

Road Detection Based on Illuminant Invariance

José M. Álvarez and Antonio M. López

Abstract—By using an onboard camera, it is possible to detect the free road surface ahead of the ego-vehicle. Road detection is of high relevance for autonomous driving, road departure warning, and supporting driver-assistance systems such as vehicle and pedestrian detection. The key for vision-based road detection is the ability to classify image pixels as belonging or not to the road surface. Identifying road pixels is a major challenge due to the intraclass variability caused by lighting conditions. A particularly difficult scenario appears when the road surface has both shadowed and nonshadowed areas. Accordingly, we propose a novel approach to vision-based road detection that is robust to shadows. The novelty of our approach relies on using a shadow-invariant feature space combined with a model-based classifier. The model is built online to improve the adaptability of the algorithm to the current lighting and the presence of other vehicles in the scene. The proposed algorithm works in still images and does not depend on either road shape or temporal restrictions. Quantitative and qualitative experiments on real-world road sequences with heavy traffic and shadows show that the method is robust to shadows and lighting variations. Moreover, the proposed method provides the highest performance when compared with hue–saturation–intensity (HSI)-based algorithms.

Index Terms—Driving-assistance system, illuminant invariance, road detection, shadows.

I. INTRODUCTION

ADVANCED driver-assistance systems aim to understand the environment of the vehicle contributing to traffic safety. In this context, not only active sensors (e.g., radar and laser scanner) but also passive sensors (e.g., different types of cameras) play a relevant role. Vision-based approaches are used to address functionalities such as lane marking detection, traffic sign recognition, pedestrian detection, etc.

In this paper, we focus on vision-based road detection. That is, detecting the free road surface ahead of the ego-vehicle using an onboard camera (see Fig. 1). Road detection is an important task within the context of autonomous driving. Moreover, it is an invaluable background segmentation stage for other functionalities such as vehicle [1] and pedestrian [2] detection. The knowledge of the free road surface reduces the image region to search for these targets, thus contributing to reach real time and to reduce false detection.

Manuscript received July 29, 2009; revised March 3, 2010, July 30, 2010 and August 26, 2010; accepted September 5, 2010. Date of publication October 14, 2010; date of current version March 3, 2011. This work was supported in part by the Spanish Ministry of Education and Science under Project TRA2007-62526/AUT and in part by the research program Consolider Ingenio 2010: MIPRCV (CSD2007-00018). The Associate Editor for this paper was M. Da Lio.

The authors are with the Computer Vision Center and the Department of Computer Science, Universitat Autònoma de Barcelona, 08193 Cerdanyola del Valles, Spain (e-mail: jalvarez@cvc.uab.es).

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.2010.2076349



Fig. 1. Aim of vision-based road detection is identifying road pixels in an image.

Road detection is a challenging task since roads are in outdoor scenarios imaged from a mobile platform. Hence, on the one hand, the algorithms must deal with a continuously changing background and the presence of different objects (vehicles, pedestrians, and infrastructure elements) with unknown movements. On the other hand, the algorithms must deal with a high intraclass variability since there are different road types (shape, size, and wear down) and different imaging conditions (e.g., varying illumination).

The review of related literature reveals that using a monocular color camera as a sensor [3]–[8] is the preferred option. Therefore, texture and color are potential features to characterize the road. However, the imaged road texture varies too much with the distance to the camera; thus, color analysis is the option that is chosen most. Some authors propose the use of depth as an additional cue for adding robustness, as in [9], in which depth comes from active sensors, or in [10], in which depth comes from a stereo rig.

Current road-detection algorithms use different color spaces as features, as well as different classification processes. For instance, the hue–saturation–intensity (HSI) color space is used in [3] and [7]. HSI is a color space that mitigates the influence of lighting variations [11], [12]. However, HSI does not appropriately behave when the road presents shadowed areas. Thus, additional constraints must be included to improve their performance, e.g., in [13], Sotelo *et al.* use road shape restrictions and ad hoc postprocessing to recover undetected shadowed areas. Other authors use standard red–green–blue (RGB) and put more emphasis on the classification process. For instance, a mixture of Gaussians (MoG) is used in [5] and [9] for road modeling. However, selecting the proper number of Gaussians is not straightforward, and the presence of shadows is still a problem.

In [6], Tan *et al.* use histograms in the *rg* color space to build a model for the road and another for the background. The road variability is modeled using different histograms, which are dynamically updated from frame to frame. This is not a straightforward process since empirical criteria to incorporate new histograms, remove old ones, fuse similar ones, and fix the number of histograms are required. The background is modeled



Fig. 2. Great challenges for monocular vision-based road detection are the treatment of shadows and the presence of other vehicles.

using a single histogram. This way, background pixels from one frame are used to build the background model in the next frame, which is a kind of temporal constraint. Therefore, errors are propagated, particularly when other vehicles are present in the scene.

In [4], several color spaces, i.e., UV , rg , and luminance, are combined. For each color space, two MoGs are used: one to model the road and the other to model the background. Then, the results from each color space are combined using a voting approach. However, selecting the proper number of Gaussians is an issue, as well as background modeling. In addition, the method relies on road shape restrictions.

All these works use road shape constraints, temporal assumptions, or complex models to mitigate the problem of lighting variations and shadows. However, these assumptions and models drop off the performance of the algorithms in many common driving situations. For instance, road shape restrictions are difficult to apply with complex road shapes (crossings or roundabouts) or when the road borders are not clearly visible (due to traffic or in urban scenarios). Temporal restrictions do not hold when abrupt driving changes occur as accelerations/decelerations of the ego-vehicle or appearing/disappearing of other vehicles.

In this paper, we aim to perform road detection using a monocular color camera in the presence of shadows and other vehicles (see Fig. 2). Moreover, we aim to avoid the use of road shape or temporal restrictions as part of the algorithm. This way, as a novelty, we propose to use the illuminant-invariant image introduced by Finlayson *et al.* [14] as the feature space. This invariant image is derived from the physics behind color formation in the presence of a Planckian light source, Lambertian surfaces, and narrowband imaging sensors. Our argument is that sunlight is approximately Planckian, road surfaces are mainly Lambertian, and regular color cameras are near narrowband. Then, the invariant image can be appropriate for road detection.

The computation of the illuminant invariant from a color image requires a parameter of the camera, namely, *the illuminant-invariant angle*. A few proposals for computing this invariant direction exist in the literature. In this paper, we also present an alternative that outperforms them.

The illuminant-invariant feature space is combined with a likelihood-based classifier to identify each image pixel as *road* or *nonroad* (background). A road model is automatically built for each still image, and the background model is not required.

The remainder of this paper is organized as follows: In Section II, the illuminant-invariant space is introduced, as well as our proposal to obtain the illuminant-invariant angle. In Section III, the likelihood-based region growing proposal for

pixel classification is detailed. Then, Section IV describes the overall road-detection algorithm. Qualitative and quantitative results are provided in Section V. Comparison of the illuminant-invariant space against the popular HSI is also included. The images used during such an evaluation process present shadows and other vehicles. Finally, in Section VI, conclusions are drawn.

