

# Multi-view Data Aggregation for Behaviour Analysis in Video Surveillance Systems

Paweł Forczmański<sup>(✉)</sup> and Adam Nowosielski

Faculty of Computer Science and Information Technology, West Pomeranian University of Technology, Żołnierska Str. 49, 71-210 Szczecin, Poland  
{pforczmanski,anowosielski}@wi.zut.edu.pl

**Abstract.** Detecting restricted or security critical behaviour on roads is crucial for safety protection and fluent traffic flow. In the paper we propose an algorithm for the analysis of movement trajectory of vehicles using vision-based techniques. It works on video sequences captured by road cameras in multi-view mode. We integrate methods of background modelling, object tracking and homographic projection. Individual views are projected into a single, planar surface of road surface and then the detected movement path is compared with a template associated with an illegal movement. The effectiveness of the proposed solution is confirmed by experimental studies.

**Keywords:** Visual surveillance · Tracking · Multi-view integration · Movement path comparison

## 1 Introduction

There are four major causes of the road congestion in the contemporary road traffic [1,2]: (i) poor planning of transport routes, (ii) existence of bottlenecks, (iii) lack of the adaptation of existing infrastructure to the current traffic load, (iv) accidents.

The last of these reasons is mostly associated with the lack of awareness of drivers and their tendency to omit some inconvenient traffic regulations. Similarly as in the case of speed control systems, which force the driving at a safe speed, solutions for automatic critical behaviour detection should enforce the appropriate driving thus reducing the number of potential accidents. Such problem of the analysis and identification of vehicles motion patterns is referred in the literature as Vehicle Behaviour Analysis [3].

Illegal movements of vehicles can be detected using vision-based techniques applied to video sequences captured by road cameras. The main advantage of using computer vision techniques is their non-intrusive application not requiring the installation of the sensors directly onto or into the road surface [2]. Such system is also capable of the immediate (real-time) automatic response and alerting in case of the occurrence of an unfortunate accident.

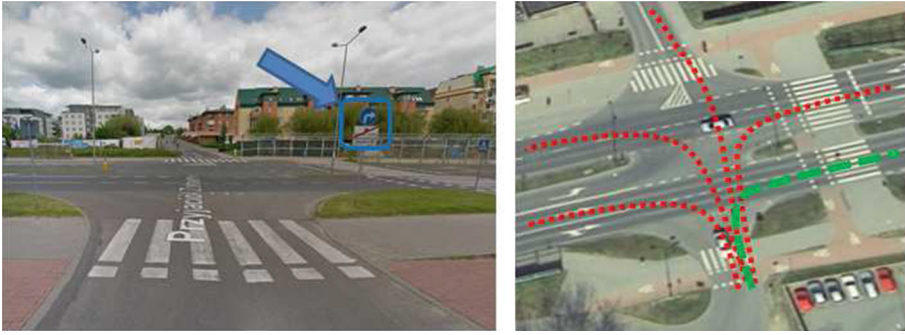
The image-based techniques have already been utilized for variety of tasks in Intelligent Transportation Systems (ITS) providing complete traffic flow information for the situations related to [3]: traffic management [4], public transportation, information service, surveillance, security and logistics management. The tasks successfully implemented with vision-based techniques include [2]: reading vehicle registration plates (ALPR - Automatic License Plate Recognition), vehicle counting, congestion calculation, traffic jam detection, lane occupancy readings, road accident detection, traffic light control, comprehensive statistics calculation, etc. Computer vision techniques are also increasingly willing utilized by driver assistant systems (ADAS - Advanced Driver Assistance Systems). Many vehicles are equipped with on board cameras which form the basis for systems such as [5,6]: TSR - Traffic Sign Recognition, CAV - Collision Avoidance (by pedestrians or surrounding vehicles detection and tracking), LDW - Lane Departure Warning (adaptive cruise control), and driver fatigue detection.

The main drawback of vision-based solutions is susceptibility to poor visibility conditions and occlusions [7]. Researchers, however, actively respond to the challenge and propose solutions that deal with those difficulties (e.g. occluded traffic signs recognition [8], behaviour analysis in multi-view environment [10]). In the paper we focus on the extension of our previous work on the automatic analysis of vehicle behaviour (presented in [2]) with the emphasis on poor visibility conditions and occlusions. The main contribution of the paper is a novel approach to the vehicle trajectory analysis on data aggregated from multiple views.

