# Feature-based Level of Service Classification for Traffic Surveillance

Felix Pletzer Roland Tusch László Böszörményi Bernhard Rinner Oliver Sidla Manfred Harrer Thomas Mariacher

***Abstract*— A novel level of service (LOS) estimation approach based on the extraction of three local visual features is presented. The feature set comprises KLT motion vectors and Sobel edges, and is fed into a Gaussian radial-basis-function (GRBF) network to classify the prevailing LOS. The whole approach is designed and implemented to run on smart cameras in real-time and has been evaluated with a comprehensive set of real-world training and test video data from a national motorway. The evaluations in daylight environments have shown an average accuracy of LOS classification of 86.2% on an Atom-based smart camera, with a maximum reachable processing frame rate of 12.5 frames/sec. Incorrect classified samples differed from the ground truth by only one level. The comparisons are done with observation data from sensors utilizing a combination of Doppler radar, ultrasound, and passive infrared technologies.**

## I. Introduction

Over the past four decades, growing traffic has lead to the ubiquitous problem of congested roads. Road operators are increasingly interested to improve the level of service (LOS) on their roads by selectively extending the road network and dynamically controlling the speed and routes of vehicles in hot spot areas. To accomplish these tasks well, careful automatic monitoring and analysis of the prevailing traffic is needed.

Various types of methods and sensors exist for monitoring the traffic on the roads. On Austrian motorways, triple-tech traffic detectors, road toll systems, and surveillance cameras are mainly used for this purpose. The triple-tech detectors utilize a combination of Doppler radar, ultrasound, and passive infrared technologies to periodically determine the individual vehicle speed, class, occupancy time, and length per road lane. The surveillance cameras are usually controlled by human operators on demand and do not perform automatic measurements. Since the triple-tech sensor infrastructure is only dense in a few hot spot areas and rather sparse in other regions, there is increasing demand for utilizing cameras as vision-based sensors for estimating the level of service.

In a case study we address this demand by using new smart cameras and existing surveillance cameras for LOS estimation within two defined test areas. For each camera, the traffic state is computed individually, using the feature-based LOS classification method discussed in this paper. The overall goal of utilizing vision-based LOS estimation is to improve the spatial and temporal accuracy of traffic messages. In subsequent processing steps, multi-sensor fusion and traffic flow modeling methods are applied to combine LOS estimations obtained from cameras with observation data obtained from other traffic sensors.

This paper presents our LOS estimation approach for smart cameras performing video analysis in the uncompressed domain in real-time. In a given analysis area within the FOV of the camera, it uses the Kanade-Lucas-Tomasi (KLT)[1], [2], [3] feature tracker for computing motion features, and Sobel-based edge detection for gathering edge features. The LOS is then estimated using a Gaussian radial-basis-function (GRBF) network [4], taking the following three local features as input:

- F1: median KLT vector length (velocity)
- F2: average KLT vector length per frame (velocity)
- F3: average block-based edge occupancy per frame (density)

In our implementation we use OpenCV[5] for feature extraction and the data mining workbench Weka[6] for LOS estimation. The estimated levels of service are finally compared to per-minute aggregated observations taken from triple-tech sensors at the same location. The implementation of the algorithm was evaluated under real-time constraints on a smart camera equipped with a low-power Intel Atom processor. The evaluations utilized a comprehensive, real-world data set in daylight conditions without any sight disturbances. The results have shown that our approach provides a correct LOS classification rate of 86.2% on average.

The remainder of this work is organized as follows. Section II discusses related work in the area of vision-based vehicle speed and density estimation. In section III, our approach for extracting the three features and classifying the level of service is presented. Experimental results on a comprehensive training and test data set are presented in section IV. Finally, section V concludes this contribution with some perspectives about our future work.

This work was supported by Lakeside Labs GmbH, Klagenfurt, Austria and funding from the European Regional Development Fund and the Carinthian Economic Promotion Fund (KWF) under grant KWF-20214/17097/24774.