II. ILLUMINANT-INVARIANT SPACE

A. Theory in a Nutshell

Image formation models are defined in terms of the interaction between the spectral power distribution of the illumination, the surface reflectance, and the spectral sensitivity of the imaging sensors. In [14], Finlayson *et al.* show that having *Planckian illumination* and *Lambertian surfaces* imaged by three *narrowband sensors* (*PLN-assumptions* hereafter), it is possible to convert a given RGB color image, i.e., $I_{RGB} = (I_R, I_G, I_B)$, into a shadow-free color image, i.e., $I_{RGB}^{sf} = (I_R^{sf}, I_G^{sf}, I_B^{sf})$. This shadow-free image has the same spatial dimensions than the original one and the *same* colors, except at the shadowed areas, where I_{RGB}^{sf} presents the underlying color of the imaged surfaces, i.e., the color if there were no shadows during image acquisition (we refer to [14] for visual examples). In fact, empirical results prove that this theory holds even for real data, i.e., using a regular camera (roughly narrowband sensors) to image real-world scenes (roughly Lambertian surfaces) under only an approximately Planckian illumination [15].

The process to compute I_{RGB}^{sf} from I_{RGB} has two main stages: First, a grayscale image \mathcal{I} is obtained from I_{RGB} . In this image, the influence of lighting variations (i.e., shadows) is greatly attenuated. Second, a shadow-free image I_{RGB}^{sf} is computed by combining \mathcal{I} and I_{RGB} using a reintegration process. In practice, obtaining \mathcal{I} is a pixel-level procedure, whereas obtaining I_{RGB}^{sf} is an image-level procedure. Hence, the computation of \mathcal{I} can be done in real time, whereas obtaining I_{RGB}^{sf} turns out in a much more time-consuming process. Therefore, in this paper, \mathcal{I} is used as a feature space to perform reliable road surface detection despite shadows and lighting variations. Notice, that we do not rely on obtaining I_{RGB}^{sf} ; thus, real time is preserved.

Let R , G , and B be the standard color channels, and let $r = \log(R/G)$ and $b = \log(B/G)$ be the corresponding log-chromaticity values using the G channel as the normalizing channel. Then, under *PLN-assumptions*, a set of color surfaces of different chromaticities imaged under different illuminations in the log-chromaticity space forms parallel straight lines, which offset corresponds to chromatic differences. Furthermore, these lines define an orthogonal axis, i.e., ℓ_θ , where a surface under different illuminations is represented by the same point, and movements along ℓ_θ imply changing the surface chromaticity (see Fig. 3). In other words, ℓ_θ is a gray-level axis, where each gray level corresponds to a surface chromaticity, independently of the illumination. Thus, \mathcal{I} is a grayscale image, invariant to lighting variations. The reason for \mathcal{I} being illuminant invariant is, in short, that nonshadow surface areas are illuminated by both direct sunlight and skylight (a sort of scattered ambient light), whereas areas in the umbra are only

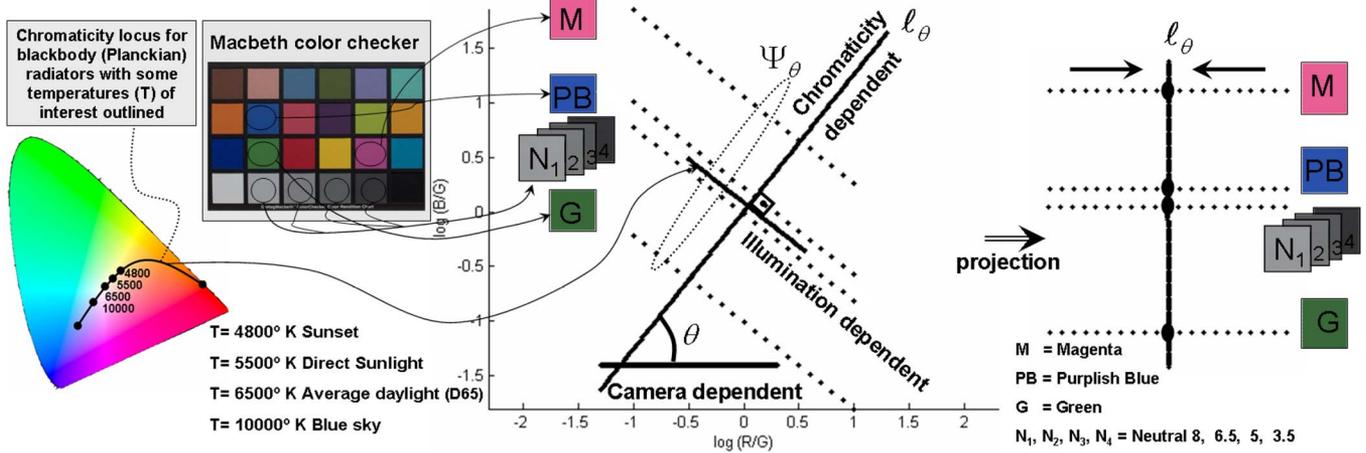


Fig. 3. Ideal log–log chromaticity plot. Each patch (Lambertian surface) under a Planckian illumination is represented using one point. The straight lines associated with each line are obtained by changing the color temperature of the illuminator. As a result, patches of different chromaticities have different lines associated with them. All these lines form a family of parallel lines, i.e. Ψ_θ . Then, ℓ_θ is the line perpendicular to Ψ_θ , and θ is its slope. The lines in Ψ_θ are projected onto ℓ_θ to obtain a one-to-one correspondence. The main property of ℓ_θ is preserving differences with regard to chromaticity but removing differences due to illumination changes.

illuminated by skylight. Since both skylight alone and with sunlight addition can be considered Planckian illuminations [16], the areas of the same chromaticity are ideally projected onto the same point in ℓ_θ , regardless of whether these areas are in shadow or not.

B. Camera Calibration

According to the theory, obtaining the main direction of ℓ_θ (camera calibration hereafter) is the key to obtaining \mathcal{I} . In fact, the angle θ is a function of the camera sensor, which does not correlate with the lighting condition or the surface material. Hence, θ can be understood as an intrinsic parameter of each color camera. Accordingly, the calibration process for a given camera needs to be done once and can be computed offline.

Two different calibration approaches can be found in the literature [17], [18]. In the first calibration approach, Finlayson *et al.* [17] use different images of a Macbeth color checker imaged under different daytime illuminations to obtain θ . The algorithm consists in analyzing the log-chromaticity plot generated from the color patches in the Macbeth color checker (see Fig. 3). In this plot, the color difference between patches is just a line offset. However, in practice, elongated clouds of points of different main-axis orientations are obtained due to deviations from the *PLN-assumptions* (see Fig. 4). Therefore, to obtain θ , each cloud of points can be shifted to a common center by subtraction of their respective mean. Then, the invariant direction is obtained by applying a robust estimator to the centrally aligned cloud of points.

