

A Collision Warning Oriented Brake Lights Detection and Classification Algorithm Based on a Mono Camera Sensor

Dario Nava, Giulio Panzani and Sergio M. Savaresi

Abstract— In this paper, a brake lights detection and classification algorithm based on a monocular camera and oriented to collision warning is described. First, the current driving lane is identified through a lane detection algorithm. Shrinking the search area to the actual driving lane, the preceding vehicle is detected exploiting YOLO object detector. Then, preceding vehicle lateral and third brake lights are identified in the L^*a^*b colorspace by harnessing brightness and color information, along with geometrical considerations. Finally, brake lights status is determined by means of SVMs classifiers, based on features computed both on light regions and the overall vehicle image, and the braking status of the preceding vehicle is determined. The algorithm is designed to work in different illumination conditions during day time: experimental results prove the robustness of the proposed strategy with respect to different illumination conditions and brake lights shape, with an overall algorithm precision of 96.3%.

I. INTRODUCTION

The acronym "ADAS" (Advanced Driver Assistance Systems) refers to a wide variety of devices with the main goal to help the driver in many challenging driving situations. Due to the undeniable benefits in terms of safety and comfort introduced by these systems, in the late years the diffusion and importance of ADAS has increased even in low-end vehicles. Moreover, thanks to the increasing in-vehicle computational power, a growing number of ADAS relies on information provided by machine vision sensors.

Among all the possible exploitations of camera based strategies in the automotive field, one of the most relevant application is related to obstacle detection and collision warning/avoidance systems. In this setup, the visual detection of a preceding vehicle braking manoeuvre by looking at its brake lights status is a challenging problem; however such information can be beneficially harnessed by different ADAS. For example, considering ACC [1] or collision warning/avoidance strategies (e.g. [2], [3]) based a single camera sensor, where an obstacle distance estimation is possible but yet inaccurate, the information on the braking status of the preceding vehicle could be used to enhance robustness in braking manoeuvre detection. Also when a distance sensor is available (e.g. lidar, radar, etc.) such information could be integrated with the distance measure to make the braking detection more reactive (visual detection is affected only by camera framerate related delay) in warning the driver about the potential dangers and/or directly acting on the brakes to avoid possible crashes.

The problem of detecting vehicle tail lights and determine their status has already been addressed in literature, both in night time and day time conditions. In the former case, the greater majority

of the works is focused on rear lights detection, that is often a mandatory step in vehicle detection strategies when the scenario lightness is very low (see the work in [4]). In the latter case instead, tail lights detection along with their status classification are two strictly related topics, frequently addressed as a single problem.

For instance, in [5] a two step brake lights detection method is proposed. Firstly, lateral brake light regions are identified in the YC_bC_r colorspace through Otsu thresholding and spatial filtering. Then, brake lights status is classified (on/off) using a Support Vector Machine trained on features computed over the brake light regions. Another SVM classifier based approach is proposed in [6]. For every frame collected by the camera, vehicles are identified using Deformable Part Model approach and their lateral tail lights detected in the HSV colorspace. Lights are classified (off/braking/turning) based on features extracted according to the Hierarchical Matching Pursuit approach. In [7] and [8], after a vehicle detection stage, lateral tail lights are extracted in the L^*a^*b colorspace looking at the a^* (red) and L^* (luminance) channels. In [7], also brake light status is determined (on/off) exploiting Fast Radial Symmetry Transform, which has the main drawback of operate effectively on radially symmetric blobs (while the shape of tail lights may not satisfy this requirement). In [9] a deep learning approach is exploited. After road area segmentation and vehicle detection based on HOG features, the extracted vehicle images are given as input to a convolutional neural network classifier, able to determine whether the vehicle is braking or not. Finally, in [10] and [11] two strategies for brake lights detection and status determination are presented, working both in day time and night time conditions. In [10], at first vehicles are detected exploiting symmetrical SURFs descriptors. Then, lateral tail lights are identified using the so called "Lamp Response" of each pixel and their status is determined using an high-pass pixel mask. In [11] candidate tail lights are extracted in YUV colorspace and selected using color and brightness thresholds. Lights activation is then detected through area and intensity tracking of such lights.

