Robotica: page 1 of 15. © Cambridge University Press 2009
doi:10.1017/S0263574709990464

Perception advances in outdoor vehicle detection for automatic cruise control

S. Álvarez, M. Á. Sotelo*, M. Ocaña, D. F. Llorca, I. Parra and L. M. Bergasa

Department of Electronics, University of Alcalá, Ctra. N-II Km. 33, Alcalá de Henares, Madrid, Spain

(Received in Final Form: July 29, 2009)

SUMMARY

This paper describes a vehicle detection system based on support vector machine (SVM) and monocular vision. The final goal is to provide vehicle-to-vehicle time gap for automatic cruise control (ACC) applications in the framework of intelligent transportation systems (ITS). The challenge is to use a single camera as input, in order to achieve a low cost final system that meets the requirements needed to undertake serial production in automotive industry. The basic feature of the detected objects are first located in the image using vision and then combined with a SVM-based classifier. An intelligent learning approach is proposed in order to better deal with objects variability, illumination conditions, partial occlusions and rotations. A large database containing thousands of object examples extracted from real road scenes has been created for learning purposes. The classifier is trained using SVM in order to be able to classify vehicles, including trucks. In addition, the vehicle detection system described in this paper provides early detection of passing cars and assigns lane to target vehicles. In the paper, we present and discuss the results achieved up to date in real traffic conditions.

KEYWORDS: Vision; Vehicle detection; Automatic cruise control; SVM (support vector machine); Intelligent transportation systems.

1. Introduction

Insufficient distance keeping is a major source of rear-end accidents as many drivers find it difficult to keep adequate headway distance because it requires taking into account both the distance to the vehicle ahead and the travelling speed of their vehicle. The importance of keeping sufficient headway for reduction of accidents is recognized by traffic authorities worldwide and is being enforced in an increasing number of countries. Headway is defined as the time it will take to reach the current position of a vehicle driving ahead, and is calculated by dividing the distance to the vehicle ahead with the travel speed of the host vehicle. Monocular vision can be used for vehicle detection and range measurement, and also to apply lane analysis in order to measure road geometry and curvature to determine the closest in-path vehicle. Headway measurement can be done by relying on the detection of the car rear in lit conditions, and on detection of taillights in dark night conditions. Situations where the vehicle would collide with another vehicle if no changes were made to the vehicle speed or direction can be detected. In this situation the vehicle can warn the driver by a warning sound and visual indication. If the time predicted to the collision is smaller than typical human reaction time emergency braking can be activated. The vision system can also detect and classify targets ahead of the host vehicle and send range and range rate (relative velocity) information to the automatic cruise control (ACC) controller to maintain a constant time gap between the host and followed vehicles. In such a case, the ACC controller automatically adjusts the speed of the host vehicle to maintain the desired headway by using throttle control and braking, and resumes to the set speed when the lane ahead is clear. In this paper, we propose a monocular vision system for vehicle detection mainly intended for ACC functionality, although the results of this system can also be used for Stop & Go and emergency braking applications.

In the context of active braking applications and ACC, long range radar information is used for very accurate range and relative velocity measurements. The vehicle path and target vehicle boundaries are used to predict the probability of an accident. Radar sensors have many advantages, such as accurate measurement capability and resistance to poor weather conditions, but this application cannot be implemented using long-range radar only. The radar ACC system uses a single sensing modality, the 76 GHz radar, to perceive the environment in which it operates. This single-sensor approach to perception problems, however, leads to single-mode failures. Although this radar is unaffected by weather and lighting conditions, sensor data from the radar is extremely limited in the context of trying to interpret an extremely complex and dynamic environment. In most cases, the combination of smart processing with radar data works well for the constrained application of ACC, but there are ACC situations where no matter how much processing is performed on the radar data, the data itself does not reflect the environment with a high enough fidelity to completely interpret the situation. Besides, spatial resolution is relatively coarse for the detected field of view, such that detections can be improperly localized in the scene and object size is impossible to determine.

* Corresponding author. E-mail: sotelo@depeca.uah.es

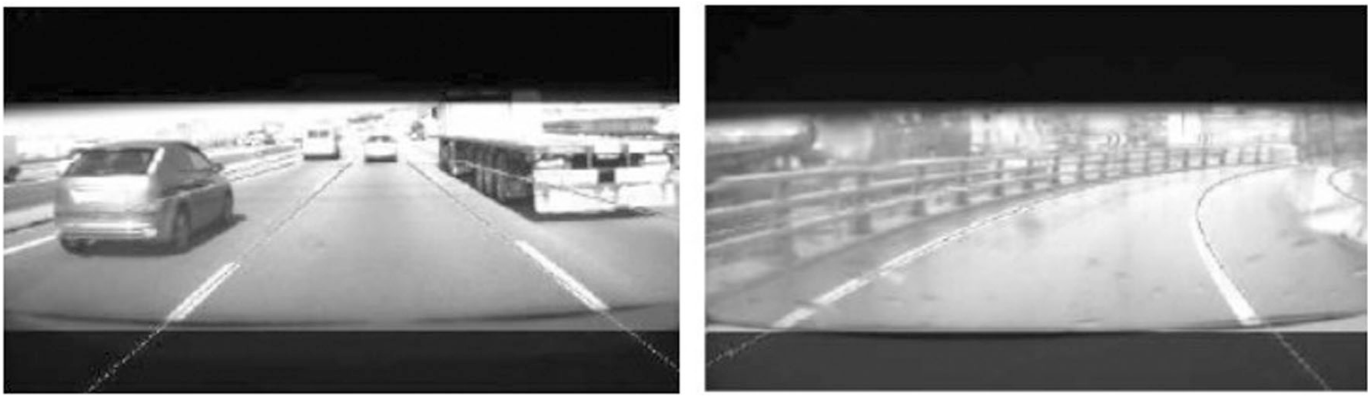

Fig. 1. Vehicle searching area as a result of lane markings analysis.

Some previous developments use available sensing methods such as radar,[21] stereo vision,[15,19,22] or a combination of stereo vision and laser[11] (Perrollaz *et al.*, 2006). In Hoffman *et al.*[10] and Bombini *et al.*[1] the authors propose the fusion of stereo vision and radar for creating a hybrid velocity adaptive control system called HACC. Only a few works deal with the problem of monocular vehicle detection using symmetry and colour features[2,3,12,20] or pattern recognition techniques[17] including support vector machines (SVM).[16] In Broggi *et al.*[2] the authors propose the use of horizontal edges and vertical symmetry together with a shape-dependent process for removing objects that are too small or too big in the image plane. Chateau and Lapreste 2004 deals with robust real-time vehicle tracking. Hilario *et al.*[9] propose the use of a geometrical model for vehicle characterization using evolutionary algorithms, assigning different geometrical models depending on the vehicle lane. Chem and Hou[5] provide night-time vehicle detection by combination with lane departure warning (LDW) in one-way roads for reducing false positive detections. Let us remark that the pattern recognition techniques used by all these systems for vehicle recognition can also be used for other ITS applications such as pedestrian detection[13] because of their generalization capability. A complete review of vehicle detection systems can be found in Sun *et al.*[18]

In the current work, the searching space in the image plane is reduced in an intelligent manner in order to increase the performance of the vehicle detection module. Accordingly, road lane markings are detected and used as the guidelines that drive the vehicle searching process. The area contained by the limits of the lanes is scanned in order to find vehicle candidates that are passed on to the vehicle recognition module. This helps reduce the rate of false positive detections. In case that no lane markings are detected, a basic *area of interest* is used instead covering the front part ahead of the ego-vehicle. The presence of collections of horizontal edges together with vertical symmetries triggers the attention mechanism.

The rest of the paper is organized as follows; Section 2 provides a description of the candidate selection method used as attention mechanism. Section 3 describes the SVM-based vehicle recognition and tracking system. Section 4 is dedicated to experimental results. In Section 5 a discussion about the system and its weak points is presented and, finally, Section 6 summarizes the conclusions and proposes future work.

2. Candidates Selection

An attention mechanism is necessary in order to filter out inappropriate candidate windows based on the lack of distinctive features, such as horizontal edges and vertical symmetrical structures, which are essential characteristics of road vehicles. This has the positive effect of decreasing both the total computation time and the rate of false positive detections. Each road lane is sequentially scanned, from the bottom to the horizon line of the image looking for collections of horizontal edges that might represent a potential vehicle. The scanned lines are associated in groups of three. For each group, a horizontality coefficient is computed as the ratio of connected horizontal edge points normalized by the size of the area being analysed. The resulting coefficient is used together with a symmetry analysis in order to trigger the attention mechanism. Apart from the detected road lanes provided by a lane departure warning system (LDWS) developed by the authors in previous works,[16] additional virtual lanes have been considered so as to cope with situations in which a vehicle is located between two lanes (e.g., if it is performing a change lane manoeuvre). Virtual lanes provide the necessary overlap between lanes, avoiding both misdetections and double detections caused by the two halves of a vehicle being separately detected as two potential vehicles. A virtual lane is located to provide overlap between two adjoining lanes. Figure 1 provides some examples of lane markings detection in real outdoor scenarios. Detected lanes determine the vehicle searching area and help reduce false positive detections. In case no lane markings are detected by the system, fixed lanes are assumed instead. The system proposed in this article can also detect vehicles in the hard shoulder although it was not originally designed to do so. The fact that the vehicle detection process is driven by lane markings detected on the road makes it viable to detect vehicles on the right-hand side hard shoulder, just next to the outer lane marking.

The first step of the process for detecting collections of horizontal edges is carried out by performing an adaptive thresholding. This process permits to obtain robust image edges, as depicted in the examples provided in Fig. 2.