In the second calibration approach, Finlayson *et al.* [18] analyze the histograms derived from the invariant images to recover θ . Let \mathcal{I}_α be the grayscale image obtained from projecting the rg pixel values of a given color image onto a line ℓ_α of slope α . Let \mathcal{E}_α be the entropy of \mathcal{I}_α , which is computed using its histogram \mathcal{H}_α as $\mathcal{E}_\alpha = -\sum_i^L H_\alpha(i) \log(H_\alpha(i))$, where L is the number of bins of \mathcal{H}_α . Then, a distribution of entropy values is obtained by varying the values of α . The underlying idea of the calibration algorithm is analyzing the

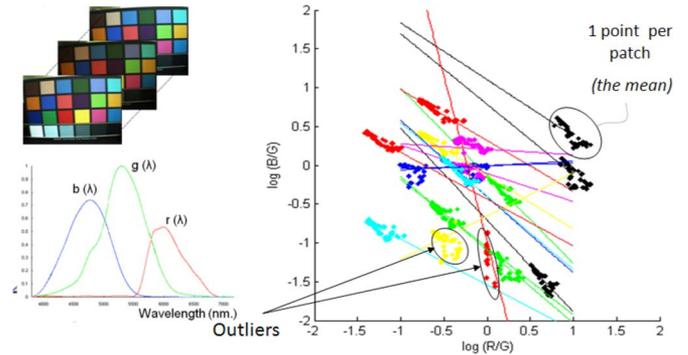


Fig. 4. Log-chromaticity plot using a Macbeth color checker imaged under different daytime illuminations using an imperfect narrowband camera. Accordingly, each color patch of the Macbeth checker generates an elongated cloud of points instead of a straight line. In addition, each elongated cloud has a different major axis and a different dispersion degree along the minor axis.

entropy distribution as follows: With a wrong α , i.e., with $\alpha \neq \theta$, similar chromaticity values of I_{RGB} are scattered along the corresponding ℓ_α , contributing to different bins of \mathcal{H}_α . This turns out in a high value of \mathcal{E}_α . With the right α , i.e., $\alpha = \theta$, similar chromaticity values of I_{RGB} contribute to the same bin of \mathcal{H}_α . Hence, a low value of \mathcal{E}_α is expected.

In our opinion, from a practical point of view, the Macbeth-based calibration is inferior to the entropy-based one. Notice that the former is quite tedious and cannot be adapted for the onboard self-calibration, whereas the latter automatically works. In addition, the entropy-based calibration only relies on image content. Hence, the onboard self-calibration is possible. However, our calibration experiments with road scenarios reveal that the estimation of θ using the entropy-based calibration is unstable. For instance, Fig. 5 shows the entropy distribution variability of images from the same sequence.

The failure analysis of the entropy-based calibration approach reveals that the key step of the algorithm is defining the bin width used for obtaining \mathcal{H}_α . This way, the authors fix the bin width by relying on Scott's rule [19], i.e., the bin width is set as $3.5N^{-(1/3)}\text{std}(\mathcal{I}_\alpha)$, where $\text{std}()$ stands for standard

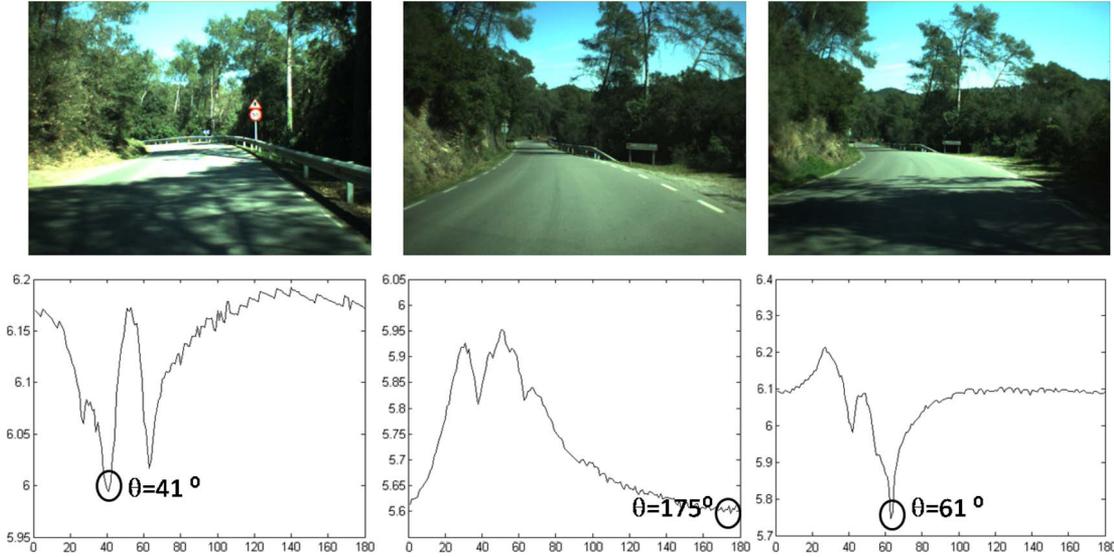


Fig. 5. (Upper row) Example images used to obtain the characteristic direction θ of our camera using the entropy-based approach in [18]. (Lower row) Variability of the corresponding entropy distributions \mathcal{E}_α . This variability influences the stability of θ .

deviation, and N is the number of pixels of \mathcal{I}_α . Moreover, the influence of outliers is reduced by considering only the middle 90% of the \mathcal{I}_α range when computing \mathcal{H}_α . Then, in this case, N is the number of pixels of \mathcal{I}_α inside the range. Notice that the bin width estimated using Scott's rule without removing the outliers would be too thick since the outliers increase $\text{std}(\mathcal{I}_\alpha)$.

Accordingly, in this paper, a new entropy-based calibration procedure is proposed to provide stable and precise θ values. The algorithm is similar to [18] in the sense of relying on entropy minimization. However, there are two main differences that improve the reliability of θ values.

- 1) **Difference 1 (D1).** The 90% of the *middle range* of \mathcal{I}_α values used to estimate \mathcal{H}_α is not fixed. Instead, Chebyshev's theorem [20] is applied to obtain more meaningful lower and upper bounds. Thus, each \mathcal{I}_α has its own adapted range.
- 2) **Difference 2 (D2).** The proposed algorithm does not rely on the entropy distribution of a single image. Instead, the analysis is performed on the entropy distributions of a set of input images. Then, the minimization is performed over the *average* entropy distribution $\hat{\mathcal{E}}_\alpha$. This *average* distribution is obtained using a robust mean approach to minimize the influence of extreme scores. This way, for each α , only the 90% middle range is used.

The summary of the algorithm is detailed here.

- 1) Select a subset of K color images from the database, i.e., $\{I_{RGB}^{(1)}, \dots, I_{RGB}^{(K)}\}$.
- 2) Select an image $I_{RGB}^{(k)}$ from the subset. Initialize the projection angle $\alpha = 0$.
- 3) Obtain a grayscale image \mathcal{I}_α projecting log-chromaticity pixel values of $I_{RGB}^{(k)}$ onto α .
- 4) Reject the outliers in \mathcal{I}_α according to Chebyshev's theorem.
- 5) Use pixels within the middle 90% range of nonoutlier pixels from \mathcal{I}_α to compute \mathcal{H}_α , i.e., the histogram of \mathcal{I}_α . The bin width of the histogram is fixed using Scott's rule.

