

# On-Road Position Estimation by Probabilistic Integration of Visual Cues

Voichita Popescu, Radu Danescu, Sergiu Nedevschi

**Abstract**—This paper addresses the problem of finding the host vehicle’s lateral position on a multi-lane road, using information obtained by processing video sequences. A very important cue for lane identification is the class of the boundaries of the current lane. This paper presents a reliable solution for lane boundary type identification, based on frequency analysis of the gray level profile of these boundaries, assuming that the current lane is already detected. The lane boundary information is combined with the obstacle information, through a Bayesian Network which will output, frame by frame, the probability of the vehicle to be positioned on each lane of the road. The probability result will be propagated throughout the sequence by a Particle Filter.

## I. INTRODUCTION

Advanced Driving Assistance Systems can significantly improve the driving experience, while also increasing the overall traffic safety. An important prerequisite for any ADAS action is the proper assessment of the situation of the host vehicle and of the surrounding traffic. Part of this situation assessment is the knowledge about the host vehicle’s position on the road. There are several systems that can help us to gain this knowledge: satellite navigation systems can provide a rough position estimate, inertial systems can fill in the gaps of satellite positioning (and filter the estimates), and lane detection systems can tell us the position within a lane. With the help of a map, we can infer an approximate position on the road, or at least we can tell we are on the side of the road corresponding to our direction of driving. Unfortunately, when we have multiple lanes for a driving direction, the problem is not that simple: the navigation systems are not precise enough to tell us on what lane we are, the lane detection systems may not detect all lanes, due to occlusions from other vehicles, and the direction of our driving does not help. In the literature, there exist several approaches for accurately positioning the host vehicle on the road, and estimating the lane on which the host vehicle is travelling on.

Manuscript received January 27, 2012. This paper was supported by the project "Doctoral studies in engineering sciences for developing the knowledge based society-SIDOC" contract no. POSDRU/88/1.5/S/60078, project co-funded from European Social Fund through Sectorial Operational Program Human Resources 2007-2013, and by the and by the POSDRU-EXCEL post-doctoral program, financing contract POSDRU/89/1.5/S/62557.

Voichita Popescu, Radu Danescu and Sergiu Nedevschi are with the Technical University of Cluj-Napoca, Computer Science Department (e-mail: {firstname.lastname}@cs.utcluj.ro). Department address: Computer Science Department, Str. Memorandumului nr. 28, Cluj-Napoca, Romania. Phone: +40 264 401484.

Generally, the solutions are based on GPS localization which is then enhanced by additional vehicle and/or on-board sensors such as inertial navigation systems, odometers, vision sensors, inter-vehicle communication systems, digital maps. Different vision enhanced lane level positioning systems are proposed in [1], [2], [3], [4]; these methods also use detailed digital maps of the environment. A method for lane level positioning based on inter-vehicle communication is presented in [5].

This paper proposed an original solution for lane level positioning on a multi-lane road, based on an on-board stereovision processing system and an extended digital map [6]. The contributions of this paper are a novel method for lane boundaries classification and an original method based on a Bayesian Network (BN) for lane estimation. The network is used for correlating the visual information with the map information; additionally, the information about other vehicles is used in the network for lane estimation. The frame by frame results are tracked using a particle filter in order to take into consideration the time evolving nature of the problem. In this approach, roads with three to six lanes per driving direction are considered. The solution is designed for structured roads (roads with marked lane boundaries). Fig. 1 illustrates the overview of the proposed solution for on-road position estimation.

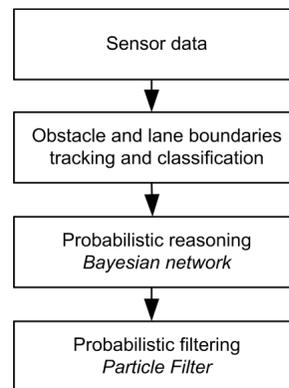


Fig. 1 Solution overview for on-road position estimation

The system contains four functional blocks. The first block delivers 3D information through stereo image processing. The second block consists of the tracking and classification of the obstacles and of the lane boundaries; this block provides the evidence for the third block in the architecture. The third block is the probabilistic reasoning block; it performs frame by frame reasoning using a Bayesian network [7], [8] approach. The fourth block performs a temporal filtering (tracking) of the instantaneous beliefs provided by the static

Bayesian Network. This ensures a more stable and reliable position estimation, taking into consideration the evolution of the driving context over time.