F. Pletzer and B. Rinner are with Institute of Networked and Embedded Systems, Alpen-Adria-Universität Klagenfurt, 9020 Klagenfurt, Austria {felix.pletzer, bernhard.rinner}@aau.at

R. Tusch and L. Böszörményi are with Institute of Information Technology, Alpen-Adria-Universität Klagenfurt, 9020 Klagenfurt, Austria {roland.tusch, laszlo.boeszoermenyi}@aau.at

O. Sidla is with SLR Engineering OG, 8020 Graz, Austria os@slr-engineering.at

M. Harrer and T. Mariacher are with ASFINAG Maut Service GmbH, 1120 Vienna, Austria {manfred.harrer, thomas.mariacher}@asfinag.at

## II. Related Work

Visual speed estimation has been studied by many research groups. Typically, estimation of traffic speed and density is based on vehicle tracking which either relies on motion analysis using background models, feature tracking, or vehicle detection.

The idea of background modeling is to segment moving foreground objects from the background. Many background models have been proposed, such as [7] or [8], that adapt to changing light and weather conditions. In [9] the authors combine simple frame differencing with a probability density function to estimate the segmentation of background and objects.

Feature-based tracking methods typically use corner features for vehicle tracking. The algorithm described in [10] employs Kalman filtering and correlation testing for tracking the features. A grouping module groups the features in order to segment the individual vehicles.

Vehicle detectors [11] are usually based on learning algorithms trained to detect vehicles in an image. In [12], [13], for instance, the authors use classifier grids with adaptive online learning for detecting cars. The velocity of vehicles is calculated using the feature-based KLT algorithm. In [11] the authors discuss the Principal Component Analysis (PCA) and Histogram of Gradients (HOG) approaches for vehicle detection.

Solutions that use background models for motion analysis usually perform well in free flow situations when there are not too many vehicles close to each other [9]. The quality of motion estimation usually degrades if traffic density is very high. Feature-based tracking algorithms, such as KLT, use feature matching to compute the optical flow. Clustering motion vectors to track vehicles is, however, a complex and error-prone task, since the number of detected features usually varies depending on contrast and lighting of the images. Vehicle trackers utilizing vehicle detectors have shown promising results. However, the detection rate of car detectors is usually affected by occlusions and difficult weather conditions. Therefore, in [14], [15] multiple sensor data is exploited and co-training to improve the classification rates of vehicle detection is performed. An alternative approach to sensor fusion for vehicle classification is presented in [16].

In [17], the authors employ texture features and edge features for traffic density estimation. Using a 21-dimension feature, Hidden Markov Models (HMM) are trained to estimate the traffic density state. The method was evaluated for different weather conditions and shows an average accuracy of 95.6 %.

In this paper we present a new approach for vision-based level of service (LOS) estimation on motorways. In contrast to most other approaches, our method does not require a background model and does not rely on vehicle tracking. Instead, we use statistical features, generated from the KLT motion vectors and edge occupancy in a small analysis area. Using the extracted features we train a GRBF network for classification of the LOS.

TABLE I

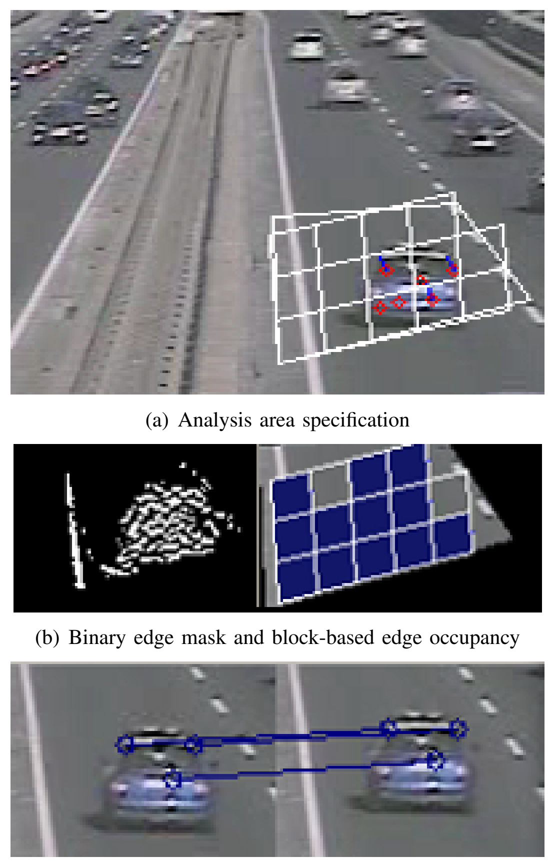
LOS level classification for a single lane.

