WILEY | Hindawi

Research Article

A Hybrid Vision-Map Method for Urban Road Detection

Carlos Fernández, David Fernández-Llorca, and Miguel A. Sotelo

Computer Engineering Department, University of Alcalá, Madrid, Spain

Correspondence should be addressed to Carlos Fernández; carlos.fernandezl@uah.es

A hybrid vision-map system is presented to solve the road detection problem in urban scenarios. The standardized use of machine learning techniques in classification problems has been merged with digital navigation map information to increase system robustness. The objective of this paper is to create a new environment perception method to detect the road in urban environments, fusing stereo vision with digital maps by detecting road appearance and road limits such as lane markings or curbs. Deep learning approaches make the system hard-coupled to the training set. Even though our approach is based on machine learning techniques, the features are calculated from different sources (GPS, map, curbs, etc.), making our system less dependent on the training set.

1. Introduction

Autonomous vehicles require a precise and robust perception of the environment. This is a crucial point in the development of autonomous vehicles because the perception layer is the base of higher level systems, such as control algorithms or path planning. One of the main issues is road detection, which corresponds to the drivable surface of the road. It has traditionally been an exhaustive topic of research in the fields of Advanced Driver Assistance System (ADAS) and autonomous driving. On the one hand, ADAS have mainly focused on increasing the safety of drivers and road users by means of driver warnings and assisted interventions. On the other hand, it is undebatable that autonomous driving has become a high priority issue on the research and commercial agendas of major car makers over recent years, with these makers intending to produce fully autonomous vehicles by 2020. The deployment of autonomous cars will bring a number of clear benefits in terms of increased traffic efficiency and reduced accident toll, resulting in unquestionably higher energy efficiency and enhanced road safety.

The Grand Challenge organized by DARPA in 2005 was the first championship of autonomous vehicles. Participants relied on precise and expensive differential GPS and IMU sensors to follow a set of waypoints along a planned route.

Dynamic obstacle detection was solved thanks to a multi-beam LIDAR. This approach demonstrated its ability to drive safely in highways and urban environments [1, 2]. However a high definition map is required. Drawbacks to this type of maps are their size (~2 GB/km), the complex process required to integrate all measurements, and how to update the map. The price of the LIDAR (~75 K) and the GPS + IMU (~25 K) is not affordable for the car industry. Therefore, our proposal is based on a low cost GPS sensor and a pair of stereo cameras (~2 K). The goal is to create a method that uses relatively low cost sensors and that detects the road even in the most challenging scenarios. Our system aims to be part of a local navigation system with a global route planner. It attempts to imitate the human way of driving: a user wants to drive from A to B and a global route planner uses a low cost GPS sensor to locate the vehicle and plan the route. When the route is ready, local navigation is required to detect the drivable area and to make the corresponding decisions.

2. Related Work

How can we reach level 5 of autonomous driving? The three pillars of autonomous driving are sensing, mapping, and driving policy (path planning). Sensing interprets the scene
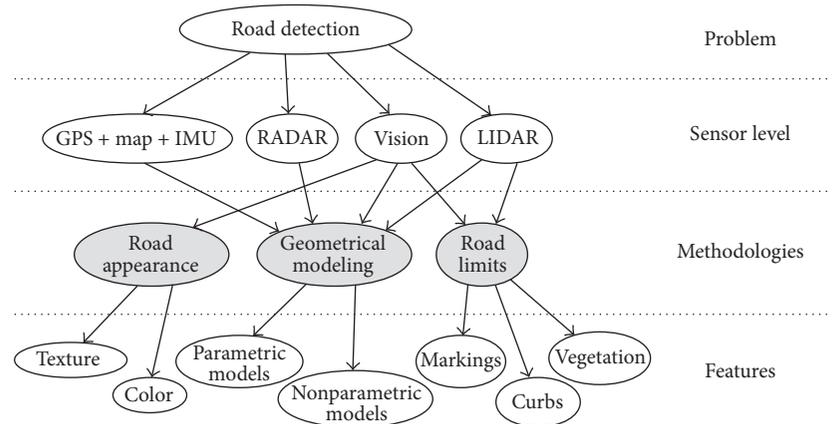
FIGURE 1: Classification of different road detection approaches depending on the sensor and the methodology.

with 360° awareness and produces an environmental model. The environmental model includes the location of the moving obstacles, road limits, curbs, and barriers. The sensing is based on cameras, RADAR, and LIDAR sensors. Cameras provide rich information of the scene with high frequency; however they are affected by illumination and weather conditions. RADARs are more resistant to dust and other weather conditions; however they cannot sense texture. Mapping, either as a part of sensing or a layer redundant to sensing, requires some sort of connectivity for the purpose of updates. There are two main types of maps; the first is navigation maps, which provide information about the steps required to reach our destination. The second is high definition maps, which provide 3D information of the environment with centimeter precision. The last pillar is driving policy, which includes the set of rules to merge in the traffic and manage driving behaviors. The driving policy needs to learn human driving behaviors in order to drive properly in mixed traffic of human and autonomous driving.

2.1. Sensing. The sensing interprets the scene with 360° awareness and produces an environmental model. It can be achieved using different types of sensors. These may be active or passive sensors. On the one hand, active sensors work at long distances and under poor weather conditions. Some examples are LIDAR and RADAR, which are able to detect obstacles at distances of over 100 meters. On the other hand, passive sensors have the main advantage of their low cost. Furthermore, visual information can be very important in some applications such as traffic sign recognition or object identification. Figure 1 shows a general classification of different approaches for road detection depending on the sensor and the methodology.

LIDAR technology has been widely used to detect curbs, road surfaces, and different types of obstacles. They can be classified into two types: single beam (2D) and multibeam (3D). 2D LIDARs are usually mounted on the front of the vehicle parallel to the ground plane for obstacle detection. When the goal is road surface or curb detection, the sensor is mounted on the top of the vehicle facing downwards. This configuration is useful for detecting height variation of

the road shape; however, the shape is only detected in a fixed distance. Therefore, it is important to integrate the measurements along the time, compensating for the vehicle's ego-motion. Thanks to measurements integration, the confidence of the scene interpretation is increased and road shape may be modeled using clothoids, splines, or other model types. This configuration is also valid for road marking detection since LIDARs provide distance information and also surface reflectivity intensity. Furthermore, there are LIDARs having 4, 6, 16, 32, and 64 layers. The resulting point cloud is sparse compared to a single beam LIDARs with temporal integration; however, they create a new scenario of possibilities since they provide 360° precise 3D information of the environment from a vertical field of view of 26°, generating a point cloud of the entire scene. They have been used for many tasks, including road marking and road, curb, and vehicle detection. Different methodologies exist for road detection. Some of them are based on road appearance learning, where the main features are texture and color information. The second approach focuses on road limits detection, assuming that the space between limits is the road surface. Finally, modeling attempts to extract compact, high level representation of the road. In order to clarify the classification, Table 1 summarizes different methodologies of road detection.

2.1.1. Road Appearance. Road appearance is the feature that makes vision sensors truly relevant. Texture and appearance can be modeled by fitting road pixel histograms with a Gaussian Mixture Model (GMM) [3]. However, shadows cause the system to fail and may classify shadowed areas as nonroads. To solve this problem, [4] a flexible number of color models are used in a modified hue that is invariant to brightness. Another strategy is to remove shadows from the images [5] by converting the original RGB image into an illuminant invariant image. Texture analysis is used to obtain a descriptor of the surface. On the one hand, Gabor filters and Histograms of Oriented Gradients (HOG) provide information regarding the orientation. The strength of the texture anisotropy describes the homogeneity in a region of interest [6]. Additional information on road appearance is extracted from a filter bank, which is an array of

TABLE 1: Classification of different road detection methodologies.

Methodology	Sensor	Feature	Examples	Refs
Road appearance	Vision	Color	Illuminant invariant, GMM	[3–5]
		Texture	HOG, anisotropy, filter bank, LBP	[6, 7]
		Automatic	CNN	[8, 9]
Road limits	Vision LIDAR	Road markings	Reflectivity intensity, adaptive threshold	[10–12]
		Road curbs	Digital Elevation Maps, difference of heights, curvatures	[13–20]
Geometrical modeling	Map + GPS RADAR Vision LIDAR	Parametric models	Parabolic curves, clothoids, B-splines, snakes	[21–25]
		Nonparametric models	Ant Colony Optimization, Dijkstra, A^*	[26, 27]

band-pass filters that usually have different scales and orientations. Finally, another common descriptor is the Local Binary Pattern (LBP), which describes the relationship between the evaluated point and its surrounding values. Those features per se are not robust for a road detection method; however they are usually included into a larger feature vector for a machine learning strategy. Some machine learning techniques will be explained in Section 2.3. Moreover, the computer vision features may be combined with other sensors, such as LIDAR, to detect the road and attain improved results [7].

2.1.2. Road Limits. This section presents an opposing approach as compared to the road appearance approach. The goal is to detect the features that describe road limits. In most of the cases, the road is limited by curbs, road markings, areas of vegetation, or parked cars. These features may be estimated using computer vision, RADAR, LIDAR, or a combination of all of them.

(i) Road markings: current ADAS have integrated road marking detection in highways since this is the most important feature to maintain the vehicle in its lane. Due to the contrast between the road and the lane markings, the problem is addressed by searching for gradients in the image [10]. Road markings are painted with reflective materials so that they are highly visible during the night. That property makes them detectable with a LIDAR sensor since they detect surface reflectivity intensity [11]. In addition to road marking detection, the correct interpretation of arrows and other symbols can be taken into account in higher level modules to ensure proper navigation [12].

(ii) Curbs: road markings are present in most roads; however in some rural roads and residential areas they may not be found. In these sites, road curbs are also a discriminant descriptor to delimit the drivable area, especially in urban environments. Depending on the city, curbs may vary significantly in size, from 3 to 12 cm or even more. Stereo cameras depend on image pair matching methods to obtain depth information. Although 2D LIDARs can directly return this information, only select curb points can be detected using this sensor [13]. 3D LIDAR provides a dense point cloud and thus makes it possible to detect a larger extent of the curb [14]. LIDARs provide precise measurements, which is a very important feature to detect small curbs. Detection range using LIDAR is up to 50 meters [15]. On the other hand, it is not possible to achieve this rate with a pair of stereo cameras, which have a detection range of approximately 20 meters. The most common approach used to detect curbs, according to the literature, is the use of a Digital Elevation Map (DEM) to integrate the 3D measurements and a posterior analysis of height variation [16]. Some of them attempt to model the curb shape with cubic polynomials or a cubic spline [17]; however, the diverse type of shapes present in urban scenarios makes these methods fail in certain scenarios. A combined detection using computer vision and LIDAR sensors is presented in [18] having accurate results thanks to the complementary features of each sensor. The main parameters for the algorithm evaluation are curb height and lateral distance with respect to the ground truth. On one side, in [19] the minimum curb height detected is 5 cm with an error of 3 cm at 20 meters. On the other side, in [16] the lateral distance is evaluated up to 20 meters with an error of 20 cm. Our novel method based on curvature values is presented in [20], where the system is evaluated on point clouds from stereovision and 3D LIDAR. The method obtains a lateral error of 14 cm at 20 meters distance and can detect curbs of 3 cm height up to 20 meters when the curb is connected to the curvature image.

2.2. Geometrical Modeling

2.2.1. Parametric Models. Road and lane detection tend to be guided in a top-down manner by fitting a geometric model to the visual features extracted from the image. The simplest geometric model used for road boundaries is straight lines [21]. Roads are not always straight. Thus, parabolic curves [22], clothoids [23], B-splines [24], or active contours (snakes) [25] are more complex models adapted to curved

road. These parametric models improve noisy bottom-up detection due to their width and curvature constraints, but urban environments are more difficult to model given the presence of parked cars that do not fit some of the width restrictions.