In this work, a novel approach to brake lights detection and status classification is proposed, specifically developed for collision warning systems in day time conditions. The first stage of the algorithm is focused on detection the preceding vehicle, which braking status must be recognised. In this phase, a lane detection algorithm and YOLO [12], [13] object classifier, as well as object 3D projection through camera calibration, are exploited so as to achieve the goal. Notice that this first step is not implemented in the great majority of literature approaches, which indeed exploit vehicle detection in order to simplify brake lights extraction procedure but take into consideration all the found vehicles and not extract the preceding one. Once the preceding vehicle is found, its brake lights are

All the authors are with Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Italy.

Corresponding author: giulio.panzani@polimi.it

extracted. Specifically, in addition to the two lateral brake lights of the vehicle, also the third brake light (*i.e.* the one usually positioned at the top or bottom side of the rear window, centred *w.r.t.* the vehicle chassis) is detected in this phase. Such third light, being turned on only in the occurrence of a braking manoeuvre, carries precious information about the braking status of a vehicle and can be used to enhance robustness of the overall strategy. Despite this fact, as far as the authors know, such information it has not been harnessed in the previous works (except indirectly in [9] where, as explained above, the input to the classifier is the entire vehicle area). Finally, the identified lights status is determined using SVMs classifiers.

The remainder of the paper is organized as follows. In Section II and Section III the lane detection strategy and vehicle detection strategy respectively are explained, aimed to the preceding vehicle identification. In Section IV the rear brake lights detection strategy is described. Section V is devoted to brake lights status classification. Finally, in Section VI experimental results are presented, along with final comments on the algorithm effectiveness.

II. LANE DETECTION

Lane detection phase is a crucial step for the brake light detection algorithm, since lane boundaries information can be exploited by the vehicle detection module so as to restrict the search area for the possible preceding vehicles (see Section III). Since the topic has been widely discussed in literature (see the survey [14]), in the following we will briefly describe the main idea behind the algorithm.

The system is based on Hough Transform, that is widely exploited in lane detection problems. First of all, for every frame collected by the RGB camera, a ROI (Region Of Interest, see Figure 2a) is chosen on the image, which contains the road plane and thus all the useful information to perform lanes identification. The ROI is then transformed from RGB colorspace to grayscale and Canny edge detector is applied, in order to obtain a binary image containing the edges corresponding to lane markings. Finally, lines corresponding to lane markings are extracted exploiting probabilistic Hough Transform (results in Figure 2b).

Once identified the lane marking candidates, a further step is necessary to distinguish the ones truly defining lane boundaries from the false positives (*i.e.* caused by road irregularities, shadows, etc.). First of all lines are projected from the 2D pixel coordinates to the 3D camera frame, so as to get rid of perspective effects (see Figure 1). Given a point $P_i^C = (X_i, Y_i, Z_i)$ in the 3D camera frame, its

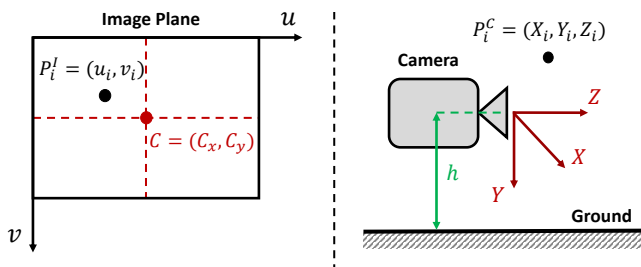


Fig. 1: Reference frames scheme: pixel coordinates (left) and 3D camera frame (right).

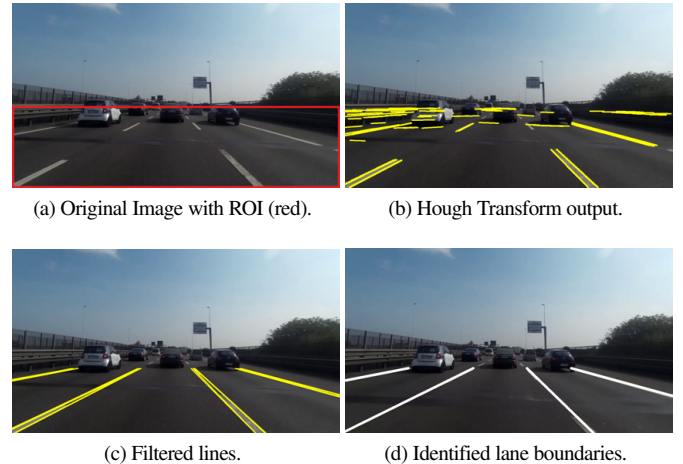


Fig. 2: Lane boundaries identification with Hough Transform.

