Accurate Ego-Vehicle Global Localization at Intersections Through Alignment of Visual Data With Digital Map

Sergiu Nedevschi, *Member, IEEE*, Voichita Popescu, Radu Danescu, *Member, IEEE*, Tiberiu Marita, and Florin Oniga

***Abstract*—This paper proposes a method for achieving improved ego-vehicle global localization with respect to an approaching intersection, which is based on the alignment of visual landmarks perceived by the on-board visual system, with the information from a proposed extended digital map (EDM). The visual system relies on a stereovision system that provides a detailed 3-D description of the environment, including road landmark information (lateral lane delimiters, painted traffic signs, curbs, and stop lines) and dynamic environment information (other vehicles). An EDM is proposed, which enriches the standard map information with a detailed description of the intersection required for current lane identification, landmark alignment, and ego-vehicle accurate global localization. A novel approach for lane-delimiter classification, which is necessary for the lane identification, is also presented. An original solution for identifying the current lane, combining visual and map information with the help of a Bayesian network (BN), is proposed. Extensive experiments have been performed, and the results are evaluated with a Global Navigation Satellite System of high accuracy (2 cm). The achieved global localization accuracy is of submeter level, depending on the performance of the stereovision system.**

***Index Terms*—Alignment, Bayesian network (BN), extended digital map (EDM), localization, stereovision, visual landmarks.**

I. Introduction

THE INTERSECTION scenario is the most complex demanding and dangerous part of all driving situations. Depending on the region and country, 30%–60% of all accidents involving injuries and up to one-third of the fatalities occur at intersections. For this reason, the INTERSAFE-2 project aimed to develop and demonstrate new systems, algorithms, and technologies able to significantly reduce injuries and fatal accidents at intersections. The analysis of the user needs for intersection safety assistance systems identified the following driving assistance functions: left-turn assistance, intersection crossing assistance, right-turn assistance, right of way, and stop-line assistance. A key requirement of such systems is the accurate global localization of the ego-vehicle. Global Navigation Satellite Systems (GNSS) equipment has the following cost–accuracy ratio disadvantage. The standard GNSS receivers that come at accessible costs provide low accuracy, on the order of meters, whereas the high-precision GNSS equipment that provides centimeter-level accuracy has a high associated cost. Therefore, neither one is a viable solution for any driver assistance systems dedicated to intersection safety. As a result, software solutions for submeter localization are being investigated.

In this paper, we will present an original approach for improving the ego-vehicle's global localization given by a standard GPS, by using a visual perception system and a more detailed digital map. The proposed method relies on discriminative visual landmarks (lane markings, curbs, painted traffic signs, and stop lines) and on their corresponding accurate positions in an extended digital map (EDM). The visual system used in this approach uses stereovision [1] to sense the environment in front of the vehicle and has a large horizontal field of view. *A priori* information about the environment is stored in an EDM that contains detailed information about the road's topology, geometry, and geography, which are necessary for the correspondence with the visual data. The localization process itself is a two-step approach. First, the ego-vehicle driving lane is identified by matching the visual information with the map information, in a probabilistic framework in the form of a Bayesian network (BN). Next, the accurate coordinates of the ego-vehicle are obtained by aligning the corresponding landmarks from the visual system and from the EDM.

The rest of this paper is structured as follows. Section II presents a literature review regarding localization and the increased use of digital maps for this task. Section III is an introduction to BNs. Section IV is an overview of the proposed solution. Section V introduces the proposed EDM. Section VI presents a novel lane-delimiter classification method. Section VII shows the proposed accurate global localization method. Section VIII is dedicated to the experimental results, and Section IX concludes this paper.

II. Literature Review

The latest approaches to improved global localization of the ego-vehicle use the idea of combining information from various input systems, such as GNSS, vehicle sensors, on-board perception systems, digital maps, infrastructure monitoring systems, and cooperative systems. The GNSS is still the core

Manuscript received April 12, 2012; revised October 25, 2012; accepted November 1, 2012. Date of publication December 20, 2012; date of current version May 29, 2013. This work was supported in part by the Romanian National Authority for Scientific Research (CNCS–UEFISCDI) under Grant PN-II-ID-PCE-2011-3-1086 and in part by the EXCEL Project under Contract POSDRU/89/1.5/S/62557. The Associate Editor for this paper was R. I. Hammoud.

The authors are with the Department of Computer Science, Technical University of Cluj-Napoca, Cluj-Napoca 400020, Romania (e-mail: sergiu.nedevschi@cs.utcluj.ro; voichita.popescu@cs.utcluj.ro; radu.danescu@cs.utcluj.ro; tiberiu.marita@cs.utcluj.ro; florin.oniga@cs.utcluj.ro).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2012.2228191

technology for global localization, but additional systems are required to combat its shortcomings such as positioning errors, loss of signal, etc. In [2], a Kalman filter is used to integrate vehicle-sensor information with differential GPS (DGPS) information for localization. In [3], vehicle sensors are integrated with low-cost GPS in an interacting multiple mode (IMM) filter for localization. The success of the method results from the proposed IMM filter that adapts the vehicle model to different driving conditions. In [4], Chausse *et al.* propose combining data from GPS, vehicle sensors, vision sensors, and a very precise digital map (NavTech) in a particle filter, for estimating the localization parameters. The same input data (visually detected landmarks, map data, and vehicle odometry) are used in [5] for localization. A more complex approach is presented in [6], where a previously built detailed digital map is used for correlation with on-the-spot light detection and ranging measurements. The map is built using state-of-the-art equipment (inertial navigation system (INS), SICK laser rangefinders, and GPS). In addition, in [7], the visual information is used together with information from a geographic information system for improving localization. A comprehensive survey about vehicle positioning and navigation can be found in [8]. A number of visual localization methods based on the correspondence between the real-world objects and their image projections are available in the literature. In [9], Haralick *et al.* propose a method for determining the camera orientation by making the correspondence between 3-D object world coordinates with 2-D object image coordinates. The method considers the simultaneous correspondence between multiple geometric features (points, lines, and ellipse–circle); the resulting redundancy improves the results. In [10], a method through which the 3-D shape and orientation of objects are recovered from 2-D images is presented. In [11], Seetharaman and Le tackled the problem of video-assisted global positioning of an airborne camera using aerial images of known landmarks (known as the perspective-n-point problem). A minimum of three collinear landmarks is required. Having the position of the camera and a minimum of three known landmarks in at least two image frames, the position of a ground object can be computed. Hardware solutions are proposed for both camera and object localization tasks.

The necessity of a detailed digital map has been widely recognized, and research has been initiated by map builders (TeleAtlas and NavTeq) and by other projects (DARPA, SAFESPOT, INTERSAFE2) in the direction of providing advanced digital map solutions for driving assistance systems (DASs). Some of the researched solutions are the Route Network Definition File (RNDF) [12], the Road Graph [13], and Local Dynamic Map (LDM) [14].

The idea of lane-level positioning has gained more attention due to the possible DAS that could benefit from it, such as lane keeping [15], lane-level navigation [16], lane maneuver recommendation [17], hazard warning for a given trajectory, and services such as intelligent speed adaptation and lane allocation [18]. Different approaches have been proposed in this direction, many of which use high-detail high-accuracy digital maps. In [19], Toledo-Moreo *et al.* present a level positioning method that fuses the information from a GNSS/European Geostationary Navigation Overlay Service, a gyroscope, an odometer, and a proposed enhanced digital map (Emap). In [15], Wang *et al.* used a high-precision GNS/INS together with a vision system and an Emap for lane-level localization; the solution is used for a lane-keeping DAS. In [20], Du and Barth propose a lane-level positioning method that uses an off-the-shelf DGPS system and a self-built high-accuracy digital map. The lane-level position is obtained through a proposed Bayesian probabilistic map-matching algorithm. In [16], an INS is aided by DGPS and vision measurements for localization in challenging environments. *A priori* measured map landmarks are used for improving the localization. In [21], a low-cost lane positioning solution is proposed, which used only GPS and intervehicle communication.

The main contribution of this paper is a visual-based global localization method dedicated to intersection scenarios. The proposed original approach consists of two steps. First, the lane on which the vehicle is driven is identified based on the visual and map information. Then, when the stop line is visually detected, it is aligned with the corresponding accurately positioned stop line from the digital map. For lane identification, an original probabilistic method based on a BN is proposed. This choice is motivated by the fact that BNs represent a framework for modeling human-like reasoning under uncertain measurements while providing a solid mathematical foundation at the same time. Compared with our previous work [22], the method proposed in this paper for lane identification brings the following significant contributions:

- an original network structure that incorporates information about both static and dynamic environments (road landmarks and other vehicles, respectively);
- a parameter learning step, which ensures the generality and applicability of the method for any road segment, with any given number of lanes, provided that the road infrastructure is *a priori* known from the EDM;
- a temporal filtering mechanism of the frame-by-frame results provided by the BN, based on a particle filter, which considerably improves the inference over time.

Another contribution of this paper is the original method for lane-delimiter classification, with the delimiter types being relevant cues for lane identification.

Hence, in this paper, an accurate global localization method is proposed, which is based on a stereovision system and an EDM. Since both technologies have known exponential developments over the last years, the proposed method is feasible and implementable, from a technological and economical point of view, in the near future.

