Automatic Detection and Classification of Road Lane Markings Using Onboard Vehicular Cameras

Maurício Braga de Paula and Cláudio Rosito Jung, *Senior Member, IEEE*

***Abstract*—This paper presents a new approach for road lane classification using an onboard camera. Initially, lane boundaries are detected using a linear–parabolic lane model, and an automatic on-the-fly camera calibration procedure is applied. Then, an adaptive smoothing scheme is applied to reduce noise while keeping close edges separated, and pairs of local maxima–minima of the gradient are used as cues to identify lane markings. Finally, a Bayesian classifier based on mixtures of Gaussians is applied to classify the lane markings present at each frame of a video sequence as dashed, solid, dashed solid, solid dashed, or double solid. Experimental results indicate an overall accuracy of over 96% using a variety of video sequences acquired with different devices and resolutions.**

***Index Terms*—Lane detection, lane marking classification, onboard vehicular cameras, driver assistance systems, pattern classification.**

I. Introduction

According to the Global Status Report on road safety [1] published by the World Health Organization (WHO), road traffic injuries are the eighth leading cause of death in the world, and current research suggests that by 2030 this will become the fifth leading cause. Furthermore, approximately 1.24 million people die every year on the roads, and another 20 to 50 million sustain non-fatal injuries as a result of road traffic crashes [1].

One direction toward the reduction of traffic accidents is the development of computer vision systems for Intelligent Transportation Systems (ITS), Driving Assistance Systems (DAS) based on onboard vehicular cameras and autonomous driving. In particular, lane changes or overtaking another vehicle are one of the most dangerous driving maneuvers, being one of the main causes for head-on or sideswipe collisions, which motivate the development of Lane Departure Warning Systems (LDWS) [2]. One key issue in such systems is to identify and classify road lane markings, since they dictate regions of the road where overtaking is possible or not.

This paper proposed a new approach for classifying road lane markings using an onboard camera. Initially, lane boundaries are detected using a linear-parabolic lane model, and an automatic on-the-fly camera calibration procedure is applied. Then, an adaptive smoothing scheme is applied to reduce noise while keeping close edges separated, and pairs of local maxima-minima of the gradient are used as cues to identify lane markings. Then, a two-stage method is used to classify the lane markings: in the first stage, a Bayesian classifier based on Mixtures of Gaussians is applied to classify the lane markings present at each frame of a video sequence as dashed, solid, dashed-solid/solid-dashed or double solid; finally, the second stage disambiguates between dashed-solid and solid-dashed.

The remainder of this paper is organized as follows. Section II revises some existing approaches for detection and recognition of types of lane markings, and our approach for road marking extraction and classification is presented in Section III. Section IV illustrates some experimental results, and concluding remarks are presented in Section V.

II. Related Work

Vision-based lane analysis is an important component for Driving Assistance Systems (DAS) and ego-vehicle localization. The generic problem of detecting and following the lane boundaries has been studied extensively by the computer vision community, and a wide variety of lane models and fitting techniques have been proposed in the past years, as shown by recent survey papers [2], [3]. However, far less effort has been devoted to the problem of lane markings classification. Next, we present some approaches that somehow tackle lane markings classification.

Lindner *et al.* [4] presented an approach to extract reliable lane markings using multi-level feature extraction and classification. The image data for the system is provided by the front camera of concept vehicle Carai.[1] First, an edge detector pre-processes the gray scale image, and subsequently a set of algorithms look for a group of four objects, like: lines, curves, parallel curves, closed objects. Based on the set of closed objects, a feature extraction algorithm calculates some information about the geometry of the object, fitting a momentum ellipse to the edge points of the linked curves. The extracted features are the number of edges, major axis

Manuscript received October 6, 2014; revised February 4, 2015 and May 15, 2015; accepted May 18, 2015. The Associate Editor for this paper was P. Grisleri.

M. B. de Paula is with the Department of Mathematics and Statistics, Federal University of Pelotas (UFPEL), 96160-000 Pelotas-RS, Brazil, and also with Institute of Informatics, Federal University of Rio Grande do Sul (UFRGS), 91509-900 Porto Alegre-RS, Brazil (e-mail: mbpaula@inf.ufrgs.br; maubrapa@ufpel.edu.br).

C. R. Jung is with Institute of Informatics, Federal University of Rio Grande do Sul (UFRGS), 91509-900 Porto Alegre-RS, Brazil (e-mail: crjung@inf.ufrgs.br).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2015.2438714

[1] http://www.carai.eu/

direction, width and height. A binary decision tree with four output classes performs the classification process. The authors chose the following markings (all dashed): long wide mark, long thin mark, short wide mark and short wide mark.

In [5], McCall *et al.* presented an approach for unifying the detection of multiple types (originally suggested in [6]) of lane marking including bott's dots, also providing robustness to shadowing. However, the multiple marks cited by the authors do not take into account the classification of markings; only the detection and tracking of the lane.

Schubert *et al.* [7] presented a lane change situation assessment system. The proposed approach can perceive the environment around the vehicle, assess the traffic situation, recognize appropriate occasions and give recommendations about lane change maneuvers to the driver. The assessment algorithm, responsible to decide whether a lane change is possible or not, classifies the lane markings into dashed or solid types. Based on the tracked lane markings, a discrete-space signal is derived and used as input to a Fast Fourier Transform (FFT). The power spectrum is calculated, and then the types of lane markings are classified. Collado *et al.* [8] also employed a frequency approach based on the power spectrum, but differently from [7], they distinguish a third type of lane marking known as merged (solid-dashed or dashed-solid, with no distinction between them).

In [9] (as cited in [10]), Collado *et al.* split the problem into a few steps: to create a birds-eye view of the road, to segment the pixels which belong to longitudinal road markings, to extract the lane boundaries using the Hough Transform, and to realize adjustments in the pitch angle. Then, lane boundaries are classified as continuous (also called solid or single-solid), broken (known as dashed) or merged, computed by the power spectrum of the FFT. This scheme needs a previous camera calibration for the initial procedure that generates the bird-eye view.

More recently, Satzoda and Trivedi [11] proposed a lane feature extraction method for embedded solutions using Inverse Perspective Mapping (IPM). Based on the bird's eye image, the authors use "scan bands" with configurable width to predict the existence of lane markings. Within these bands, image intensities are filtered by a 5 × 5 steerable filter to detect horizontal variations, and then thresholded. The two resulting binary images are combined using a shift and match technique, aiming to detect evidences of lane markings. The technique was evaluated with LISA-Q testbed data set, composed by five different data sets: freeway lanes, freeway concrete surface, urban road with shadows, freeway with vehicles and freeway circular reflectors. Three types of markings are presented in this data set: dashed, solid and circular reflectors.