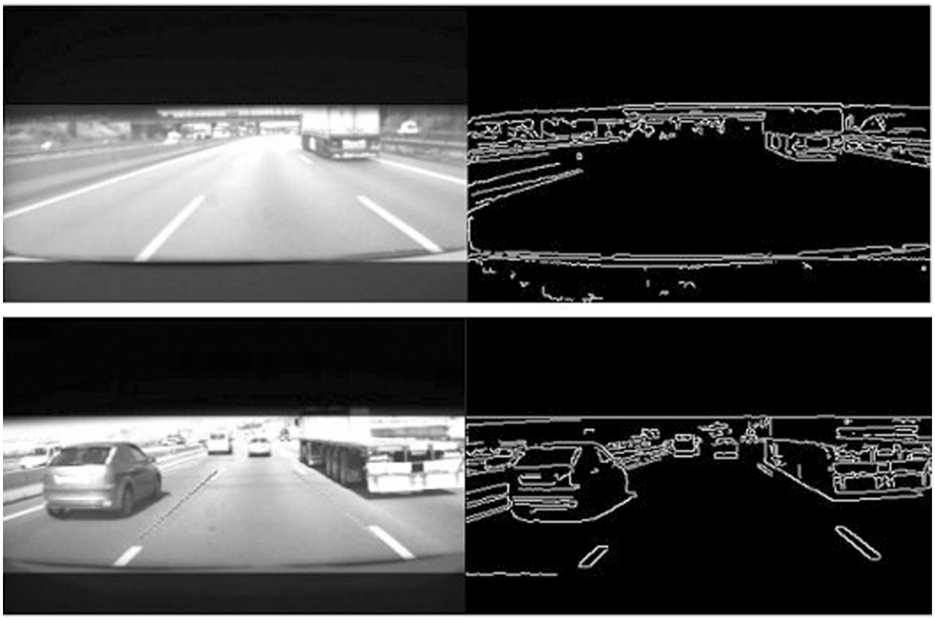

Fig. 2. Edge images after adaptive thresholding.

This adaptive process is based on an iterative algorithm that gradually increases the contrast of the image, and compares the number of edges obtained in the contrast increased image with the number of edges obtained in the actual image. If the number of edges in the actual image is higher than in the contrast increased image the algorithm stops. Otherwise, the contrast is gradually increased and the process resumed. Figure 3 summarizes the algorithm flow diagram.

After thresholding, horizontal edges in the scan regions given by the LDW system (extended to three lanes) are examined to detect the rear part of potential vehicles. In order to decide if the collection of horizontal lines represents a possible vehicle candidate, its width is compared to that of an ideal car. The ideal car width is obtained for each vertical coordinate using the camera pinhole model. Resolving the pinhole classical equation of reference changes (Eq. (1)), the searched relation is obtained (Eq. (2)):

$$u_i = f_u \frac{X_i}{Z} + u_0, \tag{1}$$

$$u_2 - u_1 = \frac{f_u}{Z}(X_2 - X_1), \tag{2}$$

where u_0 andf_u are intrinsic camera parameters, *Z* is the host-to-vehicle distance and X_i and u_i are the points to evaluate as represented in Fig. 4.

To calculate the *Z* distance, the pinhole model is used again. The origin of the vehicle coordinate system is located at the central point of the camera lens. The X_V and Y_V coordinates of the vehicle coordinate system are parallel to the image plane and the Z_V-axis is perpendicular to the plane formed by the X_V and Y_V axes. A vehicle at a look-ahead distance *Z* from the camera will be projected into the image plane at a vertical and horizontal coordinates (*u*, *v*) respectively. The vertical road mapping geometry following this nomenclature is depicted in Fig. 5. The vertical model considers the flat terrain assumption. The flat terrain assumption can be considered approximately valid during the detection stage, since first time detection takes place in short distances. During the tracking stage the distance measurement error remains bellow 5% in average. These small errors in distance measurement have little influence on the ACC system since it is based on fuzzy logic rules and variables. This is specially truth if we consider that the shorter the distance measured (and thus, the more dangerous the situation is) the smaller the distance error committed, and subsequently, the smaller the influence on the vehicle control. The camera parameters can then be calibrated using standard calibration methods. The proposed mapping uses the following parameters:

I:	image plane
Z:	look-ahead distance for planar ground (m)
$\mathbf{h}_{CAM}$:	elevation of the camera above the ground (m)
Pitch:	camera pitch angle relative to vehicle pitch axis (rad)
θ_Z:	incident angle of the preceding vehicle's contact-to-asphalt point relative to vehicle pitch axis (rad)
v:	vertical image coordinate (pixels)
HEIGHT:	vertical size of the CCD (pixels)
F_v:	vertical focal length (pixels)
F_u:	horizontal focal length (pixels)
k_v:	vertical scaling factor for the camera (pixels/mm).

According to Fig. 5, the vertical mapping geometry is mainly determined by the camera elevation $\mathrm{h_{CAM}}$ above the local ground plane as well as the pitch angle. The longitudinal axis of the vehicle is assumed to be always tangential to the road at the vehicle centre of gravity (cg). For each image scan line at *v*, there corresponds a pitch angle relative to the local tangential plane given by Eq. (3):

$$\theta_Z = \mathrm{Pitch} + a\tan\left(\frac{v}{fK_v}\right). \tag{3}$$

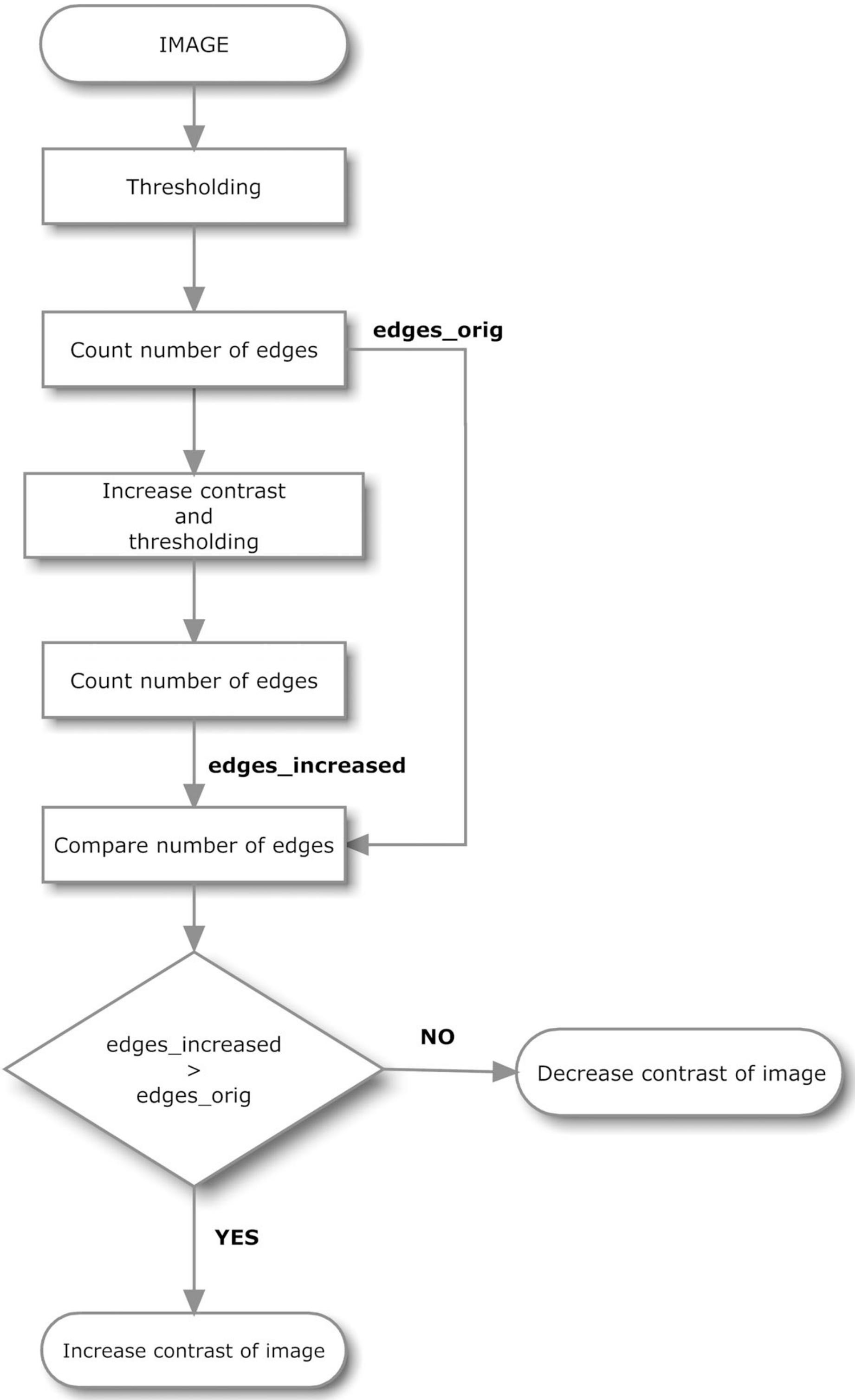
IMAGE
Thresholding
Count number of edges
edges_orig
Increase contrast and thresholding
Count number of edges
edges_increased
Compare number of edges
edges_increased > edges_orig
NO
Decrease contrast of image
YES
Increase contrast of image

Fig. 3. Block diagram of the adaptive thresholding algorithm.

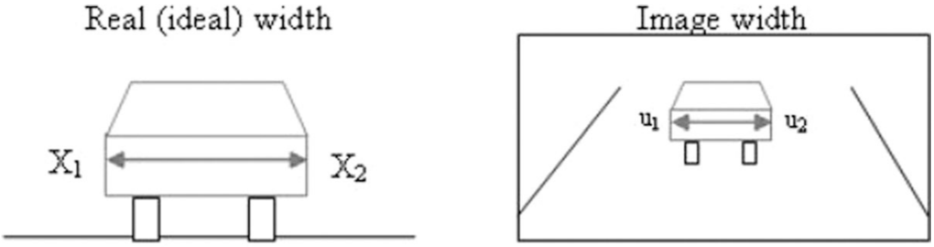

Fig. 4. Car width variables.

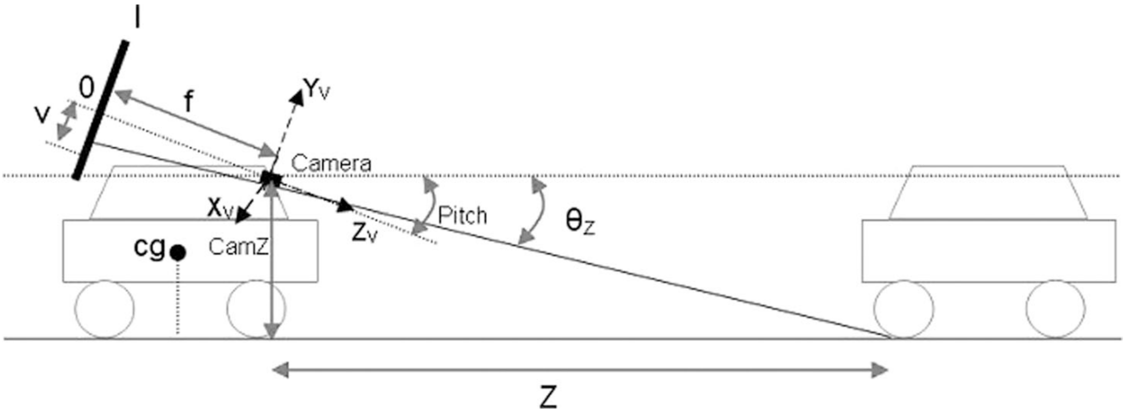

Fig. 5. Vertical road mapping geometry.

