8:8	24:17
2	22:27	20:23	1:1	0:0
3	28:30	32:32	26:24	3:3

C. Video stabilization

The developed prototype for video stabilization uses the library for KLT feature tracking[1], and the OpenCV library "Camera Calibration and 3d Reconstruction" [19] for homography estimation. The system tracks 3000 features in windows of 7×7 pixels. Large number of features and large tracking window improve stabilization but negatively affect the processing speed. The original video resolution is 1920×1080, however it has been downsampled to 960×540 in order to make the computation more tractable. Typically the system succeeds to maintain a fair correspondence with the reference frame for a few seconds. After that the results deteriorate due to the excessive number of lost features. Demonstration videos can be viewed at http://www.zemris.fer.hr/~ssegvic/pubs/culjak12original.mp4 and http://www.zemris.fer.hr/~ssegvic/pubs/culjak12stable.mp4.

The quality of results can be expressed as an average displacement of transformed image pixels in relation to corresponding pixels in the reference image, i.e. jitter. To calculate jitter, five suitable control points are manually chosen in the reference image. These control points are consequently located in 13 transformed images. The average displacement in each image is calculated as an average difference of control point coordinates between reference image and transformed images:

$$\bar{x} = \left\lceil \sum_{j=1}^{k} \sum_{i=1}^{n} \frac{|xc_i - x_{i,j}|}{nk} \right\rceil , \quad (3)$$

$$\bar{y} = \left\lceil \sum_{j=1}^{k} \sum_{i=1}^{n} \frac{|yc_i - y_{i,j}|}{nk} \right\rceil . \quad (4)$$

In the above equation $k = 13$, $n = 5$, $\bar{x}$ and $\bar{y}$ are average displacements on abscissa and ordinate, (xc_i, yc_i) are coordinates of i-th control point in the reference image and $(x_{i,j}, y_{i,j})$ are coordinates of i-th control point in the j-th transformed image. The results of evaluation are shown in Figure 5.

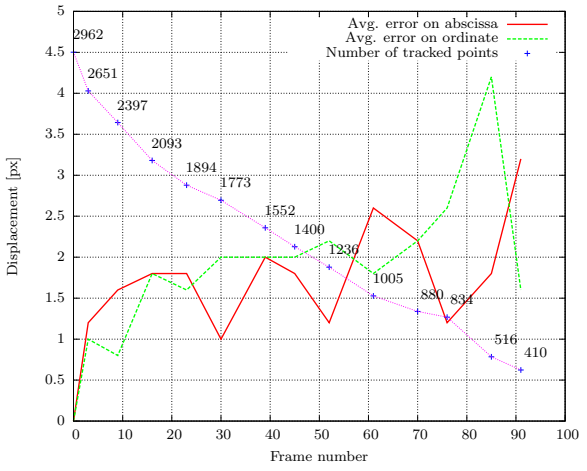


Fig. 5. Average displacement of the current image with respect to the reference image, as a function the frame distance.

VII. CONCLUSION AND FUTURE WORK

This paper considered the feasibility of estimating microscopic OD (origin-destination) matrices at complex road intersections in airborne video. Encouraging results have been obtained even with straight-forward computer vision techniques. The main problems identified by performance evaluation in the fixed-camera context include aggregated detections of nearby vehicles and occlusion by large vehicles such as tramways. The main problem identified in the hovering-camera context is the starvation of the tracked features.

The identified problems shall be addressed in our future work. In particular, advanced object detection approaches shall be tried out which would assign high-level objects to foreground pixels not only by looking at their proximity, but also by considering at their colour histograms and their image motion. Long-term stabilization of the hovering aircraft video shall be attempted by creating a map of suitable ground plane points in a SLAM fashion. The map shall be employed to restart the tracking of lost points

[1]URL http://www.ces.clemson.edu/~stb/klt/

which shall enable us to maintain an accurate relation towards the reference frame throughout the whole video.

Our future work shall also address estimating speed and headway at all points of the vehicle trajectory. These important parameters can not be reliably estimated by existing sensor technologies, since their values typically vary across the intersection. It is therefore expected that the future developments might provide significant insight into opportunities for improving the efficiency of our transportation systems.

VIII. Acknowledgements

This research has been funded by the University of Zagreb in the frame of the projects Advanced detection techniques for intelligent transportation systems, and Center for Computer Vision.

References

[1] J. Hourdakis, P. Michalopoulos, and J. Kottommannil, "Practical procedure for calibrating microscopic traffic simulation models," *Transportation Research Record*, 2003.

[2] M. Bierlaire, "The total demand scale. a new measure of quality for static and dynamic origin-destination trip tables," *Transportation Research Part B*, 2002.

[3] H. Veeraraghavan, O. Masoud, and N. Papanikolopoulos, "Computer vision algorithms for intersection monitoring," *IEEE Transactions on Intelligent Transportation Systems*, 2003.

[4] N. Buch, S. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *IEEE Transactions on Intelligent Transportation Systems*, 2011.

[5] A. Angel, M. D. Hickman, P. B. Mirchandani, and D. Chandnani, "Methods of analyzing traffic imagery collected from aerial platforms," *IEEE Transactions on Intelligent Transportation Systems*, 2003.

[6] A. C. Shastry and R. A. Schowengerdt, "Airborne video registration and traffic-flow parameter estimation," *IEEE Transactions on Intelligent Transportation Systems*, 2005.

[7] A. Ottlik and H.-H. Nagel, "Initialization of model-based vehicle tracking in video sequences of inner-city intersections," *International Journal of Computer Vision*, 2008.

[8] M. Piccardi, "Background subtraction techniques: a review," in *Proceedings of the IEEE International Conference on Systems, Man & Cybernetics*, pp. 3099–3104, 2004.

[9] A. Puri, K. P. Valavanis, and M. Kontitsis, "Generating traffic statistical profiles using unmanned helicopter-based video data," in *Proceedings of the International Conference on Robotics and Automation*, 2007.

[10] H. Grabner, T. T. Nguyen, B. Gruber, and H. Bischof, "On-line boosting-based car detection from aerial images," *ISPRS Journal of Photogrammetry & Remote Sensing*, 2008.

[11] B. Wu and R. Nevatia, "Cluster boosted tree classifier for multi-view, multi-pose object detection," in *Proceedings of the International Conference on Computer Vision*, pp. 1–8, 2007.

[12] I. Mondragón, M. Olivares-Méndez, P. Campoy, C. Martínez, and L. Mejias, "Unmanned aerial vehicles uavs attitude, height, motion estimation and control using visual systems," *Autonomous Robots*, 2010.

[13] W. H. Kraft, ed., *Traffic Engineering Handbook*. Institute of Traffic Engineers, 6th edition ed., 2009.

[14] I. I. Otković and I. Dadić, "Comparison of delays at signal-controlled intersection and roundabout," *Promet - Traffic & Transportation*, vol. 21, no. 3, pp. 157–165, 2009.

[15] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. II: 246–252, IEEE, 1999.

[16] R. Jain, R. Kasturi, and B. Schunck, *Machine Vision*. McGraw-Hill, 1995.

[17] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, (Seattle, Washington), pp. 593–600, June 1994.

[18] S. Choi, T. Kim, and W. Yu, "Performance evaluation of ransac family," in *Proceedings of British Machine Vision Conference*, 2009.

[19] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.