Vehicle Re-Identification for Automatic Video Traffic Surveillance

Dominik Zapletal, Adam Herout
Graph@FIT, Brno University of Technology
herout@fit.vut.cz

Abstract

This paper proposes an approach to the vehicle re-identification problem in a multiple camera system. We focused on the re-identification itself assuming that the vehicle detection problem is already solved including extraction of a full-fledged 3D bounding box. The re-identification problem is solved by using color histograms and histograms of oriented gradients by a linear regressor. The features are used in separate models in order to get the best results in the shortest CPU computation time. The proposed method works with a high accuracy (60 % true positives retrieved with 10 % false positive rate on a challenging subset of the test data) in 85 milliseconds of the CPU (Core i7) computation time per one vehicle re-identification assuming the fullHD resolution video input. The applications of this work include finding important parameters such as travel time, traffic flow, or traffic information in a distributed traffic surveillance and monitoring system.

1. Introduction

Obtaining accurate and up to date traffic information and statistics is increasingly in demand for multiple reasons – for collecting statistical data [10], for immediate controlling of traffic signals [9], for law enforcement [8, 13], etc. In most traffic-busy areas, many surveillance cameras are already installed. It would be advantageous to use these devices for analysis of traffic flows with no need of replacing them with some special hardware. To achieve this, re-identification on existing simple video surveillance cameras is necessary. To maintain anonymity and because of the inability to obtain all vehicle number plates from video recordings in some cases, it is desirable to base the re-identification only on visual characteristics of the vehicle [11].

The problem of re-identification consists of extraction of sufficient information from the detected vehicle in the video and of efficient use of this information to find an identical vehicle in a different set of detected vehicles. One aim of solving the problem is to minimize the cases of false positive and false negative re-identifications and to maximize the cases of positive re-identifications.

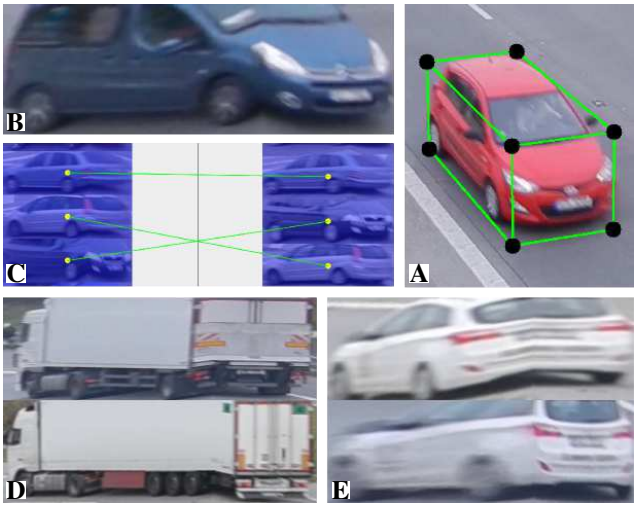


Figure 1. In this work, we solve the problem of visual re-identification of vehicles observed from different surveillance cameras. The system detects a vehicle and automatically extracts its 3D bounding box based on existing fully automatic calibration [5, 4] *(A)*. The vehicle image is "unwrapped" in order to concentrate on a normalized image of the vehicle *(B)*. Based on manual annotations *(C)* we collected a dataset for evaluation of the system. *D:* A pair of similar vehicles correctly recognized as not identical. *E:* A typical pair of matching samples recognized by the system.

Some existing works are concerned with the re-identification issue. Arth *at al.* [1] created a de-centralised vehicle re-identification engine working on a non-calibrated surveillance camera network. They used PCA-SIFT features as an appearance-based method for a vehicle data extraction. Their system represents the data extracted from a vehicle as a vocabulary tree, a so called signature. However, their approach does not take into account color information of the vehicles. Also, a simple 2D bounding box was used in their solution. Another approach was presented by Oliveira *at al.* [3]. Their paper is concentrated mostly on people re-identification. Their solution is based on local appearance features; color (HSV histograms) and SURF descriptors were used in the work.

