

Robust Road Modeling based on a Hierarchical Bipartite Graph

Marcos Nieto, Luis Salgado, and Fernando Jaureguizar
Grupo de Tratamiento de Imágenes - E. T. S. Ing. Telecomunicación
Universidad Politécnica de Madrid - Madrid (Spain)
Email: {mnd, lsa, fjn}@gti.ssr.upm.es

Abstract—Driver assistance systems based on video processing deliver a number of warnings to the driver, such as lane departure, lane invasion by other vehicles, collision prediction, etc. This has been a field of intense research for many years, providing solutions based on road models where vehicles are afterwards detected and tracked. Robustness is essential in this field of road safety where outliers represent one of the major problems for road modeling.

The motivation of this work is to provide a robust and, at the same time, flexible road model which identifies a variable number of lanes, their widths, the curvature of the road and the position of the vehicle in its lane. The major advantage of this model is that the system gives confidence measures for each lane, determining which lanes are actually present and which not. The model is structured as a hierarchical bipartite graph which simplifies information management, reduces sub-module dependencies and classifies elements of the road in different levels. At each level different strategies are applied, following four overall steps: measurement, estimation, evaluation and extrapolation, which lead to enhanced road model accuracy, reliability and flexibility. Several experimental results are provided, showing the robustness of the system, its stability and accurate results for large test paths.

I. INTRODUCTION

Driving safety means continuously acquiring knowledge of the surrounding of the vehicle, both the driver and the Advanced Driver Assistance Systems (ADAS). In the case of in-vehicle video-based ADAS, this knowledge level depends on the features that are extracted from the images and used to create a virtual model of the road. The more information is obtained, the more accurate the model is. Compared with other systems, video may go beyond gathering information about lane markings, curvature, position in the own lane, etc. Road modeling plays one of the key roles in any ADAS, as it may serve as basis for most of the services a driver might desire: e.g. lane departure warning, frontal collision warning or safety distance warning.

Although the task of extracting white lane markings from a dark surface like the road may seem easy, it leads to an enormous amount of problems due to: e.g. lighting conditions or pavement conditions dramatically change contrast between lane markings and road, vehicles may appear and occlude them, casted shadows, weather conditions, etc. Many different systems have been developed as the result of an intense research activity in this field, mainly focusing on lane modeling [1]-[4] and obstacle detection [5][6]. Most of these works apply specific strategies to focalize on detecting single features. For example, [1] works on the identification of lane changing by fusing video and vehicle status data.

An accurate 3D model of the own lane is given in [2] by analyzing the geometry of the lane markings. Free spaces in the road are detected in [4], and volumes obtained with stereo-vision depth maps in [5].

Nevertheless, none of these works establishes a structured road model on which it is possible to add or modify functionalities easily, and that may include all the information of interest about the surroundings of the vehicle. As stated, most of them focalize on solving particular aspects: i.e. no solution gathers an accurate model of the own and other lanes, their curvature, the position of the vehicle in the own lane, the position of other vehicles in the other lanes, etc.

The motivation of this work is to introduce a flexible, structured and robust road model which benefits from the implementation using graphs. Conceptually, it is based on a structured description of the information of the surroundings of the vehicle, in terms of elements (lane markings, lanes, etc.), properties (width, curvature), and links (lane marking belonging to a lane, etc.).

Typically, information structuring problems have been solved using graphs [7], as they are designed to easily manage elements, element properties and links between elements. For that reason, the model has been implemented into a hierarchical bipartite graph (HBG) which is composed of a set of bipartite sub-graphs that link different levels containing elements of the same type.

Road modeling using the HBG has a number of advantages, mainly enhancing the flexibility of the system: (i) information is structured allowing the access, storage and update of data in a simple way; (ii) each level has an associated processing stage, which depends only in the neighbor levels, thus speeding up calculations and simplifying dependencies; and (iii) modifications at each level, in terms of changing the processing algorithms, do not affect other levels.

Robustness is achieved by a four steps approach explained in detail in Section III: *measurements* related to the lane markings are taken, containing outliers and noise; *estimation* fit curves taking into account previous time instants to enhance temporal coherence; *evaluation* validates the estimates by checking them against a priori knowledge, and *extrapolation* propagates the model into areas where no observation was obtained.