The rest of the paper is structured as follows: Section II describes the novel approach for lane boundary classification; Section III presents the original approach of on-road position estimation by using the visual cues as evidence in the proposed network, while Section IV is dedicated to the temporal filtering (tracking) of the frame by frame position estimation. The results of the proposed method are illustrated and discussed in Section V, and the final conclusions are being drawn in Section VI.

## II. LANE BOUNDARY CLASSIFICATION

Knowledge about the type of boundaries of our current lane is a welcome addition to any perception system designed for driving assistance. In the context of on-road position estimation, the nature of these boundaries is an important clue about our position on a multi-lane road. Double or continuous line boundaries separate the directions of traffic; discontinuous boundaries separate lanes in the same direction, and merge-type markings (densely discontinuous markings) separate the road from road-side parking area, and are also present in intersections as direction guides.

This section presents a robust lane boundary classification technique, which relies on the frequency analysis of the intensity profile of the lane limits. Frequency based boundary classification solutions are presented in [9] and [10]. Our solution is also related to the one presented in [11], the similarity being the use of equally spaced scan lines projected in the image space, but the difference is in the processing of the resulted data – the solution presented in [11] encodes the presence of markings on neighboring scan lines as codes, sequence of codes forming regular expressions which are then analyzed in a parser-like fashion.

### A. Feature extraction

The lane border classification process starts with the extraction of basic features from the gray level perspective image. The lane geometry is already estimated by a particle filter-based lane tracker [12].

What we are interested in are the grayscale values along the lane delimiter, and their relationship with the grayscale value of the asphalt in the same region. For this reason, we will generate, in the 3D space, a set of equally spaced lines, covering a distance of 20 meters (starting from the minimum reliably visible distance). The distance between these lines is 20 cm. Thus, we will cover a length of 20 meters of lane with 100 equally spaced lines. On each line, we will generate, for each lane boundary (left and right), two search regions (segments): one which will most likely cover the painted markings, and the other that will most likely cover the asphalt inside the lane. On each segment, we will select a number of 10 equally spaced points – the points for the marking region will be spaced 2 cm apart, and the points for the asphalt region will be spaced 5 cm apart. These points are then projected in the image space, using the perspective

transformations derived from the camera parameters. The resulted image space points are seen in Fig. 2– white points for the marking search region, and black for the asphalt reference region.

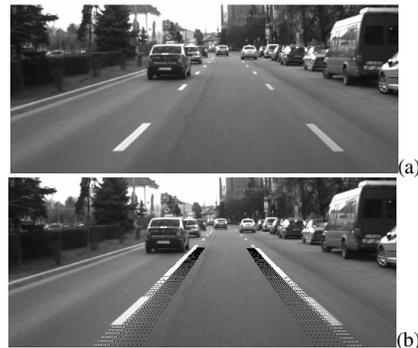


Fig. 2 Original grayscale image (top), and the search areas for the markings (white dots) and for the asphalt reference (black dots) grayscale values. The position of the search areas is given by a lane tracking result.

The grayscale values of each point generated in this way are averaged for each segment of each lane boundary. Thus, for each side we have, for each line (corresponding to a distance), two average values: the average value of the marking area, and the average value of the asphalt reference area. Then, for each distance and for each marking we compare the two averages. If the marking area average is higher than the asphalt area average (by a low, fixed threshold), we set a ‘1’ in a binary signal, and otherwise we set a zero. Fig. 3 describes the process.

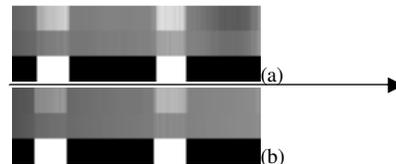


Fig. 3 Processing the search areas, for the left (a) and right (b) lane boundaries. Top row: average intensity on each marking search line; middle row: average intensity on each asphalt reference search line; bottom row: whether the marking intensity is higher than the reference intensity (plus a threshold) for each line. The arrow shows the driving direction.

The binary signal encoding the relation between the average grayscale values of the markings and the asphalt is the primary feature of our boundary classification algorithm.

### B. Temporal integration and filtering

The binary signal describes the nature of the boundary, as the alternation pattern between the 0’s and the 1’s is characteristic for the boundary type. However, a signal extracted in a single frame can be affected by some problems, such as transient errors (due to small errors in lane model fitting at the distance, or cars passing on the lane border), or errors due to a limited field of view (especially in the case of interrupted lane markings, the nature of the marking may not be well described by what we see in a single frame). For this reason, we use a temporal integration of the single frame results. The speed of the host vehicle and the time interval between the frames is used to compute an offset, which will be used to shift the past results. The results of the current frame will be averaged with the results of the past frames.