2.2.2. Nonparametric Models. Nonparametric models are less common since they only require that the line be continuous. Thus, the model may be less robust than parametric models but more flexible to adapt to the irregular shapes present in urban environments [26] or rural paths [27]. One example of a nonparametric model is Ant Colony Optimization (ACO) for finding optimal trajectories on the image plane. Furthermore, the image can be formulated as a graph, and traditional graph algorithms such as the shortest path, Dijkstra, or A^* can be applied to obtain the continuous road boundary.

2.2.3. Map-Based Models. High definition maps are a robust way to navigate [1, 2]. They are usually created by integrating several measurements of a multibeam LIDAR [28, 29] or multiple single beam LIDARs [30] and their size is ~2 GB/km, which is difficult to manage during a long trip or in a city. A higher resolution map can be updated beyond sensing. Instead of high resolution maps (~GB/km), sparse 3D maps can be created using landmarks and may be stored (~KB/km). These maps can be shared and the map can be updated using crowd sourcing. The other point of view regarding map models is digital navigation maps. An example of this type of map is the Open Street Maps (OSM), a collaborative project created and updated by a large global community. All of the information stored in the map is editable and it is freely accessible. The map consists of a listing of streets called ways. Every way is made up of a list of nodes with a location and its relationship with the other nodes and ways. Thanks to the location and relationship between nodes, the shape of the current road and the surrounding streets may be estimated [31].

2.3. Features Integration. All of the features described in previous sections are relatively weak on their own. However, when they work together they complement each other, obtaining a strong road descriptor. The fusion of the features requires some type of optimization process to give relative weights to each feature. Machine learning methods are commonly used for this task. Some examples of those techniques are the Support Vector Machine (SVM) [32], Neural Networks (NN) [33], Bayes Classifier, decision trees (DT), Random Trees (RT), Extremely Randomized Trees (ERT), and boosting [34, 35]. They receive a feature vector and the corresponding label for each pixel in the image. After the training stage, the classifier has learned the weight of every feature in the final response. The use of Convolutional Neural Networks (CNN or ConvNet) [8, 9] requires high computational requirements and its complexity should run in a Graphical Processing Unit (GPU) for the training stage. One of the most relevant features of this new technique is that instead of receiving a feature vector it can calculate its own feature vector during the training with the image as the unique input. Furthermore, its performance is significantly

better than that of other machine learning techniques [36, 37]. CNNs always have the problem of overfitting due to many connections in the full connection layer and they require a large training set with high variability of scenarios in order to learn all of them. The objective of this paper is to create a new environment perception method to detect the road in urban environments fusing stereo vision and digital maps by detecting road appearance and road limits such as lane markings or curbs. CNNs hard-couple the system to the training set. Even though our approach is based on machine learning (ML) techniques and could be affected by the overfitting problem, the features are calculated from different sources (GPS, map, curbs, etc.) and they make our system less prone to overfitting.

3. System Description

Autonomous vehicles need to detect the road as an important part of autonomous navigation. This challenging problem is addressed in this paper with a new method that is based on computer vision and digital maps. The computer vision module extracts features based on road texture, color, and geometry information. Road detection is tackled as a binary classification problem with the following labels: road/nonroad; therefore, boosting classifiers fit well since they are designed for this type of classification. Figure 2 shows a feature classification depending on the sensor. Greyscale cameras are used to detect road markings, calculate the vanishing point, and extract certain types of texture information such as LBP and HOG. Color cameras are necessary to distinguish vegetation areas that delimit the drivable space in many cases. They are also useful for obtaining an illuminant invariant image which is robust to shadows. A pair of grayscale cameras is used as a stereo vision system to estimate 3D information. 3D features are used to detect heights with respect to the ground plane. Furthermore, large obstacles such as vehicles, buildings, or trees are detected using normal vectors and curvatures. Finally, the curvature feature is the base of the curb detection module. All of the features mentioned above have been extracted from stereo or monocular cameras. Furthermore, prior knowledge of the road shape estimated from the navigation map is included. This new feature makes the classifier more robust in situations in which camera sensors fail. In Figure 2, the features marked in blue provide high level context information and those marked in red are included in a boosting classifier to detect free space. Road features are divided into three distinct types. The first type includes features that sense road appearance. The second type groups together geometry-based features and the last uses a high level set of features since they provide context information.

3.1. Appearance-Based Features. Appearance-based features enclose information related to textures and colors. The first ones are analyzed using LBP and HOG and the second ones with HSV, an illuminant invariant image, and a shadow detector. Given that LBP, HOG, and HSV are used extensively, only the illuminant invariant image is described in this section.
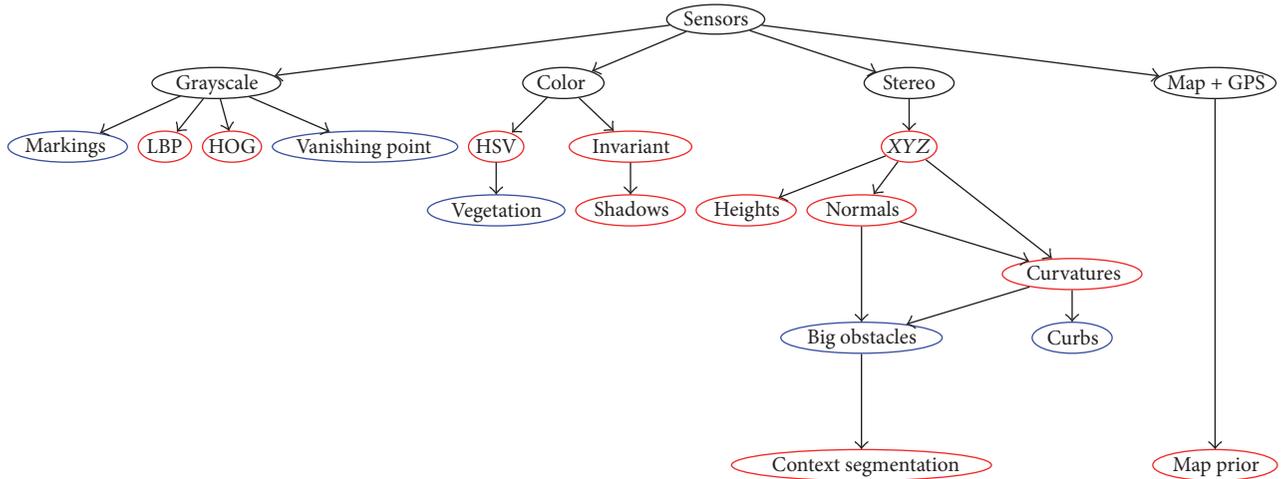
(a) Tree of features dependency. The features are classified depending on the source of information: monocular greyscale camera, monocular color camera, stereo greyscale cameras, digital navigation map, and GPS. Red nodes mean features included in the boosting classifier and the blue ones inform about the context
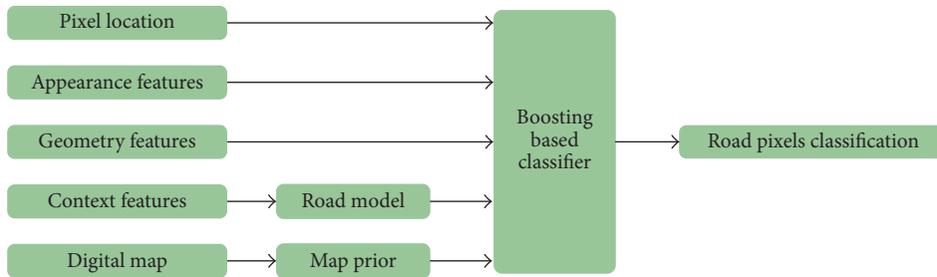
(b) General description of module connections

FIGURE 2: Features classification based on the sensor and general architecture of the proposed algorithm.

Road detection using computer vision is a challenging task, especially when the road is affected by shadows. The basis of many computer vision approaches is to consider that roads have some constant features such as color or texture that may be grouped together. Sometimes road texture and color are not homogeneous since the asphalt may have an irregular degradation or may have been partially renewed. Even in homogeneously textured roads, the road is strongly affected by shadows, creating a challenging new scenario for computer vision algorithms. Some approaches attempt to attenuate the shadow influence with an illuminant invariant space to detect the road [38, 39]. As explained in [5, 40], adopting certain assumptions regarding lights and cameras, color images can be represented in a shadow free grayscale image. The resulting illuminant invariant image is shown in Figure 3(b). During the process, the illumination of the scene is calculated, which is a very good feature to obtain the shadows of the scene; see Figure 3(c). When the shadows of the scene are detected, a wide range of possible applications appear: it may be a new feature for a classification method, it may be used as a mask for a special image processing method on shadowed areas, or it may also be used to obtain a new shadow free image.

3.2. Geometry-Based Features. Object shapes are an important characteristic for classification. The road is an almost horizontal flat surface and vertical obstacles are clearly separable using geometry descriptors. In the field of computer vision, a pair of stereo cameras is able to create a 3D representation of the scene. The factors influencing the reconstruction are focal length and the distance between the cameras. Our system is configured with a pair of cameras separated by 0.54 meters at 1.73 meters with respect to the road surface and equipped with 8 mm focal length optics. After rectification, the image size is 1242-pixel width and 375-pixels height. Using the semiglobal block matching (SGM) algorithm to estimate the disparity map, the system generates a dense 3D point cloud.

In urban scenarios, roads tend to be limited by curbs, road markings, or pavements of a different texture. Depending on the city, curbs may have a wide range of heights. Therefore, an adaptive method is necessary to detect curbs, regardless of curb height. Our approach focuses on the curvature feature. This feature describes a local surface variation and was applied to 3D semantic perception in [41]. After normalization, the curvature values vary between 0 and 1, with low values corresponding to flat surfaces. The result is a vector γ similar to normal surface vectors. Moreover, curvature vectors are more stable and robust. The objective is to detect variations in road surface; thus we only take into account the component γ_z, which is orthogonal to the road plane in our reference system (see Figure 4).

(a) Original image of the scene

(b) Illuminant invariant image

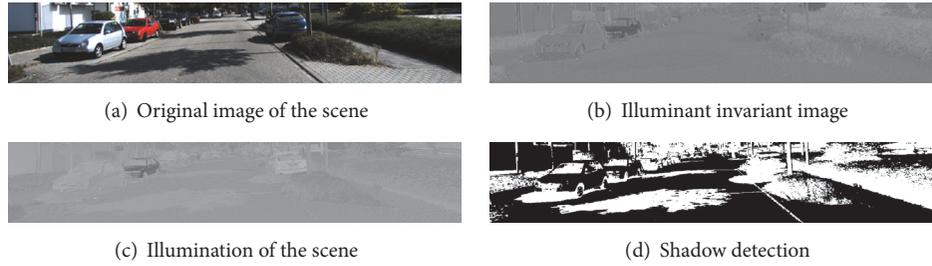(c) Illumination of the scene

(d) Shadow detection

FIGURE 3: Partial results during the shadow detection. The process generates an illuminant invariant image and also an image with the illumination of the scene. The use of both images makes the system able to detect the shadows of the scene.
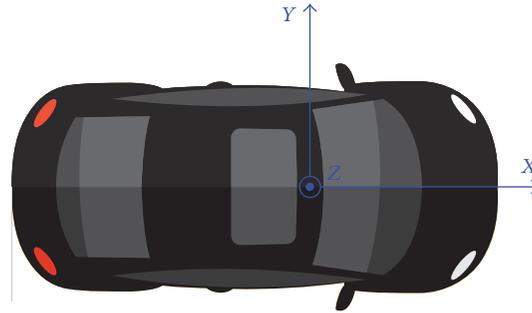

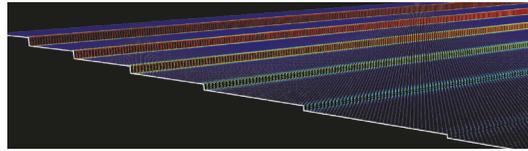FIGURE 4: Reference system of the ego vehicle.

FIGURE 5: Curvatures of an artificial 3D point cloud with curbs of different heights: 3, 5, 7, 10, 12, and 15 cm, respectively. Curvature values are represented in a color scale where cold colors correspond to low curvature values and warm colors correspond to high values.