II. ROAD MODELING STRATEGY OVERVIEW

The proposed strategy is based on the definition of the HBG and the processing stages associated to it. The next

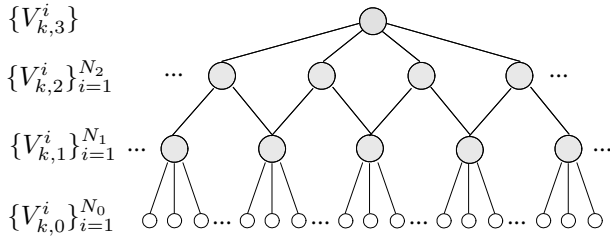


Fig. 1. Hierarchical Bipartite Graph.

TABLE I
HIERARCHICAL BIPARTITE GRAPH LEVEL DESCRIPTION.

Element-Level	Subset of Nodes	Node Information
Road	$\{V_{k,3}\}$	$V_{k,3} = \{\kappa_k\}$
Lanes	$\{V_{k,2}^i\}_{i=1}^{N_2}$	$V_{k,2}^i = \{w_k^i, \Psi_k^i\}$
Lane Markings	$\{V_{k,1}^i\}_{i=1}^{N_1}$	$V_{k,1}^i = \{\mathbf{p}_k^i, \Phi_k^i\}$
Control points	$\{V_{k,0}^i\}_{i=1}^{N_0}$	$V_{k,0}^i = \{\mathbf{x}_k^i, \{\mathbf{z}_k^{i,j}\}_{j=1}^{M_i}\}$

subsections describe this conceptual organization, and the way it has been implemented.

A. Hierarchical Bipartite Graph

There is a direct way of representing a road as a hierarchical bipartite graph, with different levels that group elements and links between them. Fig. 1 depicts the graph; as it is shown, this graph is bipartite between each consecutive pair of levels. Each level is denoted as $\{V_{k,l}^i\}_{i=1}^{N_l}$, where k is the discrete time index, l is the index of the level, and i and N_l are, respectively, the index of the node and the total number of nodes at each level. Table I shows the assignments of elements to levels, specifying the information contained at each level.

The highest level corresponds to the road node, which contains a single parameter, κ_k , the road curvature. This node is linked to an unrestricted number of lane nodes, $\{V_{k,2}^i\}_{i=1}^{N_2}$, each one containing its width w_k^i and a confidence level, Ψ_k^i . The following is the set of lane markings, which delimit each lane. These nodes contain the vector of parameters of the curve used to model the lane marking, \mathbf{p}_k^i , and a confidence level Φ_k^i . There are two lane markings per lane, and each lane marking is shared by two lanes. Therefore, the number of lane markings is $N_1 = N_2 + 1$. Finally, the base nodes, $\{V_{k,0}^i\}_{i=1}^{N_0}$ represent the control points of the curve that model each lane marking. Each control point is determined by its position \mathbf{x}_k^i and a set of measurements $\{\mathbf{z}_k^{i,j}\}_{j=1}^{M_i}$. The number of control points per lane marking, c , may depend on the type of model applied. The total number of nodes at this level is $N_0 = c \cdot N_1$. Details about these parameters are given in Section III.

B. Processing Stages

Fig. 2 depicts the processing stages applied on the HBG: (a) measurements of the position of the control points are taken at the base level. At the lane markings level these measurements are robustly fit into curves, which are used

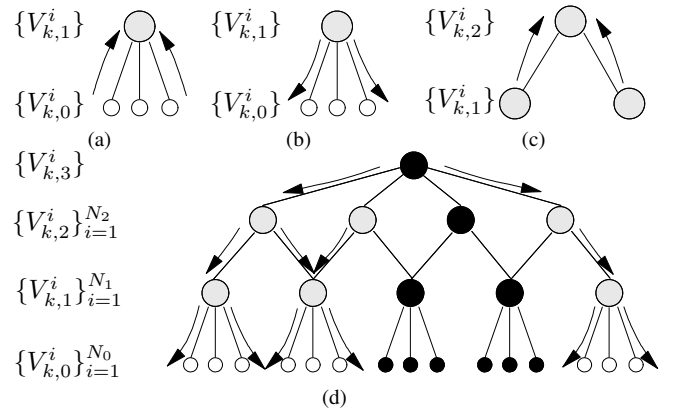


Fig. 2. Processing stages: (a) Measurement; (b) Estimation; (c) Evaluation; and (d) Extrapolation, where black nodes are those having estimations and then do not require extrapolation from higher levels.