Also, we will expand the distance interval for our time integrated signal, such that it will cover a total distance of 30 meters (20 in the visible range, and 10 meters behind the visible range). This way, we obtain a longer (150 discrete values) and more stable signal, which will better describe the marking type.

A final step is an additional smoothing of the time integrated signal, using a Gaussian kernel (Fig. 4).



Fig. 4 Time integrated binary signal for the left boundary (a), and the result of Gaussian filtering of this signal (b).

### C. Lane boundary classification

In order to classify the marking type, we will extract several features from the filtered binary signal. The most obvious feature is the number of 1's in the signal. As the signal is now continuous due to filtering, we will count as ones the signal values that exceed the value 0.5. Thus, we have our first feature for classification, which we will call *OneCount*.

Next, we have to analyze the shape of the signal. As the signal is periodic, we have to use a frequency-based analysis. Instead of Fourier transforms, as used in [9], we will use a simpler approach, which compares the signal with itself at different time intervals. For each candidate period  $t$  we will build a sum of the absolute differences of the signal values spaced by  $t$ . Equation (1) will be applied for  $t$  from 0 to 100. In this equation,  $B$  is the filtered signal, and  $N$  is the total number of values  $B$ .

$$P(t) = \sum_{k=0}^{N-t} |B(k) - B(k+t)| \quad (1)$$

The function  $P$  (we will call it a *Period Histogram*) will peak for those values of  $t$  that correspond to the spacing between the middle of the dark intervals and the middle of the white intervals of  $B$ . Thus, the first peak corresponds to the half period of the signal  $B$ , the second peak to 1.5 periods of signal  $B$ , and so on.

The next step is to find local maxima in the period histogram. For classification purposes, we will retain the smallest two values of  $t$  corresponding to distinct (not "touching") local maxima of  $P$ . Let us denote these values by  $t_{first}$  and  $t_{second}$ . The value of  $t_{first}$  describes half the period of the signal  $B$ , and the value of  $t_{second}$  is used for validation: in a periodic signal,  $t_{second} = 3t_{first}$ .

Thus, for the classification of the signal  $B$  we have three features, which we will summarize here:

- The number of 'ones' in the filtered signal, *OneCount*.
- The half-period of the signal,  $HalfPeriod = t_{first}$ .
- The peak position ratio,  $PeakRatio = t_{second}/t_{first}$ .

These simple numerical features are enough to define the nature of the lane boundary: *OneCount* should be high for a continuous line boundary, *PeakRatio* should be 3 for a periodic signal such as an interrupted line or a merge line (dense interrupted) line, and the *HalfPeriod* should have

definite values for the interrupted or merge lines. The exact rules for delimiter classification are extracted using the decision tree learning system from Weka [13]. Fig. 5 and Fig. 6 show several delimiter examples and the signal analysis for each of them.



Fig. 5 Signal analysis: left border is interrupted and right border is continuous. (a) Original grayscale image, (b) filtered binary signal  $B$ , (c) period histogram  $P$ , (d) first and second local maxima of  $P$ .

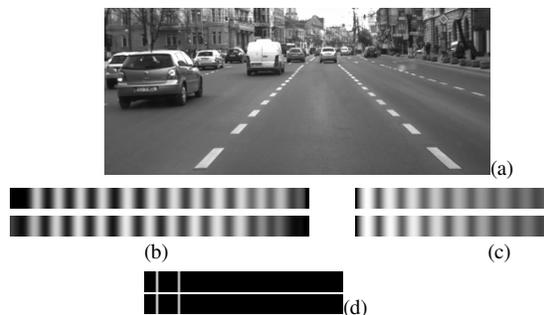


Fig. 6 Signal analysis: merge boundaries on both sides, inside an intersection. (a) Original grayscale image, (b) filtered binary signal  $B$ , (c) period histogram  $P$ , (d) first and second local maxima of  $P$ .

### D. Detection of double lines

Once the borders are classified based on their longitudinal pattern, we will analyze their lateral shape to detect the double lines. For this, we will generate search regions again, but this time they will be wider, so that double markings can be covered by them. We are interested only in the search lines that correspond to marking lines in the original non-filtered binary signal (we expect to have doubles only where markings were previously located in the narrower search region). For these region lines, we classify the pixels as markings versus non-markings, based on their intensity: if the intensity of one pixel is closer to that of an already classified marking (see section A), it is labeled as marking, and if the intensity is closer to that of the reference area, it is labeled as non-marking.