Based on this, the planar look-ahead distance corresponding to v, is obtained as

$$Z = \frac{h_{\mathrm{CAM}}}{\tan(\theta_Z)}. \tag{4}$$

Applying a coordinate change due to the fact that the image origin in our case is on the top of the image instead of in the centre, the new vertical coordinate v(top) is given by

$$v(\mathrm{top}) = 2v(\mathrm{centre}) - \mathrm{HEIGHT}. \tag{5}$$

In Eq. (6), the vertical scaling factor of the camera is introduced in the distance length parameter

$$F_v = f\, K_v. \tag{6}$$

The equation for computing the look-ahead distance Z becomes

$$Z = \frac{h_{\mathrm{CAM}}}{\tan\left(\mathrm{Pitch} + a\tan\left(\frac{2\cdot v - \mathrm{HEIGHT}}{F_v}\right)\right)}. \tag{7}$$

Once the car width is computed at the current frame it is compared to the collection of horizontal lines found after the thresholding analysis. If they are similar to some extent defined by an empirical value, a square area above the collection of horizontal lines, denoted as candidate Region Of Interest (ROI), is considered for further analysis. The aim is to compute the entropy of the candidate ROI and its vertical symmetry. Only those regions containing enough entropy and symmetry are identified as potential vehicles. Figure 6 shows a detailed block diagram of the detection procedure, and Fig. 7 depicts two examples of the detection step.

Detection distance is typically in the range 5–40 m for the first detection stage. Vehicles can then be tracked up to 60–70 m once detected. Compared to laser and radar systems, detection distance is lower since radar and laser can provide detection above 100 m. Nonetheless, information provided by these types of sensors is much less reliable than that provided by vision sensors in terms of interpretability.

Figure 7 shows an example of candidate analysis and validation. As can be observed, the detected candidate has enough horizontal edges so as to be considered a potential vehicle. In addition, symmetry analysis, conducted around the vertical central axis of the candidate bounding box, provides additional clues about the presence of a potential vehicle in the scene. A detailed zoomed image of the validated candidate is also depicted in Fig. 7.

The potential vehicle candidate might not be detected in the optimal position in the image plane, due to displacements caused by inaccurate lateral location. In order to improve the location accuracy, a further symmetry analysis is carried out. For this purpose, a region of analysis is shifted from left to right around the detected candidate. Symmetry is computed at each position of the region of analysis. The position yielding the maximum value of vertical symmetry is chosen to correct the original vehicle position in the image plane. This symmetry-based vehicle correction mechanism contributes to improve the accuracy of candidates' location. Figure 8 depicts a couple of examples in which vehicle position is corrected using this method. Blue squares represent the original position of the candidate, whereas red squares show the final, corrected position after symmetry analysis. Improving the accuracy of vehicle detection in the image plane has a significant positive impact on the classification stage.

The attention mechanism can provide lots of overlapping candidates due to the fact that potential vehicles are detected based on edge and symmetry features. Many areas of the image can exhibit vehicle-like features and attract the focus of the attention mechanism. In particular, a single vehicle in the image can yield several candidates. The next step is then to avoid candidates overlapping. For this purpose, overlapping candidates are grouped together and analysed. A coefficient is computed for each overlapping candidate, defined as the ration between Entropy+Symmetry and candidate area. Coefficients obtained for each group of overlapping candidates are compared. The candidate yielding the highest coefficient is chosen and the rest of candidates overlapping with him are discarded. Figure 9 depicts an example of overlapping candidate removal following this approach. For each group of overlapping candidates only one candidate remains. Figure 10 shows some examples of candidates selected by the attention mechanism, including vehicles and non-vehicles.

Accurate detection of the wheel-to-road contact point of the preceding vehicle is essential for assuring maximum

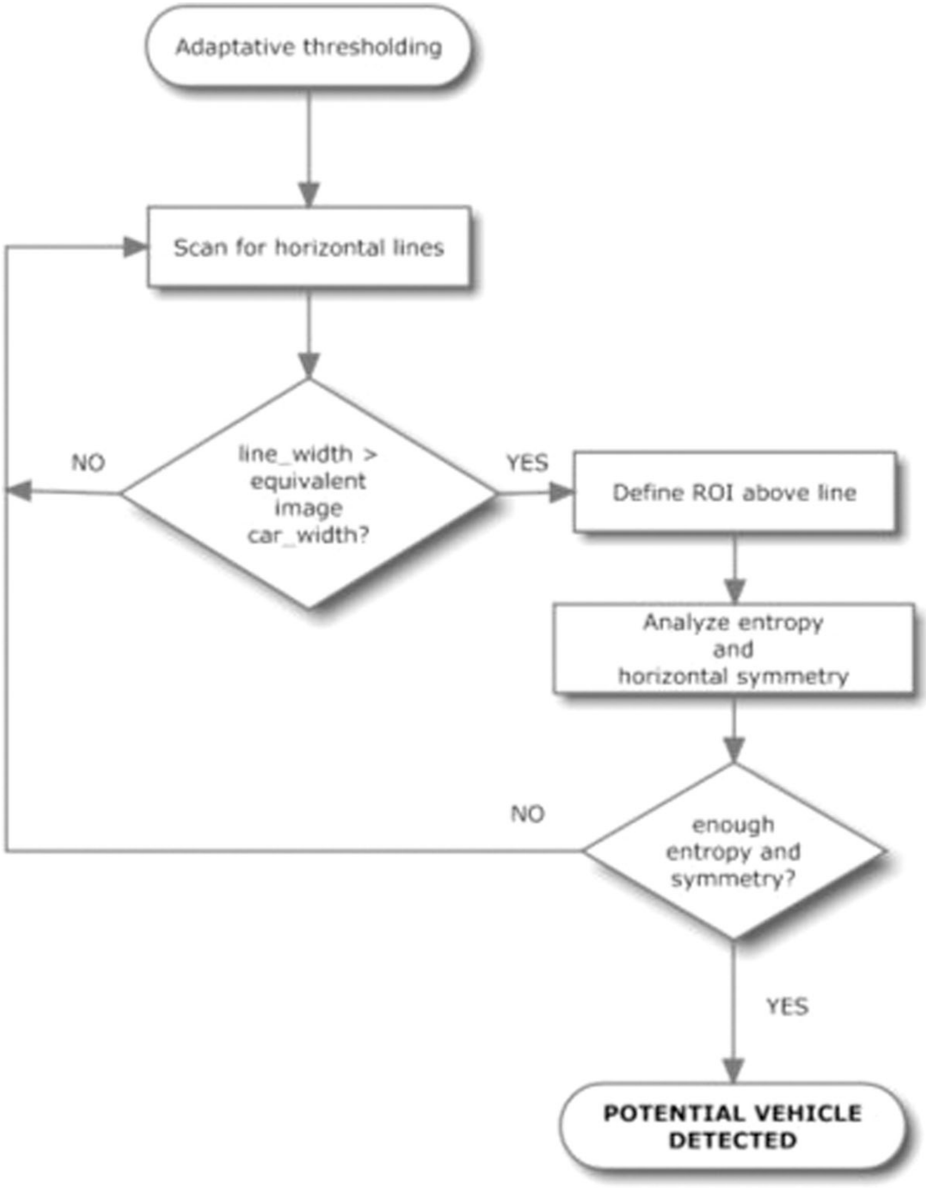
Adaptative thresholding
Scan for horizontal lines
line_width > equivalent image car_width?
NO
YES
Define ROI above line
Analyze entropy and horizontal symmetry
enough entropy and symmetry?
NO
YES
POTENTIAL VEHICLE DETECTED

Fig. 6. Block diagram of the vehicle detection mechanism.

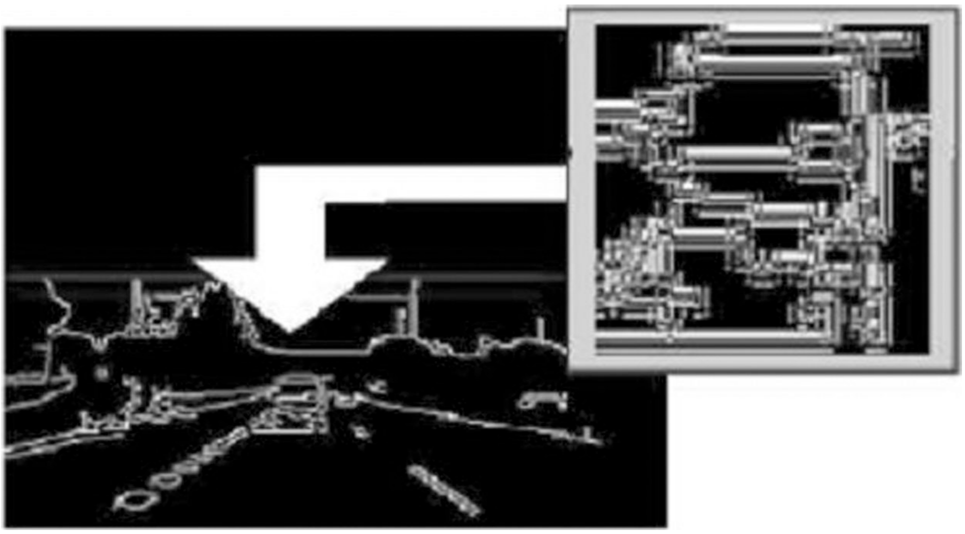

Fig. 7. Example of validated vehicle candidate, road scene and zoomed edge image.

