ON-BOARD VIDEO BASED SYSTEM FOR ROBUST ROAD MODELING

Marcos Nieto, Jon Arróspide, Luis Salgado, and Fernando Jaureguizar

Grupo de Tratamiento de Imágenes - E. T. S. Ing. Telecomunicación
Universidad Politécnica de Madrid - Madrid (Spain)
{mnd, jal, lsa, fjn}@gti.ssr.upm.es

Special Session VTMIA

ABSTRACT

In this paper, a novel road modeling strategy is proposed, defining an accurate and robust system that operates in real-time. The strategy aims to find a trade-off between computational requirements of real systems and accuracy and robustness of the results. The basis of the strategy is an adaptive road segmentation technique which ensures robust detections of lane markings and vehicles. A multiple lane model of the road is obtained by asserting hypotheses of lanes geometry based on perspective analysis and stochastic filtering. This multiple lane approach significantly improves vehicle location compared to other video-based works, as detected vehicles are accurately located within lanes.

Tests show the adaptability, robustness and accuracy of the system in daylight situations with severe illumination changes, non-homogeneous color of the pavement of the road, lane markings occlusions, shadows, variable traffic conditions, etc., performing in real-time in all cases.

1. INTRODUCTION

Video-based systems have shown their relevance within Driver Assistance Systems (DAS) compared with other systems, such as radar, ladar, etc. as they provide rich information about lanes apart from detecting vehicles. This leads to a number of services such as lane departure warning, lane deviation alert, collision prevention, etc. Moreover, they have become extremely interesting for the automotive industry due to their low cost and high fidelity results [1].

Two main requirements may be defined for any DAS: (i) real-time performance, so that the driver is informed by the system in time, i.e. with enough time to react; and (ii) quality of the information provided, which is directly defined by the accuracy of the results obtained by the system, and their robustness against real road situations.

As far as the latest requisite is concerned, real solutions have to consider that the on-board scenario is extremely variable, e.g. the color of the road may vary dramatically along the video sequence; lane markings may be visible or not, its contrast with the road may also vary due to road surface conditions, presence of oils, shadows, etc. Moreover, weather conditions affect these intensity levels, even in a small time window, e.g. when a cloud temporarily occludes the sun and the road illumination suddenly changes. The complexity of the problem is stressed due to the dynamic environment from which images are taken: the on-board scenario involves a moving vehicle capturing images of the road, which gathers both static elements, such as the lane markings, and dynamic elements, namely, other vehicles driving at different speeds.

Many works in the literature offer solutions to specific problems of this on-board scenario. Very accurate models of the lane exist, obtained by fitting curves such as clothoids [2][3], or splines [4], or by using 3D detections of lane markings [5]. On the other hand, vehicle detection strategies typically make use of stereo systems [6][7], although monocular systems offer similar results with simpler strategies [8][9].

Although some of these works focus on robustness or accuracy, there has not been an explicit effort to find a trade-off between real-time performance and quality in the results. In this paper, a novel road modeling strategy is proposed, which aims to find this trade-off through the use of a robust but simple segmentation strategy that ensures accurate detections of lane markings and vehicles for the further model fitting and vehicle tracking sub-modules. It uses a statistical modeling of the road based on Bayesian decision theory which dynamically adapts to the variations of its appearance, due to illumination changes, alteration of road elements contrast, occlusions between elements, etc.

Real-time performance is achieved by the simplification of the model fitting step, which is carried out in a corrected perspective domain, where circumference arc models are used for the lane markings. The road model is completed with an innovative multiple lane detection strategy. Based on the detected geometry of the own lane, adjacent lanes are hypothe-

This work has been partially supported by the European Commission 6th Framework Program under project IST-2004-027195 (I-WAY). This work is also supported by the Comunidad de Madrid under project P-TIC-0223-0505 (PRO-MULTIDIS) and by the Ministerio de Educación y Ciencia of the Spanish Government under project TEC2007-67764 (SmartVision).

978-1-4244-2044-5/08/$25.00 ©2008 IEEE.

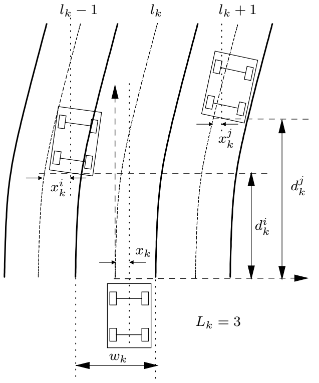


Fig. 1. On-board scenario. Model of the lanes, own vehicle and other vehicles. In this example, the number of detected lanes is $L_k = 3$.

sized, giving the capability of locating vehicles within lanes, and thus allowing to classify vehicles as potential obstacles not only by their distance to the own vehicle, but also by their relative lane position.