Figure 2. Examples of detections. **top:** Data received from the detection system are represented as seven points describing the 3D bounding box around the detected vehicle. **bottom:** Results of image extraction from the 3D bounding box.

Figure 3. An illustration of a typical "unwrapped" image. The representation used in re-identification (bottom) is much more compact compared to the original and it contains very little of the background clutter. The extraction is very efficient and preserves majority of the visual information of the vehicle.

The solution described in this paper (Figure 1) is based on the assumption that the vehicle detection in the video is already solved. In our approach, a detection system that creates 3D bounding boxes around detected vehicles is used [5, 4]. The input information for vehicle re-identification is a vector of points describing the 3D bounding box (Figure 2).

Our approach is based on linear regression which uses two trained models: one model using color histogram [12] and one model based on histogram of oriented gradients [2]. The side and front of the bounding box of the detected vehicle are projected by warping into the plane and combined together. Then, the combined image is split into a grid and color histogram values for each of the RGB channels and histogram of oriented gradients values are computed for each grid cell. Both of the models are created by training on a large amount of positive and negative data samples. The re-identification itself is performed by finding the most suitable vehicle in a detected vehicle database based on the data previously obtained.

Besides proposing the vehicle re-identification methodology, we invested effort into collecting a dataset for training and evaluation of the method. We manually annotated pairs of corresponding (identical) cars in videos captured from different cameras and different viewpoints. Then, we crowd-sourced another dataset of vehicle pairs, where human subjects annotated their perception whether the given pair "could be the same vehicle". This dataset is meant as human ground truth including stratification of clear cases of match, clear cases of non-matching vehicles, and borderline cases, where even human observers doubted or disagreed. Both these datasets are made public along with this paper for future experimenting and evaluation.

In our work, we were able to find a way of solving the vehicle re-identification problem which can be built upon in our future work and studies. We achieved accurate results that can be found useful in many ways in traffic surveillance.

2. Vehicle Fingerprinting – Feature Extraction

It is necessary to properly represent the data describing key visual characteristics of the vehicle for maximal re-identification accuracy. It is also very important to minimize the background parts of the image behind the detected vehicle in order to increase the signal/noise ratio. The data extracted has to be efficiently stored in order to speed up the whole re-identification process.

In order to reduce the amount of unwanted data extracted from video we use 3D bounding boxes as the vehicle detection representation (Figure 2). The side and front faces of the bounding box are used only, because in most cases, the rooftop does not contain any additional useful information (rooftops could be interesting in case of trucks and special vehicles, but these are relatively rare on the roads and therefore easily distinguished already by the first two sides;

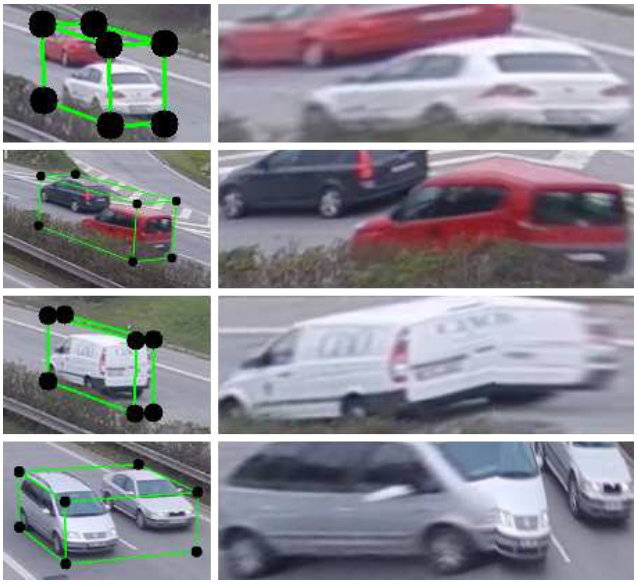


Figure 4. Some detection errors examples which can negatively affect resulting re-identification. **left**: Incorrect detection results. The bounding box is created over a group of vehicles or is deformed within a vehicle. **right**: Results of the inaccurate side and front 3D bounding box faces combination.