The next step is to count the transitions between marking and non-marking pixel sequences for each line of the search regions, as seen in Fig. 7. A normal delimiter should have two transitions, while a double one should have four. We will average, for each lane side (left and right) the number of transitions, and if the transition average is higher than 3 we declare the delimiter to be double.

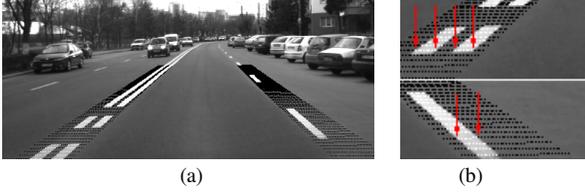


Fig. 7 Detection of double lines: (a) Search regions for counting marking/non marking transitions, (b) Transitions on a search line.

### III. POSITION ESTIMATION

The lane boundary classification obtained in this way is highly relevant information in assessing the lane that the host vehicle is travelling on. Other relevant information for estimating the host vehicle's position on the road is provided by other visually detected vehicles. Therefore, in the current approach we use the visual information as evidence for inferring the position of the host vehicle on the road.

At each time instance, the stereovision sensor processes and records measurements about the driving environment, such as: the type of lateral lane boundaries (whose classification has been presented in the previous section), the type of painted arrows [14], as well as information about the detected vehicles (speed and position) [15]. However, this data is noisy and furthermore, some quantities cannot be directly measured all the time. Therefore the sensorial data must be used in a probabilistic manner when trying to infer something about the driving context. The proposed Bayesian Network (BN) approach is a suitable one due to the following considerations: they provide a mean of modeling human-like reasoning and are capable of performing inference on uncertain and incomplete data, which is our case. Furthermore, BNs provide an instantaneous posterior probability distribution that can be filtered over time, thus taking into account the dynamic nature of the domain.

The problem of identifying the lane that the host vehicle is travelling on has been previously addressed in [6], for the purpose of accurate vehicle global localization prior to intersection. The current approach extends the previous work from [6] by introducing a higher degree of generality of the network, by improving the visual measurements used as evidence in the network (the improved technique for lane boundary classification) and by time filtering of the frame by frame results provided by the network.

The main idea of the reasoning mechanism is that the proposed network encodes the relationship between the traffic environment elements (road landmarks, other vehicles) and uses these relationships, together with the visual measurements, in order to perform reasoning about the lane that the host vehicle is driven on. There are two types of nodes in the network: observable nodes (gray) that encode variables that can be measured, and hidden nodes (white) that encode variables that cannot be measured, and whose states we are interested in. The states of the hidden variables are inferred using the network, the evidence of the observable nodes, and an inference algorithm [16]. In the proposed network (Fig. 8), the observable nodes encode information

about the static environment (the type of left and right lane boundaries, the type of painted arrows) as well as information about the dynamic environment (the behavior of other detected vehicles – relative longitudinal movement, relative lateral position). The hidden nodes of this network (including the *EgoLaneNo* node) encode the possible hypotheses for the ego-lane number, i.e. the possible lanes  $l_1, l_2, \dots, l_{MAX}$ , where  $MAX$  is the number of lanes per driving direction (the lane numbering starts from left to right). The task of the network is to infer the posterior probability distribution of the *EgoLaneNo* node over these hypotheses using the visual measurements as evidence. The intermediate nodes are used to reduce the size of the leaf node, and to simplify the computation of its conditional probability table.

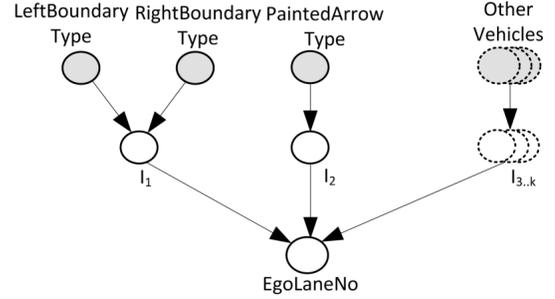


Fig. 8 Structure of the proposed BN.