2. OVERVIEW

The proposed method is a road modeling strategy that takes as input the images captured by a forward looking camera mounted inside the vehicle, near the rear-mirror. The output of the system is a set of parameters that may be of great interest for DAS, such as the horizontal position of the vehicle within its lane (the own lane), the instantaneous width of the lanes and the location of other vehicles on the road. Fig. 1 depicts these parameters within the considered model of the road. All these output data are given for each frame, with time index k, working in real-time. Temporal stochastic filtering gives temporal coherence to results, improving its robustness.

The number of detected lanes is L_k, labeling the own lane as l_k and adjacent lanes as $l_k \pm n$, where n is the number of lanes at left or right. The width of the own lane is identified as w_k. The position of the own vehicle within its lane is represented by x_k, while the relative position of other vehicle, indexed by i, is represented by its horizontal position within its lane, x_k^i, the relative distance d_k^i to the own vehicle, and the relative lane index, l_k^i.

The block diagram of the whole system is depicted in Fig. 2. From this diagram, the two main pillars of the system are the perspective analysis and the adaptive segmentation, which are used by the consecutive three sub-modules: own lane modeling, multiple lane modeling, and vehicles location.

The stretch of road ahead the vehicle is assumed to be flat, which is a typical approach [6]. This way, a perspective analysis of the vanishing point, such as in [10] allows to generate a perspective corrected representation of the road, i.e. an image displaying a bird-view of the road plane. All computations are carried out in this domain as far as models are simplified and dynamics of vehicles are linearized [10]. Basically, the perspective analysis lies in obtaining the homography matrix, H, between the image plane and the road-plane, so that points in the image coordinate system, $\mathbf{x}_i$, are directly transformed into road-plane coordinates $\mathbf{x}_r$ as $\mathbf{x}_r = \mathrm{H}\mathbf{x}_i$. In the following sections, only the road-plane coordinate system is considered, hence the r subscript is removed for the sake of clarity.

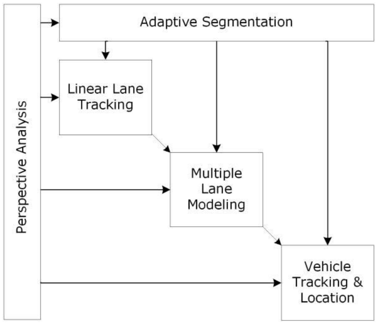


Fig. 2. Block diagram of the proposed Road Modeling system.

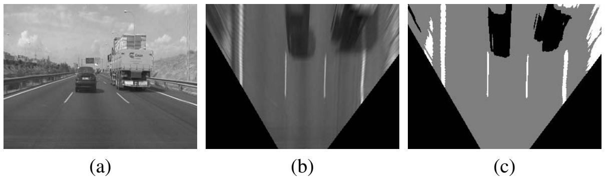
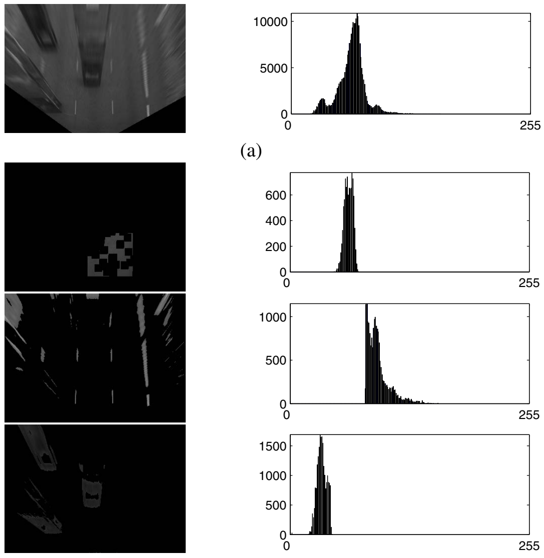


Fig. 3. Segmentation model: (a) image-plane; (b) road-plane; and (c) three-levels segmentation.

3. ROAD-PLANE IMAGE SEGMENTATION

The segmentation of the road considers that pixels belong to three types of elements within any road-plane image: pavement ("gray"), lane markings ("white"), and dark objects ("black"). Fig. 3 shows the resulting three-levels segmentation for an example image. As shown, pixels belonging to lane markings are painted in white, dark objects, mainly the wheels and shadows of vehicles are painted in black, while those belonging to the pavement are painted in gray

The objective of the segmentation is to provide a first separation of these elements, in a simple and fast way, that feed the consecutive modules, which make use of it, removing noise according to more complex assumptions of each type of element. This way, white pixels are used to model the curvature of the own lane, gray pixels allow to hypothesize the presence of adjacent lanes and black pixels are used to detect and track other vehicles. The segmentation algorithm is based on the Bayesian decision theory, defining a parametric Gaussian model of the road, from which pixels are univocaly



(b)

Fig. 4. Modified Histogram strategy: (a) Histogram of the road-plane image; and (b) the three separate histograms for gray, white and black regions.

classified, with an associated probability of error.