| Level | 1 Lane | |
|---|---|---|
| | Velocity (km/h) | Density (v/km) |
| 1 (free flow) | $[80,\infty]$ | $[0,20]$ |
| 2 (heavy) | $[80,\infty]$ | $]20,50]$ |
| 3 (queuing) | $[30,80[$ | $[0,50]$ |
| 4 (stationary) | $[0,30[$ | $]50,\infty]$ |

## III. LOS Classification Approach

### A. LOS definition and detection

LOS is a qualitative measure that describes the operational conditions of a segment or traffic stream. In real-time estimation of LOS, four classes of motorway traffic are defined: free-flow (level 1), heavy (level 2), queuing (level 3), and stationary traffic (level 4). The levels are computed in dependence of average vehicle velocity and density on the individual lanes. Table I illustrates the LOS level criteria for a single lane on the Austrian motorways. Velocities are given in km/h, densities in vehicles/km.



(a) Analysis area specification

(b) Binary edge mask and block-based edge occupancy

(c) KLT feature tracking

Fig. 1. Feature extraction using KLT-based optical flow and edges in an analysis area.

Automatic LOS detection systems typically rely on various individual or multiple sensor sources such as inductive loops, magnetic, radar, ultrasound, infrared, or laser sensors. The triple-tech sensors - taken as reference for our evaluation - use a combination of Doppler radar, ultrasound, and passive infrared sensor sources. Using velocity, vehicle count, and occupancy measurements over a certain observation period (e.g., one minute), the corresponding LOS level can be calculated. Vision-based LOS detection systems usually rely on vehicle tracking to estimate the average speed and density of the traffic. One of the main advantages of vision-based LOS detection is the usage of the ubiquitous video surveillance camera infrastructure which is already available on motorways.

### B. Classification features

In contrast to most vision-based LOS detectors, our method does not rely on vehicle tracking. Instead it uses KLT-based feature tracking and edge occupancy statistics for LOS classification. The method works on a rhomboid-shaped analysis area as illustrated in Figure 1(a). As shown, the analysis area is further divided into rhomboid-shaped sub-blocks of 256 pixel size. Instead of tracking the individual vehicles within the entire frame, our method applies KLT feature tracking locally and calculates the motion vectors within the analysis area.

For a given observation period (e.g., one minute) two motion-based features are calculated: (1) the median of the length of all motion vectors, and (2) the average length of the motion vectors per frame. For most cases, when the number of motion vectors is high compared to infrequent outliers, the median motion vector length correlates well with the average velocity of the vehicles. However, in cases where the number of motion vectors is low (free or stationary traffic), the median of vector lengths is disrupted by a small number of outliers not removed by the outlier detection. In contrast, the average motion length per frame does not provide an accurate motion estimation but reliable detection of periods where the overall amount of motion is very low.

The third feature is the average edge occupancy. Using the Sobel operator and simple thresholding, a binary edge image (Figure 1(b)) of the analysis area is calculated for every frame. Using the binary edge image, our algorithm counts the number of edge pixels for each block of the analysis area. If the amount of edge pixels exceeds the empirical threshold of 15%, a block is considered active. Using the number of active blocks in each frame, the relative amount of active blocks is averaged over the observation period. The edge occupancy feature correlates with the density of the traffic and is especially useful to distinguish between free-flow traffic, where edge occupancy is close to 0% and stationary traffic, where edge occupancy is close to 100%.

Using the statistical features of motion vectors and edges instead of vehicle tracking for LOS detection has two main advantages. First, the method is not strongly affected by occlusions and segmentation problems when traffic density is high since tracking is only performed on KLT features. And second, the method is also well suited for embedded devices such as smart cameras, since analysis is only performed in a small analysis area.

### C. Feature extraction

For computing the described classification features we developed an OpenCV-based feature extractor. The feature extractor uses video data from an image sensor or MPEG-4 stream and outputs the features used for determining the LOS. For each frame, it performs KLT-based feature tracking as well as Sobel-based edge detection for a predefined analysis area. All image processing is performed on 8-bit grayscale images. Each input frame is masked with the binary mask of the analysis area. After extracting the KLT-feature points from the masked input frame, the feature extractor calculates the motion vectors with respect to the previous frame. Therefore, a pyramidal implementation of KLT feature tracker [18] is used to calculate the matching for the KLT feature points. The matching of feature points is illustrated in Figure 1(c). Although KLT-based feature tracking is relatively robust, outliers caused by mismatches occur. For that reason our algorithm ignores feature points that lie close to the border of the analysis area to avoid potential mismatches when a vehicle exits the analysis area. Furthermore, it performs direction-based outlier detection that rejects any motion vectors not pointing in the direction of the traffic flow. To determine the direction of the traffic flow, a predefined number of vectors is collected to initialize a direction histogram. For generating the feature set, described in section IV-A, we used a direction histogram with 10 bins. Using this histogram, vectors not included in the dominant bin are rejected as outliers.