projection $P_i^l = (u_i, v_i)$ in pixel coordinates can be obtained as [15]:

$$\begin{cases} u_i &= C_x + f_x \cdot \frac{X_i}{Z_i} \\ v_i &= C_y + f_y \cdot \frac{Y_i}{Z_i} \end{cases} \quad (1)$$

where f_x and f_y represent the camera focal length and (C_x, C_y) are the coordinates of the camera principal point in pixels, which can be obtained through calibration. Under the assumption of flat road and small roll and pitch angles, we can know precisely its height h *w.r.t.* the ground plane. Now, since the road markings lie on the road plane, we know the $Y_i = h$ coordinates of their endpoints in the 3D camera frame and hence we can project them in the (X, Y, Z) coordinate system exploiting the equations in 1 in the two unknowns X_i and Z_i . The projected lines are then filtered, to discard the ones not corresponding to lane markings, based on their absolute and relative position and orientation. Indeed filtering the lines in the 3D frame is much simpler and effective than doing the same in the image plane, where parallel lines have not zero relative angles and the same lengths in pixels could correspond to different lengths in the real world.

At this point, the length of the filtered lines is normalized (see Figure 2c) and the ones that share similar position and orientation are grouped to obtain the identified lane boundaries. The algorithm, based on the lateral position of the markings identifies the current driving lane (always) and the left and right lateral lanes (if any, see Figure 2d).

The steps described so far are applied to every single frame captured by the camera. In order to correlate the information between frames, every identified marking is tracked using a Kalman Filter [14], to obtain noise reduction and robustness with respect to ambient occlusion.

III. PRECEDING VEHICLE DETECTION

Once identified the driving lane boundaries, possible obstacles (*e.g.* cars, trucks, etc.) preceding the ego vehicle in the driving lane shall be identified. To do so, we rely on an object detector based on the recently introduced YOLO (You Only Look Once) framework [12], [13]. The detector has been trained using the COCO dataset [16], and it is able to detect vehicles in square images of dimensions 416×416 pixels. Hence, the preliminary step of

vehicle detection consists in decide which are the image portions (ROIs) to be given as input to the vehicle detector. To do so, we can rely on the information derived from lane detection, since we expect the vehicle to be contained in the current driving lane. Call P_v the vanishing point defined in the image plane by the two driving lane boundaries, and P_b the middle point of the segment of length delimited by the two intersections of the lane boundaries with the bottom line of the image, whose length w_ℓ represents the width of the lane at the bottom of the frame: the segment $\overline{P_v P_b}$ identifies the driving lane center in the image plane. We define two square-shaped ROIs - a smaller one, with size 416×416 to detect distant vehicles, and a bigger one, with size 832×832 to detect near vehicles - both centred in $C_{ROI} = (u_{ROI}, v_{ROI})$ which lies on $\overline{P_v P_b}$ (see Figure 3a). This way, in every frame the two ROIs are centred with respect to the identified driving lane, and the only free parameter is the value of v_{ROI} , experimentally tuned.

Once extracted the two ROIs, both images are given as input to the YOLO detector (indeed the image from the biggest ROI is resized to fit the 416×416 input size of the CNN), obtaining the candidate vehicles. Since, doing so, the vehicles contained in the smallest ROI are identified twice, a merge strategy is necessary. Calling A_{V_i} and A_{V_j} the bounding box areas of two vehicles V_i and V_j , the bounding boxes are merged in a single detection if the index:

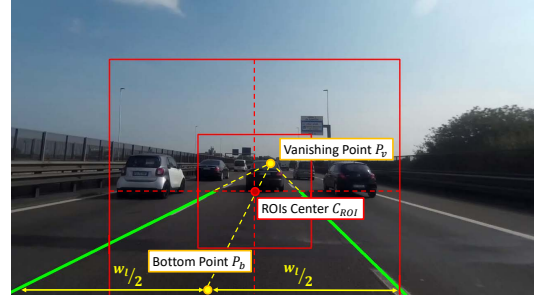
$$I_A = \frac{A_{V_i} \cap A_{V_j}}{\min\{A_{V_i}, A_{V_j}\}} \geq \tau_A \quad (2)$$

where the threshold τ_A is a tuning parameter. The results of vehicle detection and merging operation are shown in Figure 3b. As for lane detection, in order to reduce noise and increase the detection robustness, every vehicle is tracked using Kalman Filters [17].