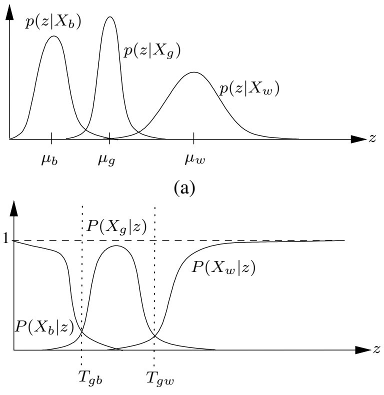
Confidence measures about the correct performance of the segmentation are also provided to further modules, which make use of them to generate confidence measures about lanes modeling.

3.1. Parametric Gaussian Model

In an ideal situation, with white lane markings, black objects and gray pavement, the histogram of the scene would be three deltas scaled by the number of pixels belonging to each category. Adding Gaussian noise due to the acquisition process transform the data into three scaled Gaussians. In a real situation, taking into account not only noise, but all the abovementioned problems of road scenes, these Gaussians are modified and distorted into a non-deterministic set of data, conforming histograms as the one shown in Fig. 4 (a).

The three Gaussian model is defined, for each frame, by the mean and standard deviation of each of the three Gaussians, which are denoted by the subscripts $\mathcal{S} = \{b, g, w\}$ respectively for the "black", "gray" and "white" Gaussians.

The problem is to assign one of these categories to each pixel z of the image at each time instant (the time subscript k is removed for the sake of clarity). Let X_i be the event that some pixel z is classified as belonging to the category $i \in \mathcal{S}$. Using the Bayesian decision theory, this classification is carried out by assigning to each pixel the category that maximizes the a posteriori conditional probability, $P(X_i|z)$, which is decomposed, by the Bayes' rule as:



(b)

Fig. 5. Parametric Gaussian Model of the road: (a) Likelihood distributions for the three defined categories; and (b) resulting posterior probabilities and the associated thresholds.

$$P(X_i|z) = \frac{p(z|X_i)P(X_i)}{p(z)} \tag{1}$$

where $p(z|X_i)$ is the likelihood distribution, that depicts the probability of a pixel belonging to a given category; $P(X_i)$ is the a priori probability of each category and $p(z)$ is the evidence, a scale factor that ensures that the posteriors sum one. It is computed as $p(z) = \sum_{i \in \mathcal{S}} p(z|X_i)P(X_i)$.

This way, each pixel is classified as the category with maximum a posteriori probability. The likelihood and the a priori probabilities are computed as described in the following subsections.

3.1.1. Likelihood distribution

The likelihood distributions are modeled as Gaussian distributions, with mean and variance extracted from the histogram of the images: $p(z|X_i) = \mathcal{N}(z|\mu_i, \sigma_i^2)$. These distributions compose the three Gaussian model, depicted in Fig. 5 (a).

The proposed algorithm separates the image into three regions, according to the a priori probability of each category so that three separate histograms are obtained. Although these histograms may not completely fit with the target Gaussian distributions, its time filtering through a Kalman filter allows to estimate very accurately these underlying distributions. from pavement and shadow pixels.

The gray mode is easy to measure by analyzing a region of interest of the lower part of the road-plane image considering only low gradient pixels (the road is expected to be more homogeneous than the lane markings or dark regions). Fig. 4 (b) shows, in the upper image, this region of interest, where high gradient pixels belonging to lane markings have been removed. Its corresponding histogram is clearly unimodal and can be modeled as $p(z|X_g) = \mathcal{N}(z|\mu_g, \sigma_g^2)$.

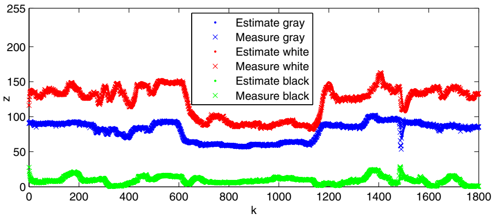


Fig. 6. Variation of black, gray and white mean values along time in an example sequence.

The white and black regions are then extracted by thresholding the road-plane image with thresholds $\mu_g \pm 3\sigma_g$, respectively. The resulting images are shown in Fig. 4 (b) central and bottom images. As it can be observed, these images contain pixels that belong to lane markings and shadows, respectively, but also may contain some pixels belonging to the pavement. Therefore, their histograms are not always unimodal. Nevertheless the means and standard deviations, $\{\mu_b, \sigma_b\}$ and $\{\mu_w, \sigma_w\}$ are used as instantaneous measurements that after the filtering provide accurate enough Gaussian distribution models.