however, for some special cases, all three sides could be considered in the same manner). These two parts are projected on a 2D surface and combined together for a compact and practical single-image representation (Figure 2, bottom three rows). The advantage of this approach is efficient background noise reduction without any extra usage of CPU computation time which would be necessary for background noise removal if simple 2D bounding box was used. Around 5 % of the combined image content is background by using 3D bounding box instead of 25 % as is illustrated by Figure 3.

The detection system is not perfect and up to 5 % of vehicles (depending on traffic density, camera viewpoint, etc.) could be mis-detected (Figure 4). These detections are unrolled into 2D representations which do not match with actual vehicles and besides introducing small extra computational load, they do not constitute a considerable harm to the system performance.

The main steps of the fingerprinting and data extraction process are visualized in Figure 5. Most of the existing solutions previously mentioned have not taken into account information about the color characteristics of the vehicle. In some cases, it could be reasonable mainly in order to reduce the data amount that has to be stored. We decided to use this information because of two reasons: it significantly speeds up re-identification process and it reduces false positive re-identification cases. Color histograms are computed for each of the RGB channels of the vehicle image. More precisely, color histogram consisting of 16 bins for each of

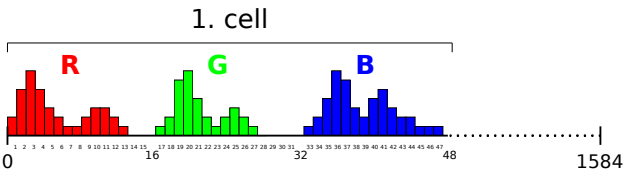


Figure 6. Color histogram vehicle fingerprinting representation result. A vector of 1,584 features (48 features per one 6×2 grid cell with 50 % overlap makes 11×3 extractions) per one side-front combined image is created. Grid cell color 16 bin histogram RGB values are concatenated consecutively into the vector.

Figure 7. Color histogram, regression result only. Relatively high regression result was computed (1.0 is the best match result, -1.0 the opposite). Color information is not sufficient for deciding the re-identification problem.

the 6×2 grid cell is computed (the cars are assumed to be big enough to be meaningfully split into the grid). Each neighboring cell has a 50 % overlap in both of the x and y axes. It follows that 1,584 color features per one side-front combined vehicle image from overall 33 grid sub-cells are computed. The number of combined images per one vehicle depends on the video shot taken (scene, frames per second, ...) and on the vehicle detection accuracy. On average, it is around 25 combined images per one vehicle. The color histogram feature vector is shown in Figure 6 for illustration.

The information about the vehicle color only is not sufficient in order to achieve a successful re-identification as can be seen in Figure 7 (though it is fast and eliminates many candidate pairs based on the vehicle color). Therefore, we added histogram of oriented gradients [2]. This information is stored separately. The histogram of oriented gradients is computed for all combined images belonging to a vehicle. The image is divided into a 12×6 grid. For each cell, the 9-bin HOG is computed with 50 % grid cell overlap. Overall, 1,449 features for one combined image from total of 161 grid sub-cells are computed.

In order to speed up the re-identification process (mainly the search in the vehicle database), average information of all combined images belonging to the vehicle for both of the color histogram and histogram of oriented gradients is computed.

