Situation Assessment for Automatic Lane-Change Maneuvers

Robin Schubert, *Student Member, IEEE*, Karsten Schulze, and Gerd Wanielik, *Senior Member, IEEE*

***Abstract*—Current research on advanced driver-assistance systems (ADASs) addresses the concept of highly automated driving to further increase traffic safety and comfort. In such systems, different maneuvers can automatically be executed that are still under the control of the driver. To achieve this aim, the task of assessing a traffic situation and automatically taking maneuvering decisions becomes significantly important. Thus, this paper presents a system that can perceive the vehicle's environment, assess the traffic situation, and give recommendations about lane-change maneuvers to the driver. In particular, the algorithmic background for this system is described, including image processing for lane and vehicle detection, unscented Kalman filtering for estimation and tracking, and an approach that is based on Bayesian networks for taking maneuver decisions under uncertainty. Furthermore, the results of a first prototypical implementation using the concept vehicle *Carai* are presented and discussed.**

***Index Terms*—Advanced driver-assistance systems, Bayesian networks, decision making, lane recognition, situation assessment, vehicle tracking.**

I. Introduction

THE MAIN goal of research on intelligent transportation systems (ITSs) is to make mobility safer, more comfortable, and more sustainable. Although current studies show that the deployment of advanced driver-assistance systems (ADASs), e.g., *adaptive cruise control* (ACC) or *lane keeping* (LK), has particularly contributed to the goal of safety, a high number of fatalities in road traffic still occur [1].

In addition to the aforementioned developments for assisted driving (e.g., in the Proreta project [2]), considerable research activities are carried out to achieve *autonomous driving*. However, although the technical feasibility has been demonstrated in different initiatives, e.g., the Defense Advanced Research Projects Agency (DARPA) Urban Challenge [3] or the research projects CyberCars and CityMobil [4], there are significant unsolved problems in terms of reliability, liability, and user acceptance, which are expected to allow the realization of this vision only in a mid- or long-term perspective [5].

Manuscript received November 14, 2008; revised October 30, 2009 and February 5, 2010; accepted April 18, 2010. Date of publication June 1, 2010; date of current version September 3, 2010. The Associate Editor for this paper was A. Amditis.

R. Schubert and G. Wanielik are with the Professorship of Communications Engineering, Chemnitz University of Technology, 09126 Chemnitz, Germany (e-mail: robin.schubert@etit.tu-chemnitz.de; gerd.wanielik@etit.tu-chemnitz.de).

K. Schulze is with the Department of Safety Electronics and Surrounding Field Sensors, IAV GmbH, 09120 Chemnitz, Germany (e-mail: karsten.schulze@iav.de).

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.2010.2049353

For this reason, there are current research activities that focus on *highly automated driving*, e.g., the research projects HAVEit [6] and SPARC [7]. The main idea of this concept is to allow the automatic execution of certain maneuvers under the supervision and control of the driver and to pass the control back to the driver in situations that cannot automatically be handled. In addition to the perception components, such systems also require the capability to assess the current traffic situation and to automatically take maneuvering decisions. Furthermore, the importance of an appropriate *self assessment*—i.e., determining if the system output is reliable by the system itself—increases.

In this paper, an approach for highly automated driving is presented on the example of an automated lane change. More precisely, a prototypical implementation of a vehicle that can perceive its environment, assess the traffic situation, and recognize appropriate occasions for a lane change maneuver is presented. The situation assessment and decision algorithm is based on a Bayesian network (BN) that enables the integrated handling of uncertainty from the perception to the decision stage.

This paper is organized as follows. Section II gives an overview about the complete system and the sensor configuration. The algorithmic background is described in Section III, including the following components:

- an image processing algorithm for detecting vehicles;
- a method for detecting and classifying lane markings;
- stochastic models for parameter estimation and tracking;
- a probabilistic situation-assessment approach;
- an algorithm for decision-making under uncertainty.

Finally, the prototypical implementation using a concept vehicle called *Carai* [8] and first test results are presented and discussed in Section IV.

II. System Architecture

The general structure of the system described in this paper is shown in Fig. 1. The elements of this diagram are derived from the sensory requirements of the system. On the one hand, knowledge about moving objects in the surrounding of the ego vehicle is necessary to avoid collisions. For that task, a 77-GHz long-range radar (LRR) mounted at the front bumper of the vehicle and two 24-GHz short-range radars (SRRs) at its rear end are used.[1]

[1]Note that the usage of short-range radars in this prototype limits the maximum relative velocity for vehicles that approach from behind, which can be allowed in the test scenarios. Furthermore, due to the existence of a blind spot area, special care needs to be taken for the vehicle tracking (see Section III-C).