To make these results more robust to outliers, a correction step is applied to these measures. Given that the most reliable mode is the gray one, additional measures about the mean of the other modes are generated. For both the black and white modes, their means, μ_b and μ_w are corrected as follows:

$$\mu_b^* = \alpha_b \mu_b + (1 - \alpha_b)\mu_{gb} \tag{2}$$

$$\mu_w^* = \alpha_w \mu_w + (1 - \alpha_w)\mu_{gw} \tag{3}$$

where $\mu_{gb} = \mu_g - \bar{d}_{gb}$ and $\mu_{gw} = \mu_g + \bar{d}_{gw}$ are the additional measures generated from the gray mode, where $\bar{d}_{gb}$ and $\bar{d}_{gw}$ are the average distances between gray and black modes, and gray and white modes, respectively, in the last m frames. The weighting parameters α_b and α_w are computed according to the distance between the measured mean values and the additional measures:

$$\alpha_b = \exp(-\tau \|\mu_b - \mu_{gb}\|^2) \tag{4}$$

$$\alpha_w = \exp(-\tau \|\mu_w - \mu_{gw}\|^2) \tag{5}$$

where τ is used to tune the weighting factors.

As a result, a three Gaussian model is generated at each time, parameterized with the means and standard deviations of the gray, white, and black modes, corresponding to the pavement, the lane markings and the shadows of the road-plane image.

A Kalman filter is used to estimate the model according to the instantaneous measures taken as explained above. The state vector is composed as $\mathrm{s}_k = (\{\mu_{i,k}, \sigma_{i,k}, \dot{\mu}_{i,k}, \dot{\sigma}_{i,k}\}_{i \in \mathcal{S}})^\mathsf{T}$, where $\dot{(\cdot)}$ means first derivative. This way, both the mean and standard deviation are estimated and their variations along time are used to predict the state of the model for the next time instant. Fig. 6 shows an example of the estimation of the model along time. Each colored curve corresponds to the variation of the mean of each mode for an example sequence. Observe that there are very significant variations in the intensity levels, mainly due to illumination changes, as for instance, between images 600 and 700, where the color of the road and lane markings change dramatically. The segmentation automatically adapts to these changes and allows to set thresholds that clearly separate lane markings.

3.1.2. Prior probabilities

The a priori probability of each of the defined categories has to be computed in order to obtain the final a posterior probability for each pixel of the image. This prior probabilities depict the prior knowledge of how likely a pixel belongs to a category before analyzing its intensity value. This information is hard to obtain in a reliable way. As previously stated, the road scene is so varying that any fixed estimation of prior probabilities may likely fail in any situation. For that reason, in this work, the prior probabilities are estimated once the pixels have been analyzed and the likelihood distributions composed. Using this information, the prior probabilities are computed as:

$$P(X_i) = \frac{N_i}{\sum_{j \in \mathcal{S}} N_j} \tag{6}$$

where N_i is the number of pixels likely belonging to each category. For the gray mode, N_g is the number of pixels within the image whose intensity value is comprised between $\mu_g \pm 3\sigma_g$. For the black mode, N_b is the number of pixels with values lower than $\mu_g - 3\sigma_g$, while the N_w is the number of pixels with values higher than $\mu_g + 3\sigma_g$.

3.2. Thresholding

The posterior probabilities of each pixel are then obtained as in (1). Fig. 5 (b) shows an example of the representation of the posterior probabilities for the range of intensity pixel values z. The figure also depicts the two inter-categories thresholds, T_{gb} and T_{gw}, which separate the regions of pixel intensity values where each category possess a posterior probability higher than the other categories. These thresholds ensure the minimum average segmentation error [11], given by $E(T_{gb}) = P(X_g)E_{gb} + P(X_b)E_{bg}$, where:

$$E_{gb} = \int_{-\infty}^{T_{gb}} p(z|X_g)dz, \;\; E_{bg} = \int_{T_{gb}}^{\infty} p(z|X_b)dz \tag{7}$$

Analogous expressions are used obtained for the pair "gray-white". Minimizing these errors leads [11] to a quadratic solution expressed by $AT^2 + BT + C = 0$ with:

$$\begin{aligned} A &= \sigma_b^2 - \sigma_g^2 && (8) \\ B &= 2(\mu_b \sigma_g^2 - \mu_g \sigma_b^2) && (9) \\ C &= \sigma_b^2 \mu_g^2 - \sigma_g^2 \mu_b^2 + 2\sigma_b^2 2\sigma_g^2 \ln\left(\frac{\sigma_g P(X_b)}{\sigma_b P(X_g)}\right) && (10) \end{aligned}$$

for the case of “gray-black” and analogous values for the case “gray-white”. This equation gives two solutions, but only the one between the two involved modes is of our interest.

The final segmentation, as the one shown in Fig. 3 (c), is obtained by thresholding the road plane with these thresholds.