3. Proposed Vehicle Re-Identification Algorithm

In our approach we used a linear SVM classifier [6] for learning the recognizer of vehicle matches. Its advantage is that it can be quickly trained on a large amount of training data with a large amount of features per one training in-

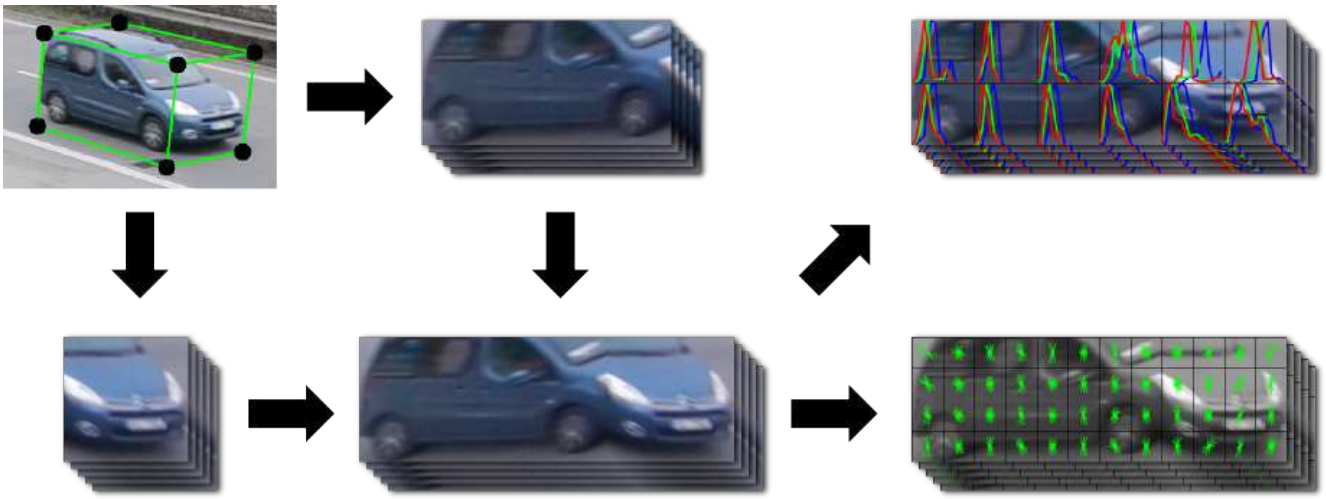


Figure 5. Illustration of the data extraction and fingerprinting process. Firstly, a vehicle is detected in the video. The detection result is represented as a 3D bounding box (green block in the top left picture). Then, the side and front faces of all bounding boxes belonging to the same vehicle are extracted and warped into a plane. Corresponding pairs of rectified side and front 3D bounding box faces are combined together. Finally, color histograms and histograms of oriented gradients are computed from each of the images in the set. The number of elements in the combined set depends on the video shot taken and on the vehicle detection accuracy. HOG and color histograms are plotted without overlap for clarity.

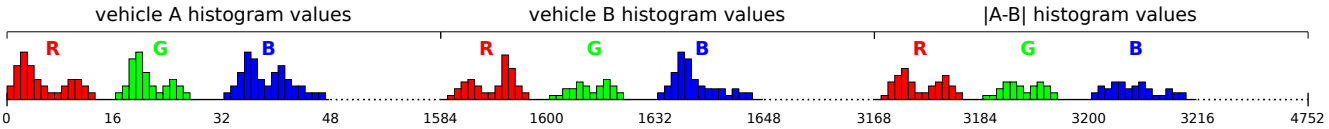


Figure 8. The color histogram feature vector concatenated from A and B vehicle feature vectors used for the training and for the regression. Vehicle A, vehicle B, and their differential color histogram feature vectors are concatenated. The resulting vector contains 4 752 values. The histogram of oriented gradients feature vector concatenation is constructed identically.

stance. The regression is also faster using the linear classifier compared to non-linear support vector machine regression.

Five video shots taken at roughly the same spot, from different angles, captured by different camcorders differently zoomed were recorded to obtain a sufficient amount of training data simulating different environments. Overall, over 800 vehicles were captured.