### 1.1 Approaches to Behaviour Analysis

Vehicle behaviour analysis in the surveillance context is mostly limited to the detection of restricted or security critical events on roads [2,9,11]. The task can be solved successfully with the analysis of moving vehicle trajectory [2,9,11]. Figure 1 presents an exemplary intersection with a compulsory right turn (adopted from proprietary Google Maps and Google Street View services). In the situation presented, the driver is obliged to follow the path approximately outlined by a green dashed line. Red trajectories denote possible but illegal and dangerous movements. As presented in the example two categories of trajectories are possible: the correct (legal) and the forbidden (illegal). Such distinction can be applied to all traffic situations. The trajectories, in terms of their geometry, may take a variety of shapes and may consist of different number of points. The key factor is the comparison with the template. Such template trajectory might be the appropriate or the forbidden one. In the first case, a discrepancy will reveal restricted behaviour. In the second case, the same discovery is possible through the similarity to a forbidden trajectory. In fact, it is specific to a road system that determines whether it is better to take as a template legal or illegal trajectory.

In video-based approaches the trajectory of moving vehicle is obtained through the vehicle detection and the successive tracking algorithm. The activity



**Fig. 1.** Compulsory right turn example and possible vehicle movements (red - illegal, green - appropriate) (Color figure online)

perception and the detection of abnormal events are detected by high level vision algorithms [9]. The following abnormal events detection can be distinguished [2]:

- illegal left and right turns,
- illegal U-turn,
- illegal lane change and violation of traffic line,
- overtaking in prohibited places,
- wrong-way driving,
- illegal retrograde,
- illegal parking.

## 1.2 Related Works

Most algorithms for behaviour analysis proposed in the literature consider the single view. Presented solutions can be divided into supervised and unsupervised methods [9]. In the first case, the manual intervention for specifying patterns of behaviour is required. In an unsupervised mode the algorithm learns abnormal activity from the sample data. The process is automatic and the outcome might be sometimes unexpected. Generally, the process requires a reasonable amount of data and is time consuming.

The trajectory-based solution for the illegal behaviour detection can be adopted to most locations. The greatest difficulty is the susceptibility to poor visibility conditions and occlusions. The use of one camera view is burdened with considerable risk when camera field of view is obscured with large or close object. Problems with vehicle detection and segmentation using the single view may adversely affect the location of a vehicle. Due to errors the vehicle may temporarily disappear and the calculated position may not always be determined reliably. Multi-view observation is resistant to such cases. When a tracked vehicle is occluded in one view and the tracking procedure is interrupted, the other views can be used to merge the information and link the interrupted parts of the trajectory.

The most similar solution to the one proposed in this paper is presented in [10]. Multiple camera views are used in [10] to remove occlusion and to extract abnormal vehicles behaviour more accurately. The vehicles trajectory analysis is based on support vector machine (SVM). The system is constructed using the distributed architecture. The analysis is performed using individual views only. Later the results are aggregated and supplemented using other views. This approach is different from the one presented here, which firstly integrates the information from all views, then calculates the trajectory.

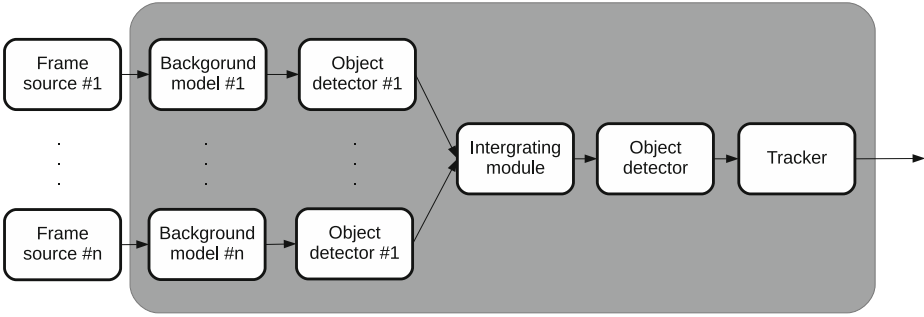
The trajectory analysis alone does not allow to discover some specific dangerous movements like: sharp brake, sharp turn or sharp turn brake. To detect these dangerous behaviours the velocity information is necessary. An exemplary solution allowing the detection of the above-mentioned events using the rate of velocity variation and the rate of direction variation has been proposed in [12].