3.3. Confidence measures

As shown in Fig. 6, the situation of the road makes the model vary. In some cases, the three Gaussians are clearly separated, corresponding to images where the contrast between lane markings, the road and the shadows is large enough to provide reliable estimations of the thresholds. Nevertheless, in other periods, this contrast decreases, and the reliability of the system also decreases. This quality assessment is provided to further modules by giving a global quality measurement of the segmentation, according to the distance between the Gaussians:

$$Q = \prod_{i \in \{\mathcal{S}-g\}} (1 - \exp(-\gamma \bar{d}_{gi})) \qquad (11)$$

where $\bar{d}_{gi} = \{\bar{d}_{gb}, \bar{d}_{gw}\}$, and γ is chosen so as the exponential has a negligible value when the distance is maximum.

4. LANE TRACKING AND MODEL FITTING

A new zoomed road-plane image containing information of the very near stretch of road is created. Fig. 7 shows the relationship between the image plane, the road-plane, and this new zoomed road-plane. As it is shown, the zoomed road-plane contains only the very lower part of the original image plane. Within this image, only the lane markings belonging to the own lane are displayed, while the presence of vehicles and other lanes are dramatically reduced. Also, lane markings can be modeled by straight lines by assuming that the vehicle moves always almost parallel to lane markings.

Lane markings are well defined by the pixels whose intensity value is higher than T_{gw}, i.e. whose posterior probability $p(X_w|z)$ is higher than the posterior of the other categories. The zoomed image is accordingly thresholded setting to zero all pixels but those belonging to the lane markings.

4.1. Lane Tracking

The lane tracker is the sub-module that analyzes the evolution of these images and determines the width of the own lane, w_k, and the position of the vehicle within it, x_k, as depicted in Fig. 1. Lane changes are also detected by analyzing the evolution of x_k and w_k.

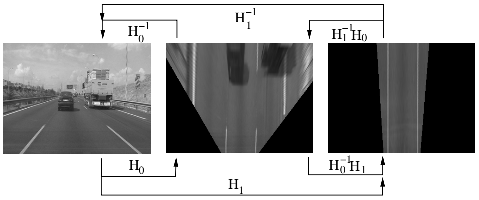


Fig. 7. Image-plane to road-plane transforms: H_0 image-plane to road-plane; H_1 image-plane to zoomed road-plane.

The problem is easily described as a dynamic linear system, and solved with a Kalman filter defined by the following state-space equation:

$$\mathbf{x}_k = \mathrm{A}\mathbf{x}_{k-1} + \mathrm{B}\mathbf{u}_k + \mathbf{w}_k \qquad (12)$$

where the state vector is $\mathbf{x}_k = (x_k, w_k, \dot{x}_k, \dot{w}_k)^\mathsf{T}$. The measurement vector is, at each instant, $\mathbf{z}_k = (x_k, w_k)^\mathsf{T}$. The transition matrix and the input control matrix are:

$$\mathrm{A} = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix};\ \mathrm{B} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad (13)$$

The control matrix is used to modify the estimation of the horizontal position of the vehicle when lane changes are detected. The input control vector is obtained as:

$$\mathbf{u}_k = \begin{cases} w_k & \text{if } \ x_k > \frac{1}{2}(W + w_k) \\ -w_k & \text{if } \ x_k < \frac{1}{2}(W - w_k) \\ 0 & \text{elsewhere} \end{cases} \qquad (14)$$

where W is the width of the image in pixels; this way, when the horizontal position rises the boundary, at left or right, of the estimated lane, the input control is activated and the horizontal position is shifted.

The measurement vector is obtained using the Hough transform on the thresholded zoomed road-plane. Two straight lines model the two lane markings. These lines intersect with the lower boundary of the image in two points. The distance between these points is w_k, while the distance between their middle point and the center of the width of the image is x_k.

4.2. Model fitting

The road-plane image is also thresholded with T_{gw} and the estimated state-vector of the lane tracker is used (matching from the zoomed road-plane to road-plane) to cluster lane marking points and to fit a curve for each of the lane markings that define the own lane.

In this work, under the road-plane domain, the circumference arc curve shows enough accuracy whereas simplicity in its computation. The circumference is governed by the following equation:

$$x^2 + y^2 + 2Ax + 2By + C = 0 \tag{15}$$

where the center is the point $\mathbf{x}_c = (-A, -B)$, and the radius is $r = \sqrt{A^2 + B^2 - C}$. The coordinates (x_i, y_i) of each pixel belonging to the lane marking are then used to compose a over-determined system from which the estimated curve is obtained in a least squares error sense:

$$\begin{bmatrix} 2x_1 & 2y_1 & 1 \\ ... & ... & ... \\ 2x_n & 2y_n & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} -(x_1^2 + y_1^2) \\ ... \\ -(x_n^2 + y_n^2) \end{bmatrix} \tag{16}$$

where n is the number of points used to fit each curve. The curve is described by its two parameters $(\mathbf{x}_c, r)^\mathsf{T}$.