As described in section III-B, we use statistical features calculated over a certain observation period for LOS classification. Therefore, our algorithm stores the vector lengths of the valid motion vectors to a sorted observation buffer. Each time the observation period expires, the feature extractor retrieves the median value from the buffer and empties the buffer. The average length of the motion vectors per frame is calculated from the average motion vector length in the analysis area. For each frame it obtains the average motion vector length and adds it to the aggregated value. When the observation period expires, the average motion vector length is obtained by dividing the aggregated value by the number of frames of the observation period.

To obtain the average edge occupancy, the algorithm applies the Sobel operator to each masked input frame. After that, it converts the resulting edge image to a binary image. For each block of the analysis area, the algorithm counts the number of edge pixels. If the number of pixels exceeds the empirical threshold of 15%, a block is considered active. The algorithm aggregates the percentage of active blocks over the observation period and calculates the average block occupancy when the observation period expires. The extracted features are finally fed into a Gaussian radial-basis-function (GRBF) network that performs the LOS classification task.

## IV. Experimental results

### A. Data set

For training and testing the LOS classification method, we used a 12 hours (from 7AM to 7PM) MPEG-4 video

TABLE II

TECHNICAL PARAMETERS OF THE SMART CAMERA.

| | |
|---|---|
| Processor | Intel Atom 1.6 GHz |
| Main Memory | 1 GB |
| Sensor | 1280 x 1024 color CCD |
| System | Ubuntu Linux 10.04 |

TABLE III

NET EXECUTION TIMES OF THE FEATURE EXTRACTOR PER FRAME.

| | |
|---|---|
| Min. | 29 ms |
| Max. | 92 ms |
| Average | 36 ms |

stream. The video stream was recorded in February 2011 on a national motorway and contains multiple occurrences of all four LOS classes. The analyzed video has a resolution of 352x288 and a frame rate of 25 frames per second. For the analysis, the frame rate was downsampled to the half frame rate to obtain motion vectors with reasonable length. The reference velocity, vehicle count, and LOS data used as ground truth were obtained from the national motorway authority. This ground truth data is derived from triple-tech traffic detectors mounted at the same location as the surveillance camera. For determining the LOS, the sensor data is aggregated over a one minute observation period.

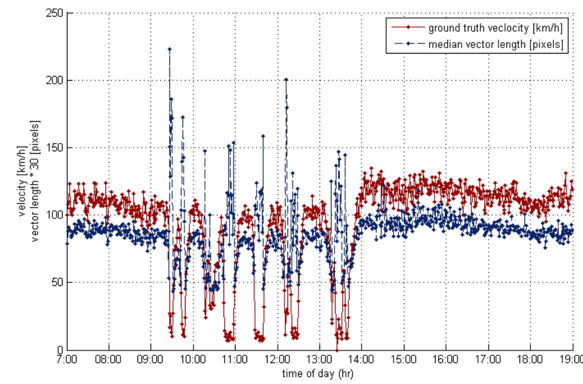
### B. Evaluation environment

The proposed approach was evaluated under real-time constraints on a custom smart camera. Table II illustrates the main technical data of this camera. It is equipped with an Intel Atom processor attached to an SXGA capable color CCD sensor. For evaluating our algorithm with the predefined analysis area, we used frames with CIF resolution (352x288 pixels).