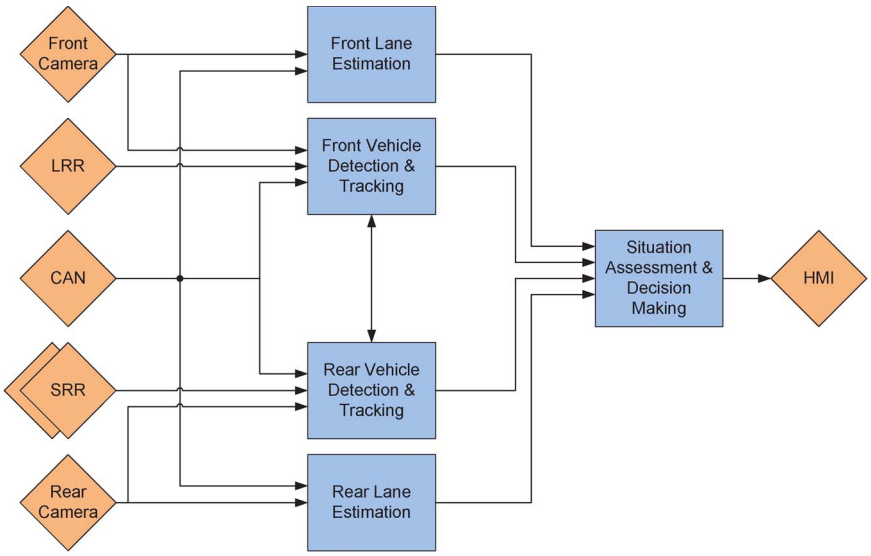

Fig. 1. General structure of the described system. The rectangular boxes represent data processing tasks, whereas sensors and the HMI are denoted by diamond-shaped elements.

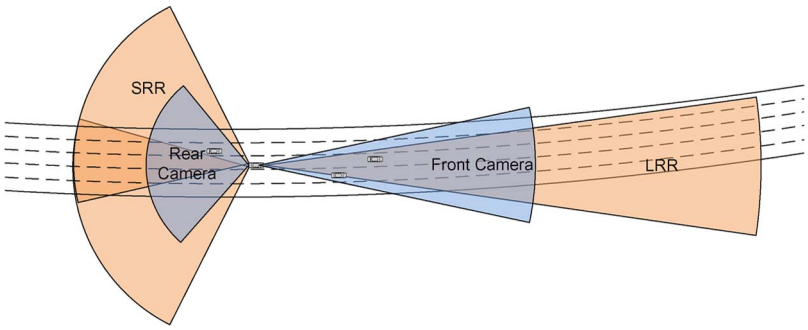

Fig. 2. Sensor configuration of the system, consisting of an LRR, two overlapping SRRs, and two cameras.

Furthermore, a good estimate of the lane parameters is necessary for the lane in which the ego vehicle travels and at least the two adjacent lanes. This requirement is derived from two reasons. On the one hand, a vehicle that will perform automatic lane-keeping and lane-changing maneuvers obviously has to know where the lanes are (or, more precisely, where the ego vehicle is located relative to the lanes). On the other hand, it is also very important to perform a reliable association of other vehicles to certain lanes to correctly assess the current traffic situation. To detect lane markings on the road surface, two gray-scale cameras with video graphics array (VGA) resolution are used, which cover the area in front of the ego vehicle and behind it, respectively (see Fig. 2).

Another important data source is the vehicle's internal controller area network (CAN) bus, which provides (among others) information about the ego motion. Measuring the velocity and the yaw rate of the ego vehicle is very important to stabilize the tracking of both the vehicles and the lane (see Section III-C for details on this so-called *ego motion compensation*).

To exploit synergies between the different elements of the sensor configuration, the cameras are also used to improve and stabilize the vehicle tracking. This approach improves the reliability of the tracking algorithms and allows us to more accurately estimate the width and lateral position of the perceived vehicles. The fusion of the radar and camera information is done on the object level (see Section III-C).

The results of the environmental perception tasks serve as an input to the situation assessment module. This part of the system evaluates all available information about the vehicle's surrounding and performs an assessment of the current traffic situation. Among others, it has to determine if traveling on the current lane is safe and if a lane change maneuver would be safe. Finally, a decision on the appropriate maneuver is taken and passed as a recommendation to the driver through a human–machine interface (HMI).

III. Algorithmic Background

A. Vehicle Detection

Detecting vehicles in camera images is a widely studied topic [9]. Numerous solutions have been proposed, which are, e.g., based on vertical symmetry [10], tail-lamp recognition [11], edge detection [12], or a combination of those features [13]. The image processing approach followed in this paper is based on [14]. Its core elements are summarized as follows.

The applied image processing method is based on contour chains, which are used to find *U-shape*-like forms (Fig. 3 shows the whole image processing data flow). This kind of

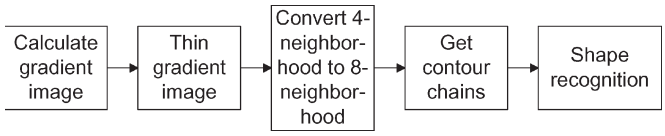

Fig. 3. Data flow of the vehicle-detection image processing approach [14].

form consists of the lower horizontal and both vertical edges of the contour and is typical for almost every type of vehicle, including trucks and buses.

To detect contour points, the gradient image is calculated. This method delivers both the magnitude and the direction of the maximum gray-value difference. The gradient operator does not necessarily produce clear edges; therefore, thinning is required, after which all lines become one pixel wide. The determined contour points are connected by a 4-neighborhood; the width of such a contour is one pixel if horizontal and vertical neighborhoods exist only. For diagonal neighborhoods, the width would be two pixels. Hence, a 4-to-8-neighborhood transformation is performed for the thinned contour points. The shape recognition requires coherent contour chains; therefore, contour chaining is applied to the 8-neighborhood contour points.

A histogram of the contour point directions is created and normalized by the overall number of contour chain points. Several thresholds have to be specified to find specific shapes; for a U-shape, the amount of vertical and horizontal points has to be in a certain ratio range. If a U-shape is confirmed, a rectangle image observation is created and incorporated into the vehicle tracking filter, as shown in Section III-C.