Curvature variation is computed on an artificial point cloud to reveal the responsiveness of the algorithm. The point cloud has several curbs of distinct heights, beginning with 3 cm in steps of 2 cm up to 15 cm. As shown in Figure 5, the feature provides sufficient information to detect variations of the curvature even for a curb of 3 cm height.

Real scenarios differ from the ideal curvature estimation shown in Figure 5. In Figure 6, a set of real scenarios is presented in which mismatching errors during the stereo computation provoke invalid curvature values (see discontinuous red areas). Because the system relies on stereo vision, curvatures are robust to illumination changes (Figure 6(b)). Residential areas are a challenging scenario for detecting road limits since curbs are small and their detection is very important to safe driving. Curvature variations are visible even for small curbs. Figures 6(c) and 6(d) demonstrate that this feature is able to distinguish curbs of 3 cm even at far distances.

The road shape can be approximated to a plane in some cases. Given that vertical obstacles have large distances with respect to the ground, the smaller the distance to the plane is, the most probable it belongs to the road. For that reason,

height with respect to the ground plane is also included in the feature set.

3.3. Context-Based Features. Some features produce higher level information than others. The ones detailed in this section are not relevant to the road description. However, they offer information regarding the context of the scene, which is very useful to our understanding of how the road is distributed.

3.3.1. Road Markings. Road marking detection is a basic task for autonomous navigation. In a multilane scenario the free space must be split in lanes and for this road markings are crucial. As explained in [42], a median filter is applied to the input image. The window size should be adjusted due to perspective. In order to maintain the window size constant, a zenithal projection of the scene (also known as bird eye view: BEV) is reconstructed. Lane markings appear parallel in the new view since they are not distorted by the perspective.

3.3.2. Vegetation. Despite the fact that the road surface has a wide range of textures and color values, green areas usually

(a) Big curbs on urban road

(b) Robustness to shadows

(c) Residential scene

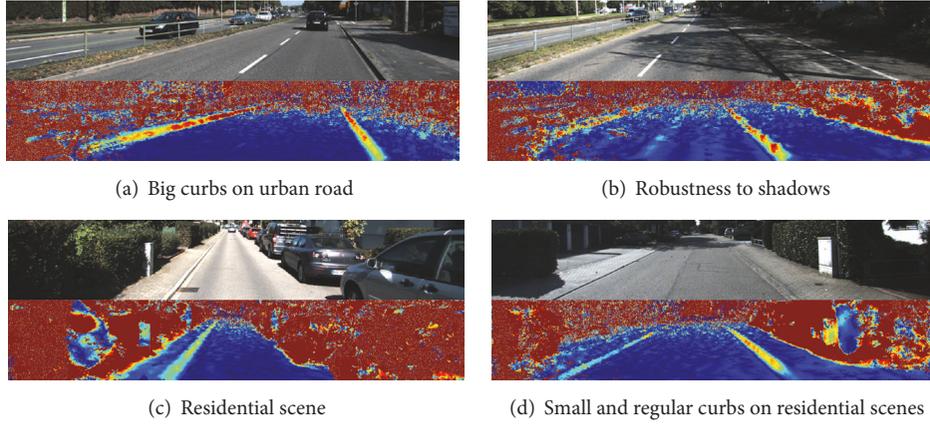(d) Small and regular curbs on residential scenes

FIGURE 6: Curvature values on different urban scenes.

correspond to vegetation in most scenarios. Therefore, the free space detection system considers these areas to be nondrivable. Variations in the green color are detected by selecting a range of HSV values.

3.3.3. Curbs.

As explained previously, the curvature variation is a good feature for the detection of curbs; nevertheless, in a real scenario, curvature values do not have homogeneous values given the mismatching errors occurring during the stereo disparity map computation. Depending on the curb height, curvature values differ in each scene. The challenge is to detect all of the curbs using the same algorithm regardless of curb height. The details of the algorithm are explained in [20] but a brief description is also included for consistency. A close relationship exists between curb height and curvature value, obtaining the values seen in Table 2. The thresholds selection is calculated offline, classifying curvatures in 5 groups. The resulting clusters are filtered independently using morphological operations and contour analysis since the image has several noisy measurements. The filtered clusters are merged back and the new clusters are considered as road curbs. This approach provides flexibility because the algorithm works on a wide range of scenes and for every type of obstacles, detecting in every case the most dominant curvature value. Those of 3 cm height are detected as well as others having larger heights. The use of fixed or empirical thresholds is avoided, given that the proposed function is automatically adapted to different scenes depending on the predominant curvature value.

3.3.4. Obstacles.

Free space is usually limited by curbs, road markings, vegetation areas, or other obstacles, such as buildings, parked cars, post lamps, traffic lights, or traffic signs. These types of obstacles are detected using 3D information from the stereo cameras. The 3D points are processed to estimate normal and curvature vectors. Points having components $n_x \geq 0.5$ or $n_y \geq 0.5$ or $\gamma_z \geq 0.5$ are considered to be large obstacles. Some pixels have noisy or unrealistic vector values. Therefore, every component is filtered independently

TABLE 2: Curb curvature values.

Description	Curvature	Color
Flat surface	$0 \leq \gamma_z < \alpha_0$	Not painted
Very small curbs (~3 cm)	$\alpha_0 \leq \gamma_z < \alpha_1$	Yellow
Small curbs (~5 cm)	$\alpha_1 \leq \gamma_z < \alpha_2$	Orange
Regular curbs (~10 cm)	$\alpha_2 \leq \gamma_z < \alpha_3$	Red
Big obstacles	$\alpha_3 \leq \gamma_z \leq 1$	Purple

by area. Afterwards, they are merged together to obtain the final result. Mismatching errors produce small holes in the detected obstacles. In order to get more robust results, columns having an obstacle are considered to be occupied from the bottom row of the obstacle until the first row of the image. The resulting image has obstacles without holes inside, obtaining a more realistic representation of the scene.

3.3.5. Road Model.

There are two main types of features: the first type describes the road and can be included directly in the classifier. The second type of features includes road limits, such as road markings, curbs, vegetation areas, or obstacles. These features provide important information regarding the scene; however they should not be included directly in the classifier (curbs and road markings) since the features do not describe either the road or the nonroad area. Figure 7 shows some examples of curb detection combined with the ground truth of the road (green). The pixels in red correspond to the curb points that lie on the road and the blue pixels are those lying outside of the road. This is why, in [43], the weights that correspond to those features were very small in the final classification response. In order to increase the weight of these features, the proposed method converts them from limit features to a new feature that describes the road.

Some state-of-the-art approaches use virtual rays starting from the bottom of the image to detect the road boundary. Analyzing the feature values along the rays, the point that satisfies certain conditions is established as the road limit for that ray. The connection of all of these creates a closed polygon that is considered free space. As demonstrated in

Figure 7: The road curb is not good feature for a road/nonroad classifier because half of the measurements are positive values of road and the others are not.

Figure 8: Example of free space detection using a set of rays starting from the bottom center of the image.

Figure 8, this point of view is sensitive to noisy measurements. In this paper, the proposed analysis relies on the vanishing point as a starting point of the rays. This point of view is more resilient to incorrect measurements and is more intuitive. Regarding the pin hole camera model, straight parallel lines converge in a vanishing point. Even in narrow roads, road limits are, in most cases, parallel. For this reason, a set of rays are estimated starting from the vanishing points. First, the vanishing point has to be detected. Several papers have been published on this topic. Since the goal in this paper is not the development of a novel method to obtain vanishing points, a public library was used for this purpose [44].

The general description of the method is explained in Figure 9. The main goal of the algorithm is to find the radial rays that may be the road limits. Firstly, a set of radial rays from the vanishing point are analyzed along the image. The sum of the features along the ray is displayed in Figure 10(a). After smoothing, the first derivative is calculated. Curbs or road marking features create two strong symmetric peaks on the feature first derivative. The obstacles and vegetation have a different distribution; they only create a single peak on the feature first derivative since their values are not symmetric with respect to the road limit. Vegetation areas are usually large surfaces outside of the road. False positives for the vegetation detector are very low, given its robustness; instead of establishing complex conditions to the feature first derivative, a fixed threshold is applied directly to the feature. After the creation of the rays vector, a second stage adds the lateral distance of each ray to the camera origin in meters. This is accomplished by converting the ray projection into a BEV. The third step is a high level filter that assumes the following assumptions:

(i) The ego vehicle is on the road.

(ii) If the road has limits from the vegetation detector, further candidates are discarded.

(iii) If the ego vehicle is on the road, then candidates from the obstacles detector which are directly in front of the ego vehicle should be removed since they are another moving vehicle instead of a free space limit.

(iv) Some road marking candidates are arrows or other road marking symbols distinct from the lines that are useful to detect road limits. Dashed and solid lines have a regular pattern along the ray; otherwise, the

symbols are isolated peaks in the ray analysis. Thus, any road marking candidates that do not satisfy this condition should also be removed.

(v) Some of the candidates are very close to each other. For example, a road marking and a curb are frequently close to each other. In order to reduce the number of candidates, both are merged into one single candidate using the mean angle of both candidates.

All of the previous assumptions significantly reduce the number of candidates. This is very important during the fourth stage, when a recursive function finds all possible adjacent lanes with a specific range of valid width. In addition to the previous condition, lane widths must be similar to one another. The algorithm is flexible and adapts to the lane range depending on the road type, since it takes information from a digital navigation map. Digital maps include the number of lanes and the type of road. Some road types include *highways*, *primary*, *secondary*, *tertiary*, or *residential types*. The last two types correspond to roads that do not typically contain road markings; for this reason lane width has less restrictive conditions on these roads.

The unfiltered lane combinations are shown in Figure 11. As in the previous step of the algorithm, the resulting road models must satisfy some high level restrictions.

(1) The ego vehicle is on the road.

(2) The number of lanes should match the information stored in the map.

(3) In residential streets having road markings, the ego vehicle drives on the right lane.

(4) The model should have a small height difference between lanes.