4.3. Multiple lanes

Once the own lane has been estimated, the adjacent lanes are hypothesized to be at w_k pixels at left and right of the own lane. Several adjacent lanes may be hypothesized this way, assuming a model defined by $(\mathbf{x}_c, r \pm n \cdot w_k)^\mathsf{T}$, where n indexes the number of hypothesized lane markings at left or right.

The verification of adjacent lanes is performed checking the percentage of pavement pixels contained at each hypothesized lane. These pavement pixels are those having values between the thresholds T_{gb} and T_{gw}. The probability of an hypothesized lane, indexed by l, to actually exist is given by:

$$P_l = \frac{Q}{N} \sum_{i}^{N_g} P(X_g|z) \tag{17}$$

where N is the number of pixels in the road-plane of the hypothesized lane and N_g is the number of pixels classified as "gray" (note that $N_g \leq N$). Any threshold may be applied to P_l to determine if the lane exists or not (e.g. lanes with $P_l > 0.5$ are declared as existing).

5. VEHICLE DETECTION

Vehicle detection, combined with the extracted lane model, allows us to locate the vehicle in the road, i.e., to know the lanes where the detected vehicles are being driven, and their relative position within the lane. The vehicle detection module is based on the segmentation described in Section 3. Within the road-plane domain, vehicles appear as dark areas, containing the shadow casted by the vehicle and its wheels. Fig. 8 (a) shows an example of the image created by thresholding the road-plane image so that pixels with intensity values lower than T_{gb} are shown as white, and the rest as black.

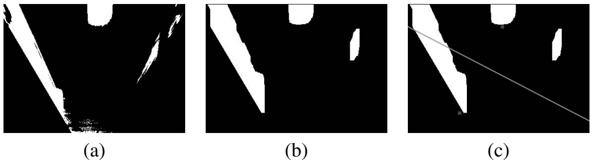


Fig. 8. Vehicle detection steps: (a) Pixels of road-plane with intensity lower than T_{gb}; (b) enhanced segmentation after morphological operations; and (c) clustering of blobs.

Connected regions within this binary image are taken to measure the position of vehicles within the road-plane at each frame. These measures are used to estimate the position of vehicles through time. However, the number of measures at each time may not match the number of objects because segmentation may produce several measures for each object. This way, the problem is to estimate the position of a variable number of objects from a different number of measures.

To address this problem, first an enhancement of the segmentation is carried out applying morphological operations, so that the fragmentation of measured objects is reduced. The image processing involves an opening morphological operation in order to enhance the relevant information, this is, the blobs of white pixels corresponding to vehicles. The initial erosion operation removes the noise introduced in the segmentation due to background clutter, shadows, etc, while the dilation restores the original appearance of the blobs. An additional dilation step is introduced using a vertical structuring element to fill possible black gaps within the blobs and therefore increase the compactness of the blobs. The result is an enhanced binary image, as shown in Fig. 8 (b), that is fed to the posterior processing steps.

Next, a blob coloring technique is performed, in order to identify the potential vehicles in the image. A perspective analysis is included at this point so as to introduce restrictions in the potential positions of the vehicles due to perspective effects. Namely, those measures that, when projected to the perspective plane, result in an occlusion by other measure, are removed. Then a temporal correlation is performed through a linear estimation using a Kalman filter with the following state vector:

$$v_k = (\{x_{k-1}^i, y_{k-1}^i, \dot{x}_{k-1}^i, \dot{y}_{k-1}^i\}_{i=1}^{n_{k-1}})^\mathsf{T} \tag{18}$$

where $\{x_{k-1}^i, y_{k-1}^i\}$ denotes the position of the vehicle i in the previous frame, and $\{\dot{x}_{k-1}^i, \dot{y}_{k-1}^i\}$ models its velocity. The prediction will consist of a set of n_{k-1} positions of the vehicles at time k, which will be compared to the actual measurements through a clustering technique.

The clustering divides the image space into n_{k-1} regions represented by the predictions, where every detected blob is assigned to the cluster with the nearest representative. Then, within each cluster, the blob that maximizes similarity (for instance, in terms of relative distance, area, etc.) with the rep-

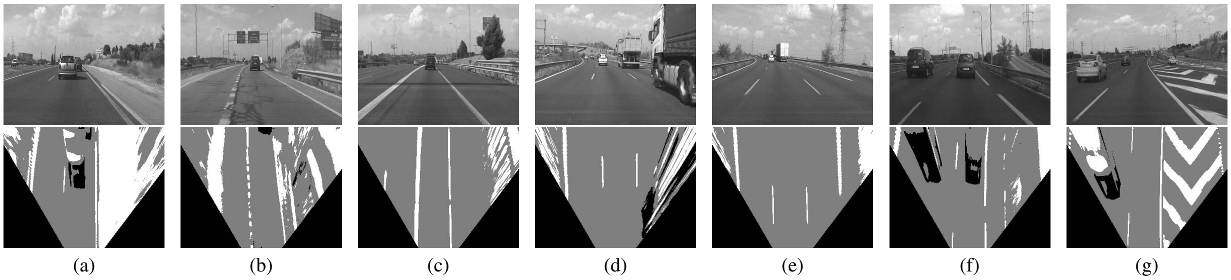


Fig. 9. Segmentation examples for a long test sequence.