back to estimate, as shown in (b), the positions of the control points. In (c) the estimated lane markings are further used to define the confidence levels of lanes which are evaluated to check the correctness of the estimations. Finally, in (d), the a priori knowledge of the road is used to extrapolate the parameters of the nodes which have not been involved in the measurement-estimation process to maintain coherence with the road model.

III. HIERARCHICAL BIPARTITE GRAPH

In the following subsections each processing stage is described in detail as well as the involved levels of the graph.

A. Measurement - Control points

The nodes of the base level correspond to the points that model the lane markings appearing in the images. They are called *control* points as they are used to estimate a curve that best fits to them, for example through the use of SVD [14] to give the least squares solution. The more control points considered, the more complex may be the curve modeling each lane marking. The number of control points per lane marking, c , models different curve types. If $c = 2$ the model could be a line [8][9] (which can be determined by a minimum of two points); $c = 3$ fits with generic second order curves, such as parabolas [1][10] circles, and constrained cubic curves approximating clothoids [11], while for $c = 4$ more complex spline shapes [12] can be estimated.

To estimate the parameters of these models most works use the likelihood maximization method [9][10]. However, RANSAC is used in this proposal, as it is a robust estimation approach that shows much better performance by removing outliers from the set of points [13]. The number of points used in RANSAC, c' , is larger than the minimum number of points required c . That way, for example, to estimate a circumference arc, $c' = 6$ can be used. Anyway, this number is still small enough not to increase the computational cost of RANSAC.

To take measures of these control points, a warped image, I'_k , with removed perspective, like the one shown in Fig. 3, is

firstly computed. It is usually denominated *inverse perspective mapping* (IPM) [6] or *plane to plane image warping* [14]. The procedure to generate it follows the approach shown in [15].

Each control point is defined by its two-dimensional position \mathbf{x}_k^i which is dynamically estimated through time. Measures are obtained as the intersection of the extended lane markings with a pattern of virtual horizontal lines. This extension is achieved by first applying an edge detection, a thresholding and a dilation on the result. The obtained mask is shown in Fig. 4 (a) where there are a number of white stripes corresponding both to correct lane markings, and to outliers (in this case a vehicle). The pattern of horizontal lines is shown in thin white lines. Blob coloring is applied on this mask and for each blob-stripe large enough a straight line is estimated using the Hough transform. The intersection of these lines with the horizontal pattern are marked as measures only for those horizontal lines close to the blob. This process is depicted in Fig. 4 (b), where estimated lines are shown in green, and measures in green circles. As it is shown, this strategy allows to obtain reliable and accurate control points even for non-continuous lane markings.

Measures are clustered around previous, in time, control points, so that each control point has an associated set of M_i measures $\{\mathbf{z}_k^{i,j}\}_{j=1}^{M_i}$, not shared with any other control point. A Delaunay triangulation is used on the immediate previous set of control points $\{\mathbf{x}_{k-1}^i\}_{i=1}^{N_0}$ to obtain the Voronoi cells which classify the image into regions of pixels which are closer to a certain control point than to any other in the image. An example of this image division is shown in Fig. 4 (c).