The first condition removes the combinations of Figures 11(a) and 11(c). The core of the algorithm is an iterative loop that calls a recursive function with different sets of parameters, beginning with hard, restrictive conditions and relaxing these conditions at each iteration. If, at the end of the loop, there is no valid combination of lanes, the number of lanes is decreased and the loop is called again. For example, a street with 2 lanes of 3 meters has no road markings and is limited by curbs. At the first iteration, the restrictions for that road anticipate 2 lanes of 3 meters each; however, there
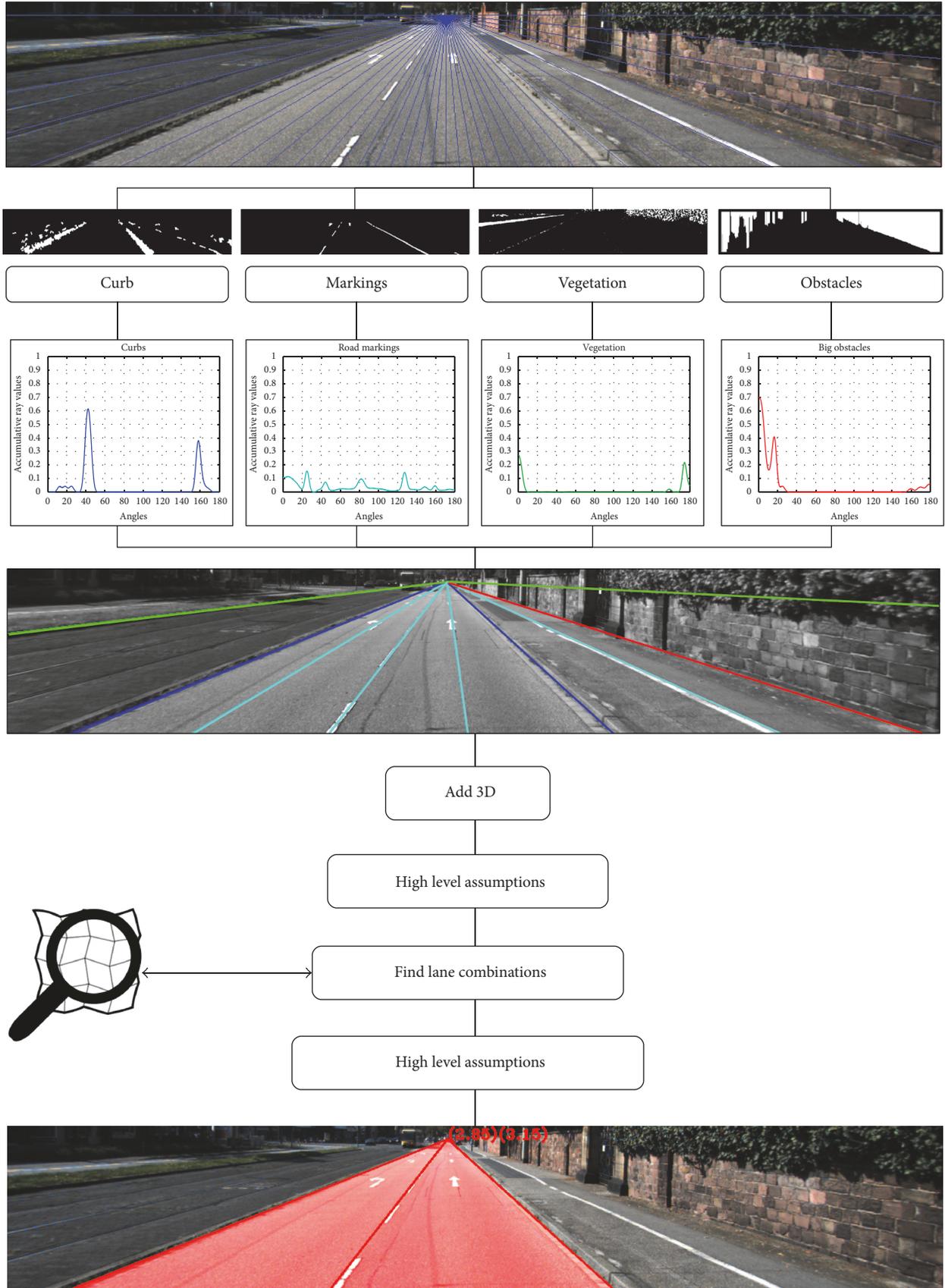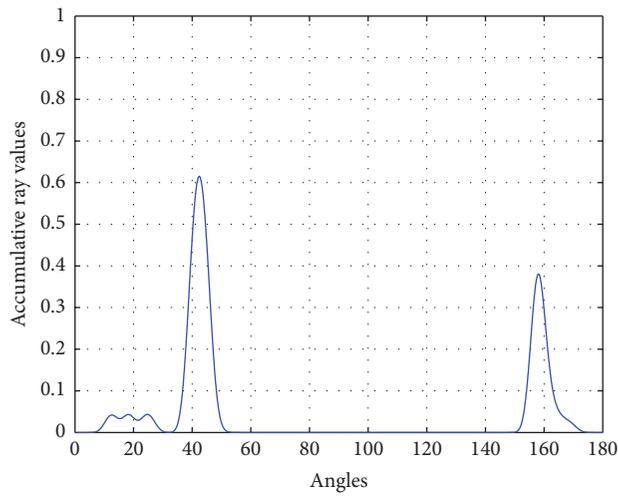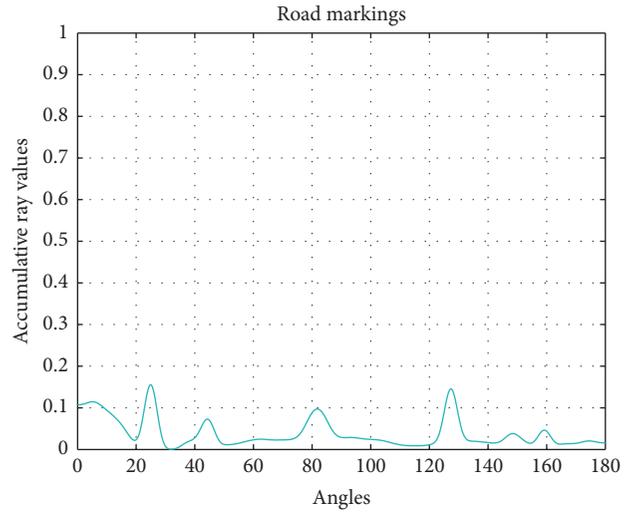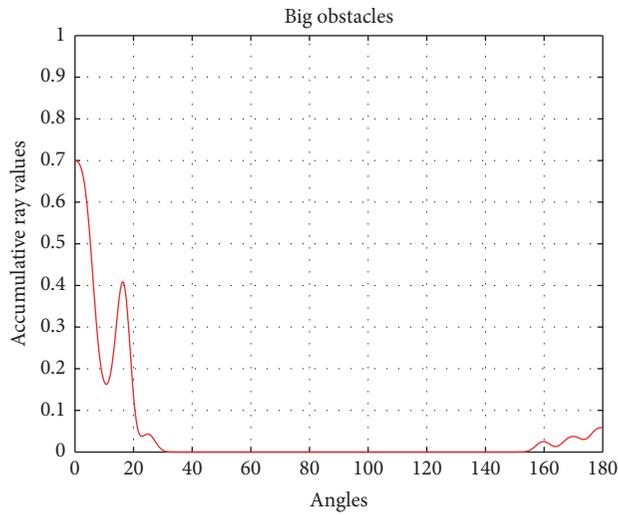
FIGURE 9: General description of the algorithm to convert a set of features that describe road limits into a new road model.
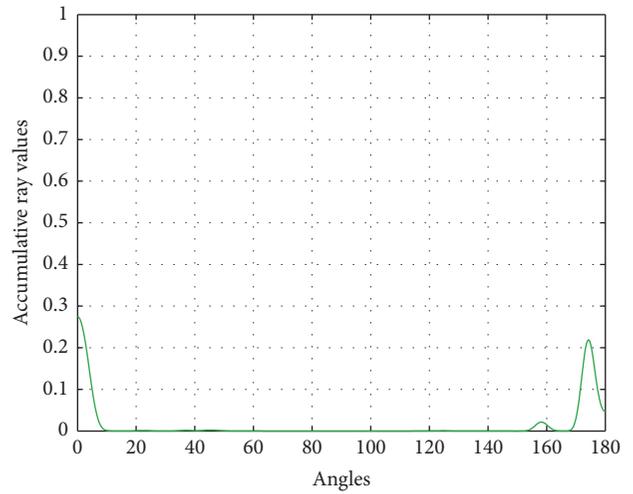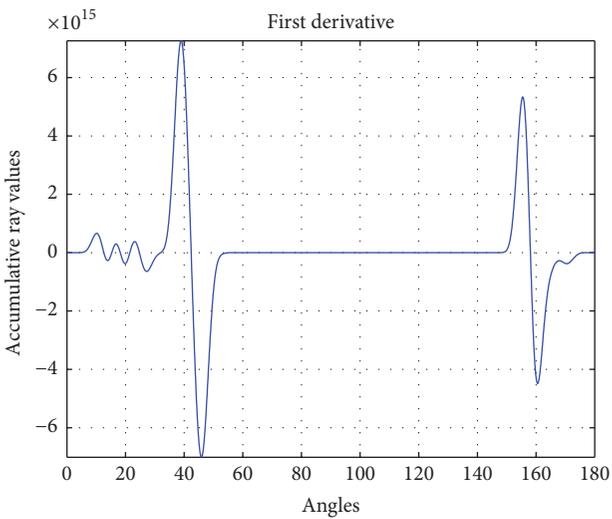
(a) Curbs

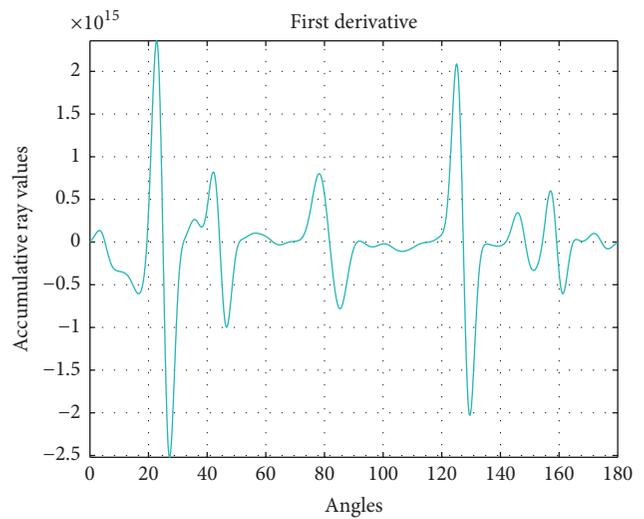(b) Road marking

(c) Obstacles
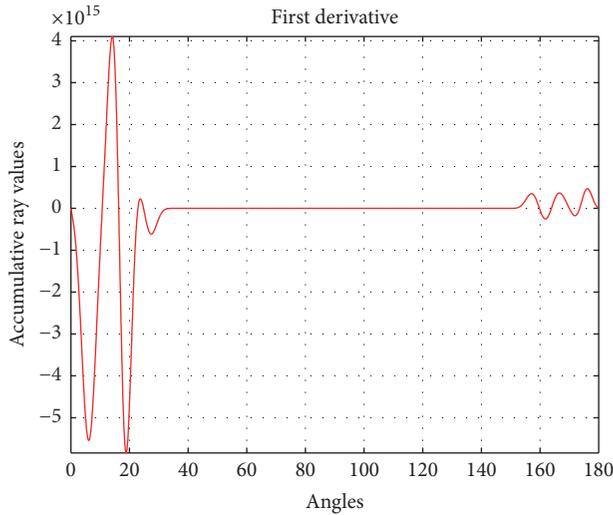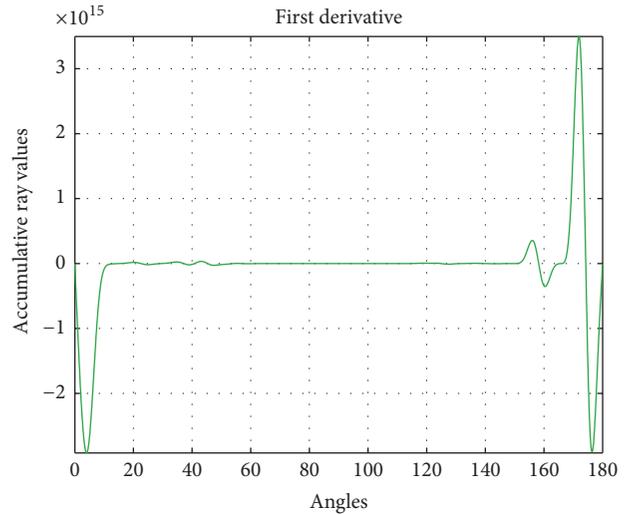

(d) Vegetation

(e) Curbs derivative

(f) Road marking derivative

FIGURE 10: Continued.

(g) Obstacles derivative

(h) Vegetation derivative

FIGURE 10: Graph analysis of the rays.

(a) One lane on the left

(b) One lane in the middle

(c) One lane on the right

(d) Two lanes on the left. Combination selected by the algorithm

(e) Two lanes on the right

(f) Three lanes

FIGURE 11: All possible lane combinations without high level filtering.

is no combination that satisfies the restrictions. In the next iteration, the algorithm finds a single lane of 6 meters width, which fits the real scene. If lane markings are not correctly detected, the possibility of finding a valid lane combination is reduced, especially when the number of lanes is greater than or equal to 3, since instead of finding 3 lanes of 3 meters each, it finds one lane of 3 meters and another of 6 meters, which does not satisfy the condition of similar width between lanes. Figures 11(d) and 11(e) are from a one-way street, however if it were a two-way street, the third condition would remove Figure 11(e). In the fourth condition, the ego lane is detected (Figure 11(b)) and a plane is fitted to its 3D points using RANSAC. The combination having the smallest mean distance from the plane to the other lanes is preserved and the others are discarded.