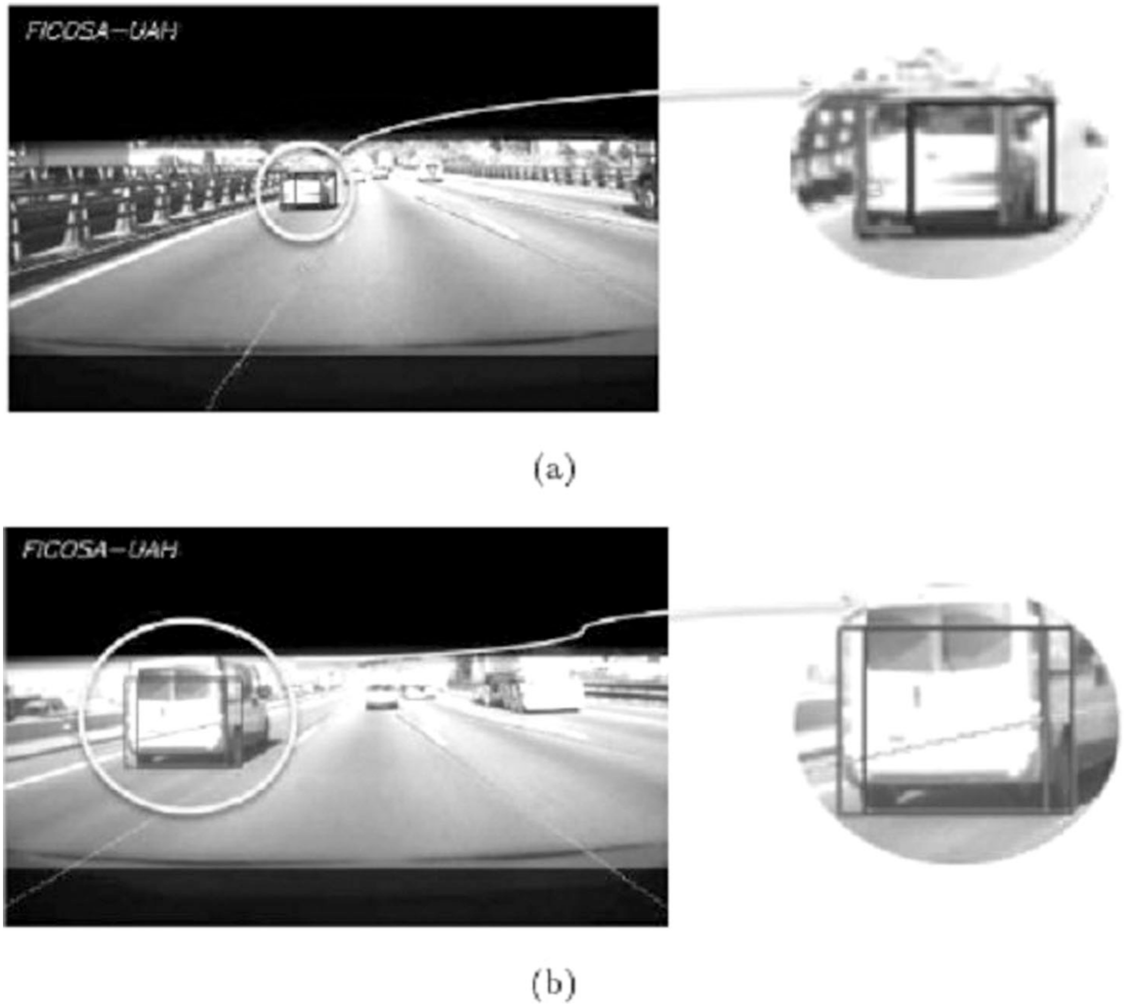

Fig. 8. Examples of symmetry-based vehicle position correction mechanism. Blue squares represent the original position of detected candidates. Red squares show the corrected position after symmetry analysis.

precision of the host-to-vehicle estimated distance. Thus, the error committed in estimating the host-to-vehicle distance Z_{err} due to a vehicle detection error of n pixels in the image plane is given by

$$Z_{\text{err}} = Z_n - Z = \frac{F_v h_{\text{CAM}}}{v+n} - Z = \frac{-nZ^2}{F_v h_{\text{CAM}} + nZ}, \quad (8)$$

where v is the vertical coordinate of the wheel-to-road contact point in the image plane, Z is the estimated host-to-vehicle distance and h_{CAM} represents the camera height (as previously defined). Considering an error of one pixel $n = 1$ and $F_v h_{\text{CAM}} \gg nZ$, Z_{err} becomes

$$Z_{\text{err}} \approx \frac{nZ^2}{F_v h_{\text{CAM}}}. \quad (9)$$

For example, for a 640 × 480 image, a focal length of 740 pixels, and a camera height $h_{\text{CAM}} = 1.2$ m, an error of 1 pixel

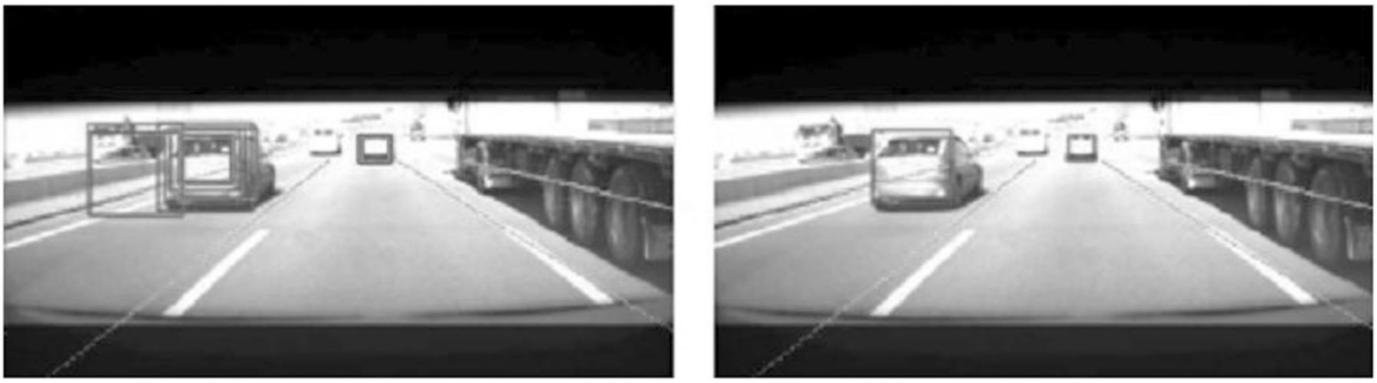

Fig. 9. Candidate overlapping removal in the attention mechanism: original candidates (left); result after overlapping removal (right).

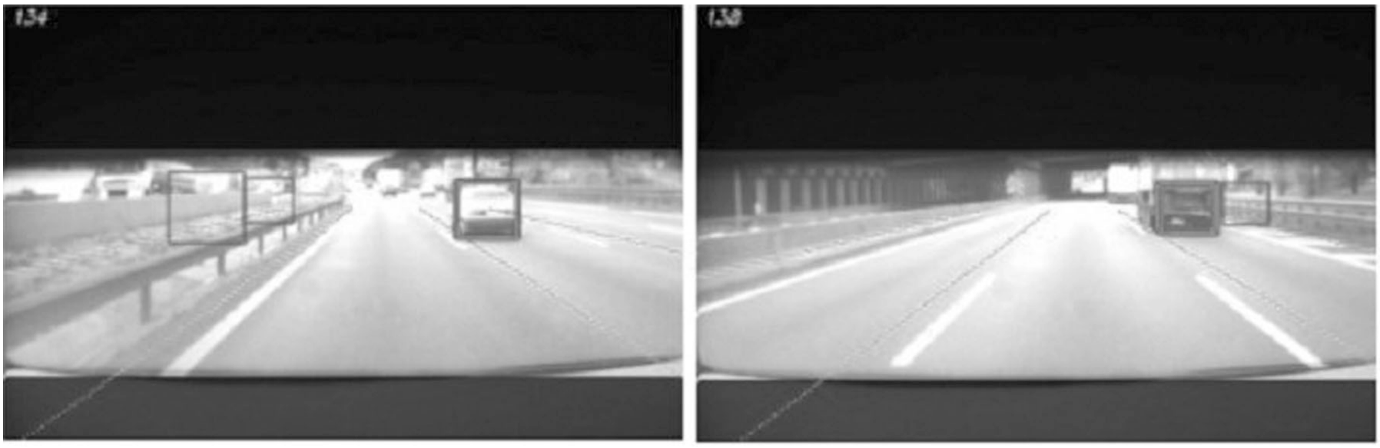

Fig. 10. Examples of initial candidates detected by the attention mechanism (vehicles and non-vehicles).

($n = 1$) becomes a relative 5% error at a distance

$$Z = \frac{Z_{\text{err}}}{Z} F_v h_{\text{CAM}} = 0.05 * 740 * 1.2 = 44\,\text{m}. \tag{10}$$

On the other hand, the error at 90 m is 10%. These values are more than enough for the ACC function. What is really important is the measurement of relative host-to-vehicle velocity. Relative velocity Rv is computed using the following equation:

$$Rv = \frac{\Delta Z}{\Delta t}. \tag{11}$$

Based on the scale change s of detected objects in the image plane, the optimal value of t that minimizes the estimation noise can be calculated. Let W denote the width (in meters) of the preceding vehicle, w and w' the width of the preceding vehicle in the image plane when it is located at distances Z and Z', respectively, with regard to the host vehicle. The scale change s can be defined as

$$s = \frac{\omega - \omega'}{\omega'}. \tag{12}$$

The estimated relative velocity can be computed as follows:

$$Rv = \frac{\Delta Z}{\Delta t} = \frac{Z\frac{\omega - \omega'}{\omega'}}{\Delta t} = \frac{Zs}{\Delta t}. \tag{13}$$

As demonstrated in Stein *et al.*,[17] the value of t that minimizes the error in the estimated relative velocity is given by

$$\Delta t = \sqrt{\frac{2Z^2 s_{\text{err}}}{F_v W a}}, \tag{14}$$

where a represents the acceleration of the host vehicle, and s_{err} is the error committed in the estimation of scale change. Building on this result, the optimal value of t for zero acceleration is infinite. In practice, it has been limited to $t = 1.8$ s.

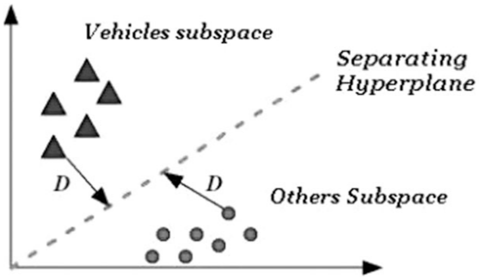

Fig. 11. Vehicle class separation using SVMs.