## 2 Method Description

In this paper we present a visual surveillance system aimed at vehicle detection and tracking. Like any typical visual surveillance system aimed at gathering information about certain phenomena in order to execute or suggest certain actions, especially in case of situations dangerous to human life, health or property, proposed system's purpose is to alarm about probable hazardous road situations. Visual surveillance in this case is often realized using a closed-circuit television system (CCTV) that consists of static camera (or cameras) aimed at one fixed point in space. In order to simplify the problem, we assume that the focal length of each camera lens is constant as well (hence Pan-Tilt-Zoom cameras are excluded from our investigations). The solution proposed in this paper integrates the information about the movement of vehicles observed by more than one camera. During the development we assumed that vehicles are tracked only in the area covered by a specified number of cameras and we consider only those vehicles that are observed by assumed number of cameras. The proposed solution consists of the following modules (Fig. 2):

- background modeling - independently detects foreground areas in each view;
- object detection - determines silhouettes of moving objects and selects vehicles in each view;
- integrator - integrates information coming from all views (homographic projection);
- object detector - detects vehicles in projected (aggregated) view;
- tracker - estimates detected vehicles trajectories.

Since most of scenes observed by CCTV cameras are not static, the process of background separation has to take into consideration many different environmental conditions, such as variable lighting [13], atmospheric phenomena and changes caused by different actions. Hence, background modeling is a crucial task since its efficiency determines the capabilities of the whole system. Until now, many methods of background modeling have been proposed. They are based



**Fig. 2.** Scheme of processing in a multi-view environment

on different assumptions and principles, however all of them can be divided into two main categories: pixel-based and block-based approaches. The former class of methods analyses each individual pixel in the image, while the latter considers an image decomposed into segments (often overlapping). For each pixel or segment certain features are calculated and used later at the classification stage (into pixels or segments belonging to background and foreground). In many typical approaches, each detected object (or blob) is also tracked. The authors often assume that the movement is constant and the direction does not change in a considerable way within certain number of frames [14, 15], which further simplifies the algorithm. The last stage of processing may involve object recognition or classification. The selection of a method used at this stage depends mainly on the object type and its invariant features. For example, in a system presented in [16] each detected object is described by mean area it occupies – a similar approach is also applied in this work.

## 2.1 Background Modeling

In our solution, the background model employs a pixel-based approach similar to the one proposed in [13]. Here, every pixel is modeled by a set of five mixtures of Gaussians in R, G and B channels. According to the research, such number of Gaussians increases the robustness of the model in the comparison to the one presented in [17]. Similar approach, successfully employed for human motion tracking has been already presented in [18].

In our case, the first 200 frames from video stream are used for learning the parameters of the background model. The further frames are processed in a stepwise manner, and the parameters of the model are updated.

During the processing loop, every pixel from the current frame is checked against the existing Gaussians in the corresponding position in the model. If there is no match the least probable Gaussian is replaced by the new one using current pixel value as a mean value. Then, the weights of all Gaussians are updated according to the following rule: weights of distributions that do not correspond with the new pixel value are decreased, while the weights of



**Fig. 3.** Two camera views and their foreground masks obtained by background modeling

distributions that suite it are increased. Parameters of unmatched distributions remain unchanged. The parameters of the distribution which matches the new observation are updated according to the following formulas:

$$\mu_t = (1 - \rho)\mu_{t-1} + \rho X_t, \quad (1)$$

$$\sigma_t^2 = (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t), \quad (2)$$

$$\rho = \alpha\eta(X_t|\mu_k, \sigma_k), \quad (3)$$

where  $X_t$  is a new pixel value,  $\eta$  is a Gaussian probability density function,  $\alpha$  is a learning rate,  $\mu$  and  $\sigma$  are distribution parameters, and  $\rho \in \langle 0, 1 \rangle$ .