This novel approach converts important features that are not suitable for direct inclusion in the classifier into a road model. It reads the number of lanes and the type of road from digital navigation maps to adapt the filtering parameters for optimal road segmentation. The detected road is estimated only from features of the stereo cameras, the color camera, and the digital navigation map, making the system free of any type of machine learning technique.

3.4. Road Shape Prior Obtained from Digital Maps. In this section, a road prior is generated from the road width estimated in Section 3.3.5 and the road shape provided by the digital navigation map. This method creates a relationship between the map and the road segmentation method where both algorithms take information from each other. As detailed in Figure 12, the road segmentation method covers road type and the number of lanes from the map and also sends the road width to the map in order to create the road prior. It is a kind of symbiosis where both functions receive benefits from the other.

Open Street Map is the digital navigation map used for our approach. These collaborative maps have been created by a large global community and all of the information
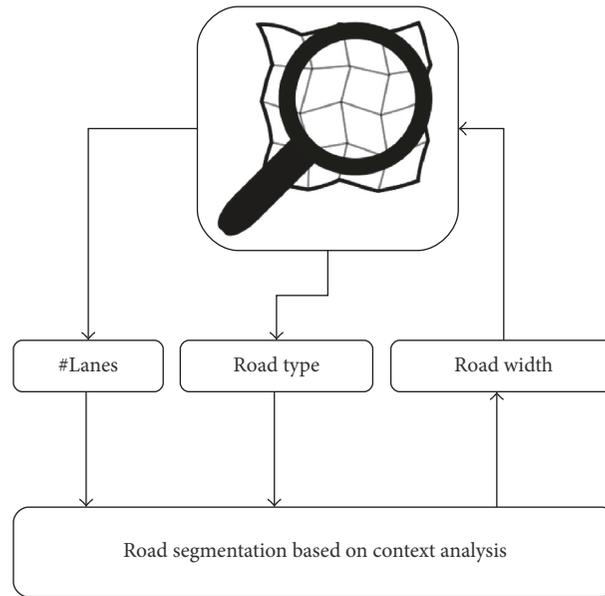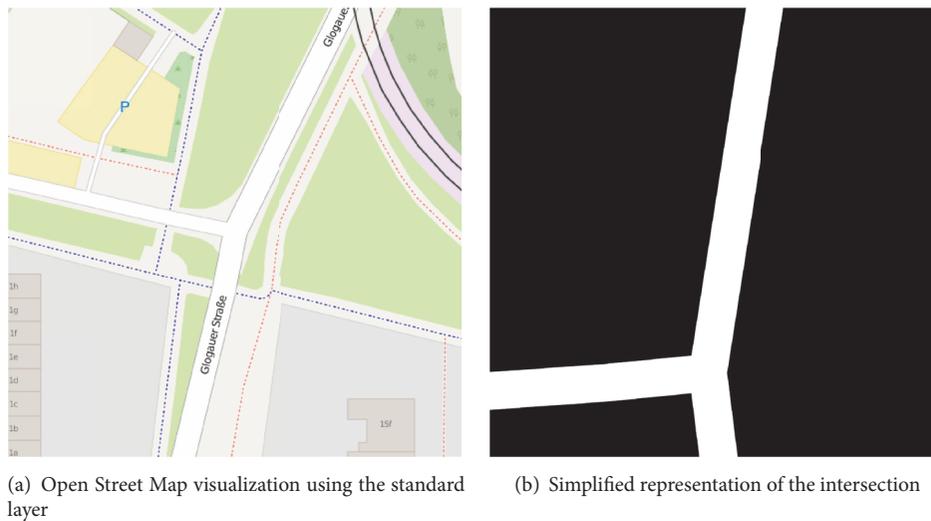
FIGURE 12: The map prior requires a road width, which is obtained from the road model presented in Section 3.3.5. Furthermore, the generated road model uses the number of lanes and the type of the road from the map. It is a kind of symbiosis where both functions take benefits from the other.

(a) Open Street Map visualization using the standard layer

(b) Simplified representation of the intersection

FIGURE 13: Standard layer map and line segment representation of an intersection. The orientation of the map is aligned with the vehicle orientation and the road width is estimated using the method presented in Section 3.3.5.

stored in the map is editable and it is freely accessible; see Figure 13(a). The map consists of a list of streets called ways. Every way is composed of a list of nodes with a location and its relationship with the other nodes and ways. As detailed in Figure 13, thanks to the location and relation between the nodes, the shape of the current street and its surroundings can be estimated.

The creation of a valid road prior requires the transformation of the map orientation to the current heading of the vehicle. The location and heading of the ego vehicle are read from a GPS/IMU sensor. These are necessary to find the current street in the map and to create a simplified map with the current street and the others that are connected to it.

The result of the road prior is shown in Figure 14, where different scenarios have been processed using the estimated road width and the shape obtained from the map. As demonstrated in Figures 14(a) and 14(b), the correct localization of the vehicle creates a good prior of the road shape. The map information is especially useful in the presence of intersections where road detection is more complex (see Figures 14(c) and 14(d)). The nodes of the way are referenced to the center of the way. However, sometimes there is a drift between the map and the real center of the street. This is the case in Figures 14(e) and 14(f). Given that the map drift is usually constant along the street, the model may be displaced along the measurements to estimate the offset and future
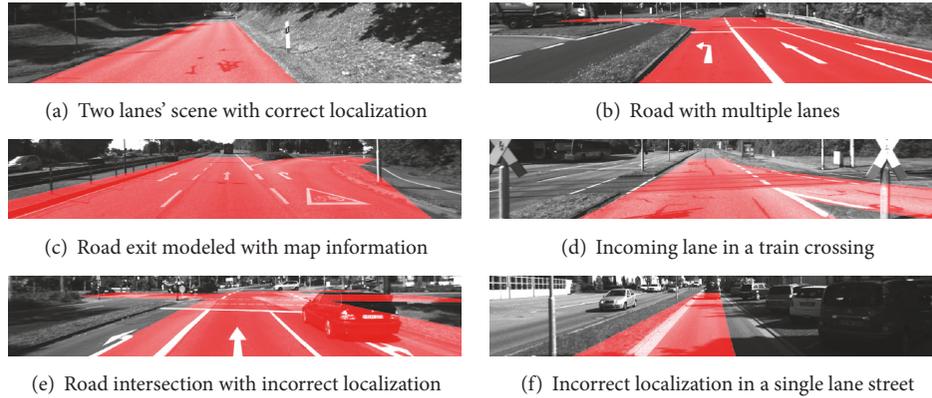
(a) Two lanes' scene with correct localization

(b) Road with multiple lanes

(c) Road exit modeled with map information

(d) Incoming lane in a train crossing

(e) Road intersection with incorrect localization

(f) Incorrect localization in a single lane street

FIGURE 14: Road models estimated from the road shape information extracted from the map.

(a) Road model of an adjacent street

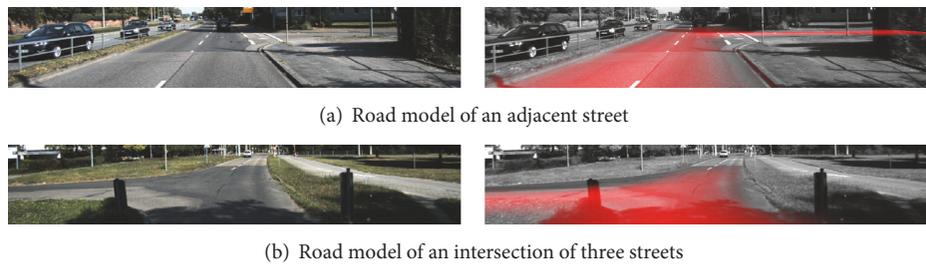(b) Road model of an intersection of three streets

FIGURE 15: Road prior obtained after modeling the uncertainty of the vehicle position and orientation.

models may be adjusted with the estimated offset. However, as the maps are updated by many collaborators, the drift can vary from one street to another, requiring the estimation of the drift at every street. In order to mitigate this offset, the final road prior is obtained by modeling the uncertainty of the vehicle position and orientation with a variability of ±2 meters and ±10 degrees, respectively. Furthermore, road width is also modeled with an uncertainty of ±1 meter. Figure 15 shows the final road prior after modeling the uncertainty of vehicle position and orientation. This model is used for the probability of a pixel belonging to a road.

3.5. Boosting-Based Classifier. Boosting techniques are becoming very relevant in the road classification problem [45]. This technique combines the performance of many weak classifiers to produce a strong classifier. The weak classifier is computationally fast and is usually a decision tree. Instead of using decision trees as weak classifiers, they may also be used for classification, with each tree leaf being marked with a class label and with multiple leaves having the same label. Random trees are a collection of decision trees and are therefore also known as a random forest. Every decision tree takes the input feature vector and classifies it and the forest output is the class label that receives the most votes. During the training stage, at each tree node, a random subset of features are used to find the best split value. On the other hand, Extremely Randomized Trees choose the feature index and the split value randomly.

In order to find the best classifier for the road detection problem in urban scenarios, the following classifiers are compared: Discrete Adaboost (BoostD), Gentle Adaboost (BoostG), Extremely Randomized Trees (ERT), Random Trees (RT), and decision trees (DT). In Table 3, different tree based classifiers are compared depending on the split value and feature selection.

The approach exploited in this paper is the robust collection of certain features, descriptors, or properties from the sensors and certain a priori information from digital navigation maps and the subsequent use of classification techniques for the final decision, regardless of whether or not there is road in each part of the captured images. Particularly, the features described in this chapter embrace appearance-based features to describe the texture and color of the road, geometry-based features to obtain the shape of the road, and the geometry of the obstacles present on the road. Furthermore, features related to context information are analyzed in a high level interpretation of the scene.

4. Results

Since this paper focuses on urban environments, the system is evaluated using the public dataset KITTI Vision Benchmark Suite [46]. The dataset provides images and information of urban scenarios from different types of sensors, such as monochrome and color cameras, multilayer LIDAR, GPS, and IMU. The evaluation method used to measure the quantitative results is the F_1-score since this score is used in the

TABLE 3: Comparison of tree based classifiers depending on the split value and the feature selection. The max depth is the maximum depth of each weak classifier.

Classifier	Feature	Split value	Max depth
DT	Best	Best	1
RT	Random	Best	N
ERT	Random	Random	N
Boosting	Best	Best	N

TABLE 4: Basic feature selection for the classifier parameters adjustment.

Feature	Length	Selected
Normals	3	✓
Curvatures	5	✓
Big obstacles	1	✓
Vegetation	1	✓
Heights	1	✓
HOG	36	✓
HSV	3	✓
Illuminant invariant	1	✓
LBP	1	✓
Pixel location	2	✓
XYZ	3	✓
Shadows	1	
From limits to road	1	
Map prior	1	
Total	*60*	*57*

KITTI dataset [47]. The dataset consists of 289 images and is divided into three types of scenes: the first is urban marked (UM) roads, the second is Urban Multiple Marked (UMM) lanes, and the third is Urban Unmarked (UU) roads.

$$F_1\text{-score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}. \quad (1)$$

The F_1-score has been used in two different images. The first is the image plane, which evaluates the performance in a pixel level. This is the most common approach found in the literature. However, in a vehicle scenario, its control stage usually occurs in a zenithal view, which is also known as a 2D bird eye view (BEV). The KITTI benchmark has a ranking sorted by F_1-score calculated on the BEV images. In order to compare our system with other algorithms of an international level, the same evaluation method is adopted. In the image plane, every pixel carries the same weight in the global statistics. Therefore, a false positive (FP) at 7 meters has the same effect as the one at 40 meters. Nevertheless, when a pixel in image plane is converted to the BEV, the further pixels receive more importance in the global score.

Our proposal may not be evaluated in the test images of the KITTI benchmark since the GPS information is not available for the test images. Thus, performance statistics are computed on the training dataset dividing 50% of the images for training and 50% for testing.