3. Vehicle Recognition and Tracking

Detected candidates are classified as vehicles or non-vehicles depending on features obtained from the vehicle ROI using SVM.[6] SVMs are a set of related supervised learning methods used for classification and pattern recognition. One special property of SVMs is that they simultaneously minimize the empirical classification error and maximize the geometric margin. Hence they are also known as maximum margin classifiers. SVMs map input vectors to a higher dimensional space where a maximal separating hyperplane is constructed, by maximizing the distance between both classes. An assumption is made that the larger the margin or distance between these parallel hyperplanes the better the generalization error of the classifier will be. An example of this can be seen in Fig. 11.

The output of the SVM, D, is simply the signed distance of the test instance from the separating hyperplane. This output indicates whether the analyzed object corresponds to a vehicle (+1, in theory) or not (−1, in theory) and can be used as a threshold for separating them. Two aspects are essential in the deployment of SVM classifiers: the training strategy and the classifier structure.

3.1. Training strategy

The first step in the design of the training strategy is to create representative databases for learning and testing. The following considerations must be taken into account when creating the training and test sets.

- The ratio between positive (vehicles) and negative (others) samples has to be set to an appropriate value. A very large

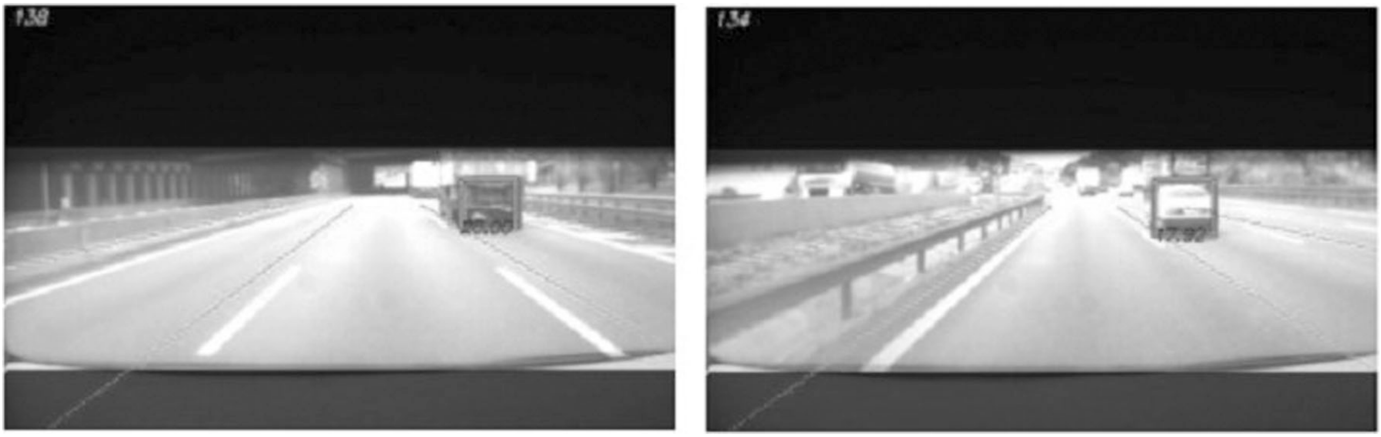

Fig. 12. Examples of final candidates classified as vehicles.

number of positive samples in the training set may lead to a high percentage of false positive detections during online classification. On the contrary, a very large number of negative samples produce misslearning.

- The size of the database is a crucial factor to take care of. As long as the training data represent the problem well, the larger the size of the training set the better for generalization purposes.
- A sufficiently representative test set must be created for verification. The content of the test set has similar characteristics to those of the training sets in terms of variability and ratio of positive/negative samples.

3.2. Classifier structure

An input vector for the classifier was defined. This vector is composed of different parameters which are computed for all candidates and define the state vector for the SVM. Those parameters are local histograms of oriented gradients (HOG).[7] The aim of this method is to describe an image by a set of local histograms which count occurrences of gradient orientation in a local part of the image (the selected candidate ROI). As a general overview, the algorithm is composed of the following steps:

- Parameters of the detected objects (HOG) are computed and used as inputs (SVM feature vector) to the SVM classifier.
- Once the parameters vector is computed, the SVM process analyzes this vector and returns a value which is simply the signed distance of the test instance from the separating hyperplane.

Figure 12 shows a couple of examples of vehicle detection after SVM classification and re-calculation of the host-to-car distance.

3.3. Vehicle tracking

After detecting consecutively an object a predefined number of times (empirically set to 3 in this work), tracking is implemented using Kalman filtering techniques. For this purpose, a dynamic state model and a measurement model must be defined. The proposed dynamic state model is simple. Let us considered the state vector $x\,[n]$, defined as follows:

$$x\,[n] = \begin{bmatrix} x \\ y \\ w \\ h \end{bmatrix}. \tag{15}$$

In the state vector x and y are the respective horizontal and vertical image coordinates for the top left corner of every object, and w and h are the respective width and height in the image plane. A dynamical model equation can be written like this:

$$x[n+1] = F \cdot x[n] + \omega_n. \tag{16}$$

In the model, F represents the system dynamics matrix and ω_n is the noise associated to the model. In this case, F has been defined as an Identity matrix, i.e. a diagonal matrix with all diagonal elements set to 1. Although simple, it proves to be highly effective in practice. The system dynamics matrix has been approximated by an identity matrix since the real time operation of the system permits to assure that there will not be great differences in distance for the same vehicle between consecutive frames. The model noise has been modelled as a function of distance and camera resolution. The state model equation is used for prediction in the first step of the Kalman filter. Physically, it means that the prediction for the next frame is that the state vector will remain basically equal to the estimation at the current frame. The next step is to define the measurement model. A measurement model equation can be established as follows:

$$z[n+1] = H \cdot x[n+1] + \varpi_n. \tag{17}$$

In Eq. (17), H represents the measurement matrix and ϖ_n is the noise associated to the measurement process. H has been defined as an Identity matrix. The purpose of the Kalman filtering is to obtain a more stable position of the detected vehicles. Besides, oscillations in vehicles position due to the unevenness of the road makes y coordinate of the detected vehicles change several pixels up or down.

This effect makes the distance detection unstable, so a Kalman filter is necessary for minimizing these kinds of oscillations. As a future idea, even though an image correction and filtering can be done, it would be much

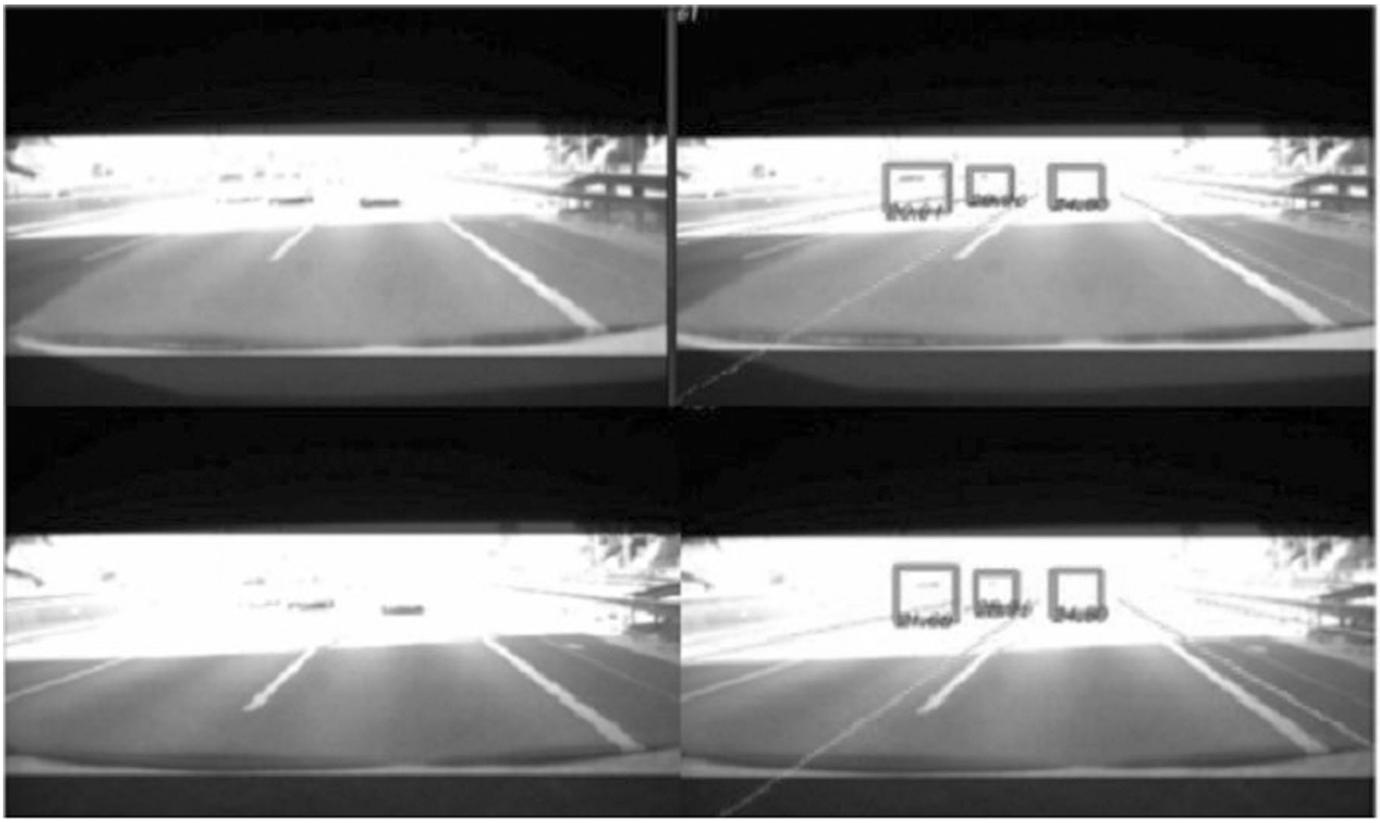

Fig. 13. Cars remain under tracking in spite of bad image conditions.

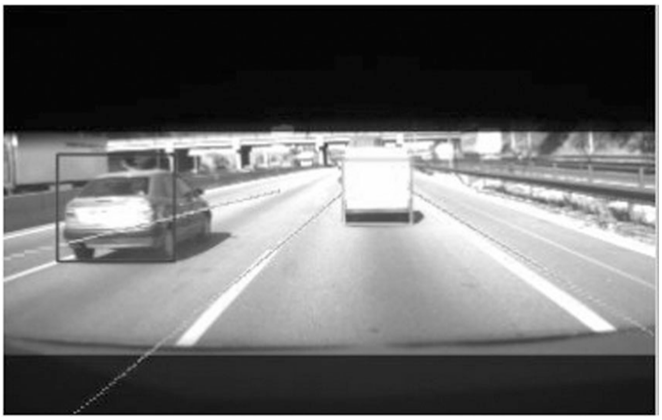

Fig. 14. Introducing a positive candidate in the database.

more efficient to go through this problem by introducing an oscillation sensor in the car. As an example of this, Figure 13 shows a case in which three cars passing beneath a bridge are not detected as potential vehicles in the original images (left images) due to bad image conditions. Nonetheless, the three cars are kept under tracking by the system using the previously described Kalman filter (right images).