B. Lane Recognition

The problem of detecting lane markings on the road surface has been studied for decades, starting from [15]. The algorithmic development in this field has reached a level that allows the deployment of vision-based lane detection systems in safety applications, e.g., *lane-keeping support* [16]. Details with regard to the image processing part of this problem can, e.g., be found in [17]–[19].

One particularity of the system described here is the fact that not only the course of the lane but also the *type* of the lane markings has to be determined. This approach is necessary because the assessment algorithm has to decide whether a lane change is possible. Although this approach obviously includes the question if the lane change is safe, it also implies determining if such a maneuver is *allowed* by traffic regulations. In most countries, such regulations are indicated by both traffic signs and dashed/solid lane markings. Traffic sign recognition is not available in the applied demonstrator; therefore, classifying lane markings into the types *dashed* or *solid* is crucial for determining correct lane change maneuver recommendations.

A brief description of the algorithm will be given in the following discussion (a more detailed treatment can be found in [20]). From the detected lane markings in the image plane, a space discrete signal is derived, which serves as input to a fast Fourier transformation (FFT). Based on the resulting coefficient vector, the power spectrum p_j is calculated. Comparing both signal classes shows that the periodicity of the dashed signal produces significant power for p_2 and p_4. The solid signal produces significant power only for the steady component.

The fact that dashed lines will always produce determined and significant power values is used to classify the spectrum by a set of rules. Therefore, the power coefficients that are characteristic for a periodical lane-marking signal are summarized as follows:

$$P_{dashed} = \sum_{j=2}^{i_{\max}} P_j. \tag{1}$$

In this formula, $i_{\max}$ is determined by the maximum number of lane markings in the field of view. Furthermore, the overall power of all alternating parts is determined as

$$P_{overall} = \sum_{j=2}^{j_{\max}} P_j. \tag{2}$$

By using the ratio of P_{dashed} and $P_{overall}$, dashed lines can further be distinguished from noise signals.

C. Parameter Estimation and Tracking

Despite the steadily increasing sensor performance, the true state of the vehicle's surrounding will never be exactly known. However, by exploiting *a priori* knowledge about the situation and explicitly taking into account model and sensor uncertainties, at least an *estimate* of the situation can be obtained. Furthermore, it is very important to handle the unavoidable uncertainties in an appropriate way during the whole data processing chain. By that approach, not only a maneuver decision but also an uncertainty measure that allows us to define a threshold for using the system output can be obtained (details about the uncertainty of the generated decision are given in Section III-E). Finally, due to the blind-spot area of the prototype vehicle (see Fig. 2), the position of the vehicles needs to be interpolated while they pass this zone.[2]

Three components are necessary for the estimation process:

1) a probabilistic framework for estimating system states;
2) a model of the systems dynamic behavior;
3) a model of the sensor properties.

The most common estimation and tracking framework is the well-known Kalman filter [22]. In this system, a special derivative of this algorithm is applied, which is called the *unscented Kalman filter* (UKF). This filter yields advantages in propagating Gaussian probability distributions through nonlinear transformations and is, in addition, easier to implement than the most commonly used extended Kalman filter (EKF). Details about the UKF can be found in [23] and [24].

In the following discussion, three different system models are described. The first and second system models are used to estimate the motion of the ego vehicle and other vehicles, respectively. The third model represents a mathematical description of the lane.

[2]A detailed treatment about tracking vehicles in the blind-spot zone is outside the scope of this paper. More information can be found in [21].

1) Ego Vehicle Model: To improve the estimation of ego motion parameters, *a priori knowledge* about vehicles can be used. For instance, it is well known that a vehicle cannot arbitrarily move (e.g., sideway) but is subject to the so-called nonholonomic constraints [25]. In addition to such physical constraints, a model can also contain assumptions about *usual* motion patterns (e.g., constant velocities on highways). An experimental evaluation of different motion models and their suitability for highways is given in [26], where the *constant turn rate and acceleration* (CTRA) model is proposed as an appropriate compromise between reality and simplification. This model is also used for modeling the ego motion in this paper and will be explained in the following discussion.

The state space of this models consists of six quantities:

$$\mathbf{x}_e(t) = (\, x \quad y \quad \theta \quad v \quad a \quad \omega \,)^T \tag{3}$$

where x, y, and θ denote the position and heading of the vehicle. Velocity, yaw rate, and longitudinal acceleration are represented by v, ω, and a, respectively. The state transition equation for this model is[3]

$$\mathbf{x}_e(t+T) = \mathbf{x}_e(t) + \begin{pmatrix} \Delta x(T) \\ \Delta y(T) \\ \omega T \\ aT \\ 0 \\ 0 \end{pmatrix} + \mathbf{v}_e(t) \tag{4}$$

with[4]

$$\begin{aligned} \Delta x(T) = \frac{1}{\omega^2} \big[& (v(t)\omega + a\omega T)\sin(\theta(t) + \omega T) \\ & + a\cos(\theta(t) + \omega T) \\ & - v(t)\omega \sin\theta(t) - a\cos\theta(t) \big] \end{aligned} \tag{5}$$

$$\begin{aligned} \Delta y(T) = \frac{1}{\omega^2} \big[& (-v(t)\omega - a\omega T)\cos(\theta(t) + \omega T) \\ & + a\sin(\theta(t) + \omega T) \\ & + v(t)\omega \cos\theta(t) - a\sin\theta(t) \big]. \end{aligned} \tag{6}$$

The equations show that—as the name suggests—longitudinal acceleration and yaw rate are assumed to be constant quantities. To account for deviations from this assumption (i.e., accelerating and lane-changing vehicles), a statistical disturbance term (called the process noise vector $\mathbf{v}$) is part of the model. The adjustment of this process noise determines which amount of vehicle dynamics can still be handled by the filter.