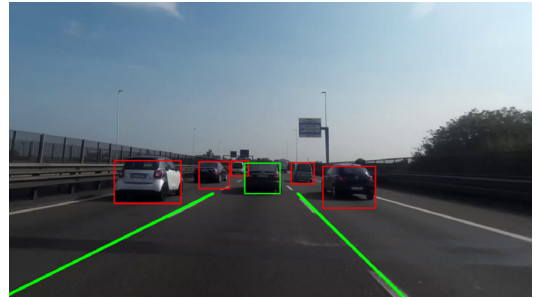
Now, the algorithm must select among all the detections the one preceding the ego vehicle an driving in the same lane. Here, we once again exploit Equation (1). Considering the bottom edge of the bounding boxes, we assume it to represent the contact line between the corresponding vehicle and the road plane and thus, to know its $Y=h$ coordinate in the 3D camera frame. Hence, projecting in 3D frame the left and right endpoints of the bottom edge associated to the i -th detected vehicle v_i - namely $P_{B_i}^l = (X_{B_i}^l, h, Z_{B_i}^l)$ and $P_{B_i}^r = (X_{B_i}^r, h, Z_{B_i}^r)$ - we can obtain the distance $d_i = Z_{B_i}^l = Z_{B_i}^r$ between the ego vehicle and v_i . Furthermore, $X_{B_i}^l$ and $X_{B_i}^r$, that give us information about the width and X direction displacement (*w.r.t.* the ego vehicle) of v_i , can be merged with the estimate of the driving lane position and width extracted in Section II. The preceding vehicle is identified as the closest to the ego vehicle and completely contained in the current driving lane (Figure 3b).

IV. BRAKE LIGHTS DETECTION

Following the preceding vehicle identification, tail lights are detected and their status classified. To perform the detection, we draw inspiration from the works [8] and [7], adding however significant features. We start from the portion of image containing the preceding vehicle, as depicted in Figure 4a, that is rescaled to a dimension of 416×416 pixels for normalization purposes. The image is then converted into $L^*a^*b^*$ colorspace and the a^* channel - containing red and green color components - is extracted (Figure 4b). The portions of the a^* channel characterized by an high red



(a) Scheme of ROIs center computation.



(b) Overall detections (in red) and preceding vehicle (in green).

Fig. 3: Vehicle detection procedure using YOLO detector and distance estimation with contact point detection.

chromaticity are identified by image binarization using Otsu thresholding (Figure 4c). Finally, tail light candidates are extracted applying connected component analysis to the binary image. Such candidates are represented on the original segmented image in Figure 4d. It can be seen that, in addition to the lateral and third break lights, also the two areas corresponding to reflectors are erroneously identified as tail lights. Hence, in the following, a set of rules to identify only the true detections will be explained in details.

A. Lateral Brake Lights Detection

The pair of lateral brake light is detected based on geometrical considerations. We assume that two candidate regions R_i and R_j ($i \neq j$) - a candidate pair - must be situated at a similar height in the image plane. Calling $C_R^i = (u_R^i, v_R^i)$ and $C_R^j = (u_R^j, v_R^j)$ the centers of the two regions respectively, this translates in the condition: $\text{abs}(v_R^i - v_R^j) < \tau_v$.

Furthermore, the two candidates R_i and R_j , must have similar dimensions and shape. This can be expressed through the *interception over union* score, computed after translating the two regions so that $C_R^i \equiv C_R^j$:

$$I_S^{i,j} = \frac{|R_i \cap R_j|}{|R_i \cup R_j|} \geq \tau_S \quad (3)$$

where the operator $|\cdot|$ means the cardinality of a set of pixels. Assuming the preceding vehicle to have only a single pair of tail lights, we need to discriminate among the possible multiple detections. To do so, we introduce two more scores. The first one is related to the candidate couple dimensions, assuming tail lights to be the biggest red chromaticity regions in the preceding vehicle image:

$$I_D^{i,j} = \frac{|R_i| + |R_j|}{\sum_{k=1}^{N_p} |R_k|} \quad (4)$$

where the total area $|R_i| + |R_j|$ of the pair itself is normalized *w.r.t.* the sum of all the candidate pairs area (N_p is the number of high red chromaticity regions which also form a candidate pair). Finally, in ideal conditions, we expect the two lateral brake lights center C_R^i and C_R^j to be one to the left, one to the right and equidistant *w.r.t.* the vertical axis of symmetry of the image itself. These two conditions can be summarized with the following score, considering R_i and R_j to be the left and right brake light respectively, namely $u_i < u_j$:

$$I_U^{i,j} = \max \left\{ 0, \frac{\min\{W_I/2 - u_R^i, u_R^j - W_I/2\}}{\max\{W_I/2 - u_R^i, u_R^j - W_I/2\}} \right\} \quad (5)$$

where W_I is the width in pixels of the vehicle image. Notice that scores (3) and (4) are inspired by the work [7], while the score (5) has been added to increase the robustness of the detection strategy. Finally, we combine all the scores into an overall score for lateral brake lights detection:

$$I_{lb}^{i,j} = \lambda_S \cdot I_S^{i,j} + \lambda_D \cdot I_D^{i,j} + \lambda_U \cdot I_U^{i,j} \quad (6)$$

so that, among all the candidate pairs (R_i, R_j) , we can choose the one which corresponds to the higher value of $I_{lb}^{i,j}$ to define the couple of lateral brake lights of the preceding vehicle, namely (R_l, R_r) (see Figure 4e and 4g). The weights λ_S , λ_D and λ_U are chosen to minimize the number of wrong detections, and satisfy the relationship $\lambda_S + \lambda_D + \lambda_U = 1$.

B. Third Brake Light Detection

The detection of the third brake light is highly dependent on the previous lateral lights detection. In fact, we expect such *third* light to be found in an higher position on the back of the car *w.r.t.* the lateral brake lights, and approximately in the middle of these last two. The first statement is a necessary condition for a detected region R_k (see Figure 4g) to be considered as a candidate third brake light, and can be formalized as follows:

$$v_R^k \leq \frac{v_R^l + v_R^r}{2} \quad (7)$$

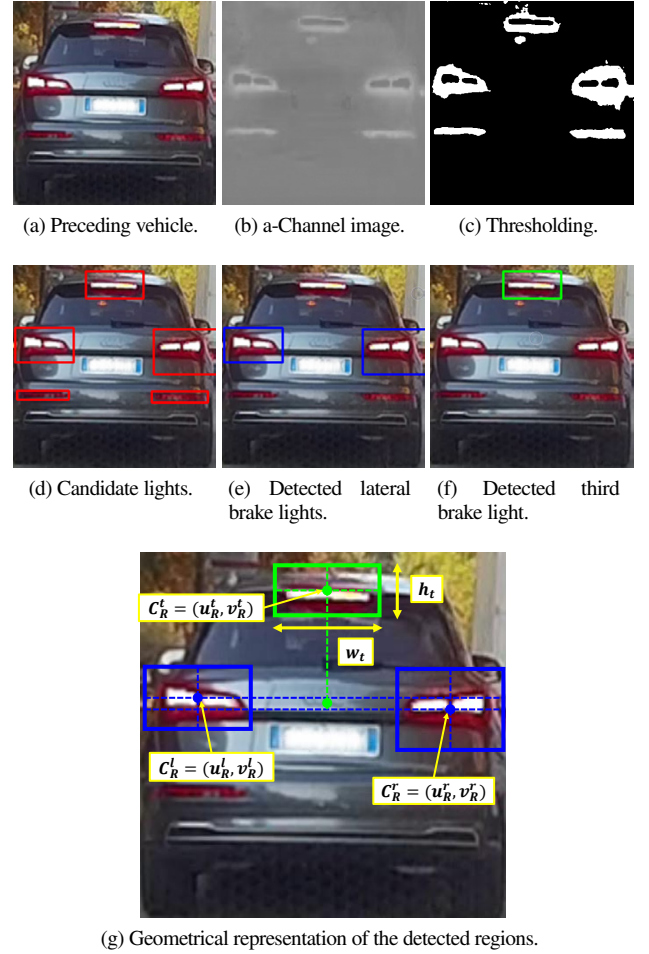
where $C_R^k = (u_R^k, v_R^k)$ is the center of the region R_k and C_R^l and C_R^r are the centres of the previously identified left and right lateral brake lights respectively. Regarding the second condition instead, it can be translated in the following score:

$$I_{tb}^k = \frac{\min\{u_R^k, (u_R^l + u_R^r)/2\}}{\max\{u_R^k, (u_R^l + u_R^r)/2\}} > \tau_{tb} \quad (8)$$

that takes value in $[0,1]$ and must be greater than a minimum value τ_{tb} . Finally, by common experience we expect the third stop light to have a width/height ratio bigger than one. Thus, calling w_R^k and h_R^k the width and the height of the k -th candidate third brake light, it must hold: $w_R^k/h_R^k > 1$. In the case that multiple candidate regions, the one with the highest score I_{tb}^k is taken as third stop light of the preceding vehicle, namely R_t (see Figure 4f and Figure 4g).