The structure of the proposed network is semi-fixed as illustrated in Fig. 8. While the variables encoding the static environment information have a fixed dependence relationship, the variables encoding the dynamic environment information are added to the network only when sensorial measurements about the other road users become available. In Fig. 8, the solid line nodes are permanent nodes of the network, while the dashed line nodes are occasional nodes in the network, are added only when dynamic environment information becomes available. While the structure of the network is semi-fixed, the parameters of the network (denoted by  $\theta$ ), however, change according to the road's infrastructure. We assume that the information about the road infrastructure (number of lanes per way, widths of the lanes, type of lanes' delimiters and painted arrows) is known a priori from an extended digital map [6]. Under this assumption, the training set  $D$  is complete. What we intend to do is to find the network that best matches the data in  $D$ , i.e. for each node to find the initial probability that best matches the data in  $D$ . According to [17] this is done using Maximum Likelihood Estimation (MLE). Since we have a set of complete data as training set, the log-likelihood is given by equation (2):

$$l(\theta | D) = \sum_l \log P(D_l | \theta) \quad (2)$$

that is equivalent to equation (3),

$$l(\theta | D) = \sum_{i,k} \sum_j m_{ijk} \log \theta_{ijk} \quad (3)$$

where  $m_{ijk} = \sum_l \mathcal{X}(i, j, k : D_l)$  is the sufficient statistics.

The sufficient statistics is a function of data that summarizes the relevant information for computing the likelihood. In other words  $m_{ijk}$  is equal to the number of cases when the node  $X_i$  has the state  $j$  and the  $X_i$ 's parents' configuration is  $k$ . By maximizing the log-likelihood from equation (3) the following estimate is obtained:

$$\theta_{ijk}^* = \frac{m_{ijk}}{\sum_j m_{ijk}} \quad (4)$$

For the construction of the network the a priori information about the environment is used. Therefore the network is rebuilt for each road segment according to the road's infrastructure, and according to the other vehicles information. The evidence for the observable nodes, however, comes from the visual measurements. At each time instance this evidence is used to compute the instantaneous belief of the *EgoLaneNo* hidden node:

$$P(l_1, l_2, \dots, l_{MAX}) = \{w_1, w_2, \dots, w_{MAX}\} \quad (5)$$

where  $\sum_{i=1}^{MAX} w_i = 1$ . This discrete posterior probability

distribution over the lane hypotheses of *EgoLaneNo* node is passed on as measurement to the particle filter, which is described in the next section.

#### IV. POSITION TRACKING

The lane number probability density is approximated for a time instance  $t$  by a set of  $N$  weighted particles  $p_t(L) \approx \{L_t^i, \pi_t^i, i = 1 \dots N\}$ . Each particle will retain a hypothesis  $l_k$ , having a discrete value from  $l_1$  to  $l_{MAX}$ . The particle will also carry a weight  $\pi_t^i$ , encoding the strength of this hypothesis.

The cycle of this (simplified) particle filter is carried out in three main phases: *Resample*, *Diffusion*, and *Measurement*. The *Resample* phase uses the past weighted particle population to generate a new population, by making  $N$  random selections and using the weight of the particles as a measure of the probability of it being selected. This way, higher weight particles are selected more than once, while lower weight particles may be selected less frequently or never. During the resample phase, a small fraction  $fN$  of the particles,  $f < 0.1$ , will be drawn from the general distribution, meaning that they will get a random hypothesis from  $l_1$  to  $l_{MAX}$ . The parameter  $f$  controls the reaction of the filter to changing conditions, a large value making it faster to adapt to a situation change, the cost being a lower stability of the estimation.

After resampling, the *Diffusion* process alters the state of the new population of particles. Each particle will get a chance of  $r$  to alter its lane hypothesis to its left neighboring lane ( $L_t^i - 1$ ), a change of  $r$  to alter its hypothesis to its right neighboring lane ( $L_t^i + 1$ ), and a chance of  $1 - 2r$  to keep its hypothesis unchanged. The parameter  $r$  is in the range of 0.025 to 0.1, a smaller value leading to a behavior when the

filter is aggressively pursuing the best hypothesis, under the penalty of disregarding other possibilities.

The *Measurement* step will assign to each particle a new weight. Each particle will get as weight the probability of its lane hypothesis, computed in Section III, i.e. for  $L_t^i = l_k$ ,  $l_k \in \{l_1, l_2, \dots, l_{MAX}\}$  the corresponding weight will be  $\pi_t^i = w_k$ . After the measurement step, the weights of the particles are normalized, so that their sum equals 1, and the cycle can start again.

The probabilities for each lane hypothesis  $l_k \in \{l_1, l_2, \dots, l_{MAX}\}$  can be estimated by adding the weights of the particles holding that particular hypothesis, as in equation (6).

$$P_t(l_k) = \sum_{L_t^i = l_k} \pi_t^i \quad (6)$$

#### V. EXPERIMENTAL RESULTS

##### A. Lane boundary classification results

The lane boundary classification system was tested using a sequence of 9830 frames, which record a drive of 8 km through the city of Cluj-Napoca, Romania. We have tested the classification results for the left lane boundary, against ground truth generated by manual labeling of intervals in the frame sequence. The reason the left boundary was chosen for performance evaluation is that throughout the sequence this lane side passes through all the classes we are looking for: interrupted line, continuous line, merge line, no line, double continuous line and double merge line. Some of the classes are forbidden for the right lane side, at least while we obey the traffic laws.