Therefore, the estimation of the control point depends on the number of measures that fall inside its corresponding cell. The estimated value is computed as follows:

$$\mathbf{x}_k^i = \begin{cases} \mathbf{x}_{k-1}^i & \text{if } M_i = 0 \\ \mathbf{z}_k^{i,*} & \text{if } M_i = 1 \\ \mathbf{z}_k^{i,*} & \text{elsewhere} \end{cases} \quad (1)$$

where $\mathbf{z}_k^{i,*}$ is the measure with lower distance to the previous control point:

$$\mathbf{z}_k^{i,*} = \min_{\mathbf{z}_k^{i,j}} \{\|\mathbf{x}_{k-1}^i - \mathbf{z}_k^{i,j}\|\} \quad (2)$$

B. Estimation - Curve fitting

Among the control points $V_{k,0}^i$ linked to each lane marking node $V_{k,1}^i$, those having an estimation \mathbf{x}_k^i obtained from a set of measures with $M_i \geq 1$ are used to fit a curve. As it was stated previously, any type of curve may be used here. Typical approaches use parabolic models [1], which offer enough accuracy for both IPM and natural images. However, for IPM images, generic circumference arc models show better performance for most situations. The parametric expression of a circumference is given by:

$$x^2 + y^2 + 2Ax + 2By + C = 0 \quad (3)$$



Fig. 3. Inverse Perspective Mapping of a road scene: (a) I_k original perspective image; and (b) I_k' image warping.

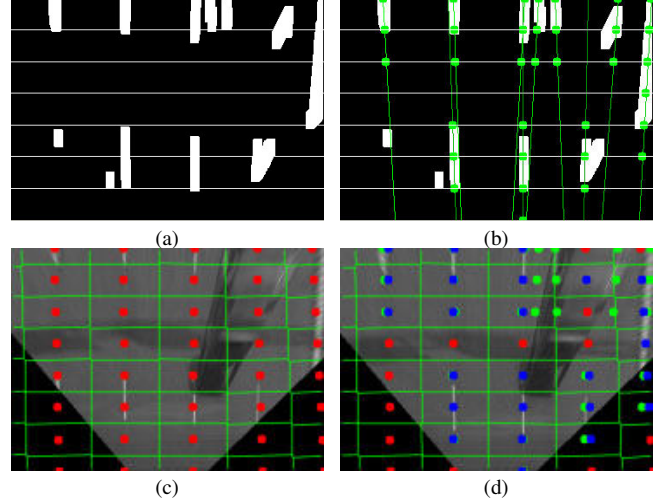


Fig. 4. Measures generation: (a) Blobs from dilated edges; (b) Intersection of Hough lines from each blob with horizontal lines; (c) Previous image with its Voronoi cells division in green solid lines, and the set of points \mathbf{x}_{k-1}^i in red circles; (d) Current image: (green) measures $\mathbf{z}_k^{i,j}$; (blue) estimated $\mathbf{z}_k^{i,*}$; and (red) no measures, using \mathbf{x}_{k-1}^i .

where the center is the point $\mathbf{x}_C = (-A, -B)$, and the radius is $r = \sqrt{A^2 + B^2 - C}$. Therefore, the vector of parameters of the node is expressed as $\mathbf{p}_k = (x_C, y_C, r)^\top$.

The minimum number of control points required is $c = 3$, but for enhancing robustness against outliers, RANSAC method is used for any set of points: $c \geq 3$, with a number of outliers removed denoted as \tilde{c} .

The accuracy of the model is improved if one particular assumption is applied: the vehicle is moving approximately parallel to the lane markings, so that the curve follows a vertical straight line in the lower part of the IPM image. The model is accordingly restricted to have its center at the known $y = H$ coordinate corresponding to the height of the image. This can be implemented in several ways, such as stating $B = -H$ and reducing one degree of freedom the expression in (3), or, equivalently, duplicating the control points as the specular image around the $y = H$ edge. Fig. 5 depicts this idea: for the left marking, the non-restricted estimation of circumferences (red) gives an erroneous circumference estimation due to the outlier at the top of the image; however, for the duplicated set of points (blue), the impact of outliers is highly decreased and the curve fits correctly.