In [12], Nedevschi *et al.* introduced the use of periodic histograms to determine the type of lane markings in order to improve the ego-vehicle global localization with respect to an approaching intersection. Although the main focus of their work was ego-localization, they are also able to classify lane boundaries into four types: (i) single solid; (ii) double-solid; (iii) dashed; (iv) merged.

de Paula and Jung [10] presented an approach for recognizing road lane boundaries using an uncalibrated onboard camera. They extract cues on the existence of lane markings based on training and test patches, and explore the temporal variation of the extracted cues to classify the lane boundaries as dashed, dashed-solid, solid-dashed, single-solid and double-solid. Suchitra *et al.* [13] presented a modular approach to detect and distinguish three main types of lane markings, namely dashed, solid and zigzag. The fundamental unity of their process is the Basic Lane Marking (BLM), which particularly treats lane markings as dashed or solid, based on continuity properties. The image frame is divided into blocks and from the position of the lane boundaries and the existence of a marking (arising from the BLM), an edge map is created. As in [10], temporal information is used to identify lane markings, resulting in delayed detection of lane markings when the marking type changes.

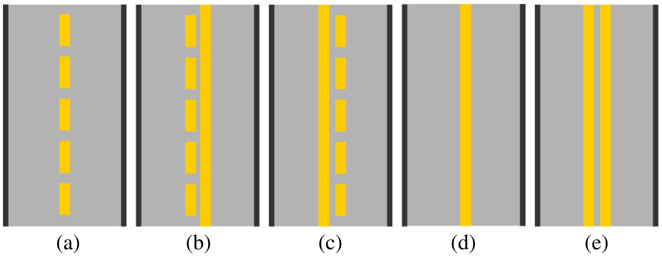


Fig. 1. Types of lane boundaries: (a) Dashed. (b) Dashed-solid. (c) Solid-dashed. (d) Single solid. (e) Double solid.

It is interesting to notice that the types of lane markings to be detected vary considerably along the different approaches, and there is no standardized database with ground truth data to allow a fair comparison between different methods (in fact, the literature review shows that most papers do not show comparisons with potentially competitive techniques). Also, approaches that explore temporal information (e.g., [10]) tend to produce lags when detecting transitions between different lane marking types. Methods that explore IPM (e.g., [10], [11]) assume a calibrated camera, which might be a limitation when detachable devices are used, such as mobile phones.

In this paper, we present a new method to classify road lane markings using an onboard camera inside a moving vehicle into five possible types: dashed, dashed-solid, solid-dashed, single-solid and double-solid, as shown in Fig. 1. It requires an onboard camera with known intrinsic camera parameters (which can be done off-line using publicly available calibration tools, such as [14]), allowing the use of detachable devices. Also, it presents a frame-wise classification of the lane markings without needing a large set of temporal data (as in [10]), producing classification results without significant lags during transitions of lane marking types. As an additional contribution, we also provide a set of video sequences with ground truth data that can be used as a benchmark to compare different methods for lane markings classification. The proposed method is detailed next.

III. The Proposed Approach

The first step in our approach is to detect and track the lane boundaries. Despite the existence of a great variety of approaches, we adopted the model proposed in [15], due to its simplicity and robustness. In such approach, the lane

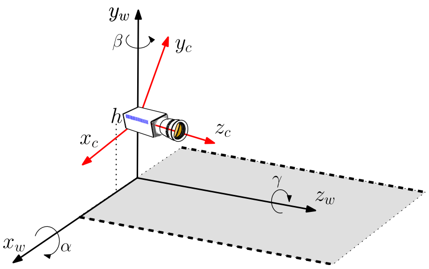


Fig. 2. 3-D world and camera coordinate systems.

boundaries are modeled as a linear function in the near field (to robustly estimate the local orientation) and a parabolic function in the far field, so that curved portions of the road can be tracked. More precisely, the mathematical model is given by

$$f^k(v) = \begin{cases} a^k + b^k(v - v_m), & \text{if } v > v_m \\ a^k + b^k(v - v_m) + c^k(v - v_m)^2, & \text{if } v \le v_m \end{cases} \quad (1)$$

where $k \in \{r, l\}$ denotes which lane boundary we are referring to (right or left), v_m defines the boundary in the image between the near and far fields, and v is the vertical pixel component. Given the lane models at frame t, the model at frame $t+1$ is obtained by computing the gradient magnitudes within a neighborhood of the current model, and then fitting Eq. (1) using least squares weighted by the obtained gradient magnitudes.

The second step is to detect lane marking cues over a set of 1-D scan line segments, equally spaced in world coordinates along the main axis of the road, which requires a calibrated camera. For that purpose, we used the self-calibration scheme presented in [16], which works on-the-fly and is also based on the linear-parabolic model. Finally, the lane marking cues are fed to a two-layer classifier. These steps are described next.

A. The Camera Setup

Let us consider that we have an onboard vehicular camera positioned at $(0, h, 0)$ in the world coordinate system (WCS), as illustrated in Fig. 2, where α (pitch), β (yaw) and γ (roll) are the Euler angles related to the x, y and z axis, respectively.

As in [16], we also consider that there is no roll (since such rotation is typically is prevented by the windshield), and assume a pinhole camera model. Then, the calibration procedure explores the expected lane geometry within a straight portion of the road, allowing to compute the rotation angles α and β, as well as the camera height h. In such model, given a point $\boldsymbol{x}_w = (x_w, y_w, z_w)^T$ in the world coordinate system, the corresponding point $\boldsymbol{x}_c = (x_c, y_c, z_c)^T$ in the 3-D camera coordinate system is given by a rigid transformation

$$\boldsymbol{x}_c = \boldsymbol{R}(\boldsymbol{x}_w - \boldsymbol{x}_0) \quad (2)$$

where $\boldsymbol{x}_0$ is the position of the camera system, and

$$\boldsymbol{R} = \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ -\sin\alpha\sin\beta & \cos\alpha & -\cos\beta\sin\alpha \\ \cos\alpha\sin\beta & \sin\alpha & \cos\alpha\cos\beta \end{bmatrix} \quad (3)$$

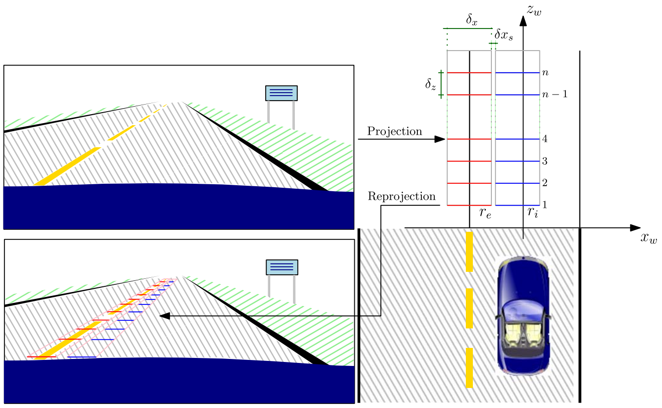


Fig. 3. Projection of the lane boundaries from image to world coordinate. Definition of the line segments in the WCS. Reprojection of the line segments to ICS.

is the rotation matrix (with no roll). The corresponding projection of $\boldsymbol{x}_c$ into an image pixel $\boldsymbol{u} = (u, v)^T$ is given by

$$u = \frac{f s_u x_c}{z_c}, \quad v = \frac{f s_v y_c}{z_c} \quad (4)$$

where f is the focal length of the camera, s_u and s_v relate to the pixel dimensions, and $(u, v)^T$ are image coordinates relative to the optical axis $O = (u_o, v_o)^T$ of the camera. Since all processing is applied to points on the ground plane (i.e., with $y_w = 0$), we can map the location of each pixel $\boldsymbol{u} = (u, v)^T$ to a world-coordinate point on the ground plane $\boldsymbol{x}_g = (x_w, 0, z_w)$ by combining Eqs. (2)–(4), setting y_w to zero, and solving for x_w and z_w, yielding

$$x_w = -\frac{h\,((u - u_o)\cos\beta + \sin\beta(f\cos\alpha - v\sin\alpha + v_o\sin\alpha))}{f\sin\alpha + v\cos\alpha - v_o\cos\alpha} \quad (5)$$

$$z_w = -\frac{h\,((u_o - u)\sin\beta + \cos\beta(f\cos\alpha - v\sin\alpha + v_o\sin\alpha))}{f\sin\alpha + v\cos\alpha - v_o\cos\alpha}. \quad (6)$$

For more details on the calibration scheme the reader should refer to [16].