After that, each weight of each distribution is updated as follows:

$$\omega_t = \begin{cases} (1 - \alpha)\omega_{t-1} + \alpha & \text{if a pixel fits the distribution} \\ (1 - \alpha)\omega_{t-1} & \text{otherwise.} \end{cases} \quad (4)$$

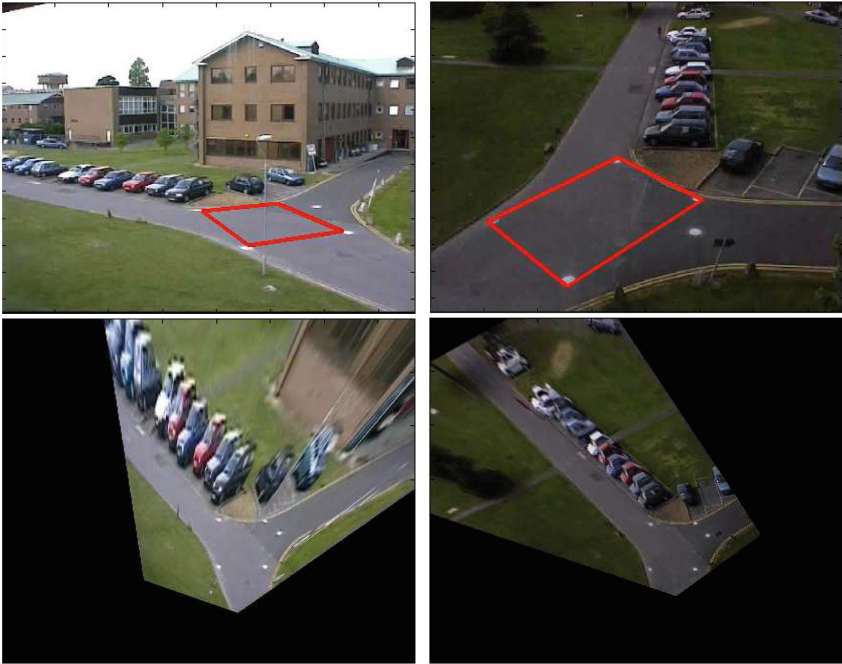
Background subtraction operation results in a binary image mask of possible foreground pixels which are grouped using connected components (see Fig. 3). Unfortunately, this approach does not suppress shadows and certain reflections from being considered as moving objects, thus it can cause certain serious problems, namely false detections of non-existent objects. In the proposed system we use a shadow detection and elimination method based on [19]. It assumes that casted shadow lowers the luminance of the point while chrominance is unchanged. This observation is valid for HSV color space, used in our solution.

## 2.2 Homographic Transformation

The proposed solution of integrating information from multiple cameras uses a simplified method of projection, originally described in [20]. As it was already mentioned, projective transformation allows for mapping of one plane to another. In our algorithm, we use it to find ground position of vehicles seen in different camera views. Projecting transformation is expressed by the equation:

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix}, \quad (5)$$

where  $x_1$  and  $y_1$  are the coordinates of a single point on the input plane,  $x_2$  and  $y_2$  are the coordinates on the output plane,  $H$  is a transformation matrix.