III. Bayesian Network Overview

BNs [23], [24] are part of the family of probabilistic graphical models, combining principles from mathematical and engineering domains, such as probability theory, statistics, and graph theory. They have an intuitive graphical representation in the form of a directed acyclic graph. Each random variable of the problem domain is represented by a node of the graph, with each node having a set of probable values called states. The nodes are connected with edges, indicating the direction of influence. The parameters of the network are the probabilities

of the nodes: *prior probabilities* for nodes with no parents, and *conditional probability tables* for nodes with parents. The importance of the graph is that it encodes the *conditional independence* assumption that each variable is independent of its nondescendents, given its parents in the graph. The BN itself encodes the joint probability distribution (JPD) over the finite set of variables represented as nodes in the graph. Due to the *conditional independence* property provided by the graph, the JPD can be decomposed into a product of conditional probability distributions over each variable given its parents in the graph, i.e.,

$$P(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} P\left(x_i \mid \text{parents}(x_i)\right). \quad (1)$$

The advantage of having the JPD in a factored form is that it makes it possible to evaluate all inference queries and that the time of computation is significantly reduced.

The goal of a BN is to infer the states of the unobservable (immeasurable) variables, given the evidence of the observable (measurable) variables; this is achievable in a BN through the *inference* mechanism that propagates the evidence of the observable nodes in the entire network and updates the beliefs of the other nodes. There are several inference algorithms that perform well in complex BNs such as the variable elimination algorithm, polytree (Pearl's algorithm restricted to a graph with no cycles), variation message passing, relevance tree, and others.

Due to its flexibility and its intuitive graphical representation, while providing a solid probability theory, the applications of the BN have grown from artificial intelligence to computer vision and autonomous mobile robots [25]–[28], domains that are dealing with uncertainty. In [28], Choi *et al.* propose the use of a BN model in order to fuse the information from a visual system with the information from a database to locate an approaching intersection. This idea is further developed by Choi *et al.* in [29], where they propose the use of a similar BN model for data fusion but, this time, to detect the driving lane.

IV. Solution Overview

Fig. 1 shows the overview of the proposed approach for an accurate global localization; the original contributions of this paper are highlighted with gray hashed boxes. The technologies employed in this approach are a stereovision perception system, a standard GPS receiver, and a proposed EDM; they are shown in Fig. 1 with gray rounded boxes. While the stereovision perception system provides real-time information about the surrounding environment, the EDM is populated ahead of time and provides *a priori* knowledge about the road's infrastructure.

The stereovision system's perception functions provide information about static and dynamic environments. The visual information used for localization consists of lane information [30], lane-delimiter localization and classification, painted-arrow localization and classification [31], curb localization [32], stop-line localization [33], and information about the other vehicles (relative position and velocity) [34]. Most of the visual perception functions providing these data are the result of

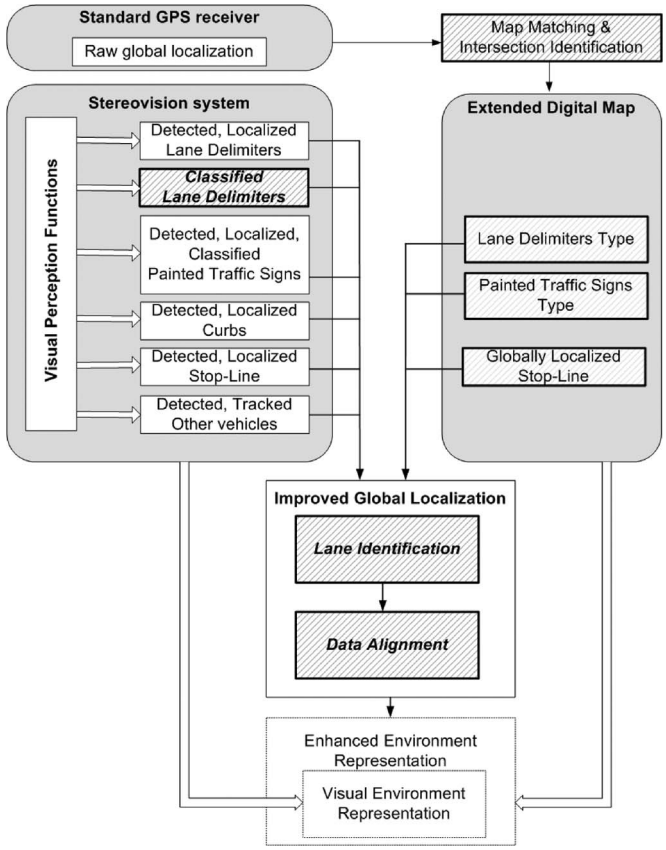


Fig. 1. Overview of the proposed approach for improved global localization.

previous work of the Image Processing and Pattern Recognition Group at the Technical University of Cluj-Napoca. The visual perception function responsible for lane-delimiter classification represents a novelty and a contribution of this paper.

The necessity of high-detail digital maps for accurate localization has been widely recognized and accepted by the intelligent transportation systems community. In this approach, we used our proposed EDM to correlate the visual landmarks with the landmarks in the map for position estimation. The proposed EDM was built based on *OpenStreetMap*, which is augmented with additional information about the geometry, topology, and geography of the road's landmarks.

For lane identification and subsequent data alignment, the position of the ego-vehicle on the map is required. The road segment that the ego-vehicle is located on is determined through a map-matching method, and the intersection toward which the ego-vehicle is heading is determined using the vehicle's orientation. In this way, the *a priori* information about the road segment that the ego-vehicle is currently on is obtained. Since these are straightforward approaches, they will not be detailed in this paper.

The goal of this paper is to provide a vision-based localization method for intersection scenarios. The original two-step localization method is also an important contribution of this paper. The steps are as follows.

1) The first step is lane identification, i.e., determining the lane on which the ego-vehicle is traveling. This is achieved through a probabilistic approach in the form of a BN, whose frame-by-frame results are time filtered

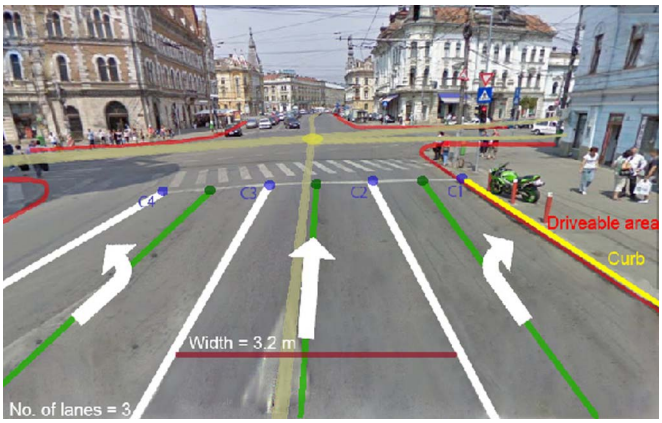


Fig. 2. Example of study case intersection. Extra features introduced in the proposed digital map are necessary for global localization.

using a particle filter. The structure and the parameters of the network are automatically learned based on the map information.

2) The second step is data alignment, i.e., determining the accurate global position of the ego-vehicle, by aligning the visual landmarks (stop line bordered by lateral delimiters) with the corresponding accurately positioned map landmarks. The detection and 3-D localization of the stop line used for alignment is presented in the earlier work of the authors [33].

V. Proposed Extended Digital Map

In this approach, we present a viable solution for extending an existing digital map, with detailed lane-level information, such as the one available in the RDNF format. As a starting point, the *OpenStreetMap* (www.openstreetmap.org) navigation map is used. This map was chosen because it is a free editable map of the world that allows viewing, editing, and using the geographical data. The proposed EDM uses the database structure provided by *OpenStreetMap* as the basic layer. On top of it, a second layer was constructed that supports all the new extra features that provide lane-level detail information. A utility tool was built for the management of the EDM data. Hence, we have extended the road-level detail of the digital map to a lane-level detail. Each *lane* entering the intersection has the following features: the lane width, the type of lane delimiters, the type of painted arrows, and the lane axis (a list of ordered nodes that are placed at the center of each lane). The lane axis is the centerline of a lane and has one of the end points on the painted stop line and the other end point at a distance of 30 m (the green lines in Fig. 2); the coordinates of these points are in a world geodetic system (latitude longitude height (LLH): WGS84). Fig. 2 shows one of the case study intersections used in this approach, i.e., a perspective view of the initial map data together with the new features introduced in the proposed EDM. The EDM is a solution on how to improve a standard digital map used for navigation with extra features containing information about the road's topology, geometry, and geography. The advantage of this proposal is that it brings the familiar structure of RNDF to a commercial free digital map application used for vehicle navigation, i.e., *OpenStreetMap*.

```
MarkingRegionLeft(1..100) = 0
MarkingRegionRight(1..100) = 0
ReferenceRegionLeft(1..100) = 0
ReferenceRegionRight(1..100) = 0
For i=1 to 100 do
   Zi = Z0 + i* 20 cm
   (XL, YL, XR, YR) = GetLaneGeometry(Z)
   For k = 0 to 10
   MarkingRegionLeft(i) += Img(Projection(XL-k*2 cm, YL, Z)
   MarkingRegionRight(i) += Img(Projection(XR+k*2 cm, YL, Z)
   ReferenceRegionLeft(i) += Img(Projection(XL+k*5 cm, YL, Z)
   ReferenceRegionRight(i) += Img(Projection(XR-k*5 cm, YL, Z)
   End For
End For
MarkingRegionLeft(1..100) /= 100
MarkingRegionRight(1..100) /=100
ReferenceRegionLeft(1..100) / = 100
ReferenceRegionRight(1..100) /= 100
```

Fig. 3. Pseudocode for basic feature extraction from the grayscale image.