B. Road Marking Extraction

After computing lane boundaries and calibrating the camera, the next step is to extract cues related to the local geometry of lane markings. As noted in [11], the computation of intensity variations along the horizontal direction within scan bands can be useful to extract lane markings. However, their approach explores scan bands over the whole image, and a predefined threshold is used to retrieve relevant "edges," which is quite dependent on illumination conditions and color/quality of painting. In our approach, the main idea is to place 1-D scan line segments centered at the detected lane boundary (as explained before, lane boundaries are detected using [15]), and to evaluate the behavior of the first derivative along these segments.

Given the input image with the detected lane boundaries, we compute the bird's eye view and place $p_{\max}$ horizontal line segments r_e^p with length δ_x meters centered at the lane boundary, for $p = 1, 2, \ldots, p_{\max}$, and equally spaced by δ_z

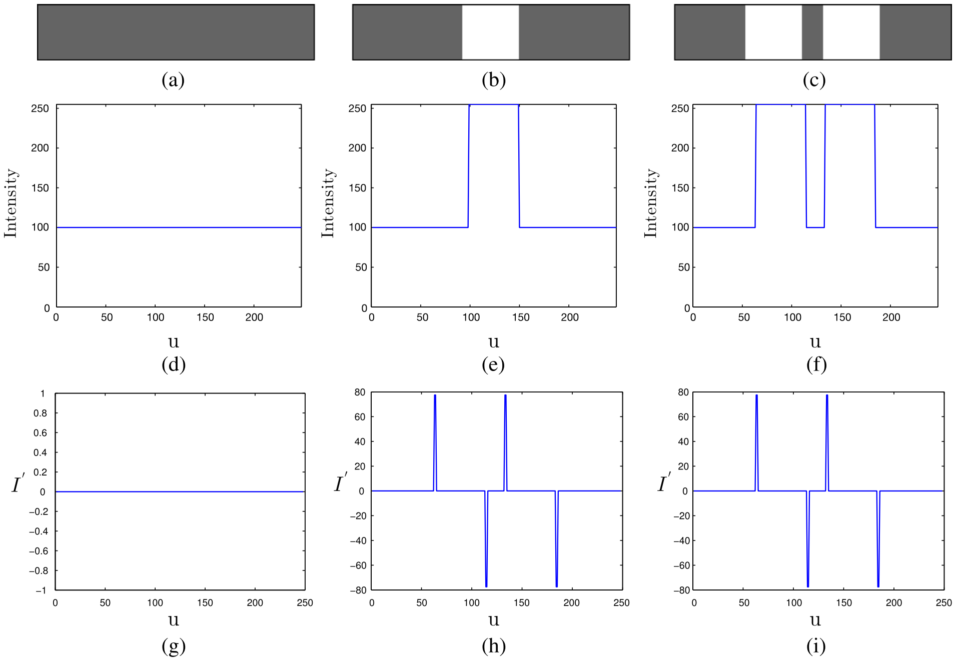


Fig. 4. (a) A simple image with no marks. (b)-(c) An image with one and two marks, respectively. (d)-(f) 1-D horizontal gray level along the center of the image. (g)-(i) Derivative of each image line.

meters. We also use a set of internal line segments r_i^p, with the same dimensions and spacing as r_e, but displaced δx_s units to the interior of the lane. The scan lines r_e and r_i are then projected back to the original image, as illustrated in Fig. 3.

Along r_e^p, the image profile $I_e^p(u)$ along the horizontal image coordinate u should relate to the local geometry of the lane marking: if no marking is present, $I_e^p(u)$ should be approximately constant, so that its derivative $I_e^{p'}(u)$ should be low and roughly flat. If a single marking is present (as in solid and dashed marking types), $I_e^{p'}(u)$ should present one strong local maximum and another strong local minimum (i.e., a pair of peaks), and if two laterally adjacent markings are present (as in double solid, dashed-solid and solid-dashed markings), $I_e^{p'}(u)$ should present two pairs of peaks. These three situations are illustrated in Fig. 4. Fig. 4(a)–(c) show three synthetic images road patches with different types of local lane markings. Fig. 4(d)–(f) show the gray level intensity $I_e^p(u)$ along a horizontal scan, and the respective derivatives $I_e^{p'}(u)$ are shown in Fig. 4(g)–(i).

However, peak detection (or edge detection in the generic case) is not a trivial task in practical scenarios. Illumination conditions, color of the markings and weak painting can generate peaks in $I_e^{p'}(u)$ with low height; conversely, noise, texture of the pavement and other artifacts may produce spurious peaks. Image/signal smoothing is a classical solution, but the choice of the kernel size is an issue: larger kernels filter out more noise, but may blur together nearby edges, as predicted by the scale-space theory [17]. Hence, the kernel should be large enough to remove noise, but sufficiently small to preserve neighboring edges, which is a common scenario when detecting lane markings [see Fig. 4(f)]. Fig. 5 illustrates the possible problem of edge interference when a large kernel is used.

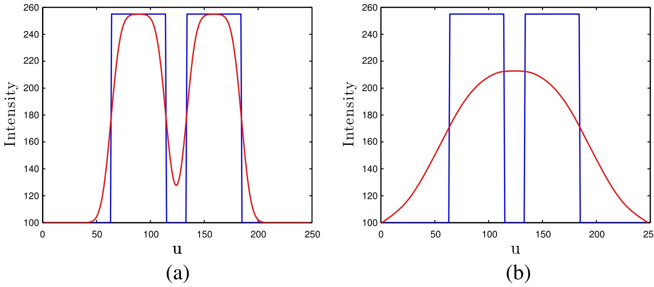


Fig. 5. Effects of using a Gaussian filter: (a) Original (blue) and filtered (red) 1-D line patch with an adequate σ value. (b) Original and filtered 1-D line patch with a larger σ, merging neighboring edges into a single one.

In the IPM image, the kernel size should be based on the lateral distance between lane markings (in double solid, dashed-solid and solid-dashed types), which is constant along the whole image. In our approach, however, the whole processing is performed directly in the original image domain. In the projected image, such inter-marking spacing varies along the image lines, due to camera perspective. Fortunately, the use of a calibrated camera allows us to map the maximum kernel size in the WCS into the image space. More precisely, for each line segment p we filter the image profile $I_e^p(u)$ using a Gaussian filter $g_p(u)$, given by

$$g_p(u) = \frac{1}{\sqrt{2\pi\sigma_p^2}} e^{-\frac{u^2}{2\sigma_p^2}} \tag{7}$$

where $\sigma_p = \text{proj}(\sigma_w, v_p)$ is the length of a projected horizontal line segment with original length σ_w (set to 0.020 m in our experiments) in WCS into the ICS using the vertical image coordinate v_p, based on Eqs. (2)–(4).