4. Implementation and Results

The system was implemented on a PC Pentium IV at 2.4 GHz onboard a car-like robot (modified Citroën C4) and tested in real traffic conditions using a 640 × 320 CMOS camera. For this purpose, a training strategy based on a large and representative database for learning and testing has been devised. The training database contains 10,000 representative samples while the test set has 3,000 samples. In both cases, a positive/negative ratio of 1:2 has been observed. We have tested that this ratio ensures a low percentage of false positive detections during on-line classification. The size of the database (10,000 samples) represents a crucial factor to take care of. The large size of this training set represents the problem well, and it permits to achieve a reasonable solution for generalization purposes. To obtain a sufficiently representative set we have taken cars and trucks as positive samples, and crash barriers, median strip, pieces of road, etc, like negative samples. The samples have been taken in different weather conditions (with and without rain, shadows, etc). The content of the test set has similar characteristics to those of the training set in terms of variability and ratio of positive/negative samples. The size of the test set (3,000 samples) is appropriate for verification of the overall system. To create the samples sets, we have developed a tool called 'ACC Database'. This tool represents an extended option of the main software used for vehicle detection. The tool allows entering the candidates extracted by the car detection system as positive or negative samples in the database. To illustrate the process, the next figures show how candidates detected by the attention mechanism can then be manually introduced in the database as positive, negative or not valid candidate (when the user is not sure about the type of candidate). Figure 14 shows the introduction of two positive candidates in the database. The first candidate was extracted in the central lane of the road and upon validation the tool labels the candidate in blue colour. The next extracted candidate is assigned to the adjoining left lane and labelled in green colour.

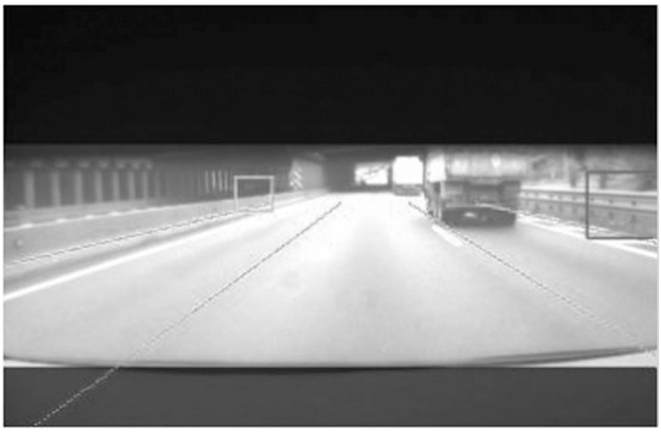

Fig. 15. Introducing a negative candidate in the database.

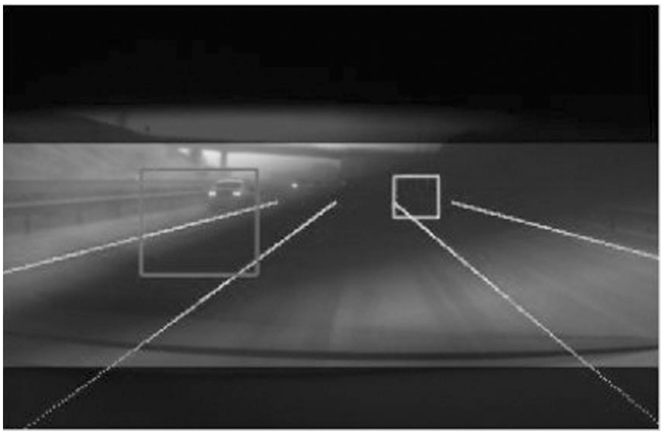

Fig. 16. Introducing a negative candidate in the database (red) and cancellation of a non-sure candidate (green).

Figure 15 shows the introduction of a negative candidate in the database. The first detected candidate is a crash barrier extracted on the right-hand side of the image. After manual validation as negative sample, the tool labels the candidate in red. The next extracted candidate is on the left-hand side of the image. It is also introduced as a negative sample and labelled in green.

Figure 16 shows the introduction of a negative candidate in the database, on the left part of the image, and the cancellation of a non-sure candidate. The first candidate was extracted on the left adjoining lane of the road while the second was assigned to the adjoining right lane. The first candidate is validated as a negative sample (red), while the second one was not included in the database because the user was not sure about the type of candidate due to bad image conditions.

Using ACC database tool an intensive training stage was accomplished. The next table shows the number of samples obtained for training and testing from several video sequences recorded in real traffic conditions.

After training, a SVM model using 2281 support vectors is obtained. This means that the classification process does not

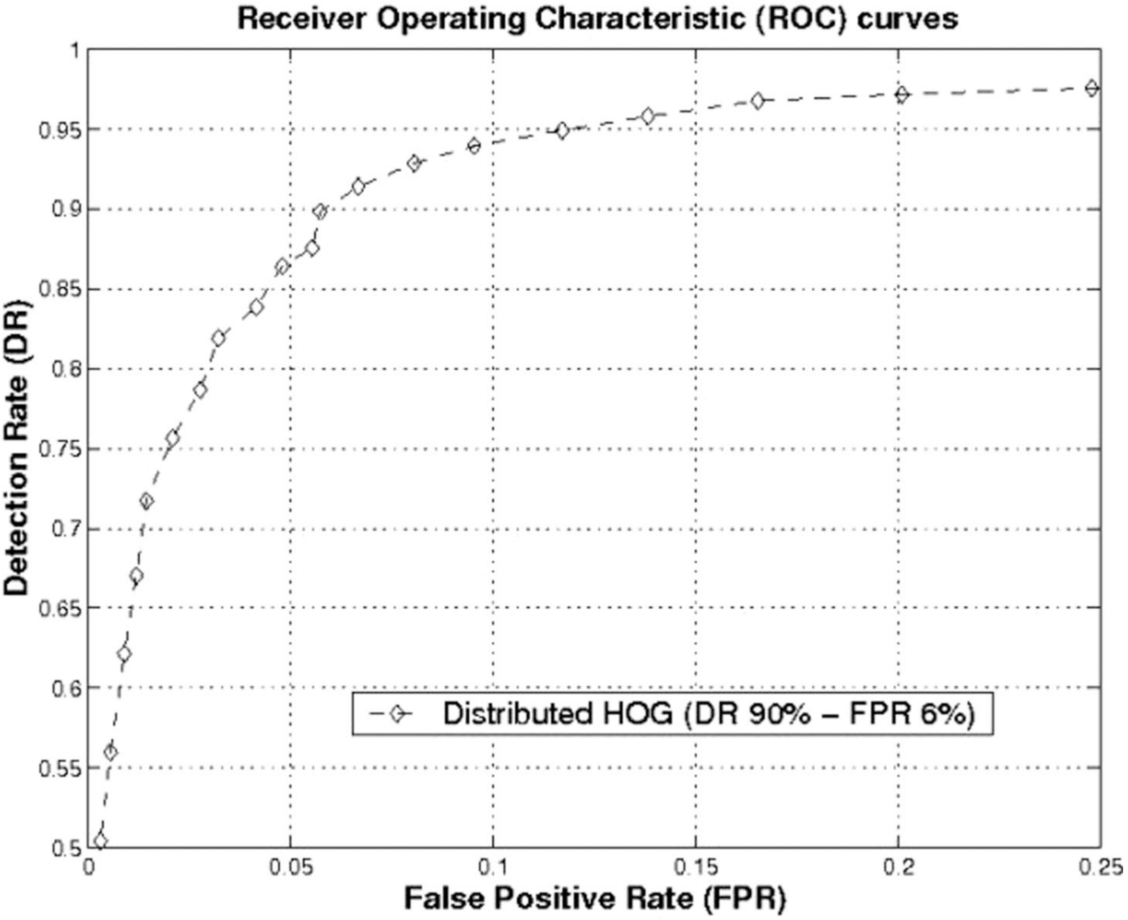

Fig. 17. Receiver operating characteristic curve (ROC curve).

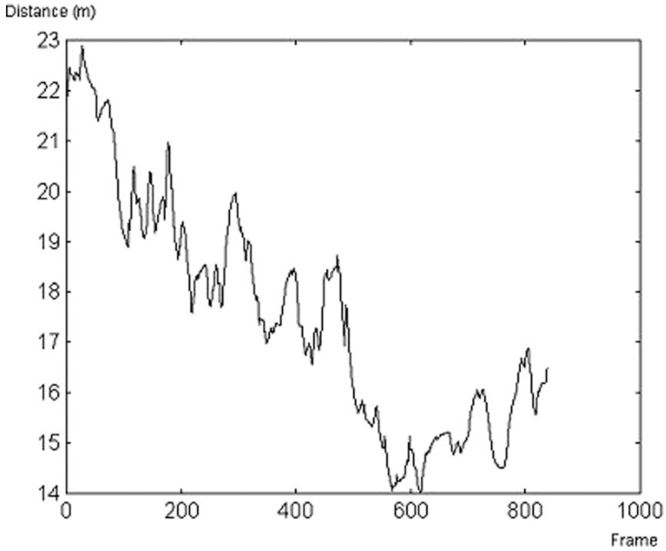

Fig. 18. Host-to-vehicle distance in a real experiment.