- 6) Compute the entropy of \mathcal{I}_α using $\mathcal{E}_\alpha = -\sum_i^L \mathcal{H}_\alpha(i) \log(\mathcal{H}_\alpha(i))$, where L is the number of bins of \mathcal{H}_α .
- 7) Obtain the distribution of entropy values for $I_{RGB}^{(k)}$ by repeating steps 3–6, varying α within the $[0, 1, \dots, 179]$ range.
- 8) Repeat steps 2–7 for each image in the subset ($k \in [1, 2, \dots, K]$).
- 9) Obtain the average distribution of entropy using the robust mean method. That is, for each α , exclude the highest and the lowest 5% of the data.
- 10) Obtain θ as the absolute minimum of the average distribution of entropy values.

Fig. 6 shows our calibration results corresponding to the original images in Fig. 5. As shown, the proposed method leads to a reliable estimation of θ . Using adapted ranges for each image (D1) improves the similarity between entropy distributions of images acquired using the same camera. Furthermore, slight differences between the minimum for each distribution are compensated using multiple images (D2).

III. LIKELIHOOD-BASED CLASSIFICATION

In this section, the classifier used to identify road pixels is introduced. The classifier must exploit the illuminant-invariant properties of the feature space introduced in Section II. According to these properties, road pixels are expected to be similar to each other, even when shadows are present. Furthermore, there are other important requirements that must be considered: 1) being as simple as possible (*Occam's razor*) to enable real-time computation; 2) not using road shape or temporal constraints; and 3) enabling self-adaptation to process still images.

Therefore, we propose using a likelihood-based classifier. This classifier assigns labels to each pixel according to two different classes, namely, road surface and background. The former refers to pixels representing the free road surface. The latter refers to anything not being road (vehicles, buildings,

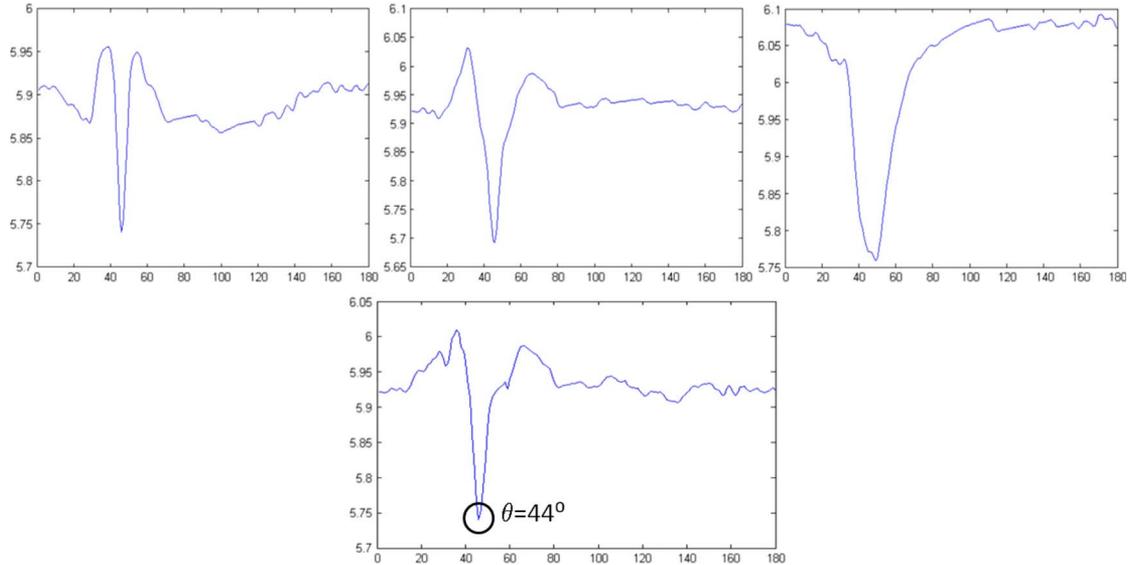


Fig. 6. For each image in the upper row of Fig. 5, the corresponding entropy distributions obtained with the entropy minimization we propose for a single image are shown. The lower row shows the averaged entropy distribution used to obtain the final estimator of θ .

infrastructures, trees, sky, and so on). Hence, this class is quite heterogeneous and, therefore, difficult to model. Consequently, modeling only road pixels is desirable.

The proposed classifier uses two entities to decide whether a pixel belongs or not to the road class. The former is a similarity measure between each pixel and a road model. The latter is a fixed threshold λ on this measure. In particular, we use a class-conditional function $p(\mathcal{I}(p)|road)$, providing the probability of pixel p being *road* according to its illuminant-invariant value $\mathcal{I}(p)$. Hence, a pixel is assigned to the road class according to the following classification rule:

$$\begin{cases} p \text{ is road,} & \text{if } p(\mathcal{I}(p)|road) \geq \lambda \\ p \text{ is background,} & \text{otherwise.} \end{cases} \quad (1)$$

It remains to estimate $p(\mathcal{I}(p)|road)$, i.e., the road model. One possibility consists in fitting a parametric model such as MoG, as it is done in [4] and [5]. However, nonparametric models are more appropriate in this case since they present two clear advantages: 1) They are fast in training and usage, and 2) they can fit any shape of the likelihood function (i.e., data distribution). Notice that, ideally, the histogram of values of \mathcal{I} at pixel imaging road surface $\mathcal{H}_{\mathcal{I}_{road}}$ is expected to be unimodal with low dispersion and skewness, even in the presence of shadows. Nevertheless, the road surface itself can present some color variations (e.g., worn out asphalt and nonuniformly wet road), which can introduce dispersion and skewness in $\mathcal{H}_{\mathcal{I}_{road}}$. Thus, parametric models using the first-order statistics do not always summarize $\mathcal{H}_{\mathcal{I}_{road}}$ properly. In fact, works on other types of applications suggest the superiority of histograms compared with parametric models [12], [21]. Therefore, we use the normalized histogram as an empirical form of the probability distribution for a random variable [22].

In particular, under the assumption that the bottom area of a road image shows the road surface, the road model is built using a set of N_s seeds in this area of each image. These seeds are placed in \mathcal{I} following an equidistant distribution along two



Fig. 7. (Left) Color image I_{RGB} . (Middle) Corresponding \mathcal{I} computed using the invariant angle θ . (Right) Nonparametric road model built using the normalized histogram formed with the surrounding region of several seeds (nine in this case) placed at the bottom part of \mathcal{I} .

rows R_s at the bottom part of each image. Then, the normalized histogram, i.e., $p(\mathcal{I}(p)|road)$, is computed using squared pixel areas surrounding each seed. In particular, we use regions of $K_s \times K_s$ pixels in \mathcal{I} . This process is exemplified in Fig. 7.

It is worth to mention that the assumption of considering the bottom part of the images showing the road surface is not too restrictive. This region usually corresponds to a distance of a few meters from the own vehicle; thus, the assumption is right if the vehicle keeps a gap with preceding ones for safety driving. For the sequences in this paper, this distance is of about 4 m away from the camera. Other works in the field [6], [13] follow similar assumptions.

IV. ROAD-DETECTION ALGORITHM

In this section, a road-detection algorithm is devised by combining the illuminant-invariant feature space and the likelihood-based classifier. The incoming color image is converted to the illuminant-invariant space, and the road model is estimated. Applying this model results in a road probability map that is binarized using λ . Then, a connected-component algorithm is applied to the binary image using the same set of seeds used to build the road model. The results present small holes usually placed from the middle to the bottom part of the images, i.e., where the road is imaged with more detail, thus actually representing just small road patches. Hence, a postprocessing

step using a fill-in hole procedure is included. This procedure is simply based on mathematical morphology operations: a closing with a structuring element e_{fi} followed by a flood-fill operation that does not require parameters.