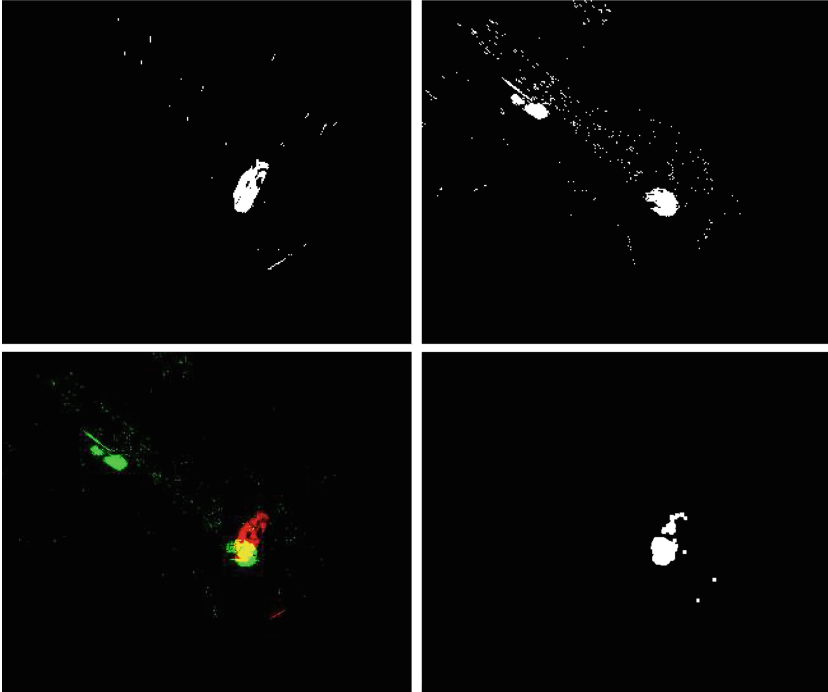


**Fig. 4.** Two camera views (with calibration rectangle marked) and their projections onto a planar surface

In order to calculate  $H$  by means of least squares method we need to collect four pairs of so called calibration points. Such an approach is a compromise between computational complexity and the quality of the resulting transformation. More precise results can be obtained by means of non-linear mapping, described i.e. in [20]. Exemplary calibration points, forming rectangles in two camera views are presented in Fig. 4.

### 2.3 Data Integration

The individual projections of foreground masks are taken as an input for data integration module. We assume that each foreground mask resembles a shadow of a moving object cast onto the ground plane. If we integrate (superimpose) all individual projections we obtain a complex image, where the common part represents the object viewed by each camera. All false detections, like shadows or reflections are often visible in single view only, hence they are not taken into consideration by this method.



**Fig. 5.** Foreground masks for camera 1 and 2 (upper row), their superposition (lower left) and output, projected, foreground mask (lower, right)

The blobs detected at this stage are described by their geometrical properties, among others by their centroids, which represent their ground position. An exemplary mapping for the data from PETS benchmark is presented in Fig. 5. As it can be seen, a car that is closer to the viewer is observed in two camera-views, hence its integrated blob is depicted in the final foreground mask. The other, further car, is visible in the second camera view only, hence it is not taken for the further processing. In the presence of multiple views, different strategies of joining individual views can be applied. Taking into consideration the environmental conditions and purpose, voting (majority) strategy (e.g. two-out-of-three) may be used.



## 2.4 Trajectory Calculation

In the next step we calculate the trajectories of foreground objects using a simplified Object Tracker. Objects, detected in integrated projection, are tracked from frame to frame in a stepwise manner. For each tracked object (labelled using unique number) we store an information about its bounding box and its position in current frame. We use historical data (previous frames) to decide about the object label. Besides such numerical data, the database contains, for each object, its binary mask (in each frame) and cropped video frame.

In order to match detected foreground blobs to tracked objects an association matrix similar to the one proposed in [21] is used. For all pairs of foreground blobs and tracked objects we measure Euclidean distance from last stored position of object to the center of the foreground blob. If a foreground blob intersects with last remembered bounding box of the tracked object we measure distance from the center of a bounding box to the center of the blob. After distance calculation between all pairs blob-object, object list is updated using blobs which are closest to them. In case when a blob has no matched object, a new tracked object is created. On the other hand when the object has not been associated to any foreground blob for several frames, it is removed.

## 2.5 Trajectory Comparison

The behaviour analysis based on the comparisons with the reference trajectory is frequently performed using the Hausdorff distance or its modifications (e.g. [2, 9, 22, 23]). The main advantage is the resistance of this measure to the different number of point coordinates of compared trajectories. This measure in the form of Modified Hausdorff Distance (MHD) proposed by [24] is also used in this paper. The MHD, in fact, combines (takes the maximum) two directional MHDs which are sometimes referred to as FHD (forward) or RHD (reverse). Characteristics of these measures are presented in graphical form in Fig. 6. There are two trajectories presented in these figures. The template trajectory has an upward direction (top right of Fig. 6). The analysed vehicle trajectory is a real trajectory extracted using multiple views (top left). For the presentation purposes the first few coordinates corresponding to the retrograde have been removed. The MHD measure presented in the middle of Fig. 6 has at the beginning high value which correspond to the RHD. The FHD value is comparatively small (in the first phase of the movement) since all tested trajectory points finds their equivalents in the template trajectory. Due to the similarity of the movement to template trajectory all the values are decreased.

The MHD, unfortunately, is unable to differentiate the direction of the movement. It considers only the mutual relationships of trajectory points which are treated as a set. To discriminate the trajectory direction we further analyse the X and Y projections of all coordinate points. The bottom parts of Fig. 6 presents projections of trajectory points onto X and Y axes. As it can be seen, both projections coincide. Hence we can decide about movement direction.