2) Obstacle Vehicle Model: To model the motion of other vehicles, the same motion model is used. However, two particularities occur. First, to exploit the capability of the camera to determine the lateral width of a vehicle, this quantity has to explicitly be estimated. Second, the position of the vehicles needs to be corrected by the motion of the ego vehicle between the last and the current filter time steps. To take these issues into account, the state vector is augmented by the width d and the ego motion parameters x_e, y_e, and θ_e, i.e.,

$$\mathbf{x}_o(t) = (\, x \quad y \quad \theta \quad v \quad a \quad \omega \quad d \quad x_e \quad y_e \quad \theta_e \,)^T. \tag{7}$$

The state transition equation is

$$\mathbf{x}_o(t+T) = \mathbf{x}_o(t) + \begin{pmatrix} f_e\begin{pmatrix} \Delta x(T) \\ \Delta y(T) \\ \omega T \end{pmatrix} \\ aT \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \mathbf{v}_o(t) \tag{8}$$

where $f_e(.)$ denotes the ego motion compensation, i.e.,

$$\begin{pmatrix} \cos\theta_e \cdot (\Delta x(T) - x_e) + \sin\theta_e \cdot (\Delta y(T) - y_e) \\ \cos\theta_e \cdot (\Delta y(T) - y_e) - \sin\theta_e \cdot (\Delta x(T) - x_e) \\ \omega T - \theta_e \end{pmatrix}. \tag{9}$$

By augmenting the state vector, the uncertainty of the ego motion is implicitly taken into account.

3) Lane Model: Usually, the lane geometry is modeled by an approximated clothoid, i.e., by a third-order polynomial (e.g., in [17]). In this paper, however, the lane is modeled as a set of parallel circle arcs that can be distorted relative to the ego vehicle [27]. The rationale behind this choice is that the detection range of automotive cameras with VGA resolution is limited (depending on the focal length) to a value around 60 m [28]. To perform a lane association of vehicles beyond this distance, the lane estimate needs to be extrapolated. As the clothoid model implies more degrees of freedom, an extrapolation may lead to significant deviations from reality. In contrast, the circular model provides a more stable basis for extrapolation; however, this advantage comes at the price of a less accurate representation at near distances (see [29] for an analysis of the approximation errors of different lane models).

The state vector consists of the following quantities:

$$\mathbf{x}_\ell = (\, c \quad o \quad \varphi \quad w \quad x_e \quad y_e \quad \theta_e \,)^T. \tag{10}$$

In (10), $c = 1/r$ denotes the curvature of the circle,[5] o denotes the lateral offset, φ denotes the distortion of the lane (which can, depending on the perspective, also be regarded as the yaw angle of the vehicle inside the lane), and w denotes its width. The model assumes that the center point of the circles are located on the y-axis of a Cartesian frame at $x = 1/c$. This frame is assumed to be distorted by φ relative to the vehicle coordinate system (see Fig. 4, left).

For the dynamic behavior of the system, it is assumed that, apart from process noise disturbances, the system state remains constant over time. That is, the state transition equation of the lane model basically performs ego motion compensation.

[3]The augmentation of the state space by the so-called *error variables* to account for process noise, which is often applied when using the UKF [23], is omitted here for clarity. Instead, the zero-mean additive noise vector $\mathbf{v}$ is used to account for the process noise.

[4]For $w = 0$, the limit of the state transition equation has to be used instead.

[5]The usage of curvature instead of a radius allows us to model straight lines.

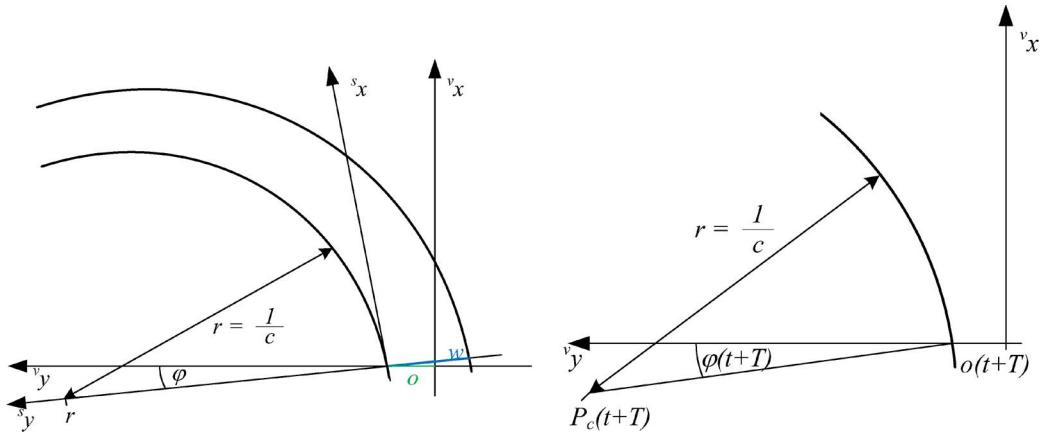

Fig. 4. Geometrical interpretation of the state space for the lane estimation (left) and its reconstruction from the ego motion compensated circle center (right). The superscripts s and v denote the segment and the vehicle coordinate system, respectively.