4.1. Classifier Selection. As mentioned previously, the following classifiers have been compared: Discrete Adaboost (BoostD), Gentle Adaboost (BoostG), Extremely Randomized Trees (ERT), Random Trees (RT), and decision trees (DT). According to the basic feature vector detailed in Table 4, an analysis of how the classifier parameters affect performance is explained in this section.

The most important parameters to be adjusted are as follows:

(i) *Type of classifier*: the chosen classifiers are decision trees (DT), Random Trees (RT), Extremely Randomized Trees (ERT), Discrete AdaBoost (BoostD), and Gentle AdaBoost (BoostG).

(ii) *Number of weak classifiers*: the analyzed values for this parameter are 50, 100, 250, and 500.

(iii) *Maximum depth*: the maximum depth of each weak classifier: the analyzed values for this parameter are 5, 10, and 25.

The classifiers have been trained with the same number of samples (~4.5 M) and features (57). The most important aspects to consider when choosing the best classifier are memory requirements and performance. The selection of the best classifier is decided in three steps: The first step evaluates performance. Given Figure 16(a), the classifiers with the best performance are Gentle Adaboost and Discrete

TABLE 5: Performance comparison of unary potentials trained with different combination of features.

Feature set	Image plane				Bird eye view			
	UM	UMM	UU	All	UM	UMM	UU	All
Basic	91.25	89.20	85.92	88.73	86.35	85.88	77.07	82.99
Basic + shadow	91.43	89.32	86.19	88.92	86.83	86.40	77.17	83.35
Basic + context	92.76	90.44	88.25	90.43	89.02	86.03	79.09	84.60
Basic + map	91.49	92.03	88.42	90.61	87.76	89.30	80.49	85.76
Full set	92.59	93.17	89.69	91.78	88.94	89.86	81.36	86.63

(a) F-score calculated in the image plane

(b) Memory usage during training
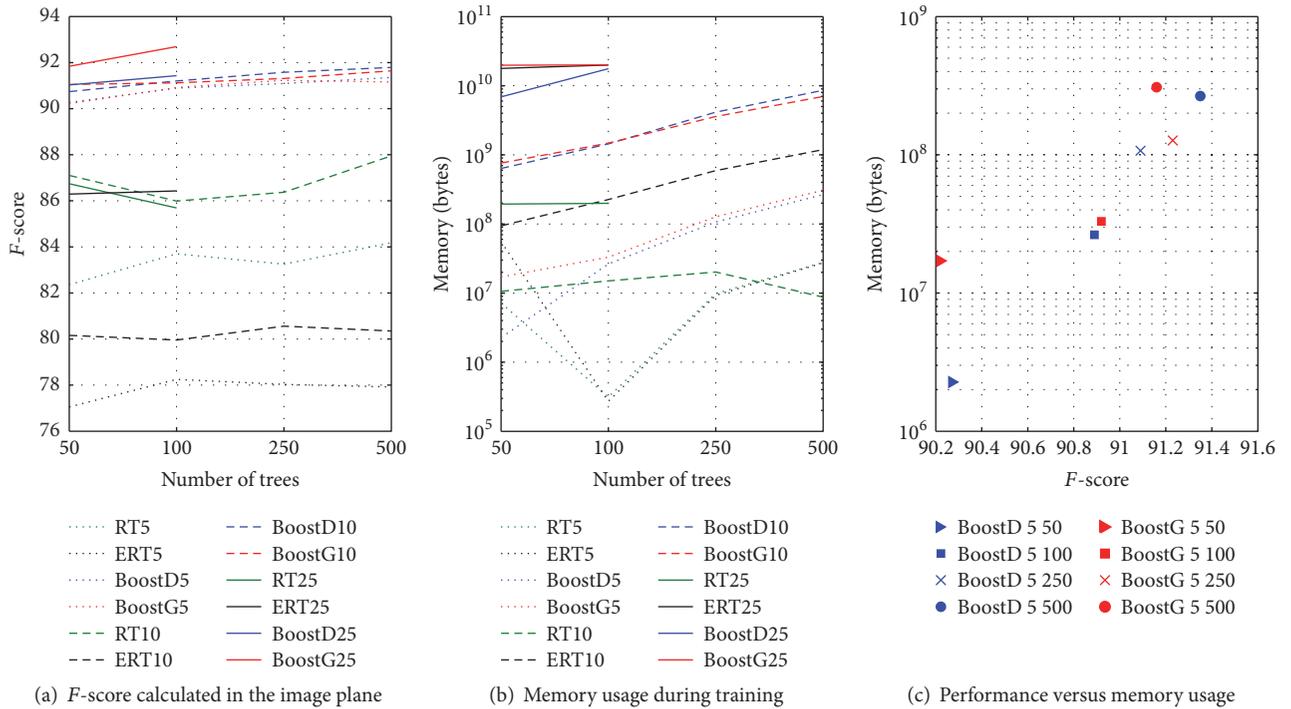
(c) Performance versus memory usage

FIGURE 16: Selection of the best classifier.

Adaboost, regardless of the selected tree depth. The second step evaluates memory usage. Figure 16(b) shows the memory requirements for all classifiers. It is observed that boosting classifiers with depths exceeding 5 require high amounts of memory. In our system we assume a maximum of ~1 GB for the road classifier, which discards classifiers having more than 250 trees and depths exceeding 5. The tradeoff between performance and memory requirements is represented in Figure 16(c), where the classifier Gentle Adaboost with 250 trees and a depth of 5 is the one having the best balance between memory usage and performance. The results discussed in the rest of the document have been obtained using this classifier.

4.2. Comparative Features. The classifier has been trained with 50% of the available training data and the other 50% is used for testing. In most of the images, the number of samples

per image is ~465 K pixels. The amount of samples for the 150 images is intractable (~69 M); therefore a subsampling technique is applied to reduce the number of samples by up to ~30 K per image. The first step is to remove samples above row 155, which is over the horizon line. The second is to take 1/3 of the pixels in the horizontal and vertical dimensions. Finally, the amount of samples used in the training stage is reduced to (~4.5 M).

Given the basic feature set described in Table 4, three more features have been tested to evaluate their influence on the final response. Table 5 shows the quantitative results of each feature combination in the image plane and BEV perspective on UM, UMM, and UU scenes.

On average, the shadow detector increases the performance by 0.36% as compared to the basic feature set. The improvement is similar in all of the scenes. Despite this,

(a) Lanes estimation based on road limit features and the vanishing point

(b) Prior knowledge of the road shape based on a digital navigation map

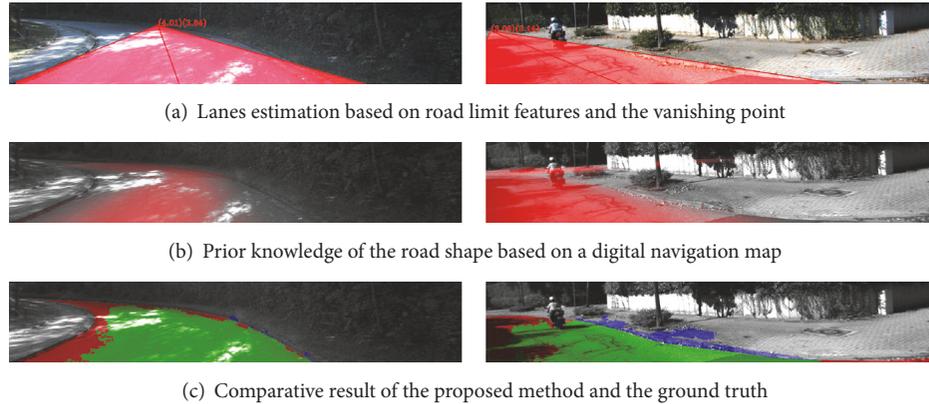(c) Comparative result of the proposed method and the ground truth

Figure 17: Qualitative results of the proposed method in narrowed roads. The meaning of the colors is as follows: TP: green; FP: blue; and FN: red.

the road segmentation based on context features increases the score, especially in scenes having two lanes with road markings (UM). The improvement of 2.67% and 2.02% in UM and UU scenes, respectively, contrasts with the 0.15% of the UMM scenarios. The road detector based on context information adds the lateral distance for each limit and attempts to find a combination of lanes that satisfies the restrictions on number of lanes and similar lane width. In scenarios such as UMM, most of the images have 2, 3, or more lanes. If road markings are not correctly detected in each lane, the function could generate an inaccurate road model. For example, if the road has 3 lanes of 3-meter width and the system detects one lane of 3 meters and another of 6 meters, the lanes combination will be discarded due to the difference between lane widths. That is the reason for the imperceptible improvement in UMM scenes. The map prior obtained from the digital navigation map offers important information of road shape. This feature increases the performance by 3.42% in UMM and UU scenarios and 1.41% in UM with respect to the basic feature set. The combined feature set of *basic + shadow + context + map prior* obtains a score of 88.94% in UM scenes, which is 0.08% below the *basic + context* combination. Nevertheless, on average, the full set of features increases performance by 3.64%.
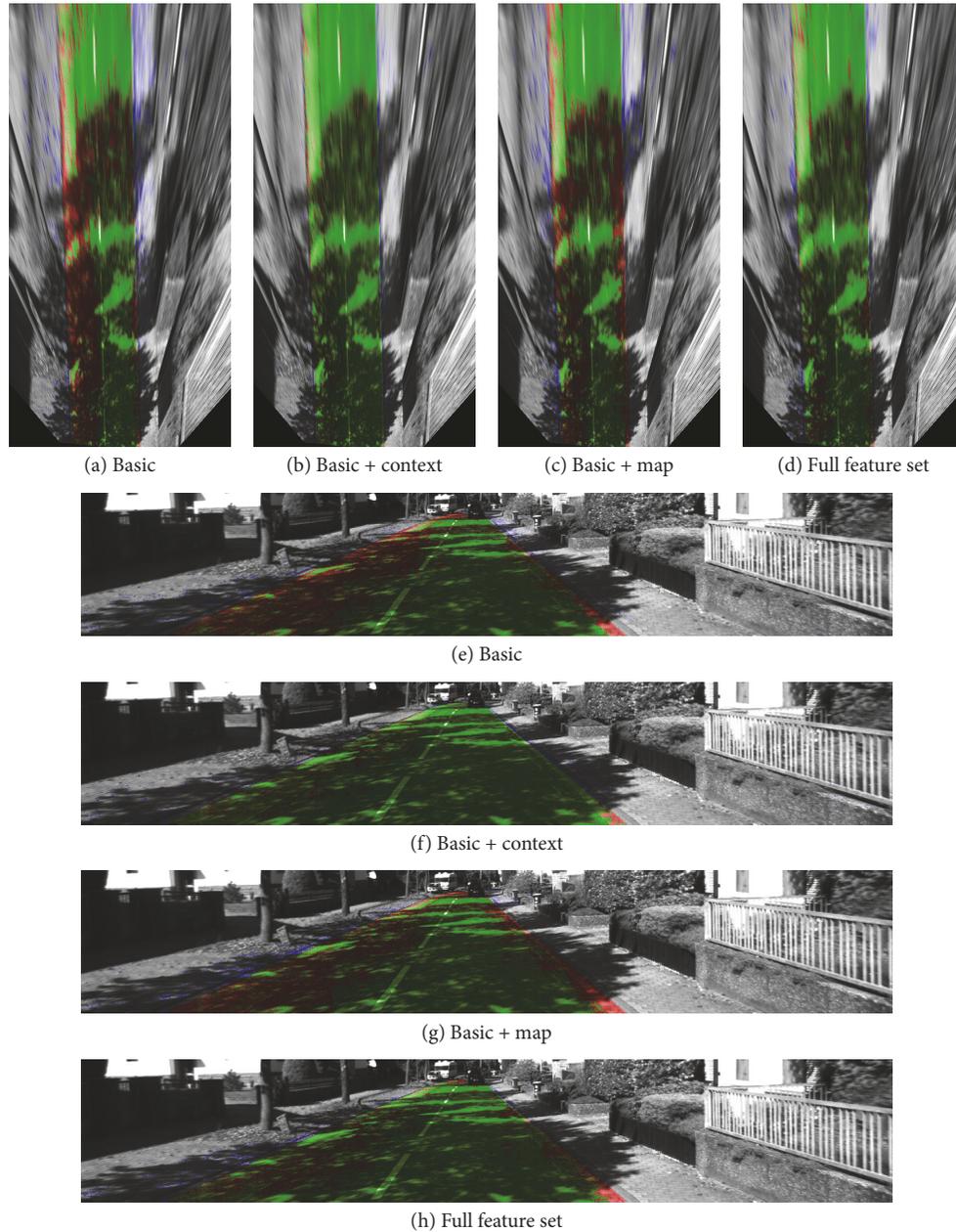
The proposed method is not limited to straight roads since the shape of the road is extracted from a digital navigation map. In scenarios having very narrow curves, the vanishing point detection could fail but the shape extracted from the map, the geometric-based features, and the appearance-based features mitigate the failure of the vanishing point based feature (see Figure 17).