Given the set of inliers $\{x_i, y_i\}_{i=1}^c$, the curve is estimated

as the least squares solution to the following system of equations:

$$\begin{bmatrix} 2x_1 & 2y_1 & 1 \\ \dots & \dots & \dots \\ 2x_c & 2y_c & 1 \\ 2x_1 & 2y'_1 & 1 \\ \dots & \dots & \dots \\ 2x_c & 2y'_c & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} -(x_1^2 + y_1^2) \\ \dots \\ -(x_c^2 + y_c^2) \\ -(x_1^2 + y_1'^2) \\ \dots \\ -(x_c^2 + y_c'^2) \end{bmatrix} \quad (4)$$

where y'_i is the height of the specular points, computed as:

$$y'_i = y_i + 2(H - y_i) \quad (5)$$

Once $\{V_{k,1}^i\}$ has been estimated, this processing stage updates the control points of nodes $\{V_{k,0}^i\}$ as the intersection points between the estimated curves and the abovementioned horizontal lines. For each horizontal line defined by its coordinate y_i , the intersection with the circumference is given by:

$$x_i = \frac{1}{2} \left(-2A \pm \sqrt{4A^2 - 4(y_i^2 + 2By_i + C)} \right) \quad (6)$$

which gives two solutions: the one falling inside the limits of the image is selected as the new control point, with coordinates $\mathbf{x}_k^i = (x_i, y_i)$.

The confidence level is assigned accordingly to the accuracy of the estimation and the number of detected outliers. Therefore, the confidence level is given by the following expression:

$$\Phi_k = \left(1 - \frac{\tilde{c}}{c} \right) \exp(-\tau D_i) \quad (7)$$

where the first term is the confidence factor given by the number of outliers; τ is a factor that determines how fast the exponential decreases; and D_i is:

$$D_i = \frac{1}{c} \sum_{i=1}^c d_i \quad (8)$$

where d_i is defined as the distance from the point (x_i, y_i) to the curve (A, B, C) :

$$d_i = \left| \sqrt{(x_i + A)^2 + (y_i + B)^2} - \sqrt{A^2 + B^2 - C} \right| \quad (9)$$

C. Evaluation

Evaluating the correctness of the estimations is done through the analysis of the confidence level of lanes, and their width. The estimation of the lane widths is carried out with a Kalman filter with the following state vector:

$$\mathbf{s}_k = (w_k^1, w_k^2, \dots, w_k^{N_2})^\top \quad (10)$$

where w^i is the width, in pixels, of the i -th lane in the IPM image. These values are supposed to vary smoothly within the range of road displayed on the image (usually less than 80 m), so that the dynamics of the system is reduced to:

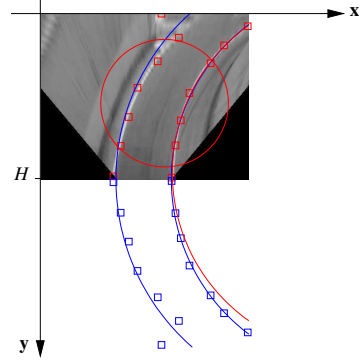


Fig. 5. Robust circumference arc fitting. Left lane marking contains one outlier control point and the LSE solution gives an incorrect circumference (in red). Applying the specular duplication the problem is fixed (blue).

$$\mathbf{s}_k = A\mathbf{s}_{k-1} + \mathbf{v}_k \quad (11)$$

where \mathbf{v}_k is Gaussian noise with zero mean and standard deviation that identifies the uncertainty in the evolution of the system, and the transition matrix, A , is the identity matrix of dimensions $N_2 \times N_2$. Measurement values are directly obtained as the distances between the modeled curves, with parameters \mathbf{p}_0 and \mathbf{p}_1 , corresponding to the lane marking nodes linked to the lane node at height $y = H$.

The a priori information introduced in the proposed strategy applied is that lanes are assumed to have similar widths, and that the central lane is usually the best estimated one (as it is the easiest lane to observe in the IPM image). Therefore, the confidence level associated to the lane is obtained as follows:

$$\Psi_k^i = \tilde{w}_k^i \prod_{n=\{0,1\}} \Phi_n \quad (12)$$

where Φ_0 and Φ_1 are the confidence levels of the left and right lane markings, respectively, linked to the lane node; and \tilde{w}_k is the confidence factor given by:

$$\tilde{w}_k^i = \exp(-\tau \|w_k^c - w_k^i\|^2) \quad (13)$$

where w_k^c is the width, at time k , of the estimated central lane, and τ is a decreasing factor. Therefore, the more the width is different to that of the central one, the more penalization is applied to the confidence of the lane.