A simple GUI interface was created for annotating the ground-truth training data. It allows to pair the same vehicles easily, while minimizing time per vehicle (Figure 9). Information about these pairs is stored and used for training afterwards.

As previously mentioned, each vehicle is described by the color histogram and the histogram of oriented gradients vectors created from a set of combined images belonging to the vehicle and one average vector for both of these kinds of vectors. Training data are based on these vectors. Respectively, they are concatenated together plus one differential vector is added (Figure 8). Concatenated vectors are created for positive vehicle pairs and negative vehicle pairs. Positive vectors are randomly combined together by choosing from the positive vehicle pairs (Figure 10) vector set only. In this case, there were around 12,000–16,000 of them generated. Negative vector concatenations are created from randomly generated vehicle pairs chosen from camera A and B. Two times more negative pairs were created than the positive ones in order to tighten and refine the resulting regression. HOG and color histogram models are stored separately after the training procedure.

The re-identification itself consists of several steps. First, the average color histogram vector of a vehicle that is to be found in another vehicle set simulating a different camera is used for the first round regression. Vehicles with positive first round regression results are tested in the second round regression where the average HOG vector is used. Vehicles with both of the regression results positive are added to another set and are considered as highly potential positive re-identification results. For example, when searching in around 300 of vehicles, the potential set contains between 5–10 vehicles depending on their unique visual characteristics. All vehicles in the set are then matched by using individual input images, not the averaged feature values. Random pairs of individual images are matched against each other; vehicle with the highest score above a positive re-identification threshold is considered to be successfully re-

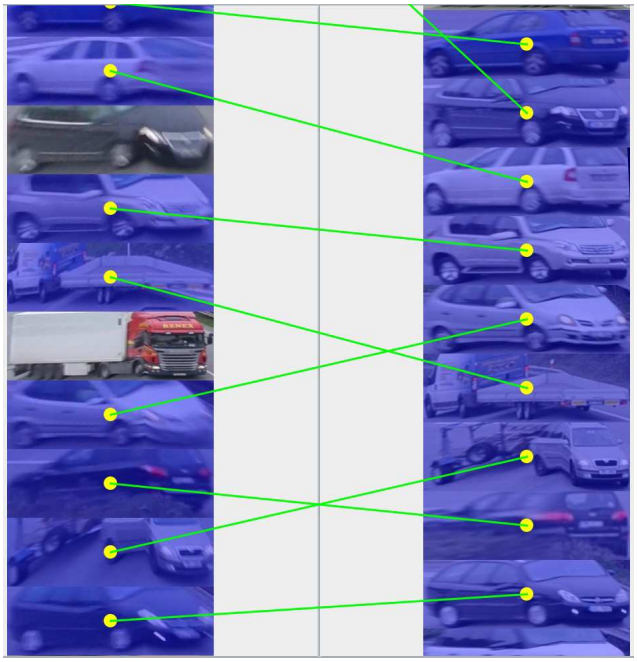


Figure 9. Ground truth annotation GUI. Creation of pairs of the same vehicles obtained from different cameras. Blue colored vehicles connected with the green line are paired. Unselected vehicles do not have its counter part and can not be paired. Data created by the GUI is used for training the linear classifier.

Figure 10. Randomly generated positive pairs of the same vehicle for the linear classifier training. **left:** Camera A, **right:** Camera B.

identified.

4. Experimental Results

Figure 11 shows examples of average results of the deciding regression consisting of five random subregressions between 10 000 randomly chosen pairs of vehicles. This illustration is meant for visual assessment of the performance of the algorithm and for showing the meaning of the regressed numerical responses.

Figure 11. Results (numbers in the middle) of deep regressions (several random regressions per one vehicle pair) from 10,000 randomly generated vehicle pairs. The more similar the vehicles are, the higher regression result is computed. From the experiment it can be assumed that almost all positive regression results belong to similar-looking vehicle pairs.