The quality of the markings in the test sequence ranged from excellent to poor, and sometimes cars or pedestrians were occluding the view. The boundary classification system was correct in 7969 cases, which means 81% of all frames. Table I describes the results for each lane boundary class. Some classes are better represented in the sequence, while others, such as the double merge class, are rarer. The table shows for each class the true positive (TP) rate (the ratio between the number of correctly detected instances of a class and the total number of instances of that class), and the false positive (FP) rate (the ration between the number of incorrectly detected instances of a class versus the number of frames that class was not actually present).

TABLE I. CLASSIFICATION RESULTS

Border class	Instances	TP rate	FP rate
<i>No marking</i>	3958	0.780	0.080
<i>Continuous</i>	1049	0.896	0.079
<i>Interrupted</i>	2413	0.804	0.052
<i>Merge</i>	144	0.792	0.014
<i>Double continuous</i>	2207	0.840	0.006
<i>Double merge</i>	59	0.831	0.002

We can see that while the true positive rate is not extremely

high, due to the conditions of the markings and some other causes of occlusion, the false positive rate is quite low, which makes the system a robust solution for boundary classification.

Some qualitative results of the lane boundary classification system are shown in Fig. 9. This figure shows in perspective view and bird-eye view the current detected lane, along with the class of its delimiters.

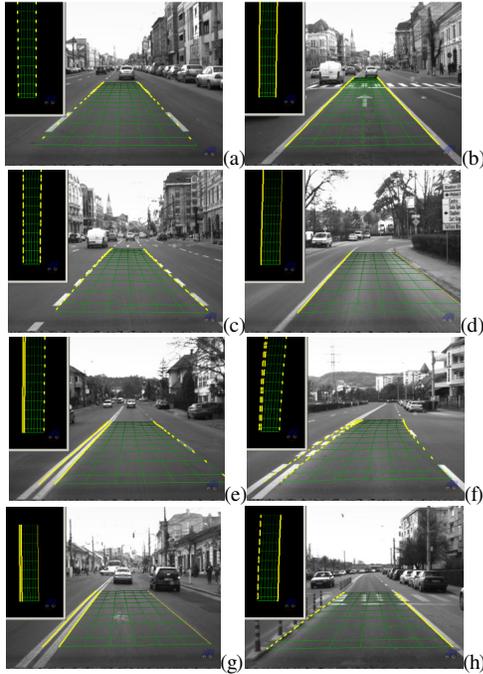


Fig. 9 Lane boundary classification sample results: (a) discontinuous lines, (b) continuous lines, (c) merge lines inside an intersection, (d) continuous line on the left, no marking on the right, (e) double continuous line on the left, discontinuous line on the right, (f) double merge line on the left, discontinuous line on the right, (g) double continuous line on the left, worn out marking on the right, not detected, (h) falsely detected merge line on the left, due to pillars, and continuous line on the right.

### B. Positioning results

The proposed position estimation method was tested on the same sequence of 8 km drive, during which we have encountered segments of road with three to six lane per driving direction. The information about the detailed road's infrastructure is available in the extended digital map. Using the proposed approach we have identified the lane on which the host vehicle was driven: the frame by frame probability identified the lane correctly in 76% of the cases, and by adding the particle filter the outcome was improved to a 89% correct lane position estimation. The following figures illustrate the instantaneous lane probabilities versus the filtered lane probabilities, together with images from the stereovision perception system that provides the visual evidence. Fig. 10 illustrates a case with three lanes per driving direction, while Fig. 11 illustrates a case with five lanes per driving direction. Our lane positioning solution produces a sharper probability for the first case, and a more dispersed probability for the second case (both instantaneous and filtered), due to the fact that more lanes have the same

lateral lane boundaries and painted arrows, making the decision process more difficult.