Since each lane marking produces one local maximum (when the intensity increases) and one local minimum (when it decreases), the next step is to find pairs of adjacent maximum-minimum peaks at each scan line, which should have similar

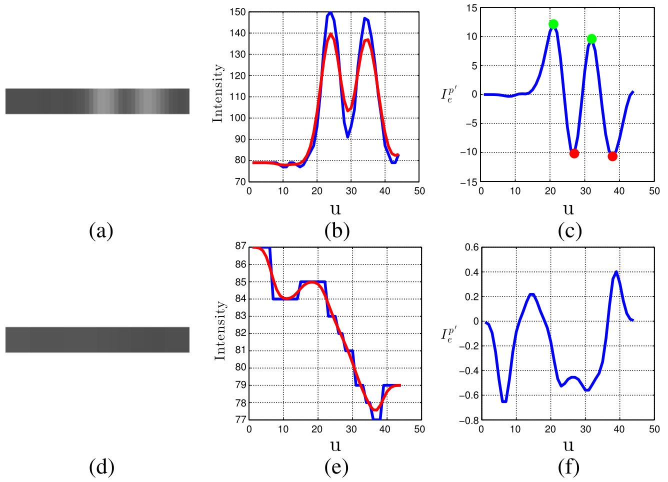


Fig. 6. Example of peak extraction along lane boundaries. (a) image profile with double solid marking, (b) Original (blue) and filtered intensities (red), (c) derivative of the filtered image profile intensity and detected pair of peaks. (d)-(f) show analogous images, but for an image profile with no marking.

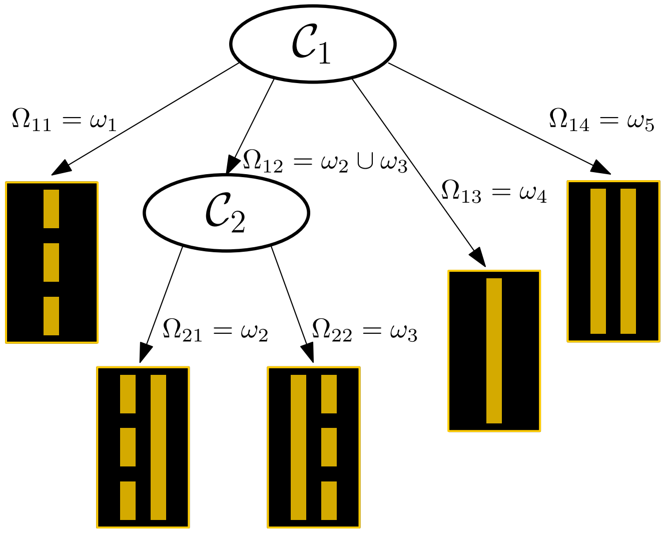


Fig. 7. Illustration of the proposed two-stage classifier.

heights (since the pavement typically presents one color and the painting of the lane markings another color). For that purpose, all the local extrema locations u_j are computed, and the global maximum $M^p_{\max} = \max_j |I^{p'}_e(u_j)|$ is extracted. Then, we retrieve only the local extrema whose height (in absolute value) is larger than $\varepsilon M^p_{\max}$, where $0 < \varepsilon < 1$ is threshold used to remove small peaks (we have defined $\varepsilon = 0.35$ experimentally).

It is important to notice that threshold ε is relative to the maximum peak height within r_e. Hence, scan lines with no lane markings will still produce small peaks related to pavement irregularities/noise that are not removed after this initial threshold. As a final validation step, we retrieve every pair of remaining adjacent maximum-minimum peaks, and compute the inter-peak height difference of the jth pair $\delta h^p_j = I^{p'}_e(u^j_{\max}) - I^{p'}_e(u^j_{\min})$, where $u^j_{\max}$ and $u^j_{\min}$ are the positions of the corresponding adjacent extrema. Then, we smooth the internal scan lines r_i (which are assumed contain pavement-related pixels) using the adaptive Gaussian kernel given in Eq. (7), and compute the derivatives $I^{p'}_i(u)$, extracting its standard deviation σ^p_i, which encodes the expected irregularity/noise of pavement-related pixels. Finally, the jth peak pair is validated if $\delta h^p_j > k\sigma^p_i$, where k is the number of standard deviations used in the rejection rule (our experiments indicated that $k = 10$ presents a good compromise between misdetections and false detections). Fig. 6 illustrates the peak extraction process.

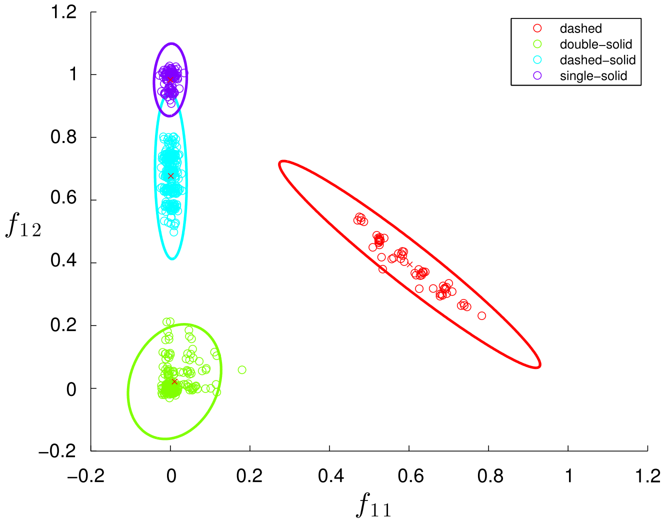


Fig. 8. Class distribution of $\boldsymbol{f_1}$ for different classes.

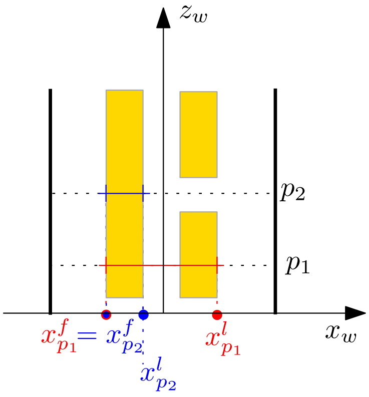


Fig. 9. Estimating the local maxima and minima abscissas.

It is important to mention that our road marking extraction process is conducted on a monochromatic image. Although the explicit use of color information could be useful, most lane markings are either white or yellow, while the pavement is typically grayscale (asphalt). Since in our work we use the luminance $Y = 0.299R + 0.587G + 0.114B$ as the input image, and the color yellow is mostly distinguishable in the red and green channels (that carry more weight in the computation of the luminance), we have sufficient contrast to detect the lane markings.