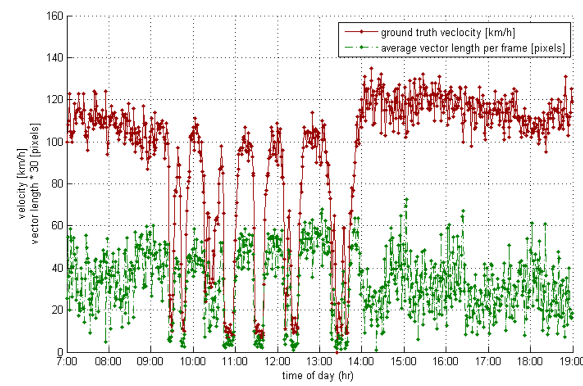
Table III shows the execution times of the feature extraction algorithm per frame. On the smart camera, the algorithm requires 36 ms on average to compute the features for a single frame. Adding 30 ms approximately needed for fetching a raw image from the camera sensor results in an average feature extraction time of 66 ms per frame. With our target frame rate of 12.5 frames/sec as reference, the average extraction time per second is 825 ms. This leaves approximately 175 ms per second for the LOS classification. Since the LOS classification is only done once per minute, there are approximately 10 seconds per minute left for this task, which is feasible for the implementation of the classifier.

### C. Velocity and density estimation

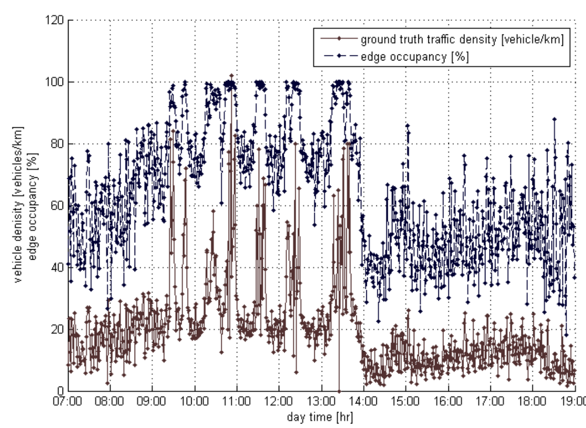
Using the developed feature extractor described in section III-C, the features were extracted from the video and synchronized with the reference values. For the feature statistics we chose a one minute time window to ensure consistency with the reference data. We defined an 3284 pixel (14 blocks) analysis area on the left lane with rear view on vehicles as shown in Figure 1(a).




(a) F1: Median vector length




(b) F2: Average vector length per frame




(c) F3: Average edge occupancy

Fig. 2. Extracted feature values from a 12 hours video recording and corresponding ground truth values. Note the different units on the y-axis.

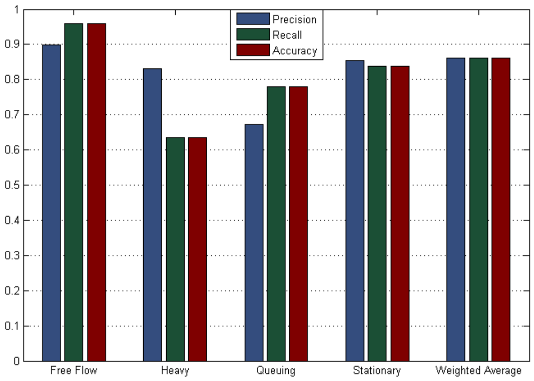
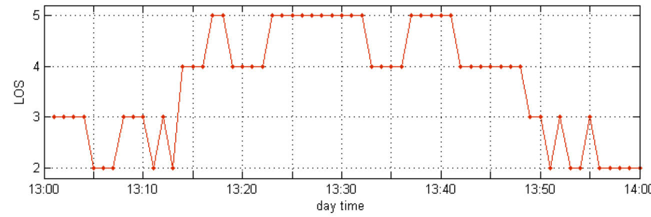



Fig. 3. Average precision, recall, and accuracy for the GRBF network trained with 10-folds cross validation

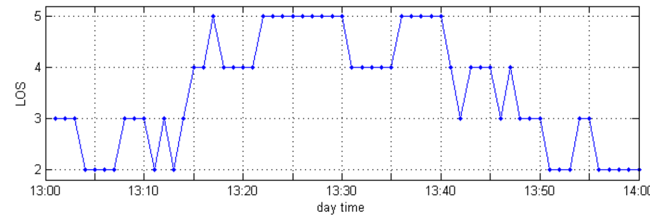
The extracted features and the corresponding reference values were exported to a file for further analysis and evaluation by the classifier. Figure 2 shows the extracted motion-based and edge-based feature values in comparison to the reference data. Figure 2(a) contains the median motion vector lengths and Figure 2(b) the average motion vector lengths per frame, both compared to the reference velocity values. The figures show that most of the time the median vector length provides a good estimation for the average velocity. However, especially in case of congestion when the average velocity is close to zero, the median vector length is not stable because of infrequent mismatches of the KLT tracker not eliminated by the outlier detection. For stationary traffic, when no motion is present, a small number of wrong motion vectors can disrupt the median value. Moreover, the figure indicates a degrading quality of motion-based features after sunset, i.e., between 6PM and 7PM.