V. BRAKE LIGHTS STATUS CLASSIFICATION

For what concerns the light status classification step, it can be assessed as a two class classification problem (lights can be on or off). Such problem can be successfully solved by means of Support Vector Machine classifiers (see [5]) which takes as



(g) Geometrical representation of the detected regions.

Fig. 4: Brake light regions detection overview

input features extracted from the previously identified candidate regions R_l , R_r and R_t .

Features are computed in the L^*a^*b colorspace, considering the L^* and a^* channels only, which in our setup carry all the useful information (pixels intensity and red chromaticity respectively). Calling $L^*(u_i, v_i)$ and $a^*(u_i, v_i)$ the L^* and a^* channel values of the generic pixel (u_i, v_i) in the image plane, the following features are extracted:

- given the identified light region R_k , the maximum value of pixels belonging to R_k in the L^* and a^* channel:

$$L_{max}^k = \max_{(u_i, v_i) \in R_k} L^*(u_i, v_i), \quad a_{max}^k = \max_{(u_i, v_i) \in R_k} a^*(u_i, v_i)$$

- the minimum value of pixels belonging to R_k in the L^* and a^* channel:

$$L_{min}^k = \min_{(u_i, v_i) \in R_k} L^*(u_i, v_i), \quad a_{min}^k = \min_{(u_i, v_i) \in R_k} a^*(u_i, v_i)$$

- the average value of pixels belonging to R_k in the L^* and a^* channel:

$$L_{avg}^k = \text{avg}_{(u_i, v_i) \in R_k} L^*(u_i, v_i), \quad a_{avg}^k = \text{avg}_{(u_i, v_i) \in R_k} a^*(u_i, v_i)$$

These features are very similar to the ones proposed in [5], with the difference of being computed in the L^*a^*b colorspace instead of YC_bC_r . Furthermore, two more features are considered:

- the percentage of pixels belonging to R_k in the L^* and a^* channel which value is bigger than a threshold:

$$L_{\%}^k = \frac{|(u_i, v_i) \in R_k | L^*(u_i, v_i) > \tau_L|}{|R_k|}$$

$$a_{\%}^k = \frac{|(u_i, v_i) \in R_k | a^*(u_i, v_i) > \tau_a|}{|R_k|}$$

where τ_L and τ_a are chosen to maximize the true positive and true negative rate during training phase;

- finally, the average value of pixels in the L^* and a^* channel computed on the whole vehicle image:

$$L_{avg}^{tot} = \text{avg}_{(u_i, v_i)} L^*(u_i, v_i), \quad a_{avg}^{tot} = \text{avg}_{(u_i, v_i)} a^*(u_i, v_i)$$

Notice that features L_{avg}^{tot} and a_{avg}^{tot} , which are computed over the whole vehicle image and thus are a measure of the average brightness and red chromaticity of the scenario, guarantee robustness with respect to lighting changes.

A training set for the SVM classifier can be created based on the features described so far. To this end, a set of 9700 car images has been collected in an experimental campaign, of which 5200 have brake lights on. Two different gaussian-kernel SVMs [18] are trained, one for the lateral brake lights and one for the third brake light status classification (results in Figure 5a and 5b). This choice is made bearing in mind that the third brake light has different (and much more standardized) shape and mounting position *w.r.t.* the lateral lights, leading to a different (and less sparse) training data-points distribution in the feature space. Considering two SVMs instead of a single one, makes the third stop light classification more robust and independent from the shape and positioning variability of the lateral stop lights in the training set. Given that each brake light status is classified independently, in order to determine if a vehicle is braking or not, we look at the classification results as a whole. We consider a vehicle to be braking if *more than a half* of his related brake lights are classified as turned on.