VI. Lane-Delimiter Classification

A robust lane boundary classification technique, which relies on the frequency analysis of the intensity profile of the lane limits, is presented here. Frequency-based boundary classification solutions are presented in [35] and [36]. Our solution is also related to the one presented in [37], with 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 of [37] encodes the presence of markings on neighboring scan lines as codes, i.e., sequence of codes forming regular expressions that 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 [30].

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 m (starting from the minimum reliably visible distance). The distance between these lines is 20 cm. Thus, we will cover a length of 20 m of a lane with 100 equally spaced lines. The following pseudocode (see Fig. 3) describes the process.

On each line, we will generate, for each lane boundary (left and right), two search regions (segments): one that 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 ten equally spaced points, i.e., 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 resulting image space points are shown in Fig. 4, with the white points for the marking search region, and the black points for the asphalt reference region.

The grayscale values of each point generated in this way are averaged for each segment of each lane boundary. Thus, for

Fig. 4. (a) Original grayscale image. (b) Search areas for the markings (white dots) and for the asphalt reference grayscale values (black dots). The position of the search areas is given by a lane tracking result.

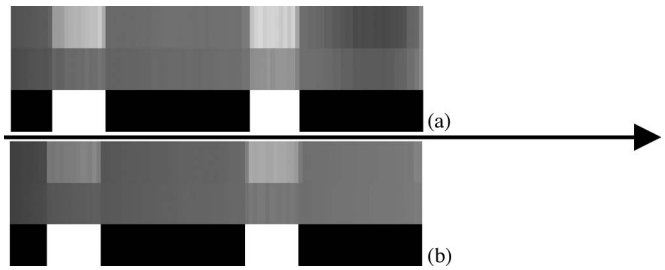


Fig. 5. Processing the search areas. For (a) the left and (b) the right lane boundaries. (Top row) Average intensity on each marking search line. (Middle row) Average intensity on each asphalt reference search line. (Bottom row) Whether the marking intensity is higher than the reference intensity (plus a threshold) for each line. The arrow shows the driving direction.

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; otherwise, we set a "0." Fig. 5 describes the process.

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. This encoding can be seen as a particular case of local binary pattern, which is a popular encoding used for pattern classification [38].

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 (particularly 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 are used to compute an offset, which will be used to shift the binary signal of the previous frame so that it becomes aligned to the binary signal extracted in the current frame. The results of the current frame will be averaged with the results of the past frames. In addition, we will expand the distance interval for our time-integrated signal, such that it will cover a total distance of 30 m (20 m in the visible range and 10 m behind the visible range). In 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 (see Fig. 6).

Fig. 6. (a) Time-integrated binary signal for the left boundary. (b) Result of Gaussian filtering of this signal.

C. Lane Boundary Classification

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 1's 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 [35], 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 (2) will be applied for t from 0 to 100 as follows:

$$P(t) = \sum_{k=0}^{N-t} |B(k) - B(k+t)| \tag{2}$$

where B is the filtered signal, and N is the total number of values B.

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, $t_{\text{second}} = 3t_{\text{first}}$.

Fig. 7. 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.

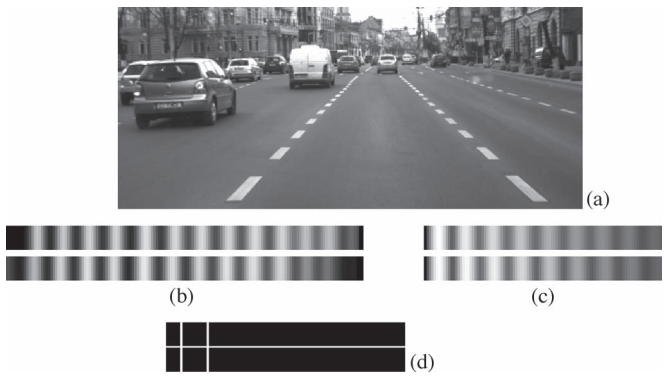


Fig. 8. Signal analysis. Merge boundaries on both sides, inside an intersection. (a) Original grayscale image. (b) Filtered binary signal B. (c) Period histogram P. (d) First and second local maxima of P.

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, $HalfPeriod = t_{\text{first}}$.
- The peak position ratio, $PeakRatio = t_{\text{second}}/t_{\text{first}}$.

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 [39]. Figs. 7 and 8 show several delimiter examples and the signal analysis for each of them.

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 nonfiltered 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 nonmarkings, based on their intensity. If the intensity of one pixel is closer to that of an already classified marking (see Section VI-A), it is labeled as marking, and if the intensity is closer to that of the reference area, it is labeled as nonmarking. The next step is to count the transitions between marking and nonmarking pixel sequences for each line of the search regions, as shown in Fig. 9. A normal delimiter should have two transitions, whereas 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. 9. Detection of double lines. (a) Search regions for counting marking/non marking transitions. (b) Transitions on a search line.

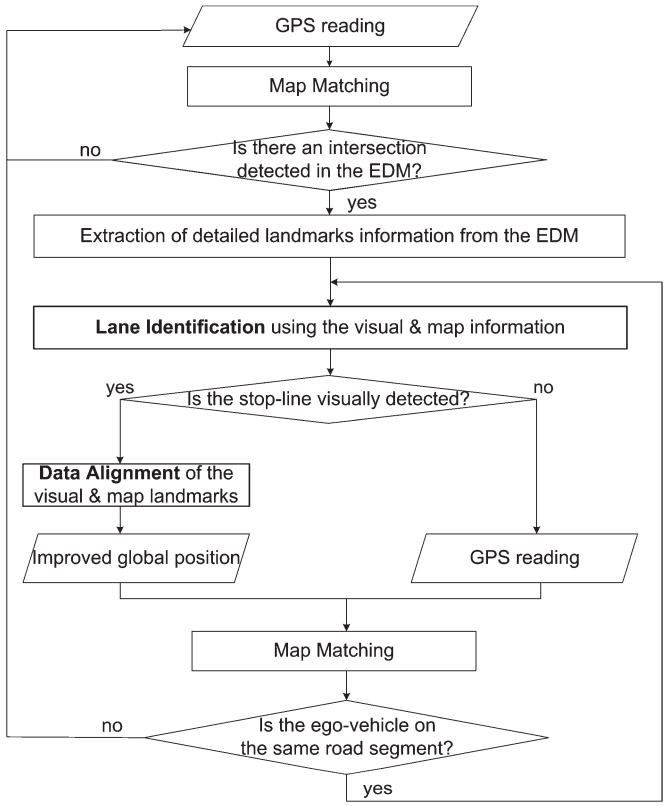


Fig. 10. Algorithm for accurate ego-vehicle localization.

VII. Accurate Ego-Vehicle Global Localization

Fig. 10 shows the basic flow of the proposed method for accurate global localization. Using the GPS receiver, the global position and the heading of the vehicle are obtained. This information is used to detect on which road segment the ego-vehicle is positioned (map matching) and the intersection that the ego-vehicle is approaching.

When the ego-vehicle is considerably close to the intersection (up to 50 m), the necessary information about the road segment on which the ego-vehicle is driven on is extracted from the EDM. This detailed map information is used together with visual evidence in the following algorithms: 1) lane identification, which uses the lateral landmarks; and 2) data alignment (when the stop line is visually detected), which uses the longitudinal landmark (stop line) bordered by the lane lateral landmarks. The result is the accurate global localization of the ego-vehicle. These two steps are repeated as long as the ego-vehicle is on the same road segment. This condition is evaluated by map matching either the global position obtained through data alignment, if the stop line has been visually detected, or the latest GPS reading, if otherwise. If the map matching returns the same road segment, the localization process continues; if not, the GPS reading continues until the ego-vehicle is approaching another intersection.

A. Lane Identification Through a BN

A key element in the localization process is lane identification. In this paper, a novel approach is proposed in the form of a BN for inferring the ego-lane, by combining the information about the static environment provided by the visual perception system with the corresponding information in the proposed EDM. The lateral landmarks are used in identifying the ego-lane. The type of lateral lane delimiters (*double, single, interrupted, and merge*) is determined by the previously presented visual perception function. When available, the type of painted arrows is also used. (The detailed presentation of the visual perception function dedicated to this task can be found in [31]).

A general remark regarding the usefulness of the information provided by the other vehicles is the following. In the case in which there are more than three lanes per driving direction, the type of lateral lane delimiters is no longer discriminatory information, and it becomes difficult to correctly and uniquely identify the lane. In such cases, the relative position and traveling direction of the detected vehicles can provide useful information in discriminating between the lanes with equal probability.

Therefore, using each detected vehicle's relative velocity and relative lateral position, the following features are extracted:

1) vehicle's traveling direction, which can be either the same or the opposite direction with respect to the ego-vehicle (Outgoing or Oncoming, respectively);
2) vehicle's lateral position, which can be on a left lane or a right lane with respect to the ego-lane (Left or Right, respectively).

Based on these two features, the vehicles detected by the stereovision perception system are classified into the following three classes: 1) LeftOutgoing (vehicle that is driving in the same direction, on a left neighbor lane); 2) RightOutgoing (vehicle that is driving in the same direction, on a right neighbor lane); and 3) LeftOncoming (vehicle that is driving in the opposite direction, on a left neighbor lane). Considering the right-side traveling convention, there are no right oncoming vehicles. The vehicles that are driving perpendicular to the ego-vehicles' driving direction are not taken into consideration. These visually detected vehicles provide important information about the lane on which the vehicle is traveling. For example, a vehicle that is traveling in the same direction as the ego-vehicle, on a left neighboring lane, is an indicator that the vehicle is not on the leftmost lane. Furthermore, depending on the relative lateral distance between the ego-vehicle and the detected vehicle, the ego-vehicle must be on one of the remaining lanes. For instance, consider the case in which the ego-vehicle is driving on a road segment with four lanes per driving direction, which are L_1, L_2, L_3, and L_4 from left to right. If the stereovision perception system detects a vehicle traveling on a left neighbor lane, in the same direction, at a lateral distance that is approximately equal to the lane's width, then the ego-vehicle cannot be on the leftmost lane L_1, and the remaining three lanes $(L_2, L_3,$ and $L_4)$ receive as evidence an equal distributed probability. This is the evidence that the stereovision perception system brings to the proposed to the proposed network, to the nodes encoding the information related to other vehicles.