The overall model of the road is evaluated at the highest level of the graph. The curvature of the road, parameterized as κ_k , is computed as $\kappa_k = r_k^{-1}$ where r_k is the radius of the circumference corresponding to the lane marking with highest confidence level.

D. Extrapolation

After the processing stages, all the parameters of the HBG have been estimated for the lanes that appear in the IPM image. Calling these lanes as the *viewed* ones, the HBG allows to compute the parameters of the nodes corresponding to the *non-viewed* lanes through an extrapolation process.

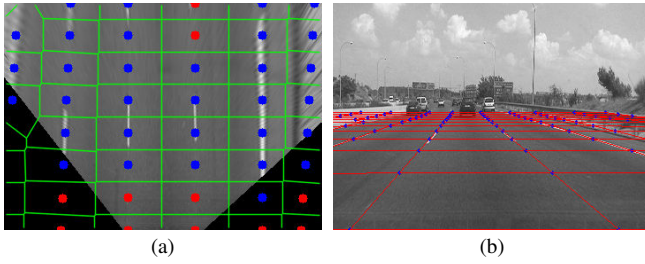


Fig. 6. Extrapolation example: (a) IPM showing the measures and estimations done on the *viewed* lanes; and (b) extrapolated grid of lanes containing *viewed* and *non-viewed* lanes.

Extrapolation is useful to have a pre-computation of lanes even when they do not appear in the images but may appear at any moment due to the movement of the vehicle between lanes.

The extrapolation is done at lanes, lane markings and control points levels: (i) it is easy to add a number of lanes at the left and right of the viewed lanes, with width equal to the nearest viewed lane; (ii) lane markings are also extrapolated at left and right of the viewed lane markings. The parameters of the curve are the same as those of the nearest lane marking, increasing the radius of the circumference w pixels, where w is the extrapolated width of the lane:

$$\mathbf{p}_{ex} = (x_C, y_C, r \pm w)^T \quad (14)$$

(iii) control points are extrapolated from these curves by computing the intersection of the curves with the horizontal lines as explained in Section III-A.

Fig. 6 shows an example of the extrapolation process. In (a), the IPM contains the *viewed* lanes, while in (b), the model, with the extrapolated *non-viewed* lanes is shown.

IV. RESULTS AND DISCUSSION

Successful results have been obtained with long test sequences recorded from a vehicle as it was driven along motorways around Madrid (Spain), gathering all the elements required to check the robustness of the model: cast shadows due to sunny weather, variable number of lanes, variable types of lane markings (continuous, discontinuous, double, etc.), presence of vehicles, severe occlusions of lane markings, changes of lanes, etc. Fig. 11 shows some images corresponding to different test sequences, with the superimposed grid of estimated control points.

An important issue is the inherent instability of the IPM images that may cause the bird-view to show non-parallel lane markings, and the detected curvature to be incorrect. For that purpose, the IPM is stabilized through the continuous estimation of the vanishing point, whose vertical component is used to determine the homography that transforms the natural image into the IPM image [15]. This vanishing point is estimated as the least squares intersection of those lane markings with high confidence level, Φ_k^i . Fig. 7 shows two images where the vanishing point is significantly different but correctly estimated. The overall variation along the whole sequence of its vertical component is shown in Fig. 8.

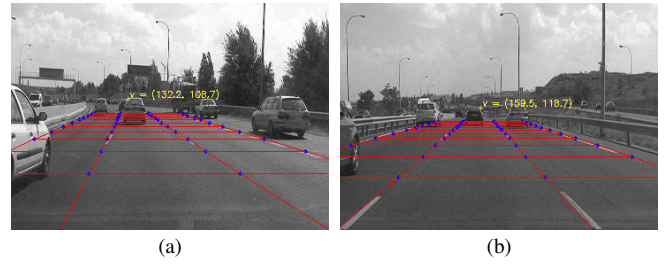


Fig. 7. Two images belonging to the same sequence, with notably different vanishing points that are correctly estimated.