For that end, the circle center point $P_c(x_c, y_c)$ at time t is calculated as

$$x_c(t) = -\frac{1}{c}\sin(\varphi) \tag{11}$$

$$y_c(t) = \frac{1}{c}\cos(\varphi) + o. \tag{12}$$

This point P_c is then transformed from time t to $t+T$ according to the ego motion as follows:

$$\begin{aligned} x_c(t+T) = {} & \cos(\theta_e)x_c(t) + \sin(\theta_e)y_c(t) \\ & + (-\cos(\theta_e))\, x_e - \sin(\theta_e)y_e) \end{aligned} \tag{13}$$

$$\begin{aligned} y_c(t+T) = {} & -\sin(\theta_e)x_c(t) + \cos(\theta_e)y_c(t) \\ & + (\sin(\theta_e))\, x_e - \cos(\theta_e)y_e). \end{aligned} \tag{14}$$

From this new circle center point, the updated offset o is determined using a circle intersection equation (see Fig. 4, right). The predicted state $\mathbf{x}_\ell(t+T)$ is then given by

$$\mathbf{x}_\ell(t) + \begin{pmatrix} 0 \\ y_c(t+T)\cdot c + \frac{\sqrt{x_c(t+T)^2\cdot c^2+1}}{c} \\ -\arctan\left(\frac{-x_c(t+T)}{o(t+T)-y_c(t+T)}\right) \\ 0 \end{pmatrix} + \mathbf{v}_\ell(t). \tag{15}$$

D. Situation Assessment

In the previous section, the tracking on object level (i.e., data fusion on JDL[6] level 1) has mainly been described; however, it is also necessary to evaluate the relationships between the tracked objects to derive an appropriate decision (data fusion on JDL level 2).

There are different general approaches for high-level data fusion, for instance expert systems, fuzzy logic, or BNs [31]. Previous work in automotive, e.g., predicts potential conflict situations using a knowledge-based approach [32] or assesses the situation based on an iterative scheme and a fuzzy-related uncertainty representation [33]. For emergency brake applications, different rule-based algorithms have been proposed [34], [35], whereas a BN-based approach for this application is presented in [36].

In this paper, a BN is used for both the situation assessment and the decision making.[7] The perception algorithms are based on Bayes' theorem (in fact, the utilized UKF is an implementation of the general Bayes filter under a Gaussian assumption [23]); therefore, the usage of a BN allows a unified way of handling uncertainties from the detection to the decision stage. A brief introduction to the concept of BNs is given as follows.

1) BNs: A BN is a directed acyclic graph that represents a set of random variables [37]. It consists of nodes that represent random variables and edges that represent probabilistic relationships.

The main advantage of BNs compared to other representations (e.g., joint probability tables) is the possibility of modeling conditional independence, which drastically decreases the size of the necessary conditional probability tables (CPTs). Furthermore, the graphical modeling makes it easier for a human designer to adapt the model to his/her perception of the system characteristics. At the same time, the concept of BN is formalized enough to allow an efficient computational evaluation.

After defining the structure of the network and the CPTs, full joint distributions may be calculated for any particular assignment to the random variables of the network. That is, given that the number of random variables (denoted by X_i) in the network is N, the joint probability distribution is given by

$$p(X_1, \ldots, X_N) = \prod_{i=1}^{N} p\left(X_i | pa(X_i)\right) \tag{16}$$

where $pa(X_i)$ represents the set of parent nodes of X_i (i.e., all nodes that posses a directed edge to X_i).

[6]The Joint Directors of Laboratories (JDL) Data Fusion Working Group tried to define a terminology related to data fusion (see [30] and the references therein).

[7]Bayesian networks that include decision and utility nodes are also called *decision graphs* [37]. However, this distinction is omitted here for clarity.

TABLE I
DECISION-RELEVANT SITUATION PARAMETERS

Node	States
LaneChangeLeft	{Impossible, Possible, Safe}
LaneChangeRight	{Impossible, Possible, Safe}
OwnLane	{Dangerous, Occupied, Free}
LeftLane	{Dangerous, Occupied, Free}
RightLane	{Dangerous, Occupied, Free}
BorderLeft	{Dashed, Solid}
BorderRight	{Dashed, Solid}

The main feature of BNs is the ability to perform *probabilistic reasoning*. That is, knowledge about the state of certain nodes (which is generally called *evidence*) can be incorporated into the network to calculate the probability distributions of all other nodes under the condition of the entered evidence. The reasoning can be performed, regardless of causality, i.e., it is possible to deduce possible reasons for observed effects, or *vice versa*. Uncertain evidence (*likelihood evidence* or *soft evidence*) can also be entered into the network.

BN can be regarded as probabilistic extensions of *expert systems* and allow a statistically sound combination of uncertain knowledge, taking into account both model and data uncertainties. For small networks, reasoning can be performed in a simple way by using (16), whereas efficient algorithms exist to allow real-time operations for larger models (see, e.g., [38]).