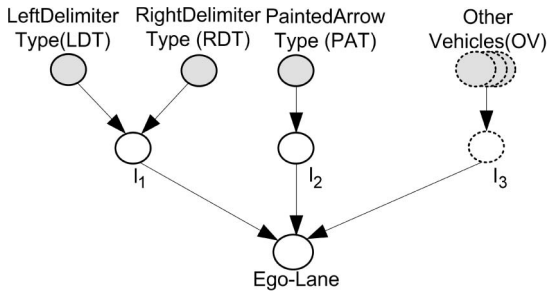


Fig. 11. BN proposed for the identification.

The BN used for inference is described in the following, which will detail the structure of the network, the parameter learning process, and the inference mechanism. The disadvantage of reasoning using a static BN is that it provides frame-by-frame results, without considering the evolution of the modeled event in time. To overcome this shortcoming, time filtering of the frame-by-frame results is introduced in the form of a particle filter, which is also described.

1) BN Structure: The construction is one of the most challenging and time-consuming aspect regarding BNs; it implies constructing the directed acyclic graph, i.e., the nodes and the edges (the relationship between the nodes), and estimating the parameters (the *a priori* probabilities and the conditional probability tables). In this case, the structure of the proposed network follows a predefined template, shown in Fig. 11. Depending on the map information and on the visual information about the other vehicles, some nodes may be or may not be added to the network. For example, if there are no vehicles detected, then the nodes encoding this information are not part of the network. The proposed network contains the following nodes.

a) The nodes that encode the information about the static environment (lateral landmarks): LeftDelimiterType (LDT), RightDelimiterType (RDT), and *PaintedArrowsType (PAT)*. The set of states for these nodes is defined by the type of left delimiters, right delimiters, and painted arrows, respectively, of all the lanes of the road segment on which the vehicle is driving on. Examples of states for the *LDT* and *RDT* are *double, single, interrupted,* and *curb*, and the examples of states for *PAT*, are

forward, left, right, forward-left, and *forward-right*. This information is provided by the EDM. These are observable nodes, and the evidence is provided by the stereovision perception system.

b) The nodes that encode the information about the dynamic environment (other vehicles): The purpose of such a node is to identify the ego-vehicle driving lane based on the position and orientation of the corresponding detected vehicles. The set of states for these nodes is $L_1, L_2, \ldots, L_n$, where n is the number of lanes per driving direction. The numbering starts from the leftmost lane in the driving direction and ends with the rightmost one. These are also observable nodes.

c) The final node is the hidden node, i.e., Ego-Lane node: Its states represent the possible lanes $L_1, L_2, \ldots, L_n$. This is an unobservable node, whose belief is computed through inference. The belief of this node is the answer to the question: "On which lane is the ego-vehicle traveling?"

d) The set of intermediate nodes $(I_k, k = 1, \ldots, 3)$*:* They have the same states as the *Ego-Lane* node $(L_1, L_2, \ldots, L_n)$ and have been introduced for the following reasons. They simplify the process of parameter learning, and they act as "parent divorcing," reducing the size of the *Ego-Lane* node's conditional probability table.

2) BN Parameter Learning: Given the structure of the network, its parameters must be learned. The parameters are estimated using the *a priori* map information about the road segment in question as training data. According to BN theory, if the training data set is complete, which is our case, the usual criterion used for parameter estimation is maximum-likelihood estimation (MLE). MLE is based on computing the probabilities that best match the training data set. Hence, the MLE method is based on finding θ^* that maximizes the likelihood of $P(D|\theta)$, where D is the training data set, and θ is the probability that the variable in question has a certain value. According to [40], for the case of discrete variables, we have

$$\theta^* = \frac{m_i}{\sum_i m_i} = \frac{\#\text{cases when } m_i \text{ discrete value appears}}{\text{total } \#\text{ cases}}. \quad (3)$$

Translating this to BNs, the *prior* probability of each state i of leaf node X, i.e., $\theta_i = P(X = i)$, is equal to

$$\theta_i^* = \frac{m_i}{\sum_i m_i} = \frac{\#\text{cases when } X = i}{\text{total } \#\text{ cases}}. \quad (4)$$

For child nodes, the conditional probabilities must be estimated, i.e., the probability for each state i of the child node, for each configuration k of its parent's node states $\theta_{\mathrm{ik}} = P(X = i|\mathrm{pa}(X) = \mathrm{k})$ must be computed. Using MLE, each parameter is computed using the following:

$$\theta_{ik}^* = \frac{m_{ik}}{\sum_i m_{ik}} = \frac{\#\text{cases } (X = i \text{ and parent}(X) = k)}{\#\text{cases (parent}(X) = k)}. \quad (5)$$

Hence, the parameters of the network in Fig. 11 are estimated using (4) and (5). For example, for a training data set of the form shown in Table I, the *prior* probabilities for *PAT* node will be $P(\text{left}, \text{forward}) = \{0.34, 0.66\}$ and the conditional probability table (CPT) of node I_2 will be the one shown in Table II.

TABLE I
EXAMPLE OF TRAINING

	PaintedArrow Type	I_2
States	*left*	L_1
	forward	L_2
	forward	L_3

TABLE II
CPT FOR I_2 NODE

PaintedArrow Type	I_2		
	L_1	L_2	L_3
left	1	0	0
forward	0	0.5	0.5

One of the most important advantages of the proposed method is the fact that the construction of the network is made automatically and generically. For each road segment that the vehicle is driven on, a corresponding BN is automatically constructed, in accordance with the map information.

3) BN Inference: While the information provided by the EDM is used for parameter estimation, the corresponding visual information provides the evidence for the leaf nodes on this network. In a BN, the reasoning is done through an *inference* mechanism, which propagates the evidence of the observable nodes in the entire network and updates the posterior probabilities of the other nodes. Thus, the belief of the *Ego-Lane* node is obtained as follows:

$$P(L_1, L_2, \ldots, L_n) = \{w_1, w_2, \ldots, w_n\}. \quad (6)$$

There are several inference algorithms that perform well in complex BNs; the one used in this approach is Pearl's algorithm [41]. A detailed example of how the BN infers the belief of the *Ego-Lane* node based on the map *a priori* data and on the visual evidence is presented in Section VIII.

4) Temporal Filtering of the BN Results: The position of the ego-vehicle on the lane is a time-evolving continuous process, and for this reason, tracking can significantly improve the final results. Therefore, the *Ego-Lane* node's beliefs are tracked over time using a CONDENSATION type of a particle filter [42]. The discrete posterior probability distribution over the lane hypotheses [see (6)] is passed on as measurement to the particle filter. The varying probability of the vehicle being located on each of the n lanes is encoded in a population of $N = 100$ particles, with each particle having value $l_i = L_1, \ldots, L_n$ (each particle is a hypothesis that the vehicle is located on lane l_i), and weight π_i. For each frame, the particle filter executes the following steps.

1) *Resampling*: From the past population of N weighted particles, we perform random extraction of a new set of N particles. The chance of a particle being selected for the new set is proportional to its weight.
2) *Diffusion*: The lane value i of each particle is changed randomly. Most of the values will remain the same, but a small percentage of these values are changed randomly to neighboring positions so that the tracking system can

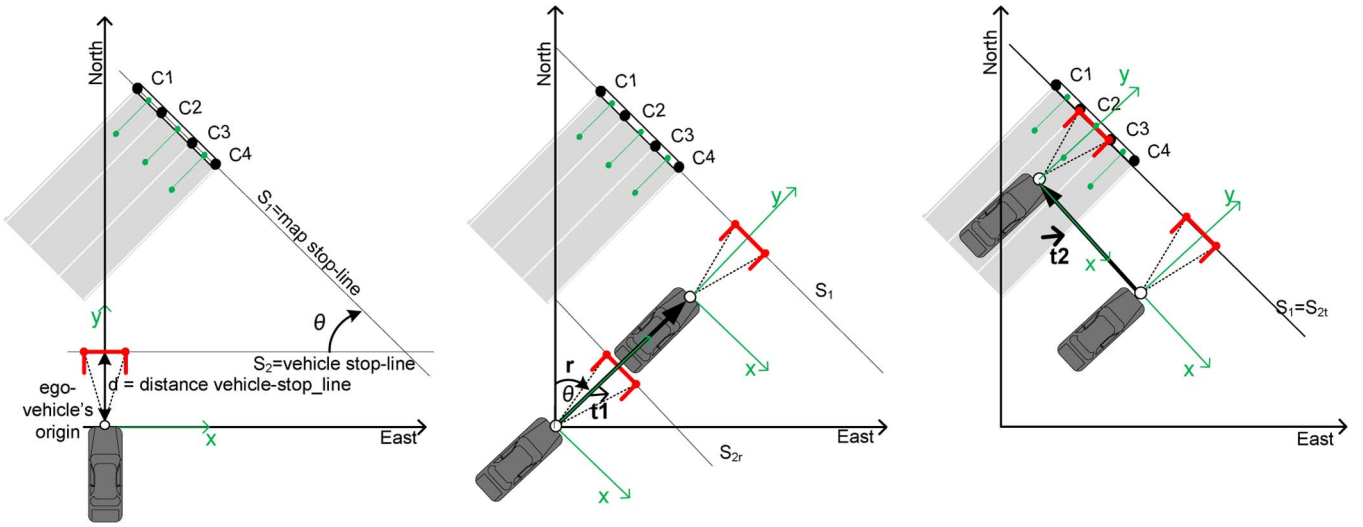


Fig. 12. (a) Initial configuration of the elements of the visual and EDM representations in the same coordinate system. The green segment represents the middle of each lane (the point on the stop line is the middle of the corresponding stop-line segment), the black points are the ends of the stop-line segments corresponding to each lane, and the red segment is the visual stop line bordered by the lateral delimiters. (b) Rotation, followed by longitudinal alignment. (c) Lateral alignment.

cope with changes in the world being tracked. Each particle will get a chance of r to alter its lane hypothesis to its left neighboring lane, a chance of r to alter its hypothesis to its right neighboring lane, 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 working toward the best hypothesis, under the penalty of disregarding other possibilities.