Thus, our road detection algorithm is summarized in the list that follows.

- 1) Compute \mathcal{I} from I_{RGB} using θ .
- 2) Build the road model $p(\mathcal{I}(p)|road)$ using the normalized histogram of the surrounding area ($K_s \times K_s$ pixels) of a set of N_s seeds placed at the bottom part of the image. This set of seeds is placed following R_s .
- 3) Obtain I_r by thresholding \mathcal{I} according to $p(\mathcal{I}(p)|road)$ and a fixed threshold λ . Set $I_r(p) = 1$ if the value of the $\mathcal{I}(p)$ th bin of $p(\mathcal{I}(p)|road)$ is higher than λ ; otherwise, set $I_r(p) = 0$.
- 4) Obtain I_{cc} by performing a connected-component procedure on I_r . The same set of N_s seeds is used as starting points.
- 5) Fill in small holes in I_{cc} : a closing with e_{fi} followed by a flood fill.

V. EXPERIMENTAL RESULTS

A. Data Set

Experiments are conducted on image sequences acquired using an onboard camera based on the Sony ICX084 sensor. This is a charge-coupled device chip of 640×480 pixels and 8 bits per pixel that makes use of a Bayer pattern for collecting color information. Standard Bayer pattern decoding (bilinear interpolation) is used to obtain RGB color images. The camera is equipped with a microlens of 6-mm focal length. The frame acquisition rate is 15 fps. Furthermore, the camera shutter is set to automatic mode to avoid global over- and underexposure, i.e., it avoids both totally saturated images and very dark ones. However, isolated image regions that are either over- or underexposed still appear.

The data set is divided in three different sequences. The first sequence, i.e., \mathcal{S}_θ , consists of 350 images, and it is used to evaluate the camera calibration algorithm. The second sequence consists of 250 images acquired during a rainy day, just after the rain stopped so that the road was wet, although there were no reflecting puddles. The third sequence consists of 450 images acquired during a sunny day to favor the existence of shadows. The second and third sequences are combined in a single sequence, i.e., \mathcal{S}_{RD} , which is used to validate the road detection algorithm. Example images of this sequence are shown in Figs. 1 and 2. The former shows an image with wet asphalt and vehicles. The latter shows images taken during a sunny day, with shadows and the presence of other vehicles. The dataset is available online at <http://www.cvc.uab.es/adas/databases>.

The sequence \mathcal{S}_{RD} is a well-balanced testing set in terms of diversity of images since it contains a good tradeoff between shadows, vehicles, and clear roads. The summary of these statistics (percentages) is listed in Table I. In this table, the column *Shadows* represents images showing heavy disturbing shadows, whereas the row *Close vehicles* (either preceding or oncoming) refers to images showing vehicles with their wheels

TABLE I
SUMMARY OF TESTING SEQUENCE STATISTICS WITH REGARD TO THE PRESENCE OF SHADOWS AND OTHER VEHICLES IN THE SCENE. THESE PERCENTAGES REFLECT THE VARIABILITY OF IMAGES IN \mathcal{S}_{RD}

	Shadows	No shadows or dim ones	Total
Close vehicles	10%	40%	50%
No vehicles or further away ones	34%	16%	50%
Total	44%	56%	100%

lying below a fixed row. For instance, an image labeled *Close vehicles* and *No shadows* is shown on the left side of Fig. 2. That is, there is a car considered close, and at the same time, this image does not present heavy disturbing shadows. An image labeled *Further away vehicle* and *Shadows* is shown on the right side of Fig. 2.

B. Camera Calibration Evaluation

In this section, two different experiments are conducted on \mathcal{S}_θ to validate the camera calibration algorithm. The number of images used for calibration i.e., the parameter K , is fixed using exhaustive search. This way, the calibration is repeated for different values of K in the $[3, 4, \dots, 50]$ range. This way, given a value of K , the calibration is performed using the first K images of \mathcal{S}_θ , and then, a sliding-window approach is followed to calibrate all the images in the data set. Therefore, given a K , we obtain a number of θ angles; thus, we can compute the mean and standard deviation for each K value (see Fig. 8). As shown, the mean is stable in the range $[43^\circ, 44^\circ]$, whereas the standard deviation stays below 1° from $K = 15$ on. Hence, K is fixed to 15 for the rest of the experiments. Notice that these 15 images correspond to an acquisition window of 1 s. Thus, the onboard self-calibration is possible.

The first experiment consists in evaluating the repeatability of the proposed calibration method. This way, all the images in \mathcal{S}_θ are calibrated using a sliding-window method with $K = 15$. For comparison, these images are calibrated using the entropy-based approach in [18] (described in Section II-B). The calibration results are shown in Fig. 9. As shown, our proposal exhibits higher stability than the entropy-based method. The output values of our method range from 43° to 45° , whereas the output values of the other method range from 40° to 65° . From these results, we can conclude that our calibration algorithm is more repetitive than the baseline.

The second experiment consists in assessing the precision of the proposed calibration algorithm. Ground truth is generated considering the Macbeth-based technique proposed in [17] (see Section II-B). This way, the Macbeth color checker is imaged under different illumination conditions, and the 2-D log-chromaticity feature space is formed by computing the median of pixel values for each color patch (see Fig. 4). Then, for each patch, the set of log-chromaticity values is centrally aligned (i.e., cluster-mean subtraction). The characteristic direction of the camera, i.e., θ , corresponds to the principal orientation of the group of center-shifted log-chromaticity values [17]. As a result, the ground-truth angle is $\theta = 42.3^\circ$. Furthermore, the

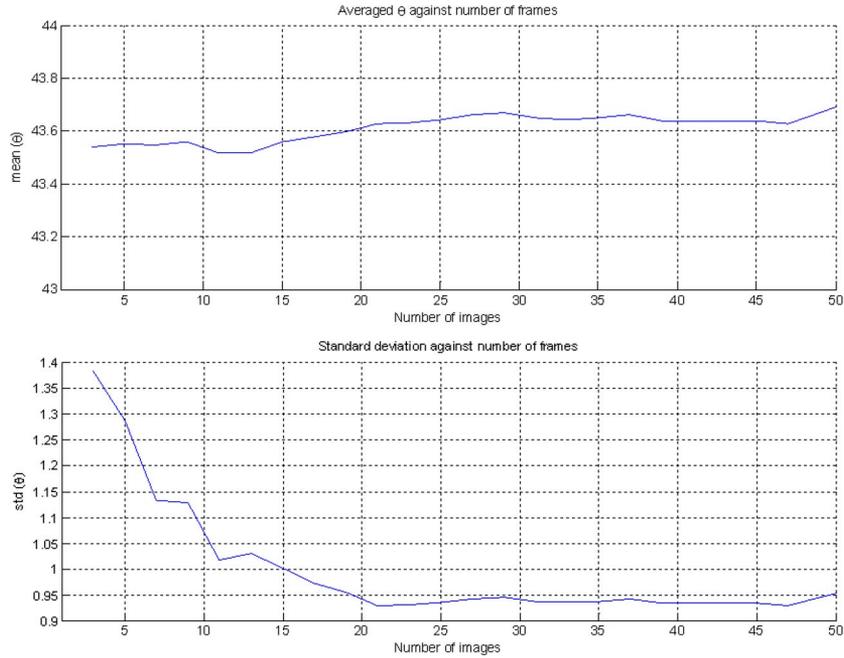


Fig. 8. (Top) Mean and (bottom) standard deviation (in degrees) of the θ angle output by our calibration method as a function of the number of images used for such calibration over the validation sequence.