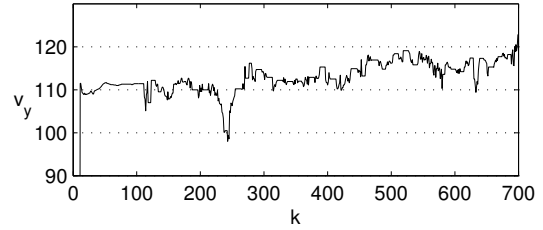


Fig. 8. Vertical coordinate of the vanishing point for a test sequence.

One of the most relevant events that has to be detected is lane changes by lateral movements of the car. Thanks to the continuing monitoring of several lanes, this issue is simple, and the position of the vehicle within its lane is computed as:

$$x_{off} = \frac{W - 2x_w}{w} \quad (15)$$

where W is the width of the image, x_w is the position of the center of the lane in the image, and w is the width of the lane, computed as explained in Section III-C. This is a value between -1.0 and 1.0 , corresponding to the maximal left and right deviation within the lane, respectively. Fig. 9 shows the evolution of this lateral offset, where two lane changes have been addressed.

The road model is supported by the confidence measures taken from the estimations. Fig. 10 shows the variation of the confidence level for eight lanes, indexed from 0 to 7, for an example test sequence. The road corresponding to this sequence has four lanes, shown in light gray. Dark shadow regions correspond to the lane that is occupied by the own vehicle at each time instant. Lane changing corresponding to the lateral offset depicted in Fig. 9 makes that lane number 1 changes from viewed to non-viewed, so that its confidence level falls to zero when the vehicle is in lane number 4. As a remark, the observed intermittenencies on the signals are mainly caused by the discontinuous lane markings, that increase and reduce the number of control points periodically.

Regarding computational issues, the whole system have been tested in a laptop CoreDuo@2GHz, working in real-time for a framerate of 10 fps for sequences with image resolution 360×256 in 8-bit grayscale.

V. CONCLUSIONS

In this paper a novel road modeling strategy has been introduced, showing robustness against outliers, caused by

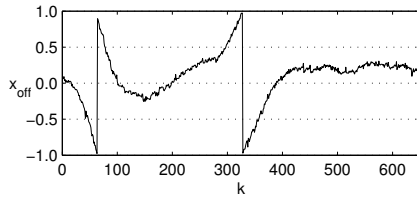


Fig. 9. Estimation of the horizontal position x_c of the own vehicle within its lane. Lane changes are denoted as large jumps between -1.0 and 1.0 .

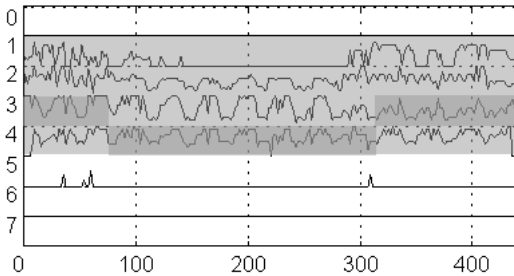


Fig. 10. Confidence levels, $\{\Psi^i\}$ for a test sequence evaluating eight lanes.

the variable lighting conditions, occlusions, shadows, ego-motion, etc., and flexibility due to a structured way of representing elements of the road.

Robustness is enhanced by the use of a probabilistic approach that assigns confidence levels to lane markings and lanes according to the quality of the measurements. Robust estimation techniques like RANSAC are also used.

The hierarchical bipartite graph ensures a working approach that simplifies dependencies, allowing the introduction of modifications or upgrades easily while keeping the coherence of the full model. The graph is used in a four steps processing, with measurement, estimation, evaluation and extrapolation steps.

Additional levels might be added naturally: e.g. vehicles level, gathering an unrestricted number of vehicles in the road, directly linked to the lanes level. The graph represents all the elements that may appear in a road scene, clearly specifying restrictions and dependencies in a flexible way allowing to configure each level independently.

ACKNOWLEDGEMENTS

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).

REFERENCES

- [1] 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, March 2006, pp. 20-37.
- [2] S. Nedeveschi, F. Oniga, R. Danescu, T. Graf, and R. Schmidt, "Increased Accuracy Stereo Approach for 3D Lane Detection", in *IEEE Proc. Int. Veh. Symp. 2006*, June 13-15, Tokyo, Japan, pp. 42-49.