3) *Measurement or reweighting*: The instantaneous belief values of the current frame are assigned as weights to the particles. Each particle having the lane value $l_i = L_k$ receives as weight the instantaneous belief of the lane L_k, i.e., $\pi_i = w_k$. After all the particles receive their assigned weights, the weights are normalized. The probability that a vehicle is located on lane L_k becomes a simple summation of the weights of the particles having the same hypothesis, as shown in the following:

$$P(l_k) = \sum_{\text{particle}.l_i = L_k} \text{particle}.\pi_k. \tag{7}$$

The ego-lane is lane L^* having the highest tracked probability $P(L_k)$, as shown in the following:

$$L^* = \underset{k \in \{1,\ldots,n\}}{\arg\max}\ P(L_k). \tag{8}$$

The proposed temporal filtering considerably improves the performance of the lane identification as shown in Section VIII.

B. Data Alignment Algorithm

The proposed data alignment algorithm consists of properly aligning the reference visual road landmarks (the stop line bordered by the lane's lateral delimiters) with the corresponding accurately positioned map road landmarks (map stop-line segment corresponding to the identified lane). The visual landmarks are represented in the vehicle coordinate system, whereas the map landmarks are represented in a global coordinate system. The vehicle coordinate system is a Cartesian coordinate system, having as origin the middle of the ego-vehicle's front bumper, which is called the ego-vehicle's origin. The global coordinate system in which the map landmarks are accurately measured is the geographic coordinate system LLH, which is a non-Cartesian coordinate system. To superimpose the visual landmarks over the map landmarks, the first requirement is to bring them into the same coordinate system, which will be the East–North–Up (ENU) Cartesian coordinate system, which is generally used in targeting and tracking applications [43]. The origin of the new coordinate system (ENU) is the ego-vehicle's origin. Fig. 12(a) shows the initial configuration of the road landmarks from the two coordinate systems, i.e., vehicle and global, brought into the same coordinate system ENU. The steps of the proposed data alignment algorithm are as follows.

1) *Rotation*: The visual stop line S_2 is rotated around the ego-vehicle's origin with θ until it becomes parallel to the map stop line S_1 [see Fig. 12(b)]. Rotation angle θ is the angle between the visual stop line and the map stop line [see Fig. 12(a)].
2) *Longitudinal translation through which the rotated visual stop* line S_{2r} is superimposed over the map stop line S_1 [see Fig. 12(b)]: The same translation is applied to the ego-vehicle's origin. The translation vector is computed as the distance between the rotated visual stop line S_{2r} and the map stop line S_1.
3) *Lateral translation through which the visual stop line S_{2t} is superimposed over the segment of map stop line S_1 that corresponds to the identified lane* [see Fig. 12(c)]: The translation vector is computed using the visual stop line S_{2t} and the segment of the map stop line that corresponds to the identified lane. The same translation is also applied to the ego-vehicle's coordinates.
4) *Final Rotation*: In the case in which the map stop line is not perpendicular on the road axis, a *final rotation* is

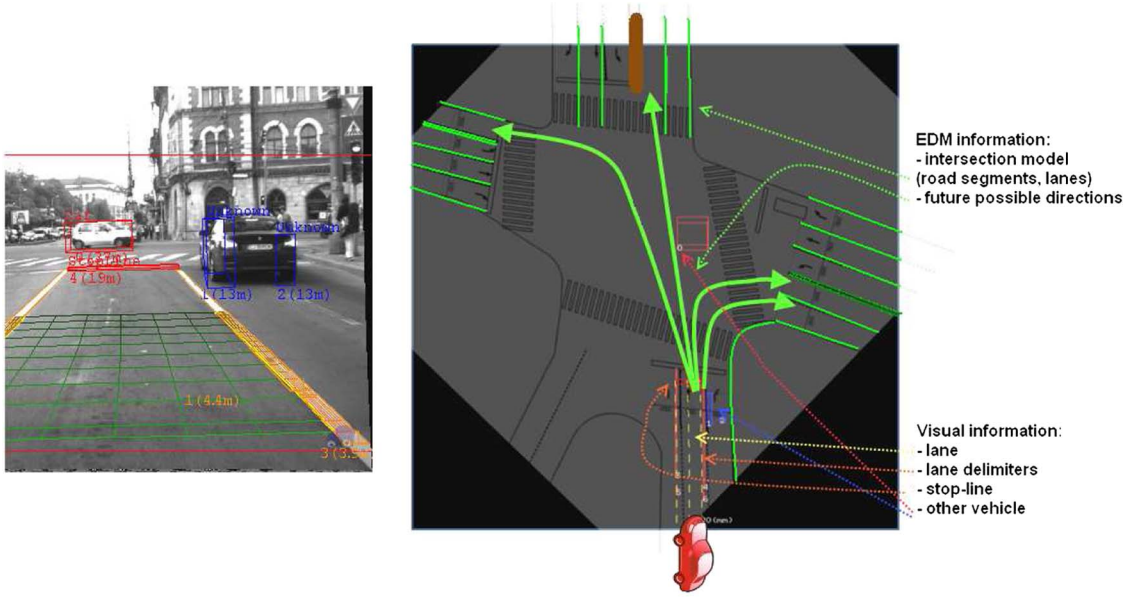


Fig. 13. (a) Visual perception outcomes. (b) Accurate global localization of the ego-vehicle prior to the intersection enables the fusion of the visual perception with EDM detailed information about the road infrastructure (road segments joint to the intersection, lane links that indicate future possible directions) resulting in an extended environment representation.

required to properly overlap the stop lines from the two representations.

Finally, the resulting ego-vehicle coordinates are transformed back from the local ENU coordinate system to the LLH global coordinate system, thus obtaining the new accurate ego-vehicle global coordinates.

C. Enhanced Environment Representation

Through this method, the accurate global localization of the vehicle with respect to the approaching intersection is obtained. Once this is achieved, the environment visual representation can be extended with the intersection representation information from the EDM, thus obtaining a more complex and complete picture about the intersection, which is useful for any safety and navigation-oriented DAS. Fig. 13(a) shows the visual perceived environment, and Fig. 13(b) shows the extended environment representation obtained by adding the intersection elements from the EDM: road segments joint to the intersection, the number of lanes for each road segment, the lane links indicating future possible traveling directions, and the trajectories through the intersection.

VIII. Experimental Results

A. Evaluation of the Lane-Delimiter Classification Method

The lane boundary classification system was tested using a sequence of 9830 frames, which records a drive of 8 km through the city of Cluj-Napoca, Romania. We tested the classification results for the left lane boundary against ground truth generated by manual labeling of intervals in the frame sequence. The reason why the left boundary was chosen for performance evaluation is that, throughout the sequence, this lane side passes through all the classes that 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.

TABLE III
Lane-Delimiter Classification Results

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

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 III describes the results for each lane boundary class. Some classes are better represented in the sequence, whereas 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 ratio between the number of incorrectly detected instances of a class versus the number of frames that the class was not actually present). It can be noticed that, while the TP rate is not extremely high, due to the conditions of the markings and some other causes of occlusion, the FP rate is quite low, which makes the system a robust solution for boundary classification.

The receiver operating characteristic (ROC) curve for the WEKA-generated decision-tree lane marking classifier is shown in Fig. 14. The curve, which is close to ideal, shows accurate classification behavior.

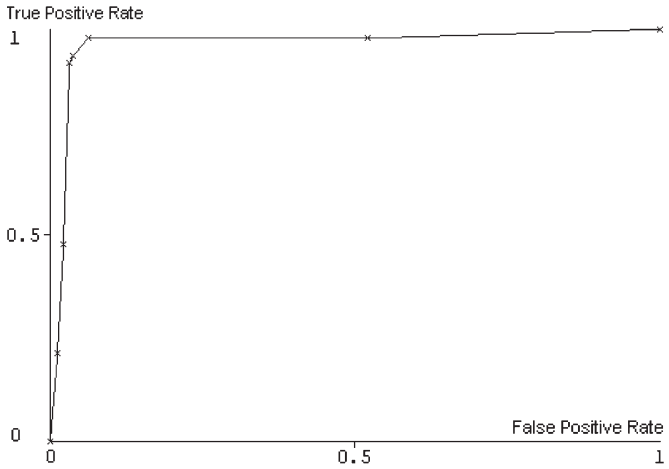


Fig. 14. ROC curve of the decision-tree lane marking classifier. The horizontal axis shows the false positive ratio, whereas the vertical axis shows the true positive ratio.