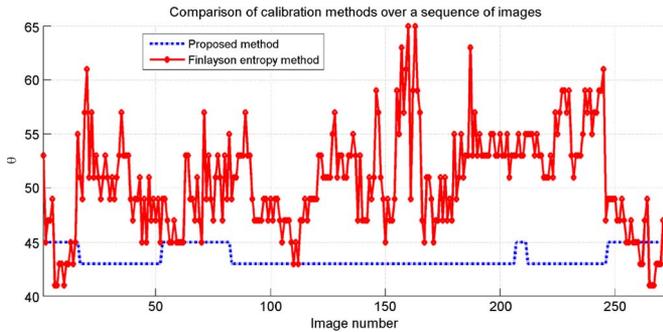


Fig. 9. Comparison of the θ obtained by setting $K = 15$ in our calibration algorithm and the θ obtained by following the approach in [18].

range of reliability of the estimation is analyzed using the 95% confidence interval [23].

Fig. 10 shows three lines corresponding to the ground-truth angle. The slope of the first line is 44° , corresponding to the averaged angle provided by our calibration method. The slope of the second line is 50.9° , corresponding to the averaged angle provided by the entropy-based approach [18]. The slope of the first line is 42.3° , corresponding to the ground truth. As shown, the output of our algorithm is within the confidence interval, whereas the result obtained using the entropy-based approach is not. Hence, we can conclude that our calibration method provides a precise estimation of θ . Furthermore, the algorithm is more accurate than the state of the art.

C. Road-Detection Evaluation

In this section, experiments are conducted on \mathcal{S}_{RD} to validate the proposed road-detection algorithm. Ground truth is generated by manually segmenting all the images in \mathcal{S}_{RD} . For

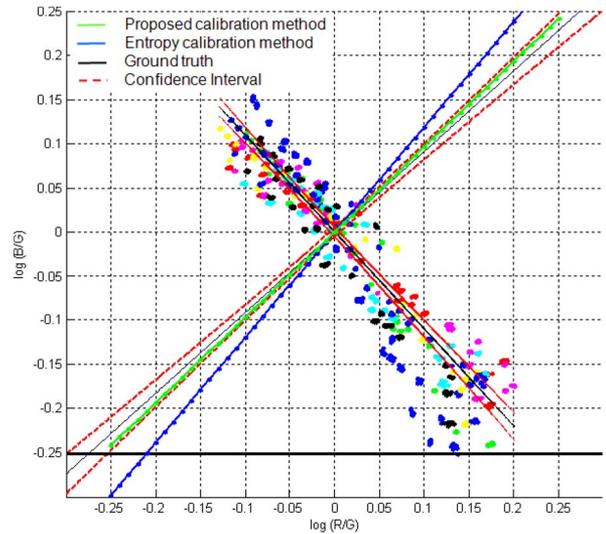


Fig. 10. Calibration results using the Macbeth color checker method. The ground truth is the main axes of the centrally aligned cloud of points from each patch in the Macbeth chart. The reliability of the estimation is shown through the 95% confidence interval. Moreover, the results provided by (green) our method and (blue) the entropy-based method [18] are also shown.

instance, Fig. 1 shows an image and its ground truth presented as a filled-in reddish polyline.

Quantitative evaluations are provided comparing the results against ground truth using three pixelwise measures: 1) precision P ; 2) recall R ; and 3) effectiveness F [24]. Precision and recall are defined as $P = (\sum \mathcal{G} \cdot I_r) / \sum I_r$ and $R = (\sum \mathcal{G} \cdot I_r) / \sum \mathcal{G}$, where \mathcal{G} and I_r are the ground-truth mask and the road detection result of a given image I_{RGB} , respectively. The symbol “ \cdot ” stands for pixelwise product, and \sum means summation over all pixels. Precision and recall provide different insights in the performance of the method: Low precision means that many background pixels are classified as road, whereas

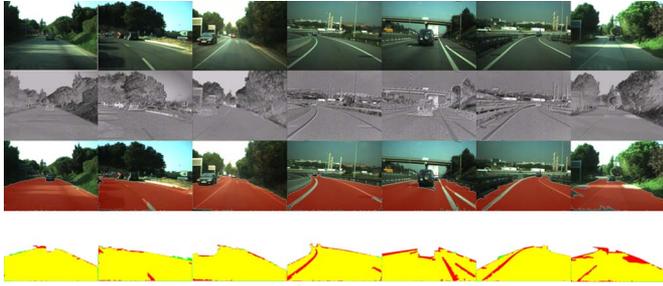


Fig. 11. Road-detection examples. (Top row) Original image. (Second row) Illuminant-invariant image. (Third row) Detected road. (Bottom row) Comparison against hand-segmented result. (Yellow) Correctly classified pixels. (Green) Falsely detected road pixels. (Red) False background pixels.

low recall indicates failure to detect the road surface. Finally, effectiveness is the tradeoff using weighted harmonic mean between precision and recall. Equally weighting precision and recall, effectiveness follows the equation $F = (2PR)/(P + R)$. All three measures range from 0 to 1, and higher values are desired.

The parameters of the algorithm are set as follows: The invariant direction θ is fixed using the results of the calibration experiment. That is, $\theta = 44^\circ$. An *equidistant distribution* along two rows in the bottom part of the image is used as the rule for placing the seeds, i.e., R_s . In particular, rows 417 and 428 are selected. Furthermore, $N_s = 9$, $b = 1$ (the minimum possible value), and $K_s = 11$. These settings (K_s , N_s , and R_s) imply that $\mathcal{p}(\mathcal{I}(p)|road)$ is built from nine small image patches equidistantly placed along the road width. Fig. 7 shows an example of the seed placement and the corresponding normalized histogram. The structuring element e_{fi} is empirically set to 5×3 pixels. Finally, a learning approach is used to fix the value of λ . This way, a set of training images is used to evaluate the effectiveness of the detection algorithm when λ varies in the range $[0.05, 0.08, \dots, 0.98]$. The optimal value of λ is the one that maximizes the average effectiveness.

Fig. 11 shows road detection results, the illuminant-invariant image, and the comparison against ground truth (hand segmentation). Other example detection results are shown in Fig. 12. More results in video format can be viewed at <http://www.cvc.uab.es/~jalvarez/RDinv.htm>. These results suggest that the algorithm can detect road pixels, even in the presence of shadows and other vehicles being in the scene. From these results, we can conclude that the proposed road detection algorithm performs well at detecting road pixels, even in the presence of shadows and other vehicles. Nevertheless, failure analysis reveals two main causes of performance drop-off (see Fig. 13). The first cause refers to the presence of undersaturated (penumbra) and oversaturated (highlights) areas in the image. In these areas, there is a lack of color information, leading to areas in the image where the invariant image is not valid. However, this issue can be addressed by improving the acquisition system (i.e., cameras with a higher dynamic range). The second cause refers to the presence of thick continuous lane markings, which stop the connected-component algorithm. This issue could be addressed by combining road detection with lane marking-detection algorithms [25].