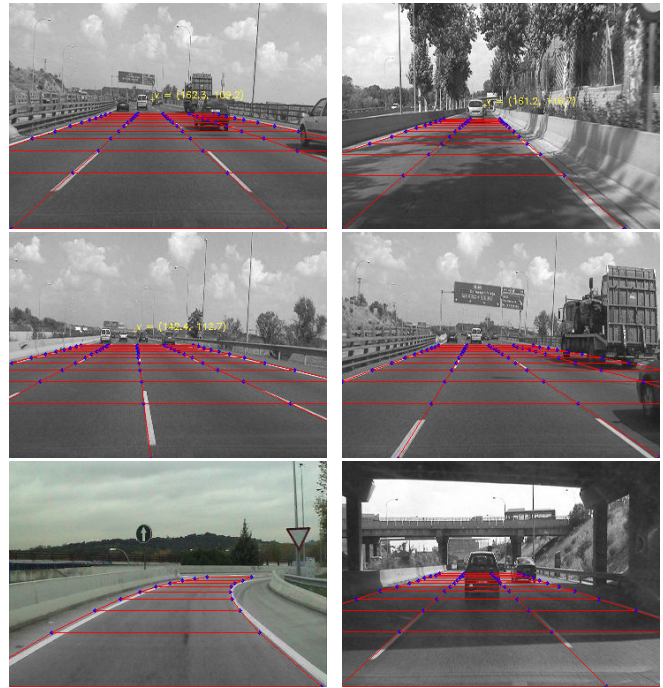


Fig. 11. Results of the road modeling. The model is proven to work well for situations with severe occlusions, lane changing, presence of vehicles, casted shadows, etc.

- [3] D. Schreiber, B. Alefs and M. Clabian, "Single camera lane detection and tracking", in *IEEE Proc. Intelligent Transportation Systems*, September 2005, pp. 302-307.
- [4] P. Cerri and P. Grisleri, "Free Space Detection on Highways using Time Correlation between Stabilized Sub-pixel precision IPM Images", in *IEEE Proc. Int. Conf. on Robotics and Automation*, Barcelona, Spain, April 2005, pp. 2223-2228.
- [5] K. Yamaguchi, T. Kato, and Y. Ninomiya, "Moving Obstacle Detection using Monocular Vision", in *IEEE Proc. Intelligent Vehicles Symposium 2006*, June 13-15, Tokyo, Japan, pp. 288-293.
- [6] M. Bertozzi, and A. Broggi, "GOLD: A Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection", in *IEEE Proc. Intelligent Transportation Systems 2000*, pp. 346-351.
- [7] V.K. Balakrishnan, "Graph Theory", McGraw-Hill, 1997.
- [8] D.-J. Kang, and M.-H. Jung, "Road lane segmentation using dynamic programming for active safety vehicles", *Pattern Recognition Letters*, December 2003, vol. 24, no. 16, pp. 3177-3185.
- [9] K. Kaliyaperumal, S. Lakshmanan, and K. Kluge, "An algorithm for detecting roads and obstacles in radar images", in *IEEE Transactions on Vehicular Technology 2001*, vol. 50(1), pp. 170-182.
- [10] K. Kluge, and S. Lakshmanan, "A deformable template approach to lane detection", in *IEEE Proc. Intelligent Vehicles 1995*, Detroit, USA, September 25-26, 1995, pp. 54-59.
- [11] C. Corridori and M. Zanin, "High Curvature two-clothoid road model estimation", in *IEEE Proc. Intelligent Transportation Systems*, Washington, D.C., USA, October 3-6, 2004, pp.630-636.
- [12] 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.
- [13] C. R. Del Blanco, F. Jaureguizar, L. Salgado, and N. García, "Aerial Moving Target Detection Based on Motion Vector Field Analysis," *Lecture Notes in Computer Science*, vol. 4678, 2007, pp. 990-1001.
- [14] R. Hartley, and A. Zisserman, "Multiple View Geometry in computer vision", Cambridge University Press, 2001.
- [15] 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 2007*, Istanbul, Turkey, June 13-15, 2007, pp. 315-320.