In order to perform a quantitative evaluation of the algorithm, we pre-selected semi-automatically 1,232 pairs likely to be matching vehicles. We constructed a web interface and crowd-sourced people's opinion of vehicles that "are

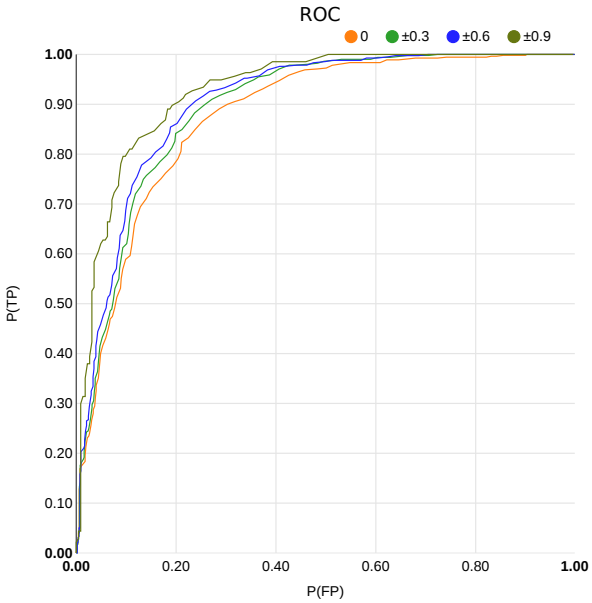
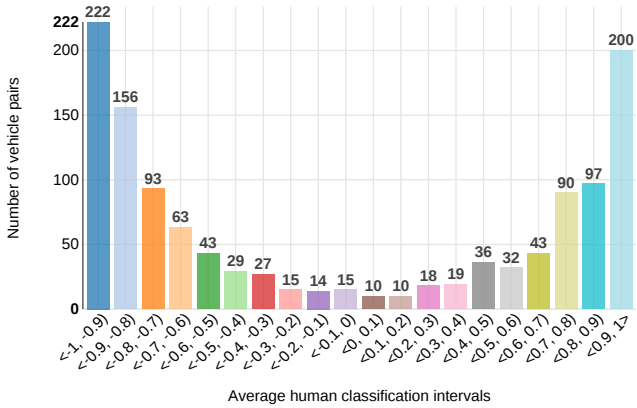


Figure 12. Four ROC curves were computed with different positive/negative average classification thresholds. For example, the blue curve was computed using an interval $< -1, -0.6)$ for negative average human classification and $(0.6, 1 >$ for positive average human classification.



Average human classification intervals

Figure 13. Histogram showing the results of people's opinion about 1,232 vehicle pairs whether "are likely to be the same vehicle". An average annotation result for each pair was computed from obtained -1 (different) and 1 (the same) values.

likely to be the same vehicle". This way, we obtained 24,000 individual annotations by approximately 500 people. The annotation results are visualized in Figure 13. The figure shows that about one third of the data were annotated unanimously, and many of the pairs were recognized with a high majority of votes. However, the dataset is quite challenging in the sense that a non-negligible part of the data was labeled ambiguously. This kind of labeling shows to be useful, because it stratifies the data according to the confidence level of the human annotators. We make this dataset

Figure 14. Examples of pairs wrongly identified as identical by the algorithm. The differences are very subtle and difficult to cover by the simple (and fast) linear model.

publicly available upon publication of this paper[1].

Figure 12 shows the receiver operating characteristic – ROC [7]. The ROC curve shows the performance of the trained classifier on the human-annotated dataset of difficult cases. It should be noted that the dataset consists of 1,232 most similar vehicles out of the 645,840 possible pairwise combinations which can be made from the original vehicle samples – 828 (camera A), 780 (camera B). The performance of the classifier (in terms of ROC) in natural use would be therefore "close to ideal" and hard to read. This dataset therefore provides a challenging and pessimistic estimation of the classifier performance. Despite that, 60 % of matches can be retrieved (TPR) with only about 10 % of false positives included. Such performance on the difficult dataset promises reliable analysis based on the proposed re-identification method. Some of the false positive re-identification results are shown in Figure 14.