The data set, containing the images, the 3-D and 2-D coordinates of the points in the search regions described in this paper, the manually labeled class of the left delimiter, the timestamps and speed of the vehicle required for temporal filtering of the intensity signal of the markings, and the projection matrix of the camera, is available at http://users.utcluj.ro/~rdanescu/itsdata.rar as a single archive file. In this file, you will find a readme.txt document describing the format of the data and a Matlab script as an example of how these files should be read.

B. Evaluation of the Lane Identification Method

The proposed solution for improving the global localization was tested in real urban traffic scenarios under normal traffic conditions in the downtown area of Cluj-Napoca, Romania. For testing purposes, more than 30 intersections in Cluj-Napoca, Romania, have been modeled and measured; their detailed information have been manually introduced in the proposed EDM. This information is used online as input data by both the lane identification and data alignment modules. The modeling of the intersections was performed using satellite images and high-precision measurements using GNSS equipment of high accuracy (Leica 1200 Series System).

The following example shows how the BN works when different pieces of evidence are added to the nodes in the network that is presented. Consider the case in which the ego-vehicle is approaching the intersection on a road segment with the landmarks' configuration shown in Fig. 15. This represents the configuration of one road segment, from one of the considered case study intersections. This information about the structure of the road is available to the ego-vehicle from the proposed EDM and is used in the automatic BN construction.

The belief of the *Ego-Lane* node fluctuates according to the evidence brought by the visual perception system. Two study cases are presented in Figs. 16 and 17. We intend to outline the following two aspects: how the belief of the correct state increases while the visual evidence is consistent [see Fig. 16(c)], and how the belief is more uniformly distributed among the states of the *Ego-Lane* node, when the visual evidence is inconsistent [see Fig. 17(c)]. In Fig. 16, the hard evidence are the following: LDT = double, RDT = interrupted, RightOutgoing = L_1, (due to the vehicle detected on the rightmost lane), and LeftIncoming = L_1. In Fig. 17, the evidence for the type of lane delimiters is the hard evidence (LDT = interrupted, and RDT = interrupted. The other detected vehicles bring the following soft evidence: RightOutgoing = L_1, L_2, i.e., $P(\mathrm{OV}_1) = \{0.5, 0.5, 0.0\}$, and LeftOutgoing = L_2, L_3, i.e., $P(\mathrm{OV}_2) = \{0.0, 0.5, 0.5\}$.

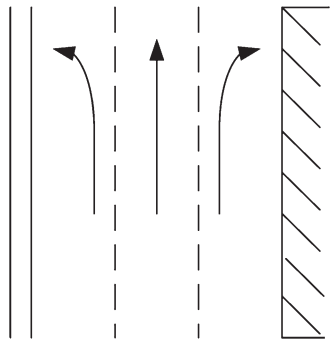
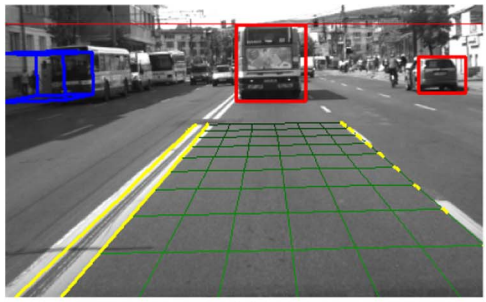
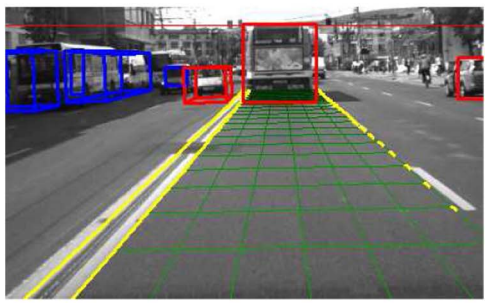


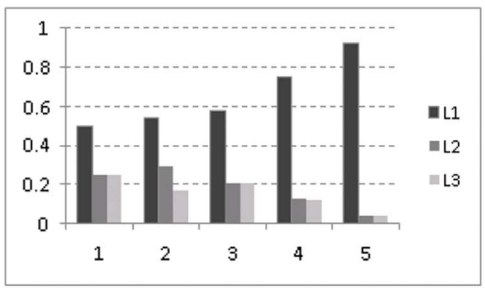
Fig. 15. Lane-level detailed information from the EDM.



(a)

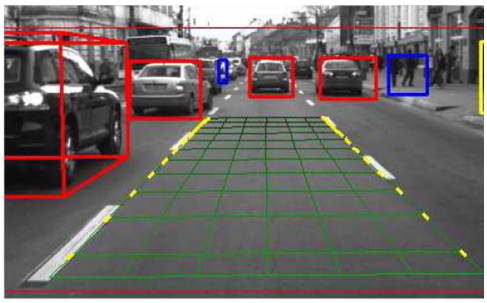


(b)

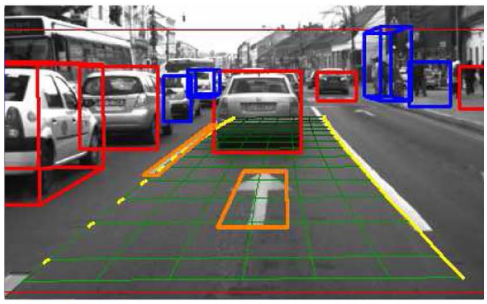


(c)

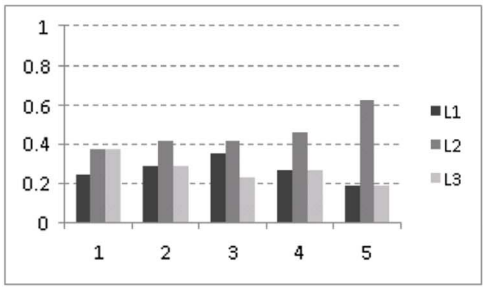
Fig. 16. Visual Evidence. (a) LDT = double, RDT = interrupted, and RightOutgoing = L_1. (b) LeftIncoming = L_1. (c) Belief of the Ego-Lane node when hard visual evidence sustains the correct hypothesis L_1.



(a)



(b)



(c)

Fig. 17. Visual Evidence. (a) LDT = interrupted, RDT = Interrupted, RightOutgoing = (L_1, L_2), and LeftOutgoing = (L_2, L_3). (b) PAT = forward. (c) Belief of the Ego-Lane node when soft visual evidence sustains both L_1 and L_2 hypotheses, with the correct one being L_2.

Finally, the PAT detected brings a significant contribution to the lane identification since this feature discriminates among the lanes, as shown in Fig. 15. Hence, the hard evidence $\mathrm{PAT} = forward$ makes the probability distribution over the set of state considerably sharper $P(L_1, L_2, L_3) = \{0.19, 0.62, 0.19\}$ [see Fig. 17(c)]. The final remark is that, in both cases, by cumulating the hard evidence and the soft evidence, the ego-lane is correctly identified, i.e., leftmost lane L_1 in the first case, and the middle lane L_2 in the second case.

The qualitative evaluation of the lane identification algorithm was performed by the assessment of the number of *correct* and *unique lane identifications, correct but multiple lane identifications,* and *incorrect lane identifications* reported by the algorithm. *Correct* and *unique lane identification* means that the algorithm correctly and uniquely identifies the lane that the vehicle is traveling on. *Correct but multiple lane identification* means that several of the lanes of the road segment have an equal posterior probability in the BN; hence, the exact ego-lane cannot be distinguished, and the actual lane is among these lanes. This happens most often in the case when there are more than three lanes per way, and the lateral landmarks are the same for several lanes. Finally, incorrect lane identification means that the algorithm has identified another lane rather than the actual lane as the ego-lane. This behavior is caused by occasional erroneous results of the stereovision sensor. Real urban traffic scenarios were used for this test, among which road segments with three to six lanes per driving direction where encountered. The scenarios sum a total number of 10 155 image frames. The results of the proposed lane identification method are presented in Table IV; both the results of static BN approach and the improvements brought by adding the particle filter are illustrated.

TABLE IV
LANE IDENTIFICATION RESULTS

Lane Identification	Static BN [%]	Adding the particle filter [%]
Correct and unique lane identification	79.1	91.1
Correct but multiple lane identification	15.3	5.4
Incorrect lane identification	5.6	3.5

C. Evaluation of the Data Alignment Algorithm

The test case scenario consists of the ego-vehicle driving through the study case intersections on one of the adjacent road segments whose detailed information has previously been modeled and entered in the EDM. The initial ego-vehicle GPS position is obtained with a standard GPS receiver, with a position precision degree of 5 m and an update rate of 1 Hz. The proposed method is designed to improve the accuracy of this position estimate.

Fig. 18 shows the results of the proposed algorithm for accurate global localization, on a sequence in which the vehicle is approaching one of the case study intersections. Several frames from the sequence are considered, in which the vehicle is at different distances from the stop line, as shown in the first column. The second column shows the correctly identified ego-lane (together with the lane belief resulting from the proposed method) and a perspective view of the perceived environment representation. The third column shows the new and more accurate ego-vehicle localization (green vehicle), resulting from the data alignment algorithm, versus the initial GPS localization (red vehicle), superimposed on a sketch of the intersection. The conclusion is that the proposed method considerably improves the standard GPS localization, reducing the error from meters level to centimeters level.

Table V shows the quantitative results of the proposed localization algorithm in terms of lateral and longitudinal error; the ground-truth position is provided by a Novatel OEM6 GNSS receiver, with accuracy of 2 cm. In addition, in Table V, the results of the proposed method are compared with related literature results. The conclusion is that the proposed method achieves good results with low-cost technology.