2) Situation Parameters: Data fusion on object level provides a high amount of information. For the assessment process, it is necessary to reduce these pieces of information as much as possible—without losing relevant aspects. For that end, a complete set of discrete state variables with minimum redundancy is defined and called *situation parameters*. These parameters are represented by *random variables* to account for uncertainties. That is, a parameter will never be assumed to be in *one* state. Instead, a discrete probability distribution is determined from the perception results. The parameter set is particularly designed for the target application, i.e., traffic situations that are not covered by the application (e.g., entering or leaving the highway) cannot explicitly be described by the parameters. The complete set of decision-relevant situation parameters is listed in Table I, whereas their mutual relationships are illustrated in Fig. 5. The general description of those parameters and their states is given in the following discussion, followed by details about their estimation (see Section III-D3).

a) BorderLeft/BorderRight: These parameters denote the status of the lane border, which can be either `Dashed` or `Solid`.

b) OwnLane/LaneLeft/LaneRight: These nodes describe the status of a certain lane with respect to its *occupancy*. The possible states are `Free` (i.e., there are no vehicles within a relevant distance), `Occupied` (the closest vehicle in the lane is within a relevant, but outside, a critical distance), and `Dangerous` (the closest vehicle is inside the safety margin).

c) LaneChangeLeft/LaneChangeRight: These nodes describe the feasibility of a lane change maneuver to the left or to the right. The state `Impossible` indicates that such a maneuver is, by no means, to be performed, because it is either not safe or not allowed. In contrast, the state `Safe` declares that the maneuver is allowed and that there is no relevant vehicle in the target lane. However, a high probability of this state does not necessarily mean that the maneuver is recommended—this condition is evaluated in a separate *decision node* (see Section III-E). The third state `Occupied` represents situations in which a lane change is allowed, but there is a vehicle in the target lane at a noncritical distance.

3) Likelihood Evidence Parameters: To perform probabilistic reasoning inside a BN, the evidence from the perception stage needs to be entered. Usually, this condition means setting some nodes to particular states. However, to account for the uncertainties of the perception evidence, likelihood functions need to be defined for nodes that are directly influenced by the perception results. This approach is similar to the definition of a sensor model in a Kalman filter; however, the likelihood functions in a BN are usually discrete. The likelihood nodes (which are illustrated in Fig. 5, upper row) are described as follows.

a) Observation BorderLeft/BorderRight: The situation assessment needs to account for the possibility that the lane border classification produced an erroneous result or cannot even perform the classification at all. If the probabilities for a correct or false classification are denoted by P_C and P_F, respectively, the discrete likelihood function can be represented by the conditional probability distribution represented in Table II (containing the conditional probabilities for the state of `Observation BorderLeft/BorderRight` under the condition of the state of `BorderLeft/BorderRight`).

b) Observation Left/Ego/Right Lane: The parameters `OwnLane/LaneLeft/LaneRight` are used for modeling if keeping or changing the lane can be considered safe (provided that the ego and object velocities are constant). Thus, it has to be determined which vehicles travel on the lane under consideration, which of those vehicles is the closest, and if this vehicle is within a certain *safety margin*. The first two items are rather straightforward, whereas the third one requires some attention. It has to be defined under which circumstances a vehicle is considered dangerous. For this purpose, different *threat measures* have been proposed in the literature [39], e.g., distance, *time to predicted collision* (TTPC), or *post-encroachment time* (PET). In this paper, the *deceleration to safety time* (DST) measure [40] is applied.

The DST denotes the deceleration that has to be applied to a vehicle to maintain a certain safety time (with respect to another vehicle). The latter can be regarded a safety distance that depends on the absolute velocity of the vehicle. If the safety time is, e.g., defined as 2 s, the corresponding distance for a vehicle that travels with 40 m/s is 80 m. The DST can be calculated using the following equation:[8]

$$d_{t_s} = f(\mathbf{x}_e, \mathbf{x}_o) = \frac{3(v_e - v_o)^2}{2(x - v_o \cdot t_s)} \tag{17}$$

where t_s is the safety time, and x denotes the distance between both vehicles.

[8]This equation is modified from [40] to account for the movement of the object. A derivation of (17) can be found in the Appendix.