C. Road Marking Classification

For each scan line p, the number of extracted peak pairs n_p characterizes the local (horizontal) geometry of the lane

TABLE I
VIDEO CLIPS: CAMERA CALIBRATION PARAMETERS

Clip	Name	u_0	v_0	f_u / f_v[2]	$\alpha°$	$\beta°$	h (m)	Speed (km/h)	Frames[3]
1	clip_watec_DMA291113_HMS120139	319.50	239.50	674.30	-7.70	-0.81	1.16	90	200
2	clip_watec_DMA311013_HMS143419_100kmh	319.50	239.50	674.30	-4.01	2.27	1.44	100	200
3	clip_watec_DMA300114_HMS154757_80kmh	319.50	239.50	674.30	1.17	-0.89	1.16	80	200
4	clip_watec_DMA311013_HMS145905	319.50	239.50	674.30	-4.01	2.27	1.44	100	200
5	clip_iPhone3G_1395434215_01	191.50	143.50	441.50	4.36	-0.75	1.25	unknown	–
6	clip_iPhone3G_1395431301_70kmh	191.50	143.50	441.50	4.36	-0.75	1.25	70	150
7	clip_iPad_DMA190414_1	959.50	539.50	2.03e+03	-5.90	0.60	1.08	80	100
8	clip_iPhone4_1807_1560	639.50	395.50	1.64e+03	3.89	-3.10	1.05	60	200
9	clip_i5s_0085	959.50	539.50	1.81e+03	5.99	-0.85	1.22	70	200
10	clip_i5s_0094	959.50	539.50	1.81e+03	5.99	-0.85	1.22	unknown	–
11	clip_i5s_0096	959.50	539.50	1.81e+03	5.99	-0.85	1.22	unknown	–
12	clip_i5s_0104	959.50	539.50	1.81e+03	5.99	-0.85	1.22	unknown	–

[2] The focal length on both axes is the same.

[3] Number of frames used for camera calibration.

marking: zero pairs indicate no marking; one pair relates to a single lane marking; and two pairs relate to two laterally placed markings. The characterization of the lane markings along a given image frame t under analysis is given by a composition of the $p_{\max}$ values $n_p(t)$: in dashed markings (class ω_1), $n_p(t)$ should vary between 0 and 1; for dashed-solid (ω_2) or solid-dashed (ω_3), between 1 and 2; for single-solid (ω_4), it should be mostly 1; and for double solid (ω_5), mostly 2.

Since the analysis of $n_p(t)$ cannot be used to distinguish between dashed-solid and solid-dashed, a two-level classifier was developed, as shown in Fig. 7. Next, we explain our choices for the classifiers $\mathcal{C}_1$ and $\mathcal{C}_2$. Let Ω_{ij} denote the output class of classifier $\mathcal{C}_i$, as also shown in Fig. 7.

1) Classifier $\mathcal{C}_1$: The first classifier consists of identifying the lane markings as one of four possible categories: dashed (Ω_{11}), single-solid (Ω_{13}), double-solid (Ω_{14}) and merged (Ω_{12}), known as dashed-solid or solid-dashed.

At each frame t, we have a set of $p_{\max}$ values $n_p(t)$ related to the pth scan line, with possible values 0, 1 or 2. As the pavement may contain problems due to several factors, we applied the median filter along p in $n_p(t)$ to remove scan lines with isolated peak responses, probably due to outliers. Then, we propose the following feature vector for classifier $\mathcal{C}_1$:

$$\boldsymbol{f_1}(t) = \begin{pmatrix} f_{11}(t) \\ f_{12}(t) \end{pmatrix} = \begin{pmatrix} \#\{n_p | n_p = 0\}/p_{\max} \\ \#\{n_p | n_p = 1\}/p_{\max} \end{pmatrix} \tag{8}$$

where # denotes the number of elements in a set. Hence, $f_{11}(t)$ contains the fraction of scan lines presenting zero peaks, and $f_{12}(t)$ the fraction with one pair of peaks (the fraction of scan lines with two peaks in embedded in $f_{11}(t)$ and $f_{12}(t)$, since the sum of fractions equals one).

Fig. 8 presents a scatter plot of the feature vectors $\boldsymbol{f_1}(t)$ related to the four output classes of classifier $\mathcal{C}_1$, and it can be observed that they form well separated clusters in the 2-D feature space. Fig. 8 also show the iso-probable contours of a 2-D Gaussian distribution fitted to each class, indicating that a Mixture of Gaussians (MoG) is a good choice to model the global distribution.

TABLE II
SUMMARY OF VARIABLES AND DEFAULT VALUES

Symbol	Definition	Def. val.
p_{max}	number of horizontal line segments	19
δ_z	spacing between the horizontal line segments	0.5m
δ_x	width of horizontal line segments	1.0m
δx_s	distance between patches r_e and r_i	0.12m
σ_w	standard deviation of Gaussian filter	0.020m
ε	threshold for rejecting spurious peaks	0.35
$\epsilon(\rho)$	rejection threshold for the Mahalanobis distance	7.37
T	number of frames used in the temporal window	20

Finally, we have chosen to use a Bayesian classifier based on Gaussian distributions with equiprobable priors for $\mathcal{C}_1$, so that the chosen class is Ω_{1j}, with

$$j = \arg\max_i P(\Omega_{1i}|\boldsymbol{f_1}) = \arg\max_i \frac{p(\boldsymbol{f_1}|\Omega_{1i})}{\sum_{i=1}^{4} p(\boldsymbol{f_1}|\Omega_{1i})} \tag{9}$$

with $p(\boldsymbol{f_1}|\Omega_{1i}) \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{C}_i)$ being the 2-D Gaussian distribution of class Ω_{1i} with mean vector $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{C}_i$. Section IV describe in details the training and test data used in our experiments.

2) Classifier $\mathcal{C}_2$: As shown in Fig. 7, classifier $\mathcal{C}_2$ is fed with samples that have both dashed and solid components. So, its aim is to distinguish dashed-solid ($\Omega_{21} = \omega_2$) from solid-dashed ($\Omega_{22} = \omega_3$) markings. For that purpose, we evaluate the location of local maxima and minima along the different scan lines within a frame in the WCS. In a solid-dashed type, the first maximum tends to occur roughly at the same x_w coordinate, whereas the location of the last minimum varies along that axis, as illustrated in Fig. 9. For dashed-solid markings, the variation would occur in the first maximum.

To detect such behavior, we identify the locations of the first maximum x_p^f and the last minimum x_p^l of each scan line in the WCS, and compute their variances (along p) $\sigma^2(x^f)$ and $\sigma^2(x^l)$. Then, a simple rule is adopted for classifier $\mathcal{C}_2$: if $\sigma^2(x^f) < \sigma^2(x^l)$, the sample frame is assign to class ω_3 (solid-dashed); otherwise, it is assigned to class ω_2 (dashed-solid).