resentative vehicle is selected. Fig. 8 (c) shows an example of a clustering step: red points depict the predicted positions of the objects detected in the previous image, while the green line separates the image into two regions containing pixels closest to each of these objects. In the upper cluster of the example, the correct blob is associated to its corresponding vehicle, discarding the small blob produced by noise.

The positions of the selected blobs are fed to the measurement vector in the updating phase of the Kalman filter, that corrects the predictions using these measurements, to yield an updated estimation of the positions.

A specific module is devoted to the management of vehicle entries and exits. An exit is detected whenever the position estimated with the Kalman filter exceeds the limits of the image. As for the entries, new blobs are sought in three predefined regions of interest of the image (uppermost, leftmost and rightmost), where new vehicles may appear. The vector of positions is updated according to the detected exits and entries.

Finally, these positions are merged with the road model to extract the lanes where the vehicles lie, and converted to relative positions within those lanes.

6. RESULTS

Test have been done on an integrated software system running in a general purpose PC, working at 2 GHz, with 1024 MB of RAM memory in a CoreDuo processor. The images resolution is 360×256 pixels and the frame-rate is 10 fps. All tests have been carried out while driving, recording the sequences, storing the results that are afterwards analyzed on-lab. The sequences have been recorded driving one hour through different motorways and highways in Madrid. The illumination conditions varied due to some clouds periodically occluding the sun, the type of road changed from new to old pavement, and the lane markings where, in some cases well painted, and in other cases almost non-visible.

Therefore, the images of these sequences show very different contrast between elements, homogeneous and heterogeneous roads, presence vehicles, shadows, etc., offering an excellent scenario to test the robustness of the proposed segmentation algorithm, which is the base of the whole on-board system. Fig. 9 shows several examples of the segmentation of the images in the road-plane domain. In all cases, the segmentation adapts the parametric Gaussian model to achieve the highest accuracy, showing clearly lane markings, dark objects, and the pavement. Particularly, case (a) shows a very accurate segmentation of the dark elements of the vehicle ahead. In other cases, such as (b) and (c), the road is very heterogeneous in color, containing shadows, or areas with darker pavement. The rest of examples show other situations, such as the presence of a heavy truck in (d), a very dark image due to a cloud occluding the sun in (f), or several lane marking paintings in (g). In all cases, the segmentation shown is correct. Notice that under this three-categories segmentation underlies a confidence analysis, giving a set of probabilities $p(X_i|z)$ which describe the reliability measure for each pixel, which is also used in the hypothesis of multiple lanes.

The variation of the confidence measure of the segmentation, Q, defined in (11), is shown in Fig. 10. The sequence is the same as used in Fig. 6. Observe the decrease of confidence when the illumination changes between frames 600 and 1200, corresponding to the stretch where the modes of the model get closer, and then the reliability of the system decreases.

As far as the lane tracker module is concerned, Fig. 11 shows some pictures of the online computing of the position of the vehicle within the own lane (depicted with a yellow segment), and the detected own lane. In (b) the width is significantly larger than in the other examples, due to the merging of two lanes after an entrance ramp to the motorway. The width of the lane and the position of the vehicle are shown, for a stretch of road, in Fig. 13. In the upper figure, this position is shown, depicting a vehicle driving quite centered in its lane, and with two lane changes, correctly detected, and shown in the bottom figure. The width is shown in the central figure. The motorway of the analyzed sequence has a very steady lane width except for a few meters, where two lanes fuse after an entrance ramp.

Finally, Fig. 12 shows three examples of vehicles detection, with different number of vehicles in the scene, correctly detected. The strategy has shown very accurate detections and robustness under different conditions (e.g. varying illumina-

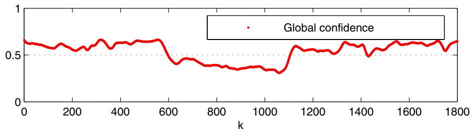


Fig. 10. Confidence level Q of the segmentation, computed for a long test sequence.

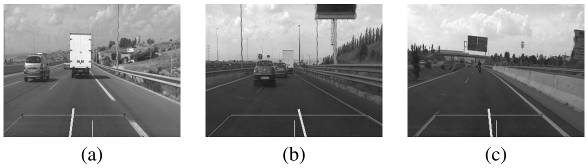

(a) (b) (c)

Fig. 11. Lane tracker examples. The own lane is depicted as the green box, with different widths at each image. The position of the own vehicle is depicted as the yellow line.