Some qualitative and quantitative results are shown in the image plane and BEV in Figures 18, 19, and 20 on the UM, UMM, and UU scenes, respectively. In Figure 18, the *basic + context* combination fits very well the real shape of the road in UM. Consequently, when the map prior is included in the feature set, the classifier interprets some of the prior information as noise or unreliable data in the final response. In scenes with multiple marked lanes (see Figure 19) the *map prior* is very important since at times the *context feature*

does not provide reliable information. Finally, in Figure 20 all of the feature combinations have good results. However, the *basic + context* combination obtains fewer false positives, especially at far distances, which is very important in the BEV evaluation.

5. Conclusions

This paper presents a novel method to detect free space. The main contributions have taken place in the field of new features development. An original method for curb detection that is based on curvature estimation improves other state-of-the-art algorithms. Others detect curbs, ranging in height from 5 cm to 10 meters; however, the proposed method does not require any parameter adjustment and it is able to detect a wide range of curbs. The minimum curb height required for a precise detection is approximately 3 cm and the detection distance is up to 20 meters when the curbs are connected in the curvature image. Otherwise a small object of 3 cm at 20 meters distance will be filtered. A novel method is presented to convert features that describe road limits into a new feature that describes road areas. Instead of creating a set of radial rays from the bottom of the image, as seen in other methods in the literature, our method presents a new approach that uses the vanishing point to create a set of radial rays that fits the road limits. This new approach improves road classifier performance, especially in urban marked scenes. Another contribution of this paper is the creation of a new way to update digital navigation maps. The innovation aims to update information on road width. The system takes the number of lanes and the road type from the digital navigation map and returns the road width. The map can be updated from several vehicles, creating a robust value of the road width. The map with the road width is used to generate a prior of the current road structure, which is very useful in intersections and narrow streets. The use of a digital navigation map together with a feature-based road detection method makes the system more robust as compared to HD-map approaches which rely only on maps.

(a) Basic　　　　　　(b) Basic + context　　　　　(c) Basic + map　　　　　(d) Full feature set

(e) Basic

(f) Basic + context

(g) Basic + map

(h) Full feature set

Feature set	Basic	Basic + context	Basic + map	Full set
BEV	73.01	94.36	77.01	91.40
Image plane	77.86	93.91	78.07	88.22

FIGURE 18: Results of unary terms in a specific UM image. The combination *basic + context* fits very well the real shape of the road in UM. Consequently, when the map prior is included in the feature set, the classifier interprets part of the prior information as noise or unreliable data in the final response.

6. Future Work

From our results and conclusions, several future lines for each treated topic may arise. They correspond to aspects that have yet to be solved or that require further analysis to improve the system's performance. In order to take advantage of on board sensors, a sensor fusion should be implemented. In multisensor approaches, redundancy is quite important. The autonomous vehicle includes RADAR, LIDAR, GPS, stereo vision, and color camera sensors. Each sensor is strong in certain situations and weak in others; therefore, the redundancy aims to obtain a robust system that can work in a

(a) Basic   (b) Basic + context   (c) Basic + map   (d) Full feature set

(e) Basic

(f) Basic + context

(g) Basic + map

(h) Full feature set

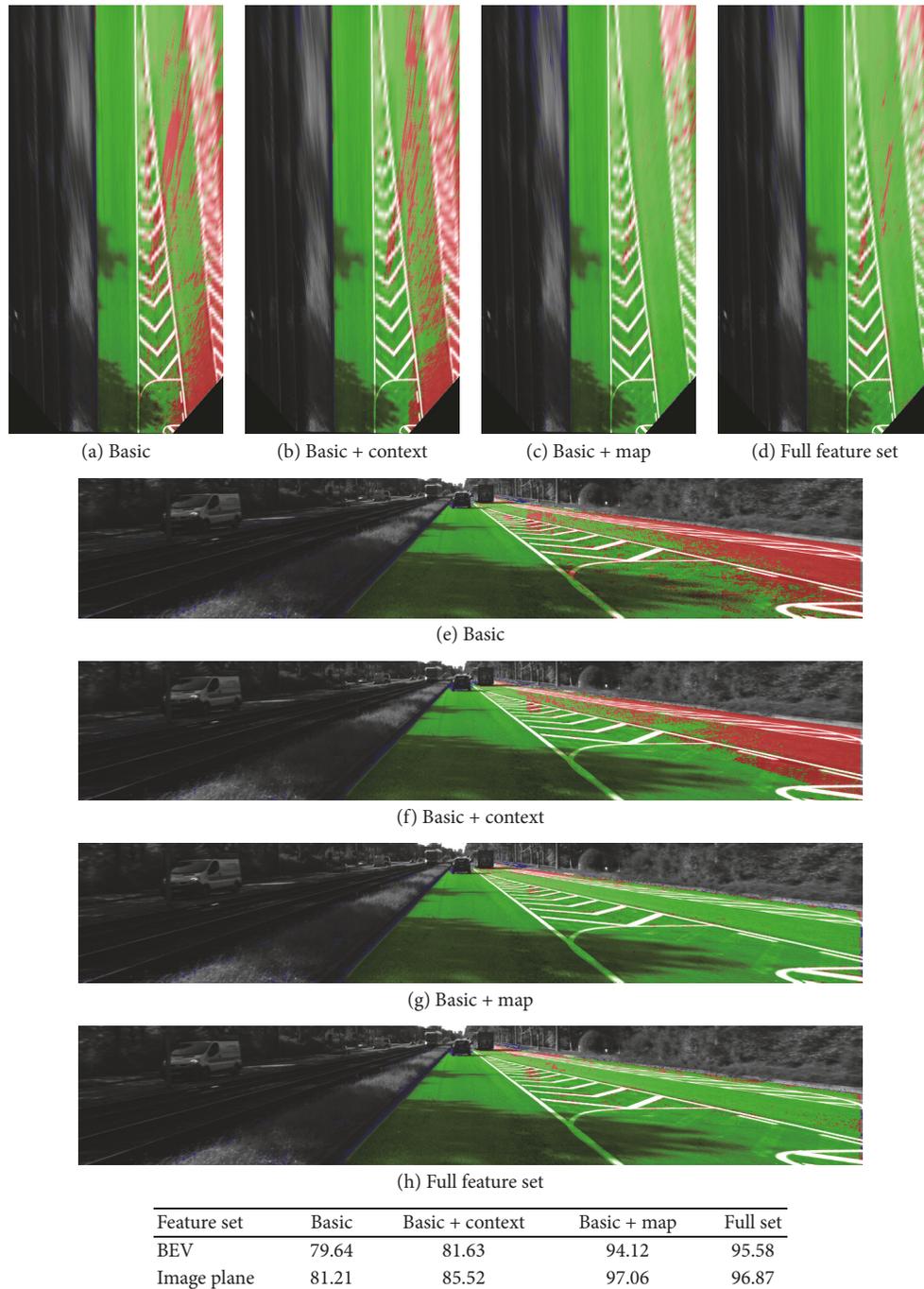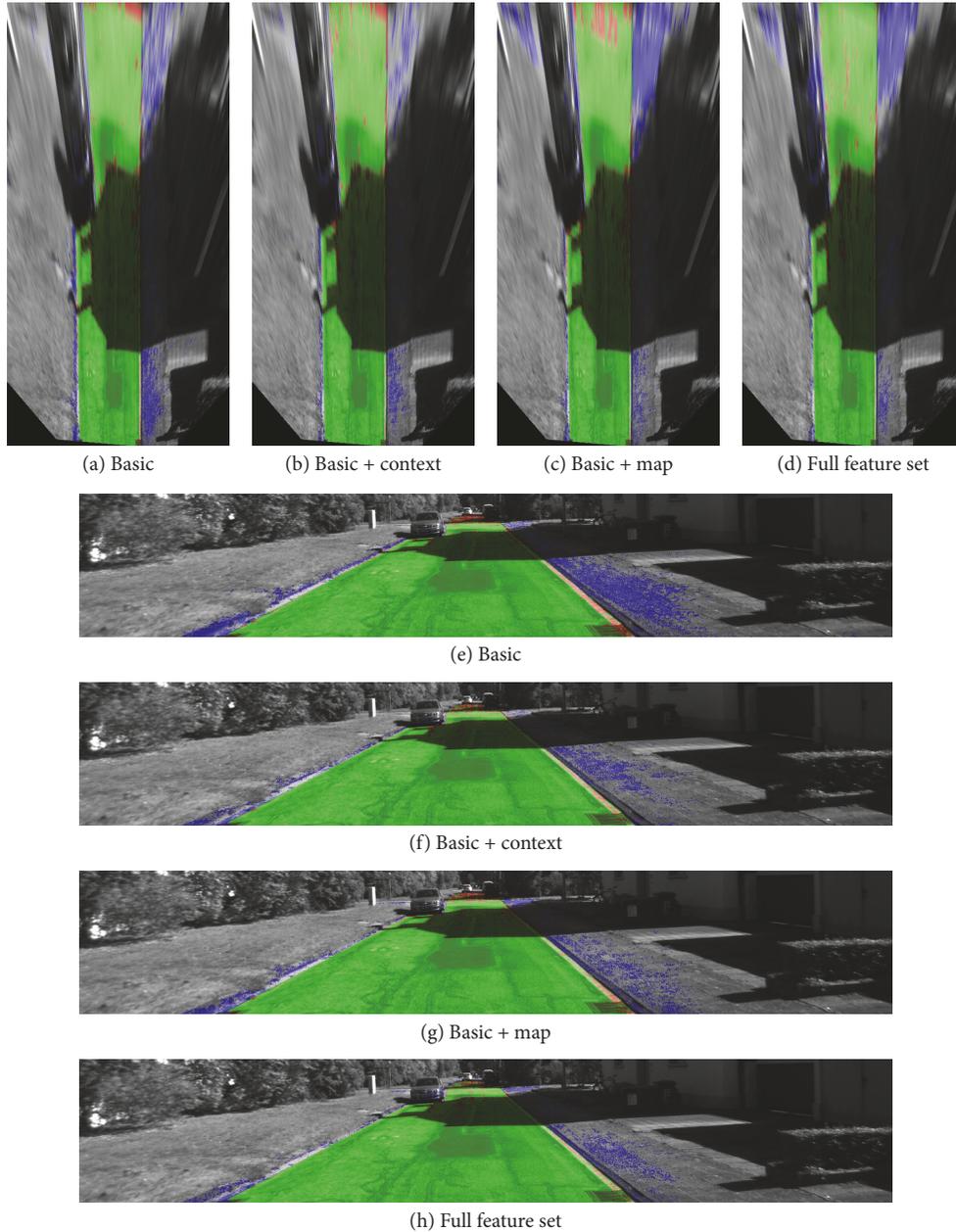| Feature set	Basic	Basic + context	Basic + map	Full set
BEV	79.64	81.63	94.12	95.58
Image plane	81.21	85.52	97.06	96.87

FIGURE 19: Results of unary terms in a specific UMM image. The *map prior* is very important because sometimes the *context feature* does not provide reliable information. The use of the *map prior* in scenarios with multiple marked lanes improves significantly the performance of the basic feature set.

variety of situations. Convolutional Neural Networks (CNN) outperform the state of the art in semantic segmentation problems. However, they should be trained with similar scenes. Our system will be integrated with a CNN classifier to increase its robustness in situations where the CNN has not been trained. Context interpretation has demonstrated an important role in the road detection problem. So, a special effort will be devoted to the improvement of this high level analysis. Due to the similarity between consecutive frames, the previous frame will be integrated as another new feature of the presented feature vector and the whole system shall be evaluated with the new feature.