For comparison, the image sequence \mathcal{S}_{RD} is processed using an HSI-based road-detection algorithm inspired in [7] and [13]. The HSI-based algorithm is summarized here.

- 1) Convert the input RGB image, i.e. I_{RGB} , onto the HSI color space, i.e., $I_{HSI} = (H, S, I)$ [26].
- 2) Compute the HSI model $\hat{\xi} = (\hat{h}, \hat{s}, \hat{i})$, averaging the values of pixel within surrounding regions of size $K_s \times K_s$ around a set of N_s seeds placed at R_s rows.
- 3) Compute the distances of each pixel to $\hat{\xi}$ using the dissimilarity measure d_{HSI} in the HSI space, which is defined as follows:

$$d_{HSI} = \sqrt{\hat{s}^2 + s^2 - \hat{s}s \cos(\varphi) + |i - \hat{i}|^2}$$

$$\varphi = \begin{cases} |h - \hat{h}|, & \text{if } |h - \hat{h}| < 180^\circ \\ 360^\circ - |h - \hat{h}|, & \text{otherwise} \end{cases}$$

where $h = H(p)$, $s = S(p)$, and $i = I(p)$ are the HSI values at pixel p .

- 4) Threshold the result using a fixed threshold γ .
- 5) Perform connected-component analysis by keeping only the pixels that passed the threshold and that belonged to the same connected component than the seeds.
- 6) Fill in small holes using e_{fi} .

For fair comparison, the parameters R_s , K_s , N_s , and e_{fi} of the HSI-based algorithm are the same as for the proposed algorithm. Finally, the threshold γ is fixed using the same learning approach than for fixing λ . However, the meaningful range of values for γ is empirically set to $[0.05, 0.005, \dots, 0.5]$ since the results outside this range are poor.

The performance of the algorithms is evaluated through a cross-validation approach. This way, subsets of 5% of the images in \mathcal{S}_{RD} are randomly selected for learning both thresholds λ and γ . The rest of the images are used for testing the algorithm. That is, in our data set, 35 are used for training and 665 for testing. The process is repeated 20 times using nonoverlapped training sets. The effectiveness of each image in the testing set is computed, as well as the mean \bar{F} and the standard deviation σ_F of the effectiveness per round. Accordingly, at the end of the process, we have 20 means and standard deviations of effectiveness. Then, the evaluation consists in comparing the performance of both algorithms in terms of average effectiveness, i.e., $\bar{\mathcal{F}} = (1/20) \sum_{i=1}^{20} \bar{F}_i$, and average standard deviation, i.e., $\sigma_{\bar{\mathcal{F}}} = (1/20) \sum_{i=1}^{20} \sigma_{F_i}$.

A summary of results is listed in Table II. As shown, the illuminant-invariant road-detection proposal outperforms the HSI-based algorithms for images containing strong shadows and other vehicles in the scene. Hence, from these results, we can conclude that the illuminant-invariant feature space introduced in this paper is more adequate than the HSI color space for road detection.

Finally, the computational cost of the algorithm is analyzed. Our current implementation is in nonoptimized MATLAB code. This code takes about 600 ms to perform road segmentation at full resolution (640×480 images) using a standard central processing unit at 2.4 GHz. This computations cost is decomposed as follows: 375 ms for obtaining the invariant image and the likelihood function; less than 1 ms doing the threshold using

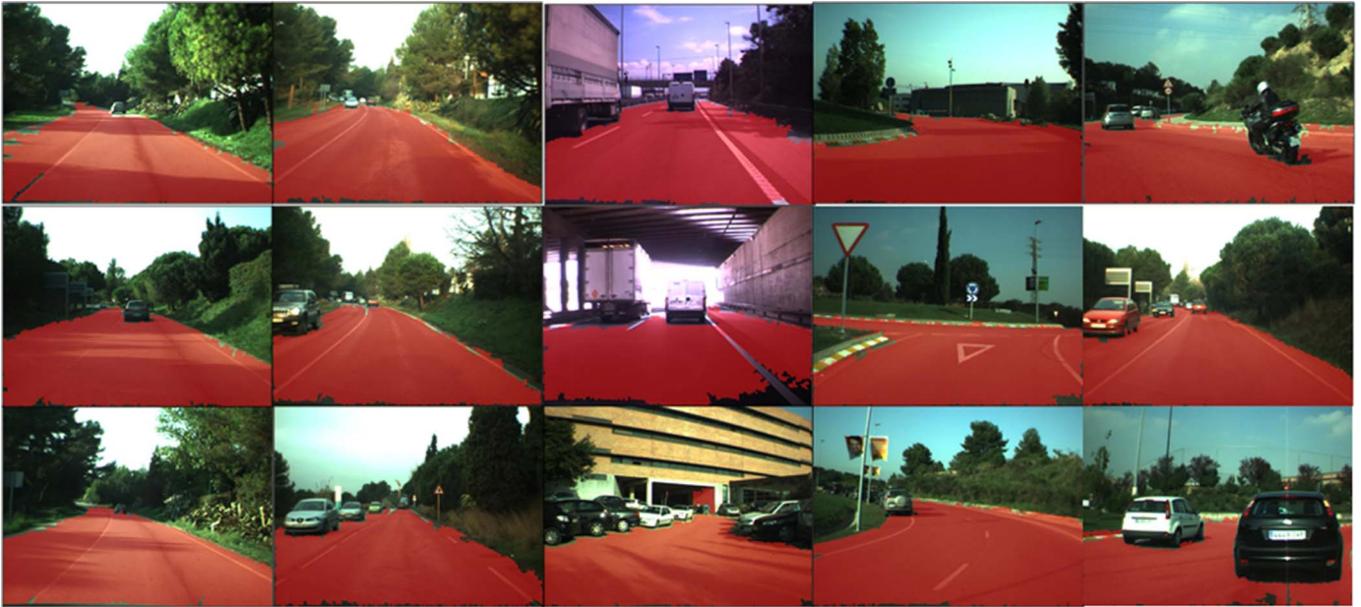


Fig. 12. Road-detection results. Example results include the following: (First column) Images from a sunny day with annoying shadows and traffic; (second column) images from a wet road without puddles with dim shadows and traffic; (third column) different scenarios; (fourth column) different road shapes; (last column) presence of other vehicles.



Fig. 13. Examples where our road-detection method currently fails.

TABLE II
PERFORMANCE OF ROAD DETECTION ALGORITHMS USING A
CROSS-VALIDATION METHOD

	\mathcal{I} based	HSI based
$\hat{\mathcal{F}}$	0.8945 ± 0.0028	0.6206 ± 0.0003
$\hat{\sigma}_{\mathcal{F}}$	0.1001 ± 0.0082	0.0343 ± 0.0002

the likelihood model; 175 ms running the connected components; and 50 ms for the fill-in of small holes. According to our experience in similar applications, we think that just by going to C++, this code can easily run in less than 50 ms. In addition, there are some easy optimizations, such as removing at least one third of the upper part of the image since it is not likely to contain the road surface. In fact, performing road detection on half-resolution images using our current implementation takes around 40 ms.