The processing time of each module, for the following system configuration, i.e., Intel Core2 Duo CPU, a 2.66-GHz processor, 2 GB of RAM, is presented in Table VI. For lane identification, the processing time depends on the amount of visual information used as evidence.

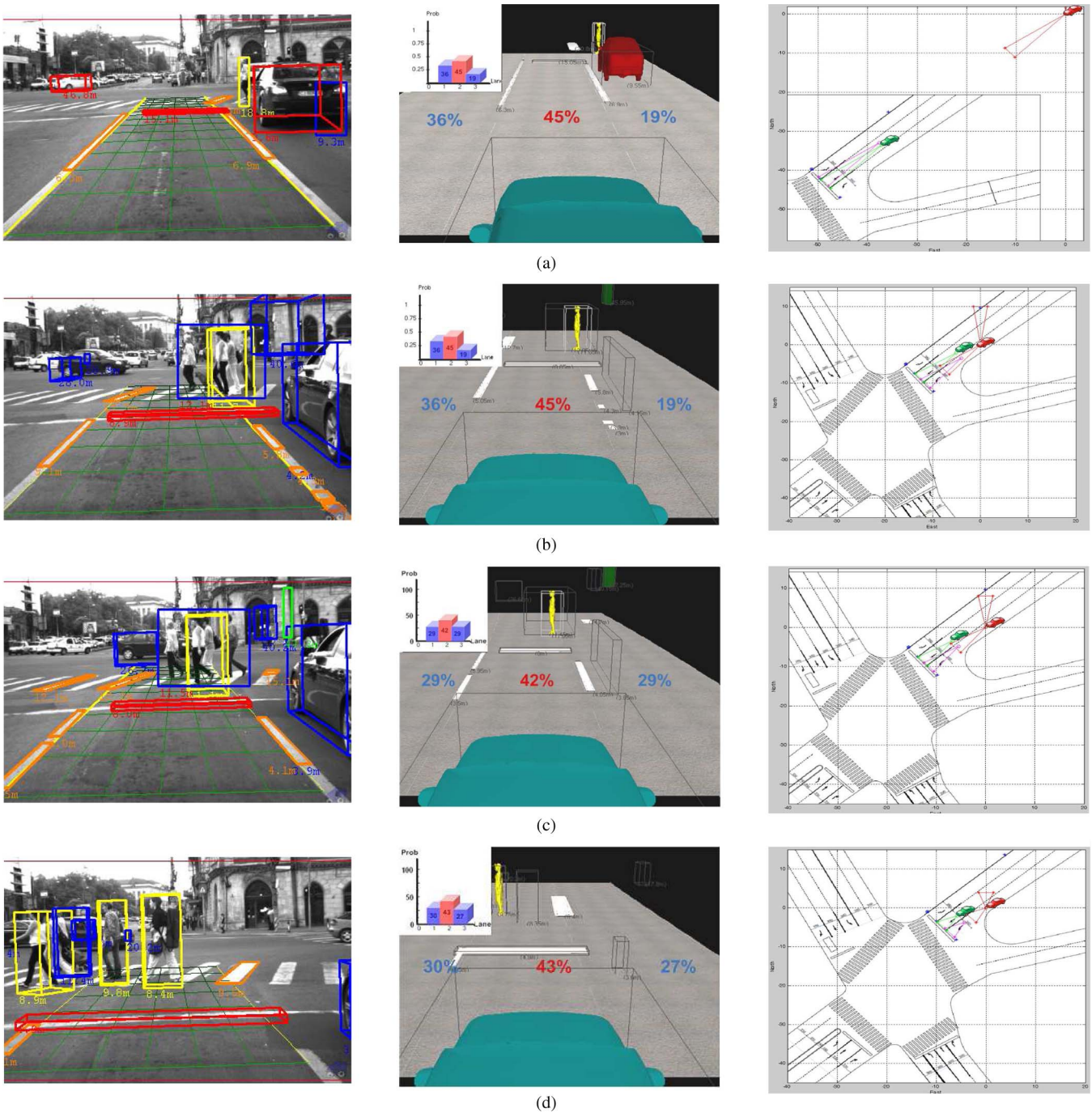


Fig. 18. Results of the proposed algorithm for accurate global localization. The different cases are for different distances to the stop line. (a) 15.1 m (b) 8.9 m (c) 8 m (d) 4.9 m. The first column shows the results of the visual perception system. The second column shows the visual environmental representation from the ego-vehicle point of view. The lane beliefs resulting from to the proposed method are the following. (a) $P(L_1, L_2, L_3) = \{0.36, 0.45, 0.19\}$. (b) $P(L_1, L_2, L_3) = \{0.36, 0.45, 0.19\}$. (c) $P(L_1, L_2, L_3) = \{0.29, 0.42, 0.29\}$. (d) $P(L_1, L_2, L_3) = \{0.30, 0.43, 0.27\}$. The third column shows the accurate ego-vehicle localization (green vehicle) versus the initial GPS localization (red vehicle).

An important remark is that the proposed method for accurate global localization relies on the visual detection of the road landmarks, and uses this information as input data for the lane identification and data alignment algorithms. In real-world conditions, occlusions happen quite frequently, leading to erroneous data or no data at all. The probabilistic approach in the form of a BN is capable of dealing with such uncertain information and, in most cases, is able to correctly identify the lane on which the vehicle is driving on, as shown in Table IV. In addition, the data alignment algorithm performs well in both good and poor traffic conditions, relying on the visual stop-line detection that has a very high detection rate (96.5%) [33].

IX. Conclusion

This paper described a method for ego-vehicle accurate global localization with respect to an approaching intersection, based on the alignment of data from the visual perception

TABLE V
DATA ALIGNMENT RESULTS

No.	Ref	Sensors	Lateral Position Error [m]	Longitudinal Position Error [m]
1		GPS	Orders of m	Orders of m
2	[4]	GPS + vehicle sensors + Map +Vision	0.08	0.48
3	[5]	GPS + odometry + Map +Vision	0.35	0.77
4	[6]	GPS + IMU + odometry + LIDAR + built map	<0.1	<0.3
5	Proposed	GPS + Stereovision + EDM	0.32	0.20

TABLE VI
PROCESSING TIME OF EACH MODULE

Module	Processing time [ms]
Painted road objects	5
Curb detection	8
Stop-line	1
Lane identification	15 – 30
Data alignment	1

system with the information from the proposed EDM. The main contributions of this paper are related to the following:

- the classification of 3-D lateral delimiters of the lane;
- the proposed EDM, which includes the map representation of the landmarks and supplementary information regarding the road geometry and intersection's configuration;
- the probabilistic method, in the form of a BN, for identifying the driving lane using the perceived visual and map information, i.e., the original structure of the network, the automatic parameter learning based on EDM information, the generic and automatic construction of the BN, the use of the visual information about the other vehicles in the network, and the temporal integration of the frame-by-frame lane identification;
- the algorithm for the alignment of the visual landmarks with the map landmarks, resulting in the improved global localization.

The method was successfully implemented and experimented on in several specific intersection scenarios. The proposed approach is able to provide increased accuracy in the global localization of the ego-vehicle.

The achieved accurate localization allows the fusion of the visual perceived information with the EDM information, increasing the completeness of the intersection representation.

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] S. Nedevschi, R. Danescu, T. Marita, F. Oniga, C. Pocol, S. Bota, M.-M. Meinecke, and M. A. Obojski, "Stereovision-based sensor for intersection assistance," in *Proc. Adv. Microsyst. Automot. Appl.—Smart Syst. Safety, Sustainability*, 2009, pp. 129–163.

[2] S. Rezaei and R. Sengupta, "Kalman filter-based integration of DGPS and vehicle sensors for localization," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 6, pp. 1080–1088, Nov. 6, 2007.

[3] K. Jo, K. Chu, and M. Sunwoo, "Interacting multiple model filter-based sensor fusion of GPS with in-vehicle sensors for real-time vehicle positioning," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 329–343, Mar. 2012.

[4] F. Chausse, J. Laneurit, and R. Chapuis, "Vehicle localization on a digital map using particles filtering," in *Proc. IEEE Intell. Veh. Symp.*, Las Vegas, NV, 2005, pp. 243–248.

[5] N. Mattern, R. Schubert, and G. Wanielik, "High-accurate vehicle localization using digital maps and coherency images," in *Proc. IEEE Intell. Veh. Symp.*, San Diego, CA, 2010, pp. 462–469.

[6] J. Levinson, M. Montemerlo, and S. Thrun, "Map-based precision vehicle localization in urban environments," in *Proc. Robot. Sci. Syst. Conf.*, Atlanta, GA, 2007.

[7] A. Rae and O. Basir, "Improving vehicle positioning and visual feature estimates through mutual constraint," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Seattle, WA, 2007, pp. 778–783.

[8] I. Skog and P. Handel, "In-car positioning and navigation technologies: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 4–21, Mar. 2009.

[9] R. Haralick, Q. Ji, M. Costa, and L. Shapiro, "A robust linear least squares estimation of camera exterior orientation using multiple geometric features," *J. Photogramm. Remote Sens.*, vol. 55, no. 2, pp. 75–93, Jun. 2000.

[10] W. J. Shomar, G. Seetharaman, and T. Y. Young, "An expert system for recovering 3D shape and orientation from a single view," in *Proc. Comput. Vis. Image Process.*, 1992, pp. 459–515.

[11] G. Seetharaman and H. V. Le, "Video-assisted global positioning in terrain navigation with known landmarks," *Int. J. Distrib. Sensor Netw.*, vol. 2, no. 2, pp. 103–119, 2006.