D. Improving the Classification Results

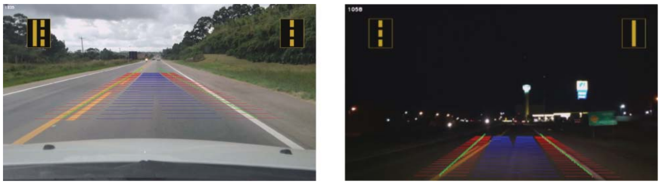
1) Outlier Detection: Frames that contain two different lane marking types (transition frames) tend to generate feature vectors $\boldsymbol{f_1}(t)$ that do not fit well into any of the four classes of classifier $\mathcal{C}_1$ shown in Fig. 8. Noise, occlusions, poor

TABLE III
RESOLUTION, FRAME RATE AND GLOBAL ACCURACY OF USED VIDEO CLIPS

Clip	Name	Resolution	FPS	Type	Frames	Train	Test	Accuracy
1	clip_watec_DMA291113_HMS120139	480 × 640	29.97	more than one	990	x	x	99.80%
2	clip_watec_DMA311013_HMS143419_100kmh	480 × 640	29.97	dashed	201	x	x	100.00%
3	clip_watec_DMA300114_HMS154757_80kmh	480 × 640	29.97	dashed	261		x	100.00%
4	clip_watec_DMA311013_HMS145905	480 × 640	29.97	more than one	498		x	100.00%
5	clip_iPhone3G_1395434215_01	288 × 384	14.64	more than one	73		x	98.63%
6	clip_iPhone3G_1395431301_70kmh	288 × 384	14.64	dashed	151		x	100.00%
7	clip_iPad_DMA190414_1	1080 × 1920	29.97	more than one	1501		x	99.57%
8	clip_iPhone4_1807_1560	720 × 1280	23.976	more than one	1190		x	99.41%
9	clip_i5s_0085	1080 × 1920	29.97	more than one	200		x	100.00%
10	clip_i5s_0094	1080 × 1920	29.97	more than one	1000		x	82.39%
11	clip_i5s_0096	1080 × 1920	29.97	more than one	1100		x	98.75%
12	clip_i5s_0104	1080 × 1920	29.97	more than one	1100		x	97.96%

TABLE IV
CONFUSION MATRIX OF CLIPS 3-12

		Predicted class					
		ω_1	ω_2	ω_3	ω_4	ω_5	Accuracy
Actual class	ω_1	**2312**	0	1	2	0	99.87%
	ω_2	2	**1225**	43	9	0	95.78%
	ω_3	193	2	**1747**	60	4	87.09%
	ω_4	130	0	73	**6555**	0	97.00%
	ω_5	0	3	0	0	**1995**	99.85%



(a) (b)

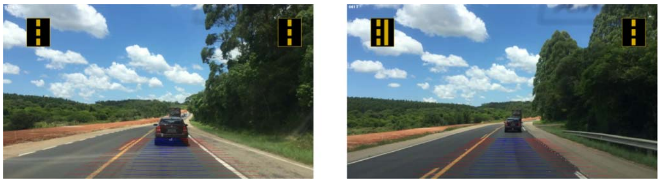
Fig. 10. Example frames with classification errors, due to problems in the lane tracker.

pavement or painting conditions may also generate spurious feature vectors. However, even in these cases, a large posterior value may be produced by Eq. (9). Since our class Probability Distribution Functions (PDFs) are bi-variate Gaussians, the squared Mahalanobis distance

$$d(\boldsymbol{f_1};\boldsymbol{\mu}_i,\boldsymbol{C}_i)^2 = (\boldsymbol{f_1}-\boldsymbol{\mu}_i)^T\boldsymbol{C}_i^{-1}(\boldsymbol{f_1}-\boldsymbol{\mu}_i) \tag{10}$$

can be used to evaluate how well a given sample vector $\boldsymbol{f_1}$ fits into a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_i;\boldsymbol{C}_i)$. In fact, the values $d\left(\boldsymbol{f_1};\boldsymbol{\mu}_i;\boldsymbol{C}_i\right)^2$ follow a chi-squared distribution with 2 degrees of freedom ($\mathcal{X}_2^2$), so that outliers can be detected if they fall on the tail of the distribution. More precisely, we select a threshold $\epsilon(\rho)$, defined based on a given quantile ρ of the chi-squared distribution (upper tail critical value) with probability level ρ as the outlier detection rule. If $c(t) \in \{\Omega_{11},\Omega_{12},\Omega_{13},\Omega_{14}\}$ denotes the class label for feature vector $\boldsymbol{f_1}(t)$ based on Eq. (9), the updated class $\hat{c}(t)$ is given

$$\hat{c}(t) = \begin{cases} \hat{c}(t-1), & \text{if } d\left(\boldsymbol{f_1}(t);\boldsymbol{\mu}_{c(t)},\boldsymbol{C}_{c(t)}\right)^2 > \epsilon(\rho) \\ c(t), & \text{otherwise} \end{cases} \tag{11}$$



(a) (b)

Fig. 11. Examples of limitations faced by our method. (a) small distance between car in front. (b) absence of right lane boundary painting.

meaning that the last reliably estimated class is kept when an outlier is detected. In all our experiments, we selected $\rho = 0.975$, leading to $\epsilon(\rho) = 7.37$.

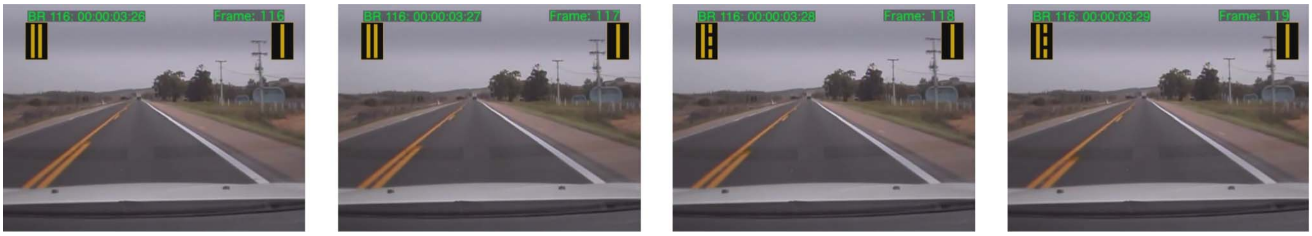
2) Temporal Information: Another important aspect of our problem is temporal continuity: road lane markings tend to belong to the same label for a set of adjacent frames, so that $\Omega_{ij}(t)$ tends to be a piece-wise constant function where each plateau corresponds to a set of adjacent frames with the same lane marking type. Although there are many alternatives (such as HMMs - Hidden Markov Models) to impose temporal coherence, in this work we adopt a simple but effective way to avoid "temporal outliers". Within a temporal window $\mathcal{T}(t) = \{t-T+1, t-T+2, \ldots, t-1, t\}$ containing the last T frames, we compute and store the type of lane marking $\hat{c}(t)$ at each frame using Eq. (11). Our temporally coherent class label $c'(t)$ at the current frame t is given by be most frequent label encountered within the temporal window, i.e.,

$$c'(t) = \underset{\tau \in \mathcal{T}(t)}{\text{mode}}\ \hat{c}(\tau) \tag{12}$$

which corresponds to a winner-takes-all classification scheme within $\mathcal{T}(t)$.