VI. CONCLUSION

In this paper, a novel approach to road detection has been presented. The algorithm combines an illuminant-invariant feature space with a road class-likelihood classifier in a frame-by-frame framework to provide reliable road detection results despite lighting variations and shadows. The novelty of the approach relies in exploiting the invariance to lighting variations of a feature space that has not been used before in the road-detection context. The main properties of the proposed method are given

as follows: It is able to run in real-time in a per-frame basis, it is not constrained to specific road shapes, the likelihood-based classifier is build online adapted to each frame, and a modeling of the background is not required.

Experiments are conducted to validate the method on a data set that includes images with heavy shadows and traffic. Furthermore, the algorithm is quantitatively evaluated and compared with an HSI-based road-detection algorithm. From the experimental results, it is concluded that the proposed method is suitable for handling shadows while still allowing one to distinguish road pixels from the background, including vehicles on the road. Moreover, the proposed algorithm outperforms state-of-the-art road-detection algorithms.

The illuminant-invariant space is camera dependent. Thus, an intrinsic parameter of the camera, i.e., the invariant direction, must be provided. Accordingly, we have proposed a method for calibrating the camera that improves the precision and repetitiveness of current algorithms. Furthermore, the method can work onboard, enabling the self-calibration of the camera.

As future work, we want to address the problem of local over- and undersaturations by improving the image acquisition system. Moreover, we want to incorporate road-detection results to improve onboard vehicle-detection algorithms [27].

REFERENCES

- [1] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using evolutionary Gabor filter optimization," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 125–137, Jun. 2005.
- [2] A. Broggi, P. Cerri, S. Ghidoni, P. Grisleri, and H. G. Jung, "A new approach to urban pedestrian detection for automatic braking," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 594–605, Dec. 2009.
- [3] M. Sotelo, F. Rodriguez, and L. Magdalena, "Virtuous: Vision-based road transportation for unmanned operation on urban-like scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 2, pp. 69–83, Jun. 2004.
- [4] O. Ramstrom and H. Christensen, "A method for following unmarked roads," in *Proc. IEEE Intell. Vehicles Symp.*, 2005, pp. 650–655.

- [5] J. Lee and C. Crane, "Road following in an unstructured desert environment based on the EM (expectation-maximization) algorithm," in *Proc. SICE-ICASE Int. Joint Conf.*, 2006, pp. 2969–2974.
- [6] C. Tan, T. Hong, T. Chang, and M. Shneier, "Color model-based real-time learning for road following," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2006, pp. 939–944.
- [7] C. Rotaru, T. Graf, and J. Zhang, "Color image segmentation in HSI space for automotive applications," *J. Real-Time Image Process.*, vol. 3, no. 4, pp. 311–322, Dec. 2008.
- [8] Y. He, H. Wang, and B. Zhang, "Color-based road detection in urban traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 309–318, Dec. 2004.
- [9] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. R. Bradski, "Self-supervised monocular road detection in desert terrain," in *Proc. Robot. Sci. Syst. Conf. (RSS)*, 2006.
- [10] R. Danescu and S. Nedevschi, "Probabilistic lane tracking in difficult road scenarios using stereovision," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 2, pp. 272–282, Jun. 2009.
- [11] N. Ikonomakis, K. Plataniotis, and A. Venetsanopoulos, "Color image segmentation for multimedia applications," *J. Intell. Robot. Syst.*, vol. 28, no. 1/2, pp. 5–20, Jun. 2000.
- [12] L. Sigal, S. Sclaroff, and V. Athitsos, "Skin color-based video segmentation under time-varying illumination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 7, pp. 862–877, Jul. 2004.
- [13] M. Sotelo, F. Rodriguez, L. Magdalena, L. Bergasa, and L. Boquete, "A color vision-based lane tracking system for autonomous driving in unmarked roads," *Auton. Robots*, vol. 16, no. 1, pp. 95–116, Jan. 2004.
- [14] G. Finlayson, S. Hordley, C. Lu, and M. Drew, "On the removal of shadows from images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 59–68, Jan. 2006.
- [15] G. D. Finlayson and S. D. Hordley, "Color constancy at a pixel," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 18, no. 2, pp. 253–264, Feb. 2001.
- [16] G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2nd ed. Hoboken, NJ: Wiley, 1982.
- [17] G. Finlayson, S. Hordley, and M. Drew, "Removing shadows from images," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 129–132.
- [18] G. Finlayson, M. Drew, and C. Lu, "Intrinsic images by entropy minimization," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 582–595.
- [19] D. Scott, "On optimal and data-based histograms," *Biometrika*, vol. 66, no. 3, pp. 605–610, Dec. 1979.
- [20] B. Amidan, T. Ferryman, and S. Cooley, "Data outlier detection using the Chebyshev theorem," in *Proc. IEEE Aerosp. Conf.*, 2005, pp. 3814–3819.
- [21] M. Jones and J. Rehg, "Statistical color models with application to skin detection," *Int. J. Comput. Vis.*, vol. 46, no. 1, pp. 81–96, Jan. 2002.
- [22] X. Li, *Probability, Random Signals, and Statistics*. Boca Raton, FL: CRC, 1999.
- [23] R. Fisher, *Statistical Methods and Scientific Inference*. Edinburgh, U.K.: Oliver & Boyd, 1956, p. 32.
- [24] C. Manning, P. R. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008, ch. 16.3.
- [25] A. López, J. Serrat, C. Cañero, F. Lumbreras, and T. Graf, "Robust lane markings detection and road geometry computation," *Int. J. Automot. Technol.*, vol. 11, no. 3, pp. 395–407, 2010.
- [26] R. Gonzalez and R. Woods, *Digital Image Processing*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2002, sec. 10.4.
- [27] D. Ponsa and A. Lopez, "Cascade of classifiers for vehicle detection," in *Proc. Int. Conf. Adv. Concepts Intell. Vis. Syst.*, 2007, pp. 980–989.



Jose M. Álvarez received the B.Sc. degree in computer science from La Salle School of Engineering, Ramon Llull University, Barcelona, Spain, in 2005 and the M.Sc. degree from the Computer Vision Center, Universitat Autònoma de Barcelona, Cerdanyola del Valles, Spain, in 2007. He is currently working toward the Ph.D. degree with the Universitat Autònoma de Barcelona, working on the project "On-board Road Detection."

His research interests include color, photometric invariance, machine learning, and fusion of

classifiers.



Antonio M. López received the B.Sc. degree in computer science from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 1992 and the M.Sc. degree in image processing and artificial intelligence and the Ph.D. degree from the Universitat Autònoma de Barcelona (UAB), Cerdanyola del Valles, Spain, in 1994, and 2000, respectively.

Since 1992, he has been giving lectures with the Department of Computer Science, UAB, where he is currently an Associate Professor. In 1996, he participated in the foundation of the Computer Vision Center, UAB, where he has held different institutional responsibilities, including, at present, responsibility for the research group on advanced driver-assistance systems by computer vision. He has been responsible for public and private projects and is a coauthor of more than 50 papers, all in the field of computer vision.