The second motion-based feature, the average motion vector length per frame, is not only correlated with the average velocity but also with the traffic density. However, as shown in Figure 2(b), the average motion vector length performs only well for indicating periods where the velocity is very low. The reason is that mismatches of the KLT tracker occur very infrequently and therefore affect only a small number of frames.

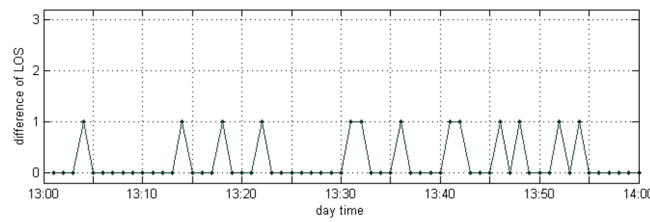
The third feature, the average edge occupancy, is shown in Figure 2(c) in comparison to the traffic density that was calculated from the reference values. For the shown analysis we used an empirically determined edge threshold of 15% (17 pixels) for setting a block active. It shows that the average edge occupancy provides a good estimate for the actual traffic density during daylight. Also, the figure shows highly uncorrelated variations of the average edge occupancy between 6PM and 7PM when no static ambient light was present. For the chosen analysis the edge occupancy was biased by 17% (corresponds to three blocks) due to the road marker on the left side (cf. Figure 1(a)).




(a) LOS values - ground truth




(b) Result of the feature based LOS classification method (output of the GRBF classifier)




(c) Absolute difference between ground truth and the feature-based LOS classification method

Fig. 4. Classification result for the one hour test set compared to ground truth

### *D. Level of service estimation*

The computed features were used to train a normalized Gaussian radial-basis-function network classifier. The GRBF network implementation [6] used for our results applies K-means clustering to estimate the centers and widths of the Gaussian functions and logistic regression for learning the classification model.

The feature set, described in section IV-C, was combined with the reference LOS data. The feature set was split into a 10-hours training and 1-hour test set. Since our LOS classifier is only intended to work during daylight, we removed all feature samples between 6PM and 7PM from the feature set. Furthermore, we defined a test set from 1PM to 2PM that contains samples of all four LOS classes to provide test data for the trained classifier. We used the feature set to evaluate the classification accuracy of several state-of-the-art classifiers, such as Bayesian network, decision tree, neural network and GRBF network. It showed that a GRBF network with $k = 4$ Gaussian functions provides the highest classification accuracy. For training the GRBF network classifier, we used a 10-folds cross-validation on the training data.

Figure 3 shows the average precision, recall, and accuracy for the individual LOS classes of the trained classifier. It shows that our LOS classifier provides an average precision of 86.1%, and an average recall and accuracy of 86.2%.

Both, free flow and stationary LOS classes show a high accuracy, while heavy and queueing LOS are more difficult to discriminate.

Also, we evaluated the trained GRBF classifier using the described comprehensive 1-hour test set. Figure 4(a) shows the classification result in comparison to the ground truth. On this test set, the classifier achieves an average accuracy of 78.3%. Figure 4(c) shows the absolute difference between the ground truth and the classification result, which is not greater than one. Therefore, on the one-hour test set, the LOS value of incorrect classified samples does never differ from the reference value by more than one level.

## V. Conclusions and future work

A vision-based level of service (LOS) classification approach utilizing a normalized Gaussian radial-basis-function (GRBF) network with three features based on Kanade-Lucas-Tomasi (KLT) motion vectors and Sobel edges was presented. All features are calculated locally in a predefined section of the frames. The features are fed into the GRBF network for estimating the four LOS classes free-flow, heavy, queuing, and stationary traffic in real-time.