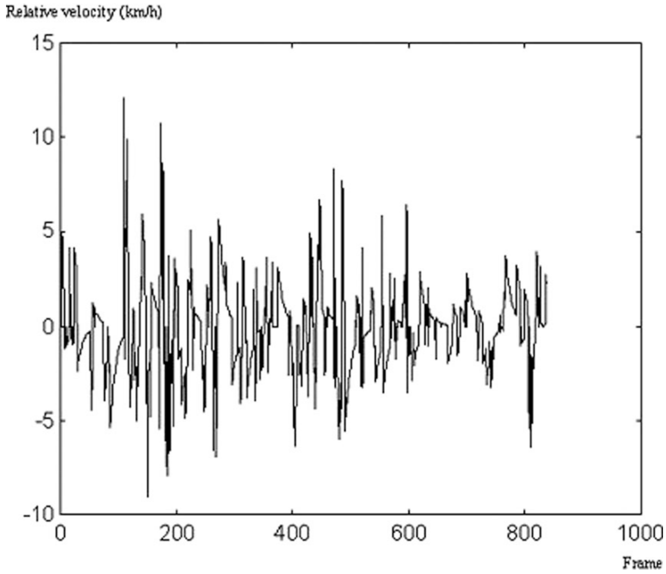

Fig. 19. Host-to-vehicle relative velocity in a real experiment.

need to use all the database vectors to obtain the hyper plane. Figure 17 shows the receiver operating characteristic curve (ROC curve) using the training and test sets illustrated in Table I. The ROC curve depicts the detection rate (DR) versus false positive rate (FPR) of the final single-frame classifier. A trade-off point has been chosen at (DR = 90%, FPR = 6%) for achieving robust vehicle detection at an acceptable level of false positive detections.

Figure 18 shows a plot of the host-to-vehicle distance measured in a sequence of 830 frames using the vision-based vehicle detection system proposed in this paper. Likewise, Figure 19 depicts the relative velocity between the host vehicle and the preceding one. As can be observed, most of the time the relative host-to-vehicle velocity is zero during the selected sequence. Nonetheless, the difference of speed goes up to 10 km/h in some moments.

Figure 20 depicts a sequence example in which a vehicle is detected and tracked by the system. In the sequence, the detected preceding vehicle is highlighted using a blue square. Other vehicles circulating along adjoining lanes are marked by a red horizontal line. Lane markings detected by the system are depicted in green.

Table I. Number of samples obtained from video sequences for training and testing.

Video	Frames	Positive	Negative	Training/test
Sequence_1 (rain)	4962	721	2183	Training
Sequence_2 (dry, bridges)	2214	954	3071	Training
Sequence_3 (dry, glare)	2215	502	765	Training
Sequence_4 (rain)	2213	297	1218	Test
Sequence_5 (cloudy)	2664	1327	423	Training
Sequence_6 (cloudy)	1338	725	808	Test

Table II. Global performance of the ACC system.

Video	# Frames	Tpf (ms)	Detected	Missed	FA	Cause
Sequence_7 (cloudy)	2.456	62	1	0	0	—
Sequence_8 (sunny)	3.765	33	9	1	0	Motorcycle
Sequence_9 (rain)	1.987	30	1	1	0	Truck (rain)
Sequence_10 (cloudy)	2.367	31	8	0	1	Road fence
Sequence_11 (rain)	2.678	53	9	1	0	Motorcycle

Table II provides a summary of statistics concerning global system performance. The table shows the results achieved using the previously described database containing 13,000 samples. The average processing time per frame (tpf) is given in ms, as well as the number of detected vehicles, missing vehicles and number of false alarms. As can be observed from Table II, not only the detection rate and false alarm rate are provided, but also the reasons that cause it.

The system operates in real time at 30 frames per second in average, as observed in Table II (tpf). All images are then processed in real time conditions. The system yields a global detection rate of 90.32% with 1 false alarm. Miss detections mainly occur with motorcycles and trucks under heavy rain. Given the fact the SVM system has been trained using only cars and trucks it can be considered that motorcycles miss detections can be solved by incorporating a sufficient number of motorcycle images in the database. For this purpose, the candidate selection mechanism should be modified in order to raise candidates with the shape and aspect of motorcycles, not only cars, trucks and buses. Indeed, if the effect of miss-detected motorcycles is neglected, a detection rate of 96.77% is achieved for cars, trucks and buses. In order to diminish the number of false alarms due to road artifacts, such as road fences, these types of elements should be included in the database as negative samples. Although these elements are already included in the current database, it should be further enlarged and enriched until proper generalization will be achieved.

5. Discussion

In this section some key problems are identified and described, together with the proposed solutions that would

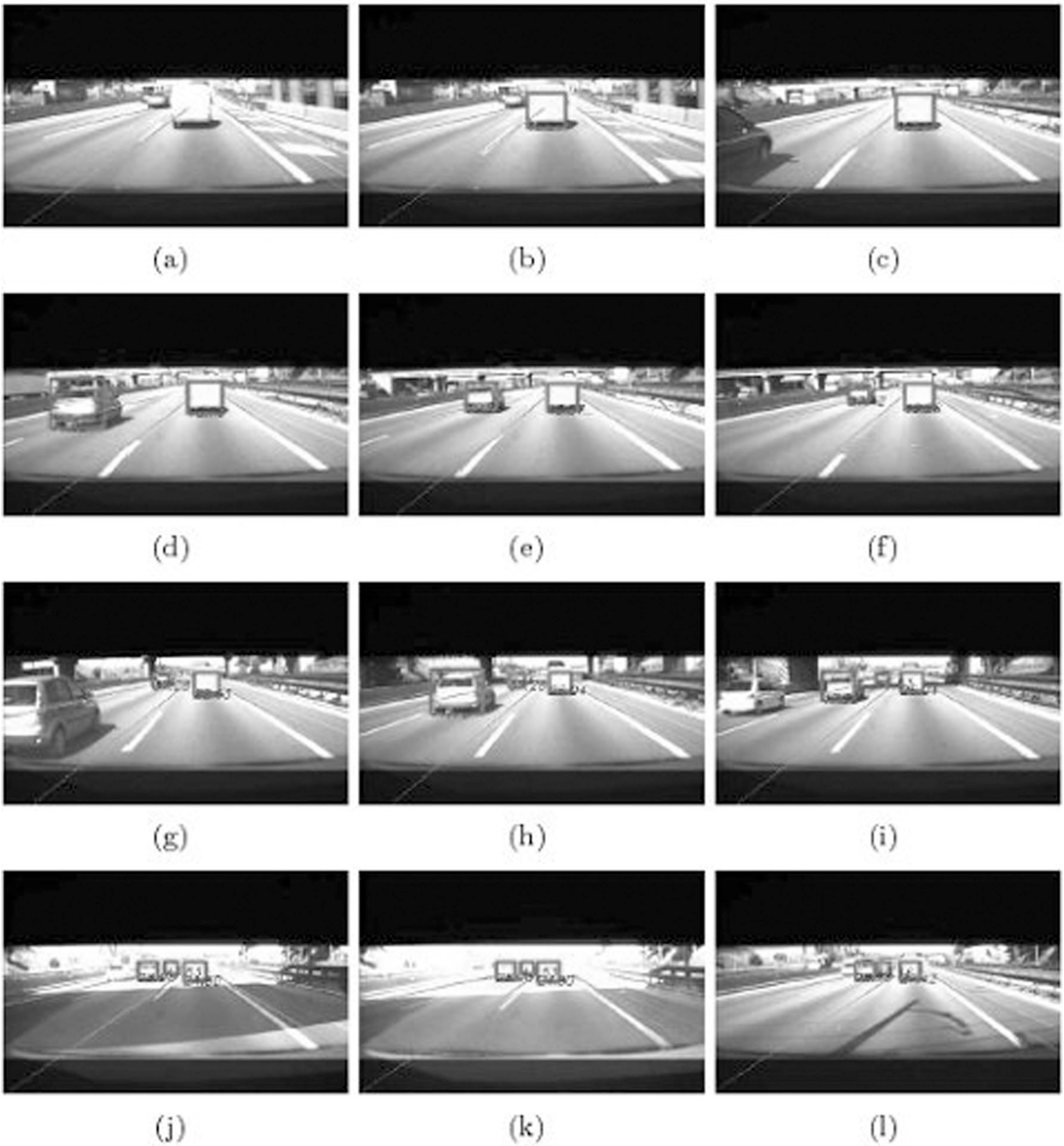

Fig. 20. Vehicle tracking example in a real sequence.

lead to sorting out the fore mentioned problems or, at least, to mitigating their negative effects in order to increase the system performance.

5.1. Single-frame classification performance

Although the attained single-frame classification results are high (DR above 90% for a FPR of 6%), the classifier performance has still to be improved in order to achieve an acceptable level of global performance for a real e-Safety application. This can be carried out by increasing the size and variability of the samples in the database, including samples of vehicles in different conditions for training and testing the SVM. It is important to notice that the training of the classifier was done with a limited amount of data. In addition, the system can be divided in three modulus, one for each road lane as depicted in Fig. 21, improving the system

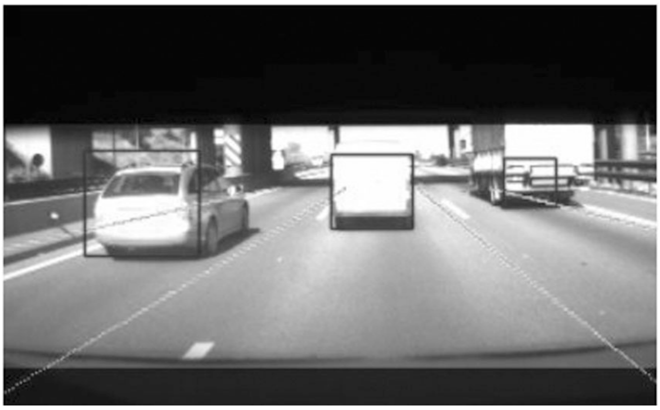

Fig. 21. SVM-based classification stage can be divided in three modulus, one for each lane.

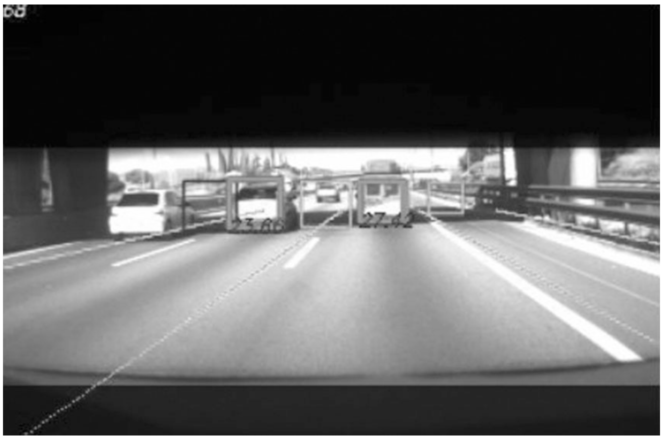

Fig. 22. Example of false vehicle detection.

performance by means of specialization and reducing the global execution time of the ACC function.