The re-identification time depends on the number of vehicles stored in the database and on the positive/negative re-identification threshold setting. It takes around 70 milliseconds of the CPU computation time to compute HOG and color histogram feature vectors for one vehicle combined image set. When searching in a database of 400 vehicles, another 15 milliseconds (on average) for an attempt to find the corresponding vehicle is needed.

5. Conclusions

In this paper, we presented a simple and effective solution to the vehicle re-identification problem. Having 3D bounding box from the vehicle detection (obtainable from an existing real-time method), we are able to extract maximum of the useful vehicle data which are represented as color histogram and histogram of oriented gradients feature vectors. The main part of the re-identification process is based on regression by a linear classifier.

Using this approach, we were able to achieve 60 % re-identification true positive rate at 10 % false positives ac-

[1] https://medusa.fit.vutbr.cz/traffic/datasets/

curacy (on a challenging subset of the test data) in 85 milliseconds of CPU (i7) computation time per one vehicle. It means that in the worst-case scenario it is possible to fluently re-identify 12 vehicles per second.

An accurate vehicle re-identification in video brings a new view on obtaining real-time traffic information, traffic statistics and other important traffic-related data. A few questions are open and leave space for future work. The combination of the feature-based data representation used in this paper is surely worth attention and can inspire other similar works. The solution presented in this work has a high potential to be deployed into full operation by constructing a higher-level system observing traffic streams and measuring interesting characteristics of the traffic flow.

Acknowledgment

This work was partially supported by TACR grant TE01020415 "V3C", by TACR project "RODOS", TE01020155, and by The Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPU II); project IT4Innovations excellence in science – LQ1602.

References

[1] C. Arth, C. Leistner, and H. Bischof. Object reacquisition and tracking in large-scale smart camera networks. In *Distributed Smart Cameras, 2007. ICDSC '07. First ACM/IEEE International Conference on*, pages 156–163, Sept 2007. 1

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893 vol. 1, June 2005. 2, 3

[3] I. O. de Oliveira and J. De Souza Pio. Object reidentification in multiple cameras system. In *Embedded and Multimedia Computing, 2009. EM-Com 2009. 4th International Conference on*, pages 1–8, Dec 2009. 1

[4] M. Dubská, A. Herout, R. Juránek, and J. Sochor. Fully automatic roadside camera calibration for traffic surveillance. *IEEE Transactions on Intelligent Transportation Systems*, PP, 2014. 1, 2

[5] M. Dubská, J. Sochor, and A. Herout. Automatic camera calibration for traffic understanding. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2014. 1, 2

[6] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. 3

[7] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, June 2006. 6

[8] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi. Traffic monitoring and accident detection at intersections. *Intelligent Transportation Systems, IEEE Transactions on*, 1(2):108–118, 2000. 1

[9] S. Lämmer and D. Helbing. Self-control of traffic lights and vehicle flows in urban road networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(04), 2008. 1

[10] J. d. Ortuzar and L. G. Willumsen. *Modelling transport*. 2011. 1

[11] M. Saghafi, A. Hussain, M. Saad, N. Tahir, H. Zaman, and M. Hannan. Appself-controlearance-based methods in re-identification: A brief review. In *Signal Processing and its Applications (CSPA), 2012 IEEE 8th International Colloquium on*, pages 404–408, March 2012. 1

[12] T. Tan and J. Kittler. Colour texture analysis using colour histogram. *Vision, Image and Signal Processing, IEE Proceedings* -, 141(6):403–412, Dec 1994. 2

[13] D. Vallejo, J. Albusac, L. Jimenez, C. Gonzalez, and J. Moreno. A cognitive surveillance system for detecting incorrect traffic behaviors. *Expert Systems with Applications*, 36(7):10503–10511, 2009. 1