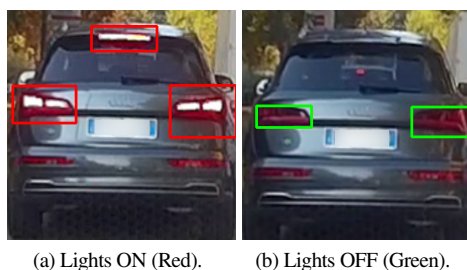


Fig. 5: Example of brake lights classification using the trained SVMs.

VI. EXPERIMENTAL RESULTS

The proposed brake lights status classification has been implemented in PythonTM using the OpenCV libraries and its performance assessed over a test dataset created using videos acquired in real driving situations. The values of the algorithm parameters, outlined in the previous Sections, are summarized in Table I.

The setup for the acquisitions is composed of a single FullHD (1920x1080 pixels, 30 fps) commercial camera mounted on the front part of a test car, with zero roll and pitch angles and at a known height \bar{h} . Data were recorded in car following scenarios

Parameter	Units of Measure	Value
τ_A	[%]	0.2
τ_v	[pixel]	60
τ_S	[%]	0.3
τ_{tb}	[-]	0.7
τ_L	[-]	150
τ_a	[-]	155

TABLE I: Algorithm parameters values.

(*i.e.* the test vehicle was following random cars) on urban and extra urban structured roads in daytime conditions. In order to compute the overall performance indices of the whole detection strategy on a set of frames numerically balanced *w.r.t.* braking and non braking situations, the following steps were performed. First of all, the lane and preceding vehicle detection algorithm, whose performance are dependent on temporal correlation of frames because of Kalman Filters exploitation, were executed on the whole dataset. From the analysed videos, a set of 4570 frames was randomly extracted. In 2390 frames the preceding vehicle has the brake lights activated, while in the remaining 2180 the preceding vehicle is not braking. Finally, the brake lights detection and status classification algorithm was executed on the extracted frames. The performances of overall strategy are summarized in Table II, by means of Precision, Recall and F1 scores. True positives (TP) and true negatives (TN) refers to frame in which the preceding vehicle is correctly identified and its brake lights are correctly classified as ON or OFF respectively. If the preceding vehicle is not detected because of failure of the lane detection and/or vehicle detection strategy, such frame is deemed as a false negative and contributes to lower the performance indices here computed. Looking at the results, high values of Precision and Recall (96.3% and 93.7% respectively), along with the consequently high F1 score, confirm the effectiveness of the proposed strategy in detecting the preceding vehicle and classifying its lights status.

In Figure 6 and 7 some qualitative results of the overall strategy are shown. More in details, in Figure 6 results of the lane detection and vehicle detection strategy are depicted. From the images it can be seen how the lane detection strategy is capable of correctly identifying the lane markings corresponding to the current driving lane in different illumination conditions and even in presence of strong shadows (top left image). Furthermore it can be noticed how the vehicle detection algorithm is able to correctly choose, even in presence of multiple vehicles, the one sharing the current driving lane with the ego vehicle, thanks to the information retrieved from the lane detection step.

In Figure 7, downstream to the lane and vehicle detection algorithm execution, some results of the brake light detection and classification strategy are shown. Looking at the true positive classification results in Figure 7a, it can be seen how the proposed

n° Frames	TP	TN	Precision	Recall	F1
4570	2292	2025	0.963	0.937	0.950

TABLE II: Overall detection and classification results.

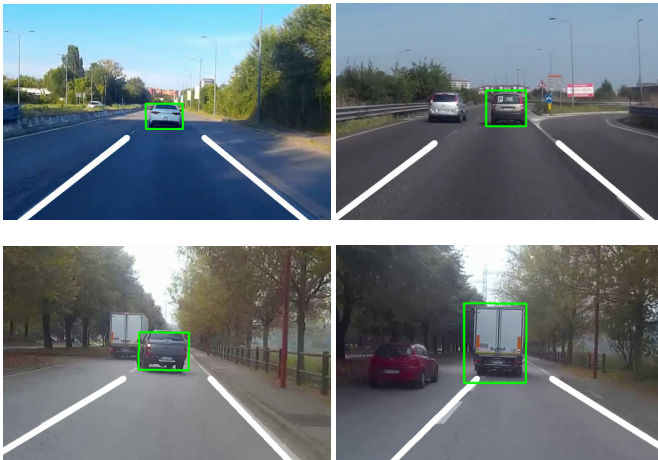


Fig. 6: Results of the lane detection (white lines) and preceding vehicle detection (green bounding-box) strategy. For the sake of clarity, frames have been zoomed-in in correspondence of the preceding vehicle.