5.2. *False vehicle candidates*

There are some cases in which the proposed features for vehicle detection, entropy and symmetry, produce values that turn out to be enough to raise a non-vehicle candidate as a potential vehicle candidate. This problem is normally fixed in the classification stage, but it implies an additional and unnecessary computational cost that can be avoided by developing a more structured detection mechanism. Figure 22 shows an example where a shadow of a bridge is incorrectly considered as a vehicle candidate before classification in three consecutive iterations.

5.3. *Shadows*

Although shadows are very useful to detect vehicles (as the horizontal edge detected at the lower part of cars and trucks), in some cases they can cause severe problems in the detection stage, especially if shadows are large (normally at sunset and sunrise). Changing illumination conditions are considered in the edge extraction stage. For this purpose, the variable Canny thresholds are adapted at all iterations as a function of illumination conditions given by the histogram properties. This allows obtaining edges even in shaded regions. Nonetheless, Figure 23 depicts an example in which a large shadow produced by a car can lead to raise a false vehicle candidate in the adjoining right lane.

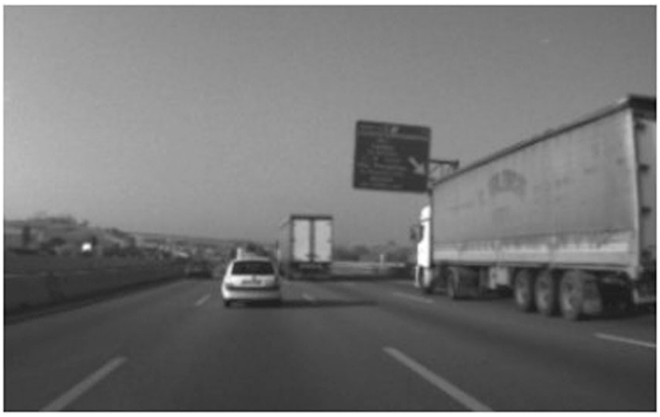

Fig. 23. Example of large shadow caused by a car.

To solve this problem, we propose to use the 'hat transform'.[23] The hat transformation is a powerful operator which permits the detection of contrasted objects on non-uniform background. There are two different types of top-hat transformations: white hat and black hat. The white hat transformation is defined as the residue between the original image and its opening. The black hat transformation is defined as the residue between the closing and the original image. The white and black hat transformations are analytically defined as follows:

$$\mathrm{WH}_T(x, y) = (f - f \circ b)(x, y) \quad \text{white hat,} \qquad (18)$$

$$\mathrm{BH}_T(x, y) = (f \bullet b - f)(x, y) \quad \text{black hat.} \qquad (19)$$

Both operators, white and black hat transform, can be used in order to modify the contrast of the image or enhancing contrast in some regions of the image. Normally, in grey scale images, the local contrast is ruled by two kinds of features: bright and dark features. The white hat image contains local peaks of intensity and the black hat image contains local valleys of intensity. The effect of shadows can be mitigated by using a combination of the white and black transformations as explained in Fernández *et al.*[8]

6. Conclusions and Future Work

We have developed and implemented a vehicle detection system based on SVM and monocular vision with the objective of providing vehicle-to-vehicle time gap measurement for ACC applications in the framework of ITS. Vehicle candidates are raised using an attention mechanism based on horizontal edges, vertical symmetry and entropy. The detected objects are passed on to a SVM-based classifier. After classification, detected vehicles are tracked using Kalman filtering. A large database containing thousands of vehicle examples extracted from real road images has been created for learning purposes. The classifier is trained using SVM in order to be able to classify cars and trucks. In addition, the vehicle detection system described in this paper provides early detection of passing cars and assigns lane to target vehicles based on the use of a LDWS. After assessment of the practical results achieved in our experiments, the following general conclusions can be summarized:

- The global performance of the monocular daytime ACC developed and described in this paper yields a detection rate above 90% for a false alarm rate around 1%.
- The combination of ACC and LDWS is possible using the same single camera as input. Indeed, the performance of ACC is significantly increased by building on the output provided by the LDWS function.
- The presence of large shadows on the asphalt due to vehicles circulating along the road produces negative effects on the candidate selection mechanism, yielding to inaccuracy in measuring the distance to the vehicles. This effect is more clearly observable when the sun is very low, at sunset or at sunrise.

Based on these conclusions, the following actions are proposed as future work:

- Fusion of vision data with data provided by a laser or radar system is strongly recommended.
- Three specialized SVM classifiers will be developed for each individual road lane in order to provide more accurate vehicle classification. The reason for that relies on the fact that vehicles circulating on adjoining lanes are not perceived in the image plane with the same perspective as vehicles circulating along the same lane (where the lateral sides of the vehicles are not perceived in the image).
- The combination of the White and Black Hat transforms will be applied to minimize the effect of shadows on the road due to other vehicles.
- Extensive experiments at nighttime conditions will be carried out in the next stage of development of the ACC application. For this purpose, detection and tracking of vehicles at night time will be performed by relying on vehicles tail lights.

Acknowledgements

This work has been supported by Ficosa International S. A. and the Spanish Ministry of Education and Science by means of Research Grant DPI2005-07980-C03-02.

References

1. L. Bombini, P. Cerri, P. Medici and G. Alessandretti, "Radar-Vision Fusion for Vehicle Detection," *Proceedings of International Workshop on Intelligent Transportation*, Hamburg, Germany (2006) pp. 65–70.
2. A. Broggi, P. Cerri and P. C. Antonello, "Multi-resolution vehicle detection using artificial vision," *IEEE Intelligent Vehicles Symposium*, Parma, Italy (2004) pp. 310–314.
3. A. Broggi, M. Bertozzi, A. Fascioli, C. Guarino Lo Bianco and A. Piazzi, "The Argo autonomous vehicle's vision and control systems," *Int. J. Intell. Control Syst.* **3**(4), 409–441 (2000).
4. T. Chateau andJ. Lapreste, "Robust Real Time Tracking of a Vehicle by Image Processing," *IEEE Intelligent Vehicles Symposium* (2004) pp. 315–318.
5. M. Y. Chem and P. C. Hou, "The Lane Recognition and Vehicle Detection at Night for a Camera-Assisted Car on Highway," *IEEE International Conference on Robotics & Automation*, Taipei, Taiwan (2003), Vol. 2, pp. 2110–2115.
6. J. C. Christopher Burges, "A tutorial on support vector machines for pattern recognition," **In** *Data Mining and Knowledge Discovery*, 2,121–167 (Kluwer Academic Publishers. 1. 1998).
7. N. Dalai and B. Triggs, "Histogram of Oriented Gradients for Human Detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego CA, USA (2005) pp. 886–893.
8. P. Fernández, M. A. Sotelo and L. M. Bergasa, "Automatic Daytime Road Traffic Control and Monitoring System," *Proceedings of the 2008 IEEE Intelligent Vehicles Symposium*, Eindhoven, The Netherlands (2008) pp. 944–949.
9. C. Hilario, J. M. Collado, J. M. Armingol and A. de la Escalera, "Visual Perception and Tracking of Vehicles for Driver Assistance Systems," *IEEE Intelligent Vehicles Symposium*, Tokyo, Japan (2006) pp. 94–99.
10. U. Hofmann, A. Rieder and E. D. Dickmanns, "Application to Hybrid Adaptive Cruise Control," *IEEE Intelligent Vehicles Symposium*, Dearborn, MI, USA (2000) pp. 468–473.
11. R. Labayrade, C. Royere, D. Gruyer and D. Aubert, "Cooperative Fusion for Multi-Obstacles Detection with Use of Stereovision and Laser Scanner," *Proceedings of International Conference on Advanced Robotics*, Coimbra, Portugal (2003) pp. 1538–1543.
12. T. Liu, N. Zheng, L. Zhao and H. Cheng, "Learning Based Symmetric Features Selection for Vehicle Detection," *IEEE Intelligent Vehicles Symposium*, Las Vegas, Nevada, USA (2005), pp. 124–129.
13. I. Parra, D. Fernández-Llorca, M. A. Sotelo, L. M. Bergasa, P. Revenga, J. Nuevo, M. Ocaña and M. A. Garrido, "Combination of feature extraction methods for SVM pedestrian detection," *IEEE Trans. Intell. Transp. Syst.* **8**(2), 292–307 (2007).
14. M. Perrollaz, R. Labayrade, C. Royere, N. Hautiere and D. Aubert, "Long Range Obstacle Detection Using Laser Scanner and Stereovision," *IEEE Intelligent Vehicles Symposium*, Tokyo, Japan (2006) pp. 182–187.
15. D. Ponsa, A. Lopez, J. Serrat, F. Lumbreras and T. Graf, "Multiple Vehicle 3d Tracking Using an Unscented Kalman," *IEEE Intelligent Transportation Systems Conference*, Vienna, Austria (2005) pp. 1108–1113.
16. M. A. Sotelo, J. Nuevo, L. M. Bergasa and M. Ocaña, "A monocular solution to vision-based ACC in road vehicles," *Lecture Notes Compute. Sci.* **3643**, 507–512 (2005).
17. G. P. Stein, O. Mano and A. Shashua, "Vision-Based ACC with a Single Camera: Bounds on Range and Range Rate Accuracy," *Proceedings of International Conference on Intelligent Vehicles*, Versailles, France (2002) pp. 120–125.
18. Z. Sun, G. Bebis and R. Miller, "On-road vehicle detection: A review," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(5), 694–711 (2006).
19. G. Toulminet, M. Bertozzi, S. Mousset, A. Bensrhair and A. Broggi, "Vehicle detection by means of stereo vision-based obstacles features extraction and monocular pattern analysis," *IEEE Trans. Image Process.* **15**(8), 2364–2375 (2006).
20. M. B. Van Leeuwen and F. C. A. Groen, "Vehicle detection with a mobile camera: Spotting midrange, distant and passing cars," *IEEE Robot. Autom. Mag.* **12**(1), 37–43 (2005).
21. G. R. Widman, W. A. Bauson and S. W. Alland, "Development of Collision Avoidance Systems at Delphi Automotive Systems," *Proceedings of International Conference on Intelligent Vehicles*, Stuttgart, Germany (1998) pp. 353–358.
22. T. Williamson and C. Thorpe, "Detection of Small Obstacles at Long Range Using Multibaseline Stereo," *Proceedings of International Conference on Intelligent Vehicles*, Stuttgart, Germany (1998) pp.311–316.
23. L. D. Ye Derong and Z. Yuanyuan, "Fast Computation of Multi-Scale Morphological Operations for Local Contrast Enhancement," *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th annual conference*, Shanghai, China (2005) pp. 3090–3092.