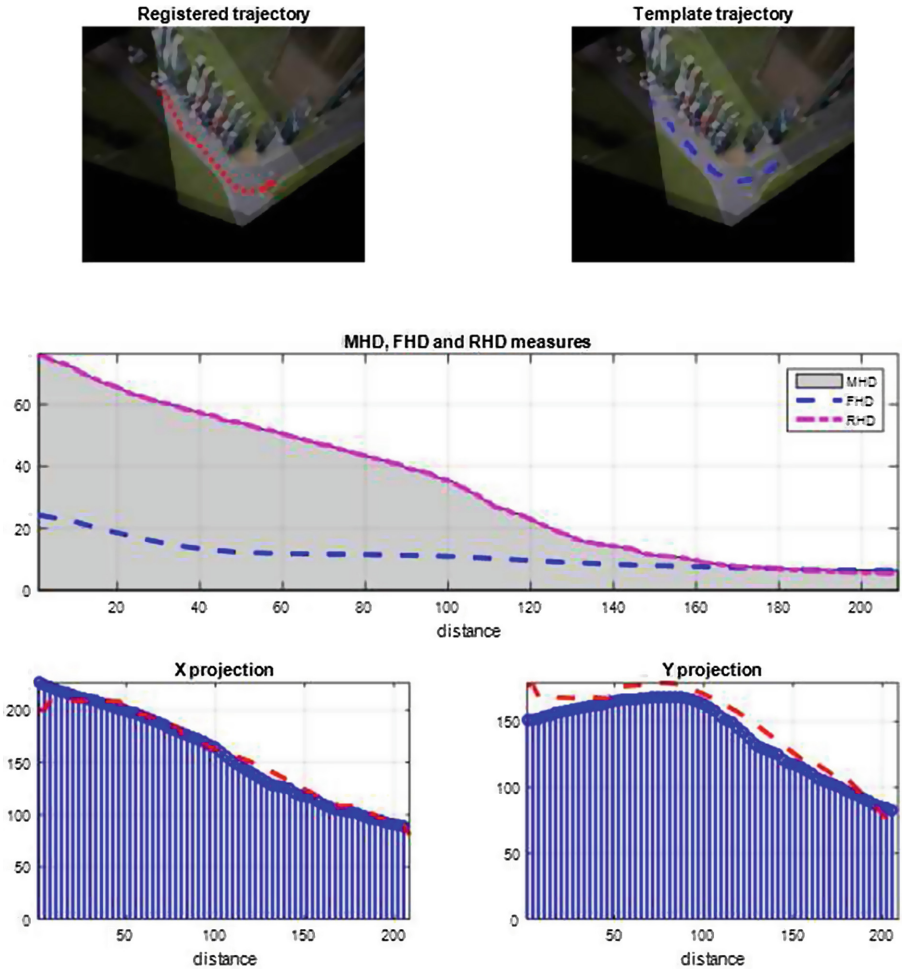


Fig. 6. Example of trajectory comparison

### 3 Conclusions

In this paper we proposed a method for the detection of restricted or security critical behaviour on roads by vehicle trajectory analysis. Our proposal contains two novel elements. The first one is a combined, multi-view background modeling approach that integrates foreground masks in order to more precisely calculate the trajectory of moving vehicles. The second element is an improvement of the original Modified Hausdorff Distance-based method by incorporating the X and Y projections in the final trajectory matching algorithm. Such solution solves the problem of the movement direction, specific trajectory configuration (found in e.g. roundabout case) and possible occlusions. Accompanied with ALPR technology the system can be a good deterrent from dangerous and illegal driving behaviour contributing for safety protection and fluent traffic flow.