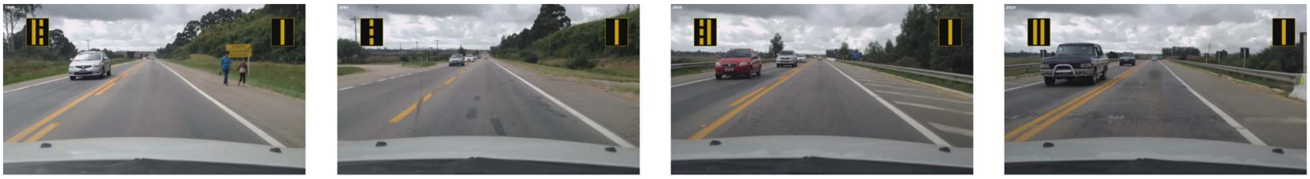
The choice of the window duration T is a compromise between smoothness and detection lag: larger values for T reduce the occurrence of isolated mis-detections, but at the cost of introducing detection lags when the lane type actually changes. We have tested different values for T and have verified that $T = 20$, which corresponds to 667 ms for videos acquired at 30 FPS, showed to be sufficient to control isolated responses (outliers) while not introducing significant lags.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



(a) (b) (c) (d)

Fig. 12. Frames of sequence clip_watec_DMA311013_HMS145905 without lags during transitions in the lane markings type.



(a) (b) (c) (d)

Fig. 13. Frames of sequence clip_iPad_DMA190414_1.

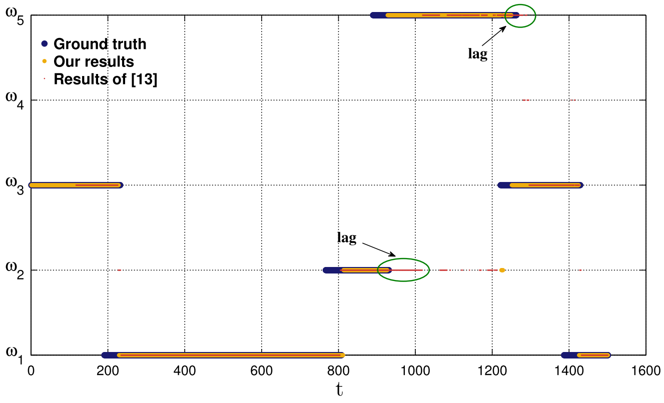


Fig. 14. Ground truth (blue), our method (orange) and results of [10] (red).

TABLE V
AVERAGE EXECUTION TIME PER FRAME

Clip resolution	Avg. Time (ms)
288 × 384	2.69
480 × 640	3.71
720 × 1280	6.53
1080 × 1920	16.59

IV. EXPERIMENTAL RESULTS

We tested our algorithm for different video sequences containing a variety of lane markings. More precisely, we have used 12 video sequences acquired with different cameras (Watec Wat 240R, iPad Mini, iPhone 3G, iPhone4, iPhone5S), resolutions and frame rates, at a variety of Brazilian roads. The only analog camera used is Watec Wat 240R, equipped with a fixed 3.8 mm lens, and all Watec videos were digitized using an AverTV iPro HD (model H869M) capture device at 29.97 FPS and resolution of 480 × 640. The data set used in this work with the corresponding ground truth data, as well as video sequences showing the results of our method, are publicly available at http://inf.ufrgs.br/~mbpaula/roadLaneMarkings/. Ground truth data contain one label for each lane boundary (left and right) at each frame. At frames when more than one lane marking is visible (i.e., during transitions), two labels are used for the corresponding lane boundary, and both of them are considered correct in the classification stage.

For all sequences, intrinsic parameters were obtained offline (once for each camera) using the Caltech Camera Calibration Toolbox for Matlab [18], and extrinsic parameters were obtained on-the-fly for each video sequence using [16], based on portions of the sequence in which: (i) the vehicle was moving along a flat and straight portion of the road, (ii) the lane markings were dashed, (iii) the speed of the vehicle in these excerpts was constant and known. It is noteworthy that with few frames, at 30 frames per second for example, we calibrated the camera in just 3 $\sim$ 5 seconds. The estimated intrinsic and extrinsic camera parameters as well as the speed of the vehicle and the number of frames used to obtain the corresponding values are presented in Table I. The extrinsic parameters for video sequences 5 and 10–12 (for which the speed is unknown) were computed using clip_iPhone3G_1395431301_70kmh and clip_i5s_0085, respectively, which present the same camera setups and known speed.

For the classification step itself, the values of all used parameters were as described in Section III, and they are summarized in Table II. Based on these parameters and, according to the Brazilian Traffic Signs Manual [19], the number of patches was defined taking into account: (i) length between two adjacent lane markings ($l_{\text{adj}} = 6$ m); (ii) length of a dashed marking ($l_{mk} = 2$ m). Thus, the number of patches used to analyze the pavement markings is expressed by $p_{\max} = \lfloor 1 + (1.5 l_{mk} + l_{\text{adj}})\delta_z^{-1} \rceil = 19$, where $\lfloor \cdot \rceil$ denotes the rounding operator to the closest integer.

To find the MoG parameters required in the first classifier $\mathcal{C}_1$, we used a total of 662 labeled samples extracted from clips 1 and 2, comprising 60 samples of class ω_1, 270 of ω_2 and ω_3, 101 of ω_4 and 231 of ω_5. The remaining 529 samples from these two clips were used to test the model, yielding a global accuracy of 100%.

In order to evaluate the robustness and generalization capabilities of the model, we also used another set of video sequences (clips 3–12) acquired with different cameras at varying resolutions, as shown in Table III. For this set, the overall classification rate was 96.36%, with the lowest per-clip accuracy of 82.39%. The confusion matrix shown in Table IV indicates that the most common problems are the mis-classification of samples from classes ω_3 and ω_4 as ω_1, and ω_4 as ω_3, which is related to classifier $\mathcal{C}_1$. In fact, most of the errors were due to problems in the lane tracker, as illustrated in Fig. 10. In Fig. 10(a), the lane tracker presented some deviation from the actual right lane boundary in the far field, and in Fig. 10(b), the left lane boundary was not correctly tracked. As a consequence, the corresponding scan line segments r_p^e were misplaced, leading to misdection of peaks and finally to misclassification. In general, situations in which the lane tracker faces limitations, such as rainy days (since windshield wipers affect the tracker) also tend to corrupt the results of our classification scheme.

Fig. 11(a) shows other examples in which the proposed approach fails. In Fig. 11(a), there is vehicle close to the camera interfering with the scan lines, which compromises the peak extraction process. In Fig. 11(b), the lack of painting on the right lane boundary also interferes with the peak extraction process: since part of the painting is solid, the overall peak distribution is similar to a dashed line style.

Despite a few classification errors, it is important to point out that the proposed approach performed well for low frame rate and low resolution sequences, such as clips clip_iPhone3G_1395434215_01 and clip_iPhone3G_1395431301_70kmh, and also for high resolution clips, such as clip_iPad_DMA190414_1. Also, some of these sequences present several transitions of lane marking types, which could possibly cause detection lags. However, the proposed method performed well even in such transitions, as illustrated in Fig. 12 for sequence clip_watec_DMA311013_HMS145905.

As shown in Table III, our method also performed very well for sequence clip_iPad_DMA190414_1, despite the presence cracks and corrugations on the pavement at some frames, as shown in Fig. 13(d). Also, one of our clips (clip_iPhone4_1807_1560) was acquired at night, and our accuracy was over 99%.