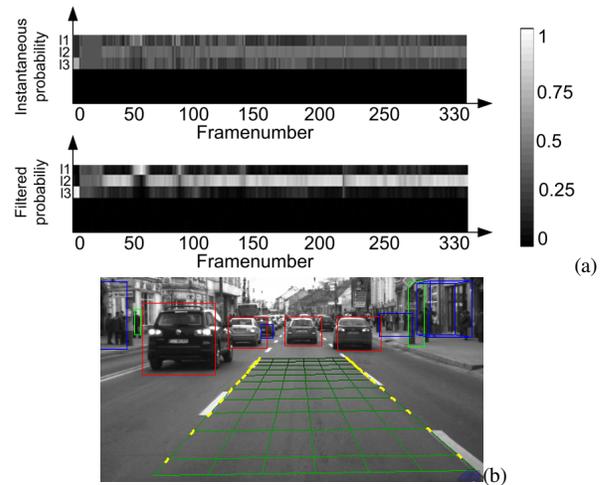


Fig. 10 Three lanes per driving direction (a) Comparative results of the instantaneous posterior probabilities provided by the BN against the filtered probabilities resulted from the particle filter (b) Example of instantaneous visual evidence (left boundary = interrupted, right boundary = interrupted, left and right outgoing vehicles) and resulted instantaneous probabilities  $P(l_1, l_2, \dots, l_{MAX}) = \{0.27, 0.46, 0.27\}$  and filtered probabilities  $P(l_1, l_2, \dots, l_{MAX}) = \{0.05, 0.935, 0.015\}$ .

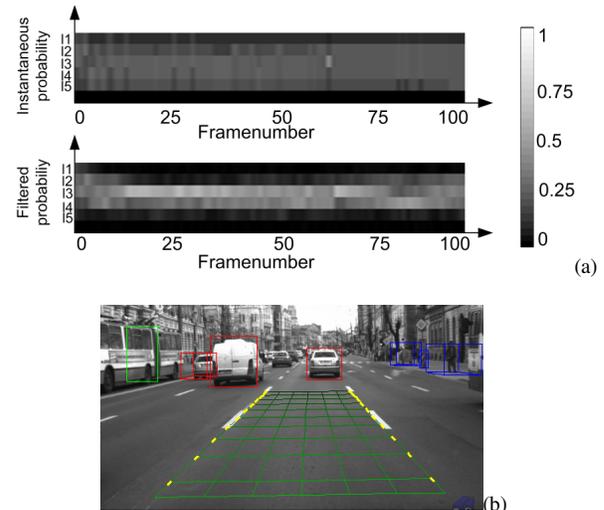


Fig. 11 Five lanes per driving direction (a) Comparative results of the instantaneous posterior probabilities provided by the BN against the filtered probabilities resulted from the particle filter (b) Example of instantaneous visual evidence (left boundary = interrupted, right boundary = interrupted, left outgoing vehicles) and resulted instantaneous probabilities  $P(l_1, l_2, \dots, l_{MAX}) = \{0.10, 0.20, 0.25, 0.25, 0.20\}$  and filtered probabilities  $P(l_1, l_2, \dots, l_{MAX}) = \{0, 0.1, 0.45, 0.42, 0.03\}$ .

Finally, Fig. 12 illustrates a case of lane change. The road has three lanes per driving direction and the host vehicle is passing from the middle lane to the leftmost lane. It can be noticed that the type of left lane boundary changes from interrupted to double, which considerably influences the Bayesian Network to identify the lane as being the leftmost one. Furthermore the particle filter emphasizes this result, as shown in the graphics of Fig. 12.

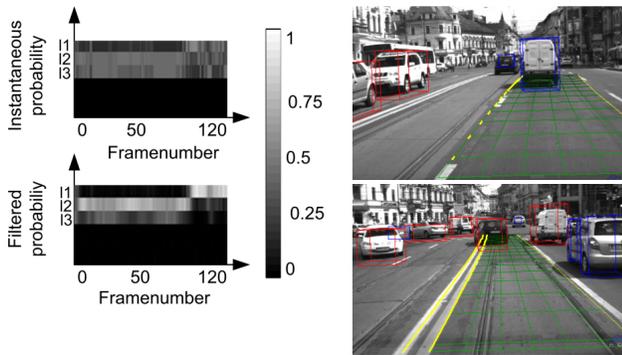


Fig. 12 Lane change example. Between frames 85 and 95 the vehicle is changing the lane from  $l_2$  to  $l_1$ ; this is much clearer from the filtered probability than from the instantaneous probability.

## VI. CONCLUSION

This paper presents a system that solves the difficult problem of lateral positioning on a multi-lane road, using simple visual cues produced by a stereovision-based sensorial system. Due to the fact that accurate classification of the types of lane boundaries is essential for road position inference, the first contribution of this paper is a novel and robust algorithm for delimiter type identification. The delimiter type information, together with other cues extracted from stereovision, such as the presence of obstacles, or the presence of arrows on the lanes, is fused into a Bayesian Network decision system which is able to estimate instantaneous probabilities for each lane position hypothesis. The results of the single frame Bayesian decision system are filtered in time, using a Particle Filter which significantly improves the overall results.