[12] Defense Adv. Res. Projects Agency, Arlington, VA, DARPA Urban Challenge, Route Network Definition File (RNDF) and Mission Data File (MDF), 2007, Last accesed: October, 2012. [Online]. Available: http://archive.darpa.mil/grandchallenge/docs/RNDF_MDF_Formats_031407.pdf

[13] H. K. Knaup Jorn, "Graph based environmental modelling and function independent situation analysis for driver assistance systems," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2010, pp. 428–432.

[14] Z. Papp, C. Brown, and C. Bartels, "World modeling for cooperative intelligent vehicles," in *Proc. IEEE Intell. Veh. Symp.*, Eindhoven, Holland, 2008, pp. 1050–1055.

[15] J. Wang, S. Schroedl, K. Mezger, R. Ortloff, A. Joos, and T. Passegger, "Lane keeping based on location technology," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 3, pp. 351–356, Sep. 2005.

[16] A. Vu, A. Ramanandan, A. Chen, J. A. Farrell, and M. Barth, "Real-time computer vision/DGPS-aided inertial navigation system for lane-level vehicle navigation," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 899–913, Jun. 2012.

[17] R. Schubert, K. Schulze, and G. Wanielik, "Situation assessment for automatic lane-change maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 607–616, Sep. 2010.

[18] R. Toledo-Moreo, D. Bétaille, F. Peyret, and J. Laneurit, "Fusing GNSS, dead-reckoning, and enhanced maps for road vehicle lane-level navigation," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 5, pp. 798–809, Oct. 2009.

[19] R. Toledo-Moreo, D. Betaille, and F. Peyret, "Lane-level integrity provision for navigation and map matching with GNSS, dead reckoning, and enhanced maps," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 100–112, Mar. 2010.

[20] J. Du and M. Barth, "Bayesian probabilistic vehicle lane matching for link-level in-vehicle navigation," in *Proc. IEEE Intell. Veh. Symp.*, Tokyo, Japan, 2006, pp. 522–527.

[21] T.-S. Dao, K. Y. Leung, C. M. Clark, and J. P. Huissoon, "Markov-based lane positioning using intervehicle communication," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 4, pp. 641–650, Dec. 2007.

[22] V. Popescu, M. Bace, and S. Nedevschi, "Lane identification and ego-vehicle accurate global positioning in intersections," in *Proc. IEEE Intell. Veh. Symp.*, Baden-Baden, Germany, 2011, pp. 870–877.

[23] F. V. Jensen and T. D. Nielsen, *Bayesian Networks and Decision Graphs*, 2nd ed. New York: Springer-Verlag, 2007.

[24] U. B. Kjrulff and A. L. Madsen, *Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis*. New York: Springer-Verlag, 2008.
[25] J. Forbes, T. Huang, K. Kanazawa, and S. Russell, "The BATmobile: Towards a Bayesian automated taxi," in *Proc. Int. Joint Conf. Artif. intell.*, Montreal, QC, Canada, 1995, pp. 1878–1885.
[26] E. Besada-Portas, J. A. Lopez-Orozco, and J. M. de la Cruz, "Unified fusion system on Bayesian networks for autonomous mobile robots," in *Proc. Int. Conf. Inf. Fusion*, 2002, pp. 873–880.
[27] C. Cappelle, M. E. El Najjar, D. Pomorski, and F. Charpillet, "Multi-sensors data fusion using dynamic Bayesian network for robotised vehicle geo-localization," in *Proc. Int. Conf. Inf. Fusion*, Cologne, Germany, 2008, pp. 1–8.
[28] K. H. Choi, S. Joo, S. I. Cho, and J. H. Park, "Locating intersections for autonomous vehicles: A Bayesian network approach," *ETRI J.*, vol. 29, no. 2, pp. 249–251, Apr. 2007.
[29] K. H. Choi, S. Y. Park, S. H. Kim, K. S. Lee, J. H. Park, and S. I. Cho, "Methods to detect road features for video-based in-vehicle navigation systems," *J. Intell. Transp. Syst.*, vol. 141, no. 1, pp. 13–26, Oct. 20, 2010.
[30] R. Danescu and S. Nedevschi, "Probabilistic lane tracking in difficult road scenarios using stereovision," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 2, pp. 272–282, Jun. 2009.
[31] R. Danescu and S. Nedevschi, "Detection and classification of painted road objects for intersection assistance applications," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Funchal, Portugal, 2010, pp. 433–438.
[32] F. Oniga and S. Nedevschi, "Polynomial curb detection based on dense stereovision for driving assistance," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Funchal, Portugal, 2010, pp. 1110–1115.
[33] T. Marita, M. Negru, R. Danescu, and S. Nedevschi, "Stop-line detection and localization method for intersection scenarios," in *Proc. IEEE Intell. Comput. Commun. Process.*, Cluj-Napoca, Romania, 2011, pp. 293–298.
[34] S. Bota, S. Nedevschi, and M. Konig, "A framewok for object detection, tracking and classification in urban traffic scenarios using stereovision," in *Proc. IEEE Intell. Comput. Commun. Process.*, Cluj-Napoca, Romania, 2009, pp. 153–156.
[35] J. M. Collado, C. Hilario, A. de la Escalera, and J. M. Armigol, "Detection and classification of road lanes with a frequency analysis," in *Proc. IEEE Intell. Veh. Symp.*, Las Vegas, NV, 2005, pp. 78–83.
[36] T. Gavrilovic, J. Ninot, and L. Smadja, "Frequency filtering and connected components characterization for zebra-crossing and hatched markings detection," in *Proc. IAPRS*, Saint-Mandé, France, 2010, pp. 43–48.
[37] S. Vacek, C. Schimmel, and R. Dillmann, "Road-marking analysis for autonomous vehicle guidance," in *Proc. Eur. Conf. Mobile Robots*, Freiburg, Germany, 2007, pp. 1–6.
[38] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognit.*, vol. 29, no. 1, pp. 51–59, Jan. 1996.
[39] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *SIGKDD Explor.*, vol. 11, no. 1, pp. 10–18, Jun. 2009.
[40] N. L. Zhang, "Irrelevance and parameter learning in Bayesian networks," *Artif. Intell.*, vol. 88, no. 1/2, pp. 359–373, Dec. 1996.
[41] J. Pearl, "Decision making under uncertainty," *ACM Comput. Surveys*, vol. 28, no. 1, pp. 89–92, Mar. 1996.
[42] M. Isard and A. Blake, "CONDENSATION—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, Aug. 1998.
[43] S. P. Drake, Converting GPS coordinates to navigation coordinates, Defence Sci. Technol. Org., Edinburgh, SA, Australia, (2012, Oct. 12). [Online]. Available: http://www.dsto.defence.gov.au/publications/2443/DSTO-TN-0432.pdf

Sergiu Nedevschi (M'99) received the M.S. and Ph.D. degrees in electrical engineering from the Technical University of Cluj-Napoca (TUCN), Cluj-Napoca, Romania, in 1975 and 1993, respectively.

From 1976 to 1983, he was with the Research Institute for Computer Technologies, Cluj-Napoca, as a Researcher. In 1998, he was appointed Professor of computer science and founded the Image Processing and Pattern Recognition Group, TUCN. From 2000 to 2004, he was a Department Head with the Department of Computer Science, TUCN. From 2004 to 2012, he was the Dean of the Faculty of Automation and Computer Science, TUCN. He is currently a Vice-Rector with TUCN. He is the author of more than 200 scientific papers and the Editor of over ten volumes, including books and conference proceedings. His research interests include image processing, pattern recognition, computer vision, intelligent vehicles, signal processing, and computer architecture.

Voichita Popescu received the Diploma Engineer and the M.S. degrees in computer science from the Technical University of Cluj-Napoca (TUCN), Cluj-Napoca, Romania, in 2009 and 2011, respectively, where she is currently working toward the Ph.D. degree in computer science.

She is currently with the Image Processing and Pattern Recognition Group, TUCN. Her research interests include localization, reasoning, and decision-making under uncertainty, as well as situation assessment with applications in driving assistance.

Radu Danescu (M'11) received the Diploma Engineer, the M.S., and the Ph.D. degrees, all in computer science, from the Technical University of Cluj-Napoca (TUCN), Cluj-Napoca, Romania, in 2002, 2003, and 2009, respectively.

He is currently a Senior Lecturer with the Department of Computer Science, TUCN, teaching image processing, pattern recognition, and design with microprocessors. He is also currently a member of the Image Processing and Pattern Recognition Group, TUCN. His main research interests include stereovision and probability-based tracking, with applications in driving assistance.

Tiberiu Marita received the Diploma Engineering and Ph.D. degrees in computer science from Technical University of Cluj-Napoca (TUCN), Cluj-Napoca, Romania, in 1995 and 2008.

He is a Cofounder of the Image Processing and Pattern Recognition Group, TUCN, where he is currently involved in multiple stereovision-related research projects as a Member or a Team Leader. His main area of expertise is camera calibration for accurate stereovision-based measurements, and he has published many scientific papers in this field. His research interests include software and hardware design, image processing, and computer vision.

Florin Oniga received the Diploma Engineer, M.S., and Ph.D. degrees, all in computer science, from the Technical University of Cluj-Napoca, (TUCN), Cluj-Napoca, Romania, in 2002, 2003, and 2011, respectively.

He is a Senior Lecturer with the Department of Computer Science, TUCN, teaching image processing, pattern recognition, and computer architecture. He is also with the Image Processing and Pattern Recognition Group, TUCN. His research interests include stereovision, digital elevation map processing, and vision-based automotive applications.