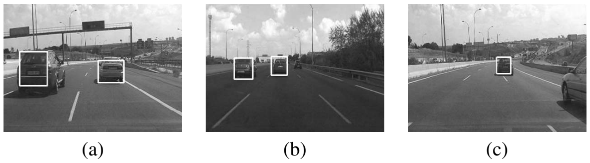

(a) (b) (c)

Fig. 12. Vehicle detection results.

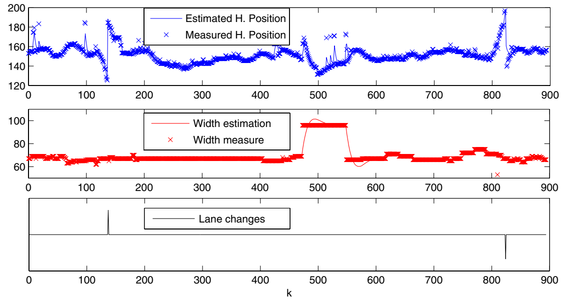


Fig. 13. Lane tracker output variables: (up) horizontal position of the vehicle (in image-plane coordinates); (middle) width of own lane; and (bottom) detected lane changes.

tion, different color of the road, etc.), as for instance, in case (b) the contrast between the road color and the shadows of the vehicles is very low.

7. CONCLUSIONS

Based on a robust and adaptive segmentation, using a parametric Gaussian model of the road, the on-board system described in this paper have shown excellent results in real on-road testing scenarios. The main advantage of the proposed strategy is its robustness against different illumination conditions, the variation of the color of the road, in terms of homogeneity of the pavement intensity level and its contrast with other elements, such as lane markings or vehicles. A complete confidence measurement framework is introduced, governing decisions taken by the system: multiple lanes are hypothesized even when their lane markings are not completely visible, based on a probabilistic analysis of the own-lane and the results of the segmentation. A very accurate vehicle detection strategy is carried out, working in the road-plane domain, hence linearizing the estimation of positions and simplifying the detection of appearing and disappearing vehicles.

The result is a road model that gathers information from the own-lane, such as the horizontal position of the vehicle within it, its widening, information about the number of lanes in the images, and finally locating vehicles within their own lanes, providing also measures of distance to the own-vehicle.

8. REFERENCES

[1] W. Liu, X. Wen, B. Duan, H. Yuan, and N. Wang, “Rear vehicle detection and tracking for lane change assist,” in *IEEE Proc. Intelligent Vehicle Symp.*, Istanbul, Turkey, June 13-15, 2007, pp. 252–257.

[2] C. Corridori and M. Zanin, “High curvature two-clothoid road model estimation,” in *Proc. Intelligent Transportation Systems*. IEEE, Washington, D.C., USA, October 3-6, 2004, pp. 630–636.

[3] Ben Southall and Camillo J. Taylor, “Stochastic road shape estimation,” in *ICCV*, 2001, pp. 205–212.

[4] Y. Wang, E. K. Teoh, and D. Shen, “Lane detection and tracking using b-snakes,” *Image and Vision Computing*, vol. 22, pp. 269–289, 2004.

[5] S. Nedevschi, F. Oniga, R. Danescu, T. Graf, and R. Schmidt, “Increased accuracy stereo approach for 3d lane detection,” in *Proc. International Vehicles Symposium*. IEEE, June 13-15, Tokyo, Japan, 2006.

[6] M. Bertozzi and A. Broggi, “Gold: A parallel real-time stereo vision system for generic obstacle and lane detection,” *IEEE Trans. on Image Processing*, vol. 7, No. 1, pp. 62–81, 1998.

[7] K. Yamaguchi, T. Kato, and Y. Ninomiya, “Moving obstacle detection using monocular vision,” in *Proc. International Vehicles Symposium*. IEEE, June 13-15, Tokyo, Japan, 2006.

[8] K. Kaliyaperumal, S. Lakshmanan, and K. Kluge, “An algorithm for detecting roads and obstacles in radar images,” *IEEE Transactions on Vehicular Technology*, vol. 50, no. 1, pp. 170–182, 2001.

[9] D. Alonso, L. Salgado, and M. Nieto, “Robust vehicle detection through multidimensional classification for on board video based systems,” in *IEEE Proc. ICIP*, San Antonio, USA, September 16-19, 2007, vol. 4, pp. 321–324.

[10] M. Nieto, L. Salgado, F. Jaureguizar, and J. Cabrera, “Stabilization of inverse perspective mapping images based on robust vanishing point estimation,” in *IEEE Proc. Intelligent Vehicles Symposium*, 13-15 June 2007, pp. 315–320.

[11] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing (3rd Edition)*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.