As seen from the testing section, the system provides reliably good results, and yet there is room for future work. The lane boundary classification system will greatly benefit from automatic identification of the marking quality. False classification results may be withheld if we knew that the markings are worn out, or if they are occluded by dirt or snow. Also, we will extend the training of the boundary classifier to more scenarios, so that all relevant conditions will be covered. The lane position decision system will be augmented with the use of a vehicle motion model, integrating speed, yaw rate, and possibly visual-based odometry, so that changes in lane position may be properly anticipated.

## ACKNOWLEDGMENT

The Bayesian Network described in this paper was created using the GeNIe modeling environment developed by the Decision Systems Laboratory of the University of Pittsburgh (<http://dsl.sis.pitt.edu/>).

## REFERENCES

[1] N. Mattern, R. Schubert, and G. Wanielik, "Lane Level Positioning using Line Landmarks and High Accurate Maps," in *Proceedings of the 16th World Congress on Intelligent Transportation Systems*, Stockholm, Sweden, 2009.

[2] J. Wang, S. Schroedl, K. Mezger, R. Ortloff, A. Joos, and T. Passegger, "Lane keeping based on location technology," *Trans. Intell. Transport. Sys.*, vol. 6, pp. 351-356, September 2005.

[3] A. Rae and O. Basir, "Improving Vehicle Positioning and Visual Feature Estimates through Mutual Constraint," in *Proceedings of Intelligent Transportation Systems Conference*, Seattle, WA, 2007, pp. 778 - 783

[4] K. H. Choi, Park, S.Y., Kim, S.H, Lee, K.S, Park, J.H., Cho, S.I., "Methods to Detect Road Features for Video-Based In-Vehicle Navigation Systems," *Journal of Intelligent Transportation Systems*, vol. 141, pp. 13 - 26, 20 October 2010.

[5] T.-S. Dao, K. Y. Leung, C. M. Clark, and J. P. Huissoon, "Markov-based lane positioning using intervehicle communication," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, pp. 641-650, December 2007.

[6] V. Popescu, M. Bace, and S. Nedevschi, "Lane Identification and Ego-Vehicle Accurate Global Positioning in Intersections," in *Proceedings of IEEE Intelligent Vehicles Symposium*, Baden-Baden, Germania, 2011, pp. 870-877.

[7] F. V. Jensen and T. D. Nielsen, *Bayesian Networks and Decision Graphs, Second Edition*. New York, USA: Springer Verlag, 2007.

[8] A. W. Moore, "Bayes Nets for representation and reasoning about uncertainty," ed, 2010.

[9] J. M. Collado, C. Hilario, A. d. l. Escalera, and J. M. Armigol, "Detection and Classification of Road Lanes with a Frequency Analysis," in *Proceedings of IEEE Intelligent Vehicles Symposium*, Las Vegas, Nevada, USA, 2005, pp. 78-83.

[10] T. Gavrilovic, J. Ninot, and L. Smadja, "Frequency Filtering and Connected Components Characterization for Zebra-Crossing and Hatched Markings Detection," in *IAPRS*, 2010, pp. 43-48.

[11] S. Vacek, C. Schimmel, and R. Dillmann, "Road-marking analysis for autonomous vehicle guidance," in *Proceedings of European Conference on Mobile Robots*, Freiburg, Germany, 2007, pp. 1-6.

[12] R. Danescu and S. Nedevschi, "Probabilistic Lane Tracking in Difficult Road Scenarios Using Stereovision," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, pp. 272-282, 2009.

[13] M. Hall, Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I. H., "The WEKA Data Mining Software: An Update," *SIGKDD Explorations*, vol. 11, 2009.

[14] R. Danescu and S. Nedevschi, "Detection and Classification of Painted Road Objects for Intersection Assistance Applications," in *Proceedings of IEEE Intelligent Transportation Systems Conference*, Madeira, Portugal, 2010, pp. 433-438.

[15] S. Bota, S. Nedevschi, and M. Koenig, "A Framework for Object Detection, Tracking and Classification in Urban Traffic Scenarios Using Stereovision," in *Proceedings of IEEE Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2009, pp. 153-156.

[16] J. Pearl, "Decision making under uncertainty," *ACM Comput. Surv.*, vol. 28, pp. 89-92, 1996.

[17] N. L. Zhang, "Irrelevance and parameter learning in Bayesian networks," *Artif. Intell.*, vol. 88, pp. 359-373, 1996.