strategy is capable of correctly identify and classify vehicle lateral and third stop lights when the preceding vehicle is braking, despite of different illumination conditions and losses in resolution due to higher distances between the preceding and the ego vehicles. The algorithms shows as well robustness to different shapes, dimensions and positioning of both lateral and third stop lights (for example in the case of the detected truck in the lower right image). Finally, looking at the lower left image, it can be seen that the strategy works correctly even if the third stop light is not detected (in this case it is off even if the vehicle is braking, probably because of a malfunctioning). Similar conclusions can be drawn looking at the true negative classification results in Figure 7b, where it is shown how the algorithm is capable of correctly identify and classify lateral brake lights also when turned off. In this case it is worth to notice that the third brake light, when turned off, is very often not recognized by the detection algorithm. This is because such light is usually positioned behind the car back window and thus not visible if not active.

REFERENCES

- [1] R. Kanjee, A. K. Bachoo, and J. Carroll, "Vision-based Adaptive Cruise Control using pattern matching," in *2013 6th Robotics and Mechatronics Conference (RobMech)*, oct 2013, pp. 93–98.
- [2] H. Y. Lin, L. Q. Chen, Y. H. Lin, and M. S. Yu, "Lane departure and front collision warning using a single camera," in *2012 International Symposium on Intelligent Signal Processing and Communications Systems*, 2012, pp. 64–69.
- [3] E. Dagan, O. Mano, G. Stein, and A. Shashua, "Forward collision warning with a single camera," in *IEEE Intelligent Vehicles Symposium, 2004*, pp. 37–42.
- [4] R. O'Malley, E. Jones, and M. Glavin, "Rear-Lamp Vehicle Detection and Tracking in Low-Exposure Color Video for Night Conditions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 2, pp. 453–462, jun 2010.
- [5] X. Wang, J. Tang, J. Niu, and X. Zhao, "Vision-based two-step brake detection method for vehicle collision avoidance," *Neurocomputing*, vol. 173, pp. 450–461, jan 2016.
- [6] Z. Cui, S. W. Yang, and H. M. Tsai, "A Vision-Based Hierarchical Framework for Autonomous Front-Vehicle Taillights Detection and Signal Recognition," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, sep 2015, pp. 931–937.



(a) True Positives



(b) True Negatives

Fig. 7: Examples of correct functioning of the brake lights detection and classification strategy (red bounding-boxes stand for brake light on, green ones brake light off).

- [7] H. T. Chen, Y. C. Wu, and C. C. Hsu, "Daytime Preceding Vehicle Brake Light Detection Using Monocular Vision," *IEEE Sensors Journal*, vol. 16, no. 1, pp. 120–131, jan 2016.
- [8] I. Cabani, G. Toulminet, and A. Benshair, "Color-based detection of vehicle lights," in *IEEE Intelligent Vehicles Symposium*, 2005, pp. 278–283.
- [9] J. G. Wang, L. Zhou, Y. Pan, S. Lee, Z. Song, B. S. Han, and V. B. Saputra, "Appearance-based Brake-Lights recognition using deep learning and vehicle detection," in *IEEE Intelligent Vehicles Symposium*, 2016, pp. 815–820.
- [10] L. C. Chen, J. W. Hsieh, S. C. Cheng, and Z. R. Yang, "Robust Rear Light Status Recognition Using Symmetrical SURFs," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, sep 2015, pp. 2053–2058.
- [11] A. Almagambetov, S. Velipasalar, and M. Casares, "Robust and Computationally Lightweight Autonomous Tracking of Vehicle Taillights and Signal Detection by Embedded Smart Cameras," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3732–3741, jun 2015.
- [12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," jun 2015.
- [13] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," dec 2016.
- [14] S. P. Narote, P. N. Bhujbal, A. S. Narote, and D. M. Dhane, "A review of recent advances in lane detection and departure warning system," *Pattern Recognition*, vol. 73, pp. 216–234, jan 2018.
- [15] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Comput. Soc, pp. 1106–1112.
- [16] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft COCO: Common Objects in Context," may 2014.
- [17] E. Cuevas, D. Zaldivar, and R. Rojas, "Kalman filter for vision tracking," *Measurement*, vol. 33, 01 2005.
- [18] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1.