(a) Basic — (b) Basic + context — (c) Basic + map — (d) Full feature set

(e) Basic

(f) Basic + context

(g) Basic + map

(h) Full feature set

Feature set	Basic	Basic + context	Basic + map	Full set
BEV	88.30	91.94	85.04	87.09
Image plane	88.62	93.84	92.61	94.15

FIGURE 20: Results of unary terms in a specific UU image. All the features combinations have good results. However, the combination *basic + context* obtains less false positives, especially at far distances, which is very important in the BEV evaluation.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] J. Ziegler, P. Bender, M. Schreiber et al., "Making bertha drive-an autonomous journey on a historic route," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 2, pp. 8–20, 2014.

[2] P. Czerwionka, *A Three Dimensional Map Format for Autonomous Vehicles [M.S. thesis]*, Intelligent Systems and Robotics, Freie Universität Berlin, Berlin, Germany, 2014.

[3] D. K. Savitha and S. Rakshit, "Gaussian Mixture Model based road signature classification for robot navigation," in *Proceedings of the International Conference on "Emerging Trends in Robotics and Communication Technologies", INTERACT-2010*, pp. 230–233, December 2010.

[4] T.-C. Dong-Si, D. Guo, C. H. Yan, and S. H. Ong, "Robust extraction of shady roads for vision-based UGV navigation," in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pp. 3140–3145, September 2008.

[5] G. D. Finlayson, S. D. Hordley, C. Lu, and M. S. Drew, "On the removal of shadows from images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 59–68, 2006.

[6] J. Zhang and H.-H. Nagel, "Texture-based segmentation of road images," in *Proceedings of the Intelligent Vehicles '94 Symposium*, pp. 260–265, October 1994.

[7] C. Rasmussen, "Combining laser range, color, and texture cues for autonomous road following," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '02)*, vol. 4, pp. 4320–4325, 2002.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 770–778, July 2016.

[9] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[10] A. Parajuli, M. Celenk, and H. B. Riley, "Robust Lane Detection in Shadows and Low Illumination Conditions using Local Gradient Features," *Open Journal of Applied Sciences*, vol. 03, no. 01, pp. 68–74, 2013.

[11] A. Hata and D. Wolf, "Road marking detection using LIDAR reflective intensity data and its application to vehicle localization," in *Proceedings of the 2014 17th IEEE International Conference on Intelligent Transportation Systems, ITSC 2014*, pp. 584–589, October 2014.

[12] B. Mathibela, P. Newman, and I. Posner, "Reading the road: road marking classification and interpretation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2072–2081, 2015.

[13] C. Fernández, M. Gavilán, D. F. Llorca et al., "Free space and speed humps detection using lidar and vision for urban autonomous navigation," in *Proceedings of the 2012 IEEE Intelligent Vehicles Symposium, IV 2012*, pp. 698–703, June 2012.

[14] G. Zhao and J. Yuan, "Curb detection and tracking using 3D-LIDAR scanner," in *Proceedings of the 2012 19th IEEE International Conference on Image Processing, ICIP 2012*, pp. 437–440, October 2012.

[15] T. Chen, B. Dai, D. Liu, J. Song, and Z. Liu, "Velodyne-based curb detection up to 50 meters away," in *Proceedings of the IEEE Intelligent Vehicles Symposium, IV 2015*, pp. 241–248, July 2015.

[16] J. Siegemund, U. Franke, and W. Förstner, "A temporal filter approach for detection and reconstruction of curbs and road surfaces based on conditional random fields," in *Proceedings of the 2011 IEEE Intelligent Vehicles Symposium, IV'11*, pp. 637–642, June 2011.

[17] F. Oniga and S. Nedevschi, "Curb detection for driving assistance systems: A cubic spline-based approach," in *Proceedings of the 2011 IEEE Intelligent Vehicles Symposium, IV'11*, pp. 945–950, June 2011.

[18] W. S. Wijesoma, K. R. S. Kodagoda, A. P. Balasuriya, and E. K. Teoh, "Road edge and lane boundary detection using laser and vision," in *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1440–1445, November 2001.

[19] F. Oniga, S. Nedevschi, and M. M. Meinecke, "Curb detection based on a multi-frame persistence Map for Urban driving scenarios," in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems, ITSC 2008*, pp. 67–72, December 2008.

[20] C. Fernandez, D. F. Llorca, C. Stiller, and M. A. Sotelo, "Curvature-based curb detection method in urban environments using stereo and laser," in *Proceedings of the IEEE Intelligent Vehicles Symposium, IV 2015*, pp. 579–584, July 2015.

[21] M. Nieto and L. Salgado, "Simultaneous estimation of vanishing points and their converging lines using the em algorithm," *Pattern Recognition Letters*, vol. 32, no. 14, pp. 1691–1700, 2011.

[22] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 1, pp. 20–37, 2006.

[23] S. Nedevschi, R. Schmidt, T. Graf et al., "3D lane detection system based on stereovision," in *Proceedings of the Proceedings - 7th International IEEE Conference on Intelligent Transportation Systems, ITSC 2004*, pp. 161–166, October 2004.

[24] A. Wedel, H. Badino, C. Rabe, H. Loose, U. Franke, and D. Cremers, "B-spline modeling of road surfaces with an application to free-space estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 4, pp. 572–583, 2009.

[25] L. Bentabet, S. Jodouin, D. Ziou, and J. Vaillancourt, "Road vectors update using SAR imagery: a snake-based method," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 8, pp. 1785–1803, 2003.

[26] J. Beck and C. Stiller, "Non-parametric lane estimation in urban environments," in *Proceedings of the 25th IEEE Intelligent Vehicles Symposium, IV 2014*, pp. 43–48, June 2014.

[27] A. V. Nefian and G. R. Bradski, "Detection of drivable corridors for off-road autonomous navigation," in *Proceedings of the 2006 IEEE International Conference on Image Processing, ICIP 2006*, pp. 3025–3028, October 2006.

[28] F. Moosmann and C. Stiller, "Velodyne SLAM," in *Proceedings of the 2011 IEEE Intelligent Vehicles Symposium, IV'11*, pp. 393–398, June 2011.

[29] R. Valencia, E. H. Teniente, E. Trulls, and J. Andrade-Cetto, "3D mapping for urban service robots," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '09)*, pp. 3076–3081, IEEE, October 2009.

[30] P. Pfaff, R. Triebel, C. Stachniss, P. Lamon, W. Burgard, and R. Siegwart, "Towards mapping of cities," in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation, ICRA'07*, pp. 4807–4813, April 2007.

[31] J. M. Alvarez, F. Lumbreras, T. Gevers, and A. M. López, "Geographic information for vision-based road detection," in *Proceedings of the 2010 IEEE Intelligent Vehicles Symposium, IV 2010*, pp. 621–626, June 2010.

[32] S. Zhou, J. Gong, G. Xiong, H. Chen, and K. Iagnemma, "Road detection using support vector machine based on online learning and evaluation," in *Proceedings of the 2010 IEEE Intelligent Vehicles Symposium, IV 2010*, pp. 256–261, June 2010.

[33] C. C. T. Mendes, V. Frémont, and D. F. Wolf, "Vision-based road detection using contextual blocks," https://arxiv.org/abs/1509.01122.

[34] G. B. Vitor, A. C. Victorino, and J. V. Ferreira, "A probabilistic distribution approach for the classification of urban roads in complex environments," in *Proceedings of the Workshop on IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[35] G. B. Vitor, A. C. Victorino, and J. V. Ferreira, "Comprehensive performance analysis of road detection algorithms using the common urban Kitti-road benchmark," in *Proceedings of the 25th IEEE Intelligent Vehicles Symposium, IV 2014*, pp. 19–24, June 2014.

[36] X. Chen, K. Kundu, Y. Zhu et al., "3D object proposals for accurate object class detection," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 424–432, 2015.

[37] C. C. T. Mendes, V. Fremont, and D. F. Wolf, "Exploiting fully convolutional neural networks for fast road detection," in *Proceedings of the 2016 IEEE International Conference on Robotics and Automation, ICRA 2016*, pp. 3174–3179, May 2016.

[38] J. M. Álvarez, A. López, and R. Baldrich, "Illuminant-invariant model-based road segmentation," in *Proceedings of the 2008 IEEE Intelligent Vehicles Symposium, IV*, pp. 1175–1180, June 2008.

[39] J. M. Álvarez and A. M. Ĺopez, "Road detection based on illuminant invariance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 184–193, 2011.

[40] G. D. Finlayson and M. S. Drew, "4-sensor camera calibration for image representation invariant to shading, shadows, lighting, and specularities," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2, pp. 473–480, 2001.

[41] R. B. Rusu, *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments [Ph.D. dissertation]*, Technische Universität München, 2009.

[42] P. Foucher, Y. Sebsadji, J.-P. Tarel, P. Charbonnier, and P. Nicolle, "Detection and recognition of urban road markings using images," in *Proceedings of the 14th IEEE International Intelligent Transportation Systems Conference, ITSC 2011*, pp. 1747–1752, October 2011.

[43] C. Fernandez, R. Izquierdo, D. F. Llorca, and M. A. Sotelo, "A Comparative Analysis of Decision Trees Based Classifiers for Road Detection in Urban Environments," in *Proceedings of the 18th IEEE International Conference on Intelligent Transportation Systems, ITSC 2015*, pp. 719–724, September 2015.

[44] M. Nieto, *Detection and tracking of vanishing points in dynamic environments [Ph.D. dissertation]*, Universidad Politécnica de Madrid (UPM), 2010.

[45] A. Torralba, K. P. Murphy, and W. T. Freeman, "Contextual models for object detection using boosted random fields," in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou, Eds., pp. 1401–1408, MIT Press, Cambridge, MA, USA, 2005.

[46] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 3354–3361, June 2012.

[47] J. Fritsch, T. Kuhnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC '13)*, pp. 1693–1700, IEEE, The Hague, The Netherlands, October 2013.

Journal of
Engineering

The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

International Journal of
Distributed
Sensor Networks

Advances in
Civil Engineering

Journal of
Control Science
and Engineering

Journal of
Robotics

Journal of
Electrical and Computer
Engineering
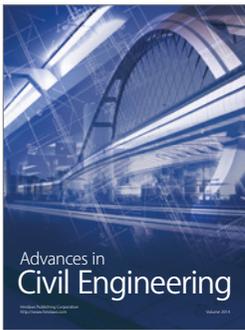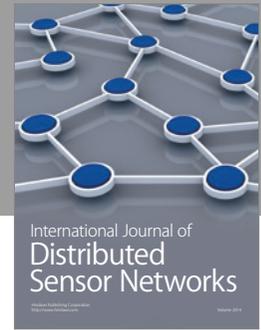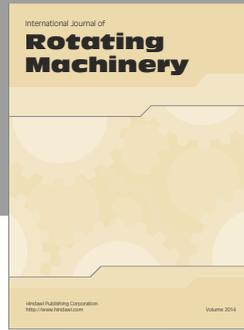
Submit your manuscripts at
https://www.hindawi.com

Advances in
OptoElectronics

VLSI Design

International Journal of
Navigation and
Observation

Modelling &
Simulation
in Engineering

International Journal of
Aerospace
Engineering

International Journal of
Chemical Engineering

International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components

Shock and Vibration

Advances in
Acoustics and Vibration