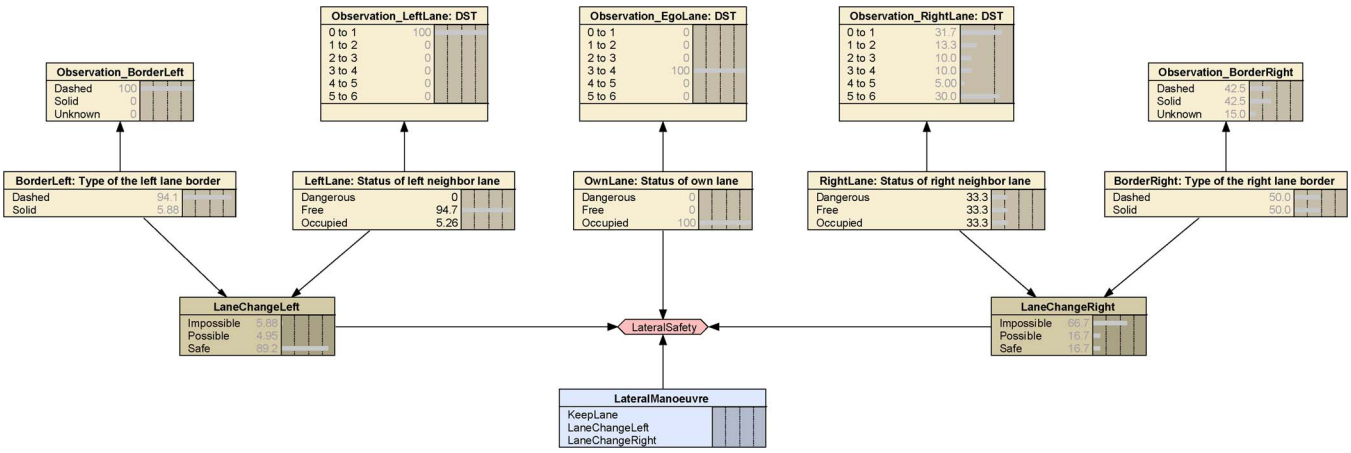

Fig. 5. BN for deriving lane-change maneuver decisions (chance nodes are colored ocher, whereas utility and decision nodes are blue and red).

TABLE II
CONDITIONAL PROBABILITY TABLE FOR THE BORDER OBSERVATION

Dashed	Solid	Unknown	BorderLeft/Right
P_C	P_F	$1-P_C-P_F$	Dashed
P_F	P_C	$1-P_C-P_F$	Solid

In a rule-based system without uncertainties, the state of the nodes `OwnLane`/`LaneLeft`/`LaneRight` can be determined by using a threshold value for the DST. In this case, the state would be `Free` if and only if

$$^{c}d_{t_s} < \overline{d}_{occ}. \tag{18}$$

In this equation, the index c denotes the closest vehicle in the lane under consideration. Similarly, the lane would be considered `Occupied` if and only if

$$\overline{d}_{occ} \leq {}^{c}d_{t_s} < \overline{d}_{dan}. \tag{19}$$

If the DST exceeds the threshold $\overline{d}_{dan}$, the state would be considered `Dangerous`.

However, the DST calculation is an estimation only. In fact, for each quantity that is required in (17), only a probability distribution is known. Thus, the likelihood function also needs to take the uncertainty of the DST into account. The state variables in the perception task are assumed to be Gaussian; therefore, the variance of the DST σ_d^2 can be conveniently approximated using the Unscented transform (UT). The UT allows the approximation of a nonlinear transformation of Gaussian probability distributions and constitutes the basis for the UKF [24]. Its main idea is to *deterministically* draw samples from the Gaussian distribution and to separately transform each of those samples. It has been shown in [24] that the accuracy of this approximation exceeds linear approximations (e. g. the first-order Taylor series expansion used in the EKF).

The likelihood function for the DST under the condition of the states `Observation` `Left`/`Ego`/`Right` `Lane` $\in [\mathbf{Dangerous}, \mathbf{Occupied}, \mathbf{Free}] = [\mathtt{D}, \mathtt{O}, \mathtt{F}]$ is defined as

$$p({}^{c}d|\mathtt{D}) \propto \begin{cases} \exp\left(-\frac{({}^{c}d-\overline{d}_{\mathrm{dan}})^2}{2\cdot\sigma_d^2}\right), & \text{if } {}^{c}d < \overline{d}_{\mathrm{dan}} \\ 1, & \text{else} \end{cases} \tag{20}$$

$$p({}^{c}d|\mathtt{O}) \propto \begin{cases} \exp\left(-\frac{({}^{c}d-\overline{d}_{\mathrm{occ}})^2}{2\cdot\sigma_d^2}\right), & \text{if } {}^{c}d < \overline{d}_{\mathrm{occ}} \\ \exp\left(-\frac{({}^{c}d-\overline{d}_{\mathrm{dan}})^2}{2\cdot\sigma_d^2}\right), & \text{if } {}^{c}d > \overline{d}_{\mathrm{dan}} \\ 1, & \text{else} \end{cases} \tag{21}$$

$$p({}^{c}d|\mathtt{F}) \propto \begin{cases} \exp\left(-\frac{({}^{c}d-\overline{d}_{\mathrm{occ}})^2}{2\cdot\sigma_d^2}\right), & \text{if } {}^{c}d > \overline{d}_{\mathrm{occ}} \\ 1, & \text{else.} \end{cases} \tag{22}$$

For reasoning within the BN, the likelihood functions need to be normalized and discretized (the latter is illustrated by the bars in Fig. 6).

The dependency between the likelihood and σ_d allows an integrated Bayesian handling of the uncertainties from the perception to the decision stage. An example is shown in Fig. 6, where the upper row shows the likelihood for $\sigma_d^2 = 0.1$, whereas the lower row illustrates it for $\sigma_d^2 = 2$. The influence of the evidence on the probability distribution of the nodes `OwnLane`/`LaneLeft`/`LaneRight` is directly determined by the shape of the likelihood function.

E. Decision Making

In addition to probabilistic reasoning, BNs also allow the derivation of decisions under uncertainty. For that, the network needs to be extended by *decision* and *utility nodes* [37]. The states of a decision node contain all possible outcomes of the modeled decision. For the system in this paper, the decision node `LateralManeuver` contains the states `KeepLane`, `LaneChangeLeft`, and `LaneChangeRight`.

To evaluate the decision alternatives, a conditional utility value distribution has to be defined. Each value of this distribution needs to be an element of the *utility scale* interval $[0, \ldots, 1]$. Similar to the chance nodes of the BN, the conditional utilities need to be defined for each combination of the states, which are defined in their parent nodes. For the lane change decision, the utility node is called `Safety` and contains a utility value for each combination of the traffic situation and the maneuver decision.