## References

1. Munuzuri, J., Cortés, P., Guadix, J., Onieva, L.: City logistics in Spain: why it might never work. *Cities* **29**(2), 133–141 (2012)
2. Nowosielski, A., Frejlichowski, D., Forczmański, P., Gościewska, K., Hofman, R.: Automatic analysis of vehicle trajectory applied to visual surveillance. In: Choraś, R.S. (ed.) *Image Processing and Communications Challenges 7. Advances in Intelligent Systems and Computing*, vol. 389, pp. 89–96. Springer, Switzerland (2016)
3. Wu, J., Cui, Z., Chen, J., Zhang, G.: A survey on video-based vehicle behavior analysis algorithms. *J. Multimedia* **7**(3), 223–230 (2012)
4. Czajewski, W., Iwanowski, M.: Vision-based vehicle speed measurement method. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) *ICCVG 2010, Part I. LNCS*, vol. 6374, pp. 308–315. Springer, Heidelberg (2010)
5. Kovacic, K., Ivanjko, E., Gold, H.: Computer vision systems in road vehicles: a review. In: *Proceedings of the Croatian Computer Vision Workshop, Year 1, Zagreb, Croatia*, pp. 25–30 (2013)
6. Nowosielski, A.: Vision-based solutions for driver assistance. *J. Theor. Appl. Comput. Sci.* **8**(4), 35–44 (2014)
7. Song, H.-S., Lu, S.-N., Ma, X., Yang, Y., Liu, X.-Q., Zhang, P.: Vehicle behavior analysis using target motion trajectories. *IEEE Trans. Veh. Technol.* **63**(8), 3580–3591 (2014)
8. Forczmański, P.: Recognition of occluded traffic signs based on two-dimensional linear discriminant analysis. *Arch. Transp. Syst. Telematics* **6**(3) (2013)
9. Desheng, W., Jia, W.: A new method of vehicle activity perception from live video. In: *International Symposium on Computer Network and Multimedia Technology*, pp. 1–4 (2009)
10. Babaei, P.: Vehicles behavior analysis for abnormality detection by multi-view monitoring. *Int. Res. J. Appl. Basic Sci.* **9**(11), 1929–1936 (2015)
11. Hu, W., Xiao, X., Xie, D., Tan, T.: Traffic accident prediction using vehicle tracking and trajectory analysis. In: *2003 IEEE Proceedings of Intelligent Transportation Systems*, vol. 1, pp. 220–225 (2003)
12. Jiang, E., Wang, X.: Analysis of abnormal vehicle behavior based on trajectory fitting. *J. Comput. Commun.* **3**, 13–18 (2015)
13. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 246–252 (1999)
14. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(5), 564–577 (2003)
15. Okarma, K., Mazurek, P.: Application of shape analysis techniques for the classification of vehicles. In: Mikulski, J. (ed.) *TST 2010. CCIS*, vol. 104, pp. 218–225. Springer, Heidelberg (2010)
16. Li, L., Ma, R., Huang, W., Leman, K.: Evaluation of an IVS system for abandoned object detection on PETS 2006 datasets. In: *Ninth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, pp. 91–98 (2006)
17. Tian, Y., Feris, R.S., Hampapur, A.: Real-time detection of abandoned and removed objects in complex environments. In: *IEEE International Workshop on Visual Surveillance (in conjunction with ECCV 2008)*, Marseille, France (2008)
18. Forczmański, P., Seweryn, M.: Surveillance video stream analysis using adaptive background model and object recognition. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) *ICCVG 2010, Part I. LNCS*, vol. 6374, pp. 114–121. Springer, Heidelberg (2010)

19. Cucchiara, R., Grana, C., Piccardi, M., Prati, A., Sirotti, S.: Improving shadow suppression in moving object detection with HSV color information. In: *IEEE Intelligent Transportation Systems*, pp. 334–339 (2001)
20. Auvinet, E., Grossmann, E., Rougier, C., Dahmane, M., Meunier, J.: Left-luggage detection using homographies and simple heuristics. In: *Proceedings of 9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS 2006)*, pp. 51–58 (2006)
21. Lv, F., Song, X., Wu, B., Singh, V.K., Necatia, R.: Left-luggage detection using bayesian inference. In: *Ninth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS 2006)*, pp. 83–90 (2006)
22. Wang, X., Tieu, K., Grimson, W.E.L.: Learning semantic scene models by trajectory analysis. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) *ECCV 2006*. LNCS, vol. 3953, pp. 110–123. Springer, Heidelberg (2006)
23. Yang, Y., Cui, Z., Wu, J., Zhang, G., Xian, X.: Trajectory analysis using spectral clustering and sequence pattern mining. *J. Comput. Inf. Syst.* **8**(6), 2637–2645 (2012)
24. Dubuisson, M.P., Jain, A.K.: A Modified Hausdorff distance for object matching. In: *Proceedings of the 12th IAPR International Conference on Pattern Recognition ICPR94*, pp. 566–568 (1994)