Regarding comparison with other methods, it is important to note that only [10] classifies lane boundaries into the five types described in this work. As an illustration, Fig. 14 shows the detection results of our approach and [10] for the central lane shown in clip_iPad_DMA190414_1, which presents several transitions of lane marking types. As it can be observed, the proposed approach does not present transition lags, opposed to [10], which also presents some isolated responses. We have also implemented the approach described in [13], which classifies lane markings into solid or dashed, and evaluated its performance in a subset of our data set containing only solid or dashed results (410 frames in total). The accuracy achieved by [13] was 96.43%, and ours (restricted to the same subset) was 100%.

The proposed system was implemented in C++ (GCC 4.8.2 compiler), using the **Open** source **C**omputer **V**ision library (shortly OpenCV) Version 2.4.9|4[2] and the **Open** **M**ulti-**P**rocessing (OpenMP)[3] API.

All experiments were conducted on a desktop with 3.40 GHz Intel Core i7-2600 CPU with 8GB RAM, and Table V presents the average execution time for each frame, considering the whole pipeline (from acquisition to classification). From this total, 24.70% is devoted to frame grabbing/video decoding, 22.89% to lane detection/following, 5.60% in overhead and 46.81% to the lane marking classification itself, so that implementation in embedded devices (e.g., smartphones) are viable.

V. Conclusion

This paper presented a novel approach for lane marking classification. A simple yet efficient lane detector/tracker [15] is used to estimate lane boundaries, and an on-line calibration scheme is used to estimate extrinsic camera parameters [16]. Then, a robust peak extraction algorithm within a search scan line is used to obtain information on the local geometry of the lane markings, which are fed to a cascaded classifier.

Experimental results showed that our method presents global accuracies over 96% considering all tested video sequences, running at over 60 FPS for full HD videos. More importantly, it showed consistently good results for video sequences acquired with different devices, at varying resolutions and frame rate, using the same training set in all examples. The few classification errors were due mainly to deviations of the lane tracker from the actual lane boundaries, so that some lane markings are not considered.

As future work, we intend to focus on better exploring the expected temporal coherence of lane markings. Among the possibilities, Hidden Markov Models (HMMs) can be a good option, since they can be trained to present a good compromise between rapid detection of transitions and removal of isolated misdetections.

Acknowledgment

C. Jung would like to thank Brazilian agency CNPq for partially supporting this work.

References

[1] "Global status report on road safety 2013: Supporting a decade of action," World Health Org., Geneva, Switzerland, 2013. [Online]. Available: http://www.who.int/

[2] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vis. Appl.*, vol. 25, no. 3, pp. 727–745, Apr. 2014.

[3] S. Yenikaya, G. Yenikaya, and E. Düven, "Keeping the vehicle on the road: A survey on on-road lane detection systems," *ACM Comput. Surveys (CSUR)*, vol. 46, no. 1, p. 2, Oct. 2013.

[4] P. Lindner, S. Blokzyl, G. Wanielik, and U. Scheunert, "Applying multi level processing for robust geometric lane feature extraction," in *IEEE Conf. MFI Intell. Syst.*, Sep. 2010, pp. 248–254.

[5] J. McCall, M. Trivedi, D. Wipf, and B. Rao, "Lane change intent analysis using robust operators and sparse Bayesian learning," in *Proc. IEEE Comput. Soc. Conf. CVPR Workshops*, Jun. 2005, pp. 59–59.

[6] S. Gehrig, A. Gern, S. Heinrich, and B. Woltermann, "Lane recognition on poorly structured roads—The bots dot problem in California," in *Proc. IEEE 5th Int. Conf. Intell. Transp. Syst.*, 2002, pp. 67–71.

[7] R. Schubert, K. Schulze, and G. Wanielik, "Situation assessment for automatic lane-change maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 607–616, Sep. 2010.

[8] J. Collado, C. Hilario, A. de la Escalera, and J. Armingol, "Detection and classification of road lanes with a frequency analysis," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2005, pp. 78–83.

[2] http://opencv.org

[3] http://openmp.org/

[9] J. M. Collado, C. Hilario, A. de la Escalera, and J. M. Armingol, "Adaptative road lanes detection and classification," in *Proc. 8th Int. Conf. ACIVS*, Berlin, Germany, 2006, pp. 1151–1162. [Online]. Available: http://dx.doi.org/10.1007/11864349_105

[10] M. Braga de Paula and C. Rosito Jung, "Real-time detection and classification of road lane markings," in *Proc. 26th SIBGRAPI*, Aug. 2013, pp. 83–90.

[11] R. Satzoda and M. Trivedi, "Vision-based lane analysis: Exploration of issues and approaches for embedded realization," in *Proc. IEEE Conf. CVPRW*, Jun. 2013, pp. 604–609.

[12] S. Nedevschi, V. Popescu, R. Danescu, T. Marita, and F. Oniga, "Accurate ego-vehicle global localization at intersections through alignment of visual data with digital map," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 673–687, Jun. 2013.

[13] S. Suchitra, R. Satzoda, and T. Srikanthan, "Identifying lane types: A modular approach," in *Proc. 16th Int. IEEE ITSC*, Oct. 2013, pp. 1929–1934.

[14] J.-Y. Bouguet, "Camera calibration toolbox for MATLAB," 2004. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/ref.html

[15] C. R. Jung and C. R. Kelber, "Lane following and lane departure using a linear-parabolic model," *Image Vis. Comp.*, vol. 23, no. 13, pp. 1192–1202, Nov. 2005.

[16] M. B. De Paula, C. R. Jung, and L. G. Da Silveira, Jr., "Automatic on-the-fly extrinsic camera calibration of onboard vehicular cameras," *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1997–2007, Mar. 2014. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2013.08.096

[17] A. Witkin, "Scale space filtering," in *Proc. 8th Int. Joint Conf. Artif. Intell.*, 1983, pp. 1019–1021.

[18] J. Y. Bouguet, "Camera calibration toolbox for MATLAB," 2008. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/

[19] C. N. de Trânsito, *Manual Brasileiro de Sinalização de Trânsito*, 1st ed. Brasilia, Brazil: DENATRAN, 2007, vol. 4.

Maurício Braga de Paula received the B.Sc. degree in computer science from Federal University of Pelotas, Pelotas, Brazil, in 2000 and the M.Sc. degree in computer science from Federal University of Santa Catarina, Florianópolis, Brazil, in 2002. He is currently working toward the Ph.D. degree at the Federal University of Rio Grande do Sul, Porto Alegre, Brazil. He is also currently an Assistant Professor with the Department of Mathematics and Statistics, Federal University of Pelotas. His main areas of interest are computer vision, image processing, and pattern recognition.

Cláudio Rosito Jung (SM'11) received the B.S. and M.S. degrees in applied mathematics and the Ph.D. degree in computer science from Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre, Brazil, in 1993, 1995, and 2002, respectively. He is currently a Faculty Member with the Computer Science Department, UFRGS. His research interests include several topics of image processing, computer vision, and pattern recognition, such as medical imaging, multiscale image analysis, intelligent vehicles, object tracking, multimedia applications, human motion analysis, multimodal signal processing, and stereo/multiview matching.