Training and evaluation of the classifier were performed with a comprehensive, real-world data set in daylight conditions (i.e., an 11 hours video stream with corresponding ground truth from triple-tech traffic detectors) and have shown promising classification results. With cross-validation training, the classifier yields an average accuracy of 86.2%. Classification of free-flow and stationary traffic shows a high accuracy, while heavy and queuing traffic are more difficult to distinguish. Further evaluations on a one-hour test set also approve the quality of the implemented method. The presented LOS estimation method was evaluated on a smart camera equipped with an Intel Atom 1.6 GHz processor. For a target frame rate of 12.5 frames per second, the algorithm runs significantly faster than real-time.

As part of ongoing project work, we plan to integrate the proposed LOS classification method into runtime environments at the motorways. Therefore, we will also evaluate the accuracy of the presented method for different weather conditions, such as rain and snow. Moreover, in order to use the LOS classification method for arbitrary settings, a function that maps the length of motion vectors to a trained reference length is required. Such a mapping can be calculated from the intrinsic and extrinsic camera parameters. Future work in this field will also include a feedback loop for the LOS estimation on the smart cameras.

Based on previous work we will investigate online-learning and co-training methods to further improve the LOS estimation accuracy.

## References


[1] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *Proc. of the 7th Intl. Joint Conference on Artificial Intelligence*, 1981, pp. 674–679.

[2] C. Tomasi and T. Kanade, “Detection and tracking of point features,” Carnegie Mellon University, Tech. Rep. CMU-CS-91-132, April 1991.

[3] J. Shi and C. Tomasi, “Good features to track,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600.

[4] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4th ed. Academic Press, 2009.

[5] G. R. Bradski and V. Pisarevsky, “Intel’s computer vision library: Applications in calibration, stereo, segmentation, tracking, gesture, face and object recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2000, pp. 2796–.

[6] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed., I. H. Witten, Ed. Morgan Kaufmann, January 2011.

[7] C. Stauffer and W. Grimson, “Adaptive background mixture models for real-time tracking,” in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 1999, pp. 246–252.

[8] Z. Zivkovic, “Improved adaptive Gaussian mixture model for background subtraction,” in *17th International Conference on Pattern Recognition*, vol. 2, 2004, pp. 28–31.

[9] S.-C. S. Cheung and C. Kamath, “Robust background subtraction with foreground validation for urban traffic video,” *EURASIP Journal on Applied Signal Processing*, vol. 14, pp. 2330–2340, 2005.

[10] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik, “A real-time computer vision system for measuring traffic parameters,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 1997, pp. 495–501.

[11] O. Sidla, E. Wildling, and Y. Lypetskyy, “Vehicle detection methods for surveillance applications,” in *Proceedings of SPIE*, vol. 6384, 2006.

[12] P. M. Roth, S. Sternig, H. Grabner, and H. Bischof, “Classifier grids for robust adaptive object detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2727–2734.

[13] M. Pucher, D. Schabus, P. Schallauer, Y. Lypetskyy, F. Graf, H. Rainer, M. Stadtschnitzer, S. Sternig, H. Bischof, J. Birchbauer, W. Schneider, and B. Schalko, “Multimodal highway monitoring for robust incident detection,” in *13th International IEEE Conference on Intelligent Transportation Systems*, 2010, pp. 837–842.

[14] M. Godec, C. Leistner, H. Bischof, A. Starzacher, and B. Rinner, “Audio-visual co-training for vehicle classification,” in *Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, ser. AVSS ’10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 586–592. [Online]. Available: http://dx.doi.org/10.1109/AVSS.2010.31

[15] H. Bischof, M. Godec, C. Leistner, B. Rinner, and A. Starzacher, “Autonomous audio-supported learning of visual classifiers for traffic monitoring,” *Intelligent Systems, IEEE*, vol. 25, no. 3, pp. 15–23, 2010.

[16] A. Klausner, A. Tengg, and B. Rinner, “Distributed multilevel data fusion for networked embedded systems,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 4, pp. 538–555, 2008.

[17] J. Chen, T. Evan, and L. Zhidong, “A machine learning framework for real-time traffic density detection,” *Int. J. Patt. Recog. Art. Intel.*, vol. 23, no. 07, pp. 1265–1284, 2009.

[18] J.-Y. Bouguet, “Pyramidal implementation of the lucas kanade feature tracker: Description of the algorithm,” Intel Corporation Microprocessor Research Labs,” OpenCV Document, 2000.