Taking a decision in a BN means calculating the expectation value of the utility for each possible outcome of the decision conditioned on the joint probability distribution of the network,

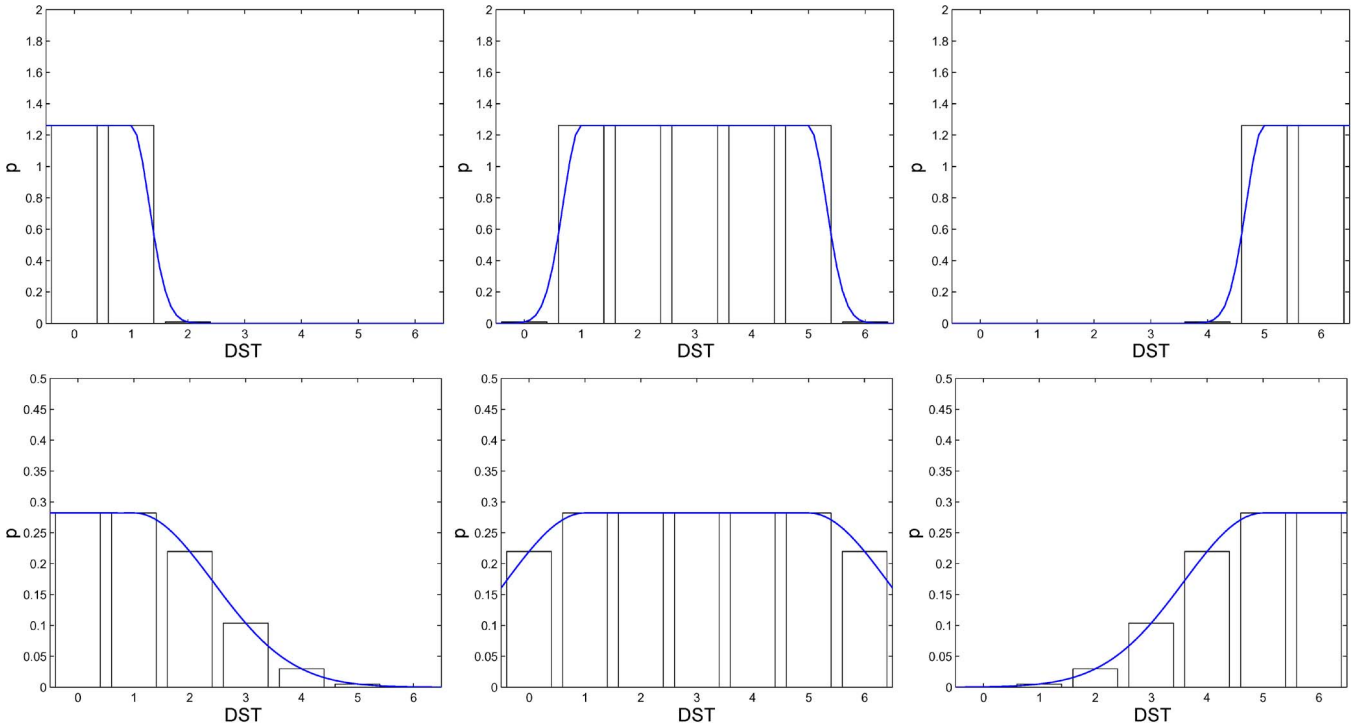

Fig. 6. Likelihood of the node `Observation Left/Ego/Right Lane` for $\sigma_d^2 = 0.1$ (upper row) and $\sigma_d^2 = 2$ (lower row) for the states `Free`, `Occupied`, and `Dangerous` (from left to right). The bars illustrate the discretized likelihood distribution, which can be used in the BN.

Fig. 7. Concept vehicle *Carai* [8] used for the evaluation.

i.e., the *expected utility* (EU). The alternative with the highest EU is considered the optimal decision, given the current evidence. To assess the confidence of this decision, the EU values of the other alternatives are taken into account. The idea behind this approach is that, if the EU values are uniformly distributed, the obtained decision appears rather ambiguous. However, if the highest EU is significantly higher than the second highest EU, the confidence in the decision may be considered higher. Therefore, the ratio between the two highest EU values is used as a confidence measure of the determined decision.[9]

IV. Results

The proposed algorithm has been implemented in a concept vehicle (see Fig. 7), which is equipped with a variety of sensors (see Fig. 2 for the sensor subset used for the presented application). In the first step, the maneuver decision of the system is delivered to the driver on an HMI. More precisely, an icon is permanently shown in a display that encodes the recommended maneuver.

Fig. 8 shows an example traffic situation from different perspectives (front camera image, rear camera image, and bird's eye view). They illustrate that the algorithm tracks four vehicles: one on the current lane in front of the ego vehicle and three on the right neighbor lane. Fig. 9 shows how these data are entered as evidence into the BN. It also illustrates the result of the probabilistic reasoning. In this example, the maneuver `KeepLane` is recommended with a high confidence value (because the expected utilities of the other alternatives are close to zero).

To evaluate the presented approach, the recommended decision (including its confidence) and the sensor data have been recorded in different test drives. This approach allowed a manual postprocessing of the data to check for false alarms (in particular, lane-change recommendations in unsafe situations). Future work will include the manual labeling of the recorded data to obtain *ground-truth* information. After this step, detailed quantitative evaluations can be performed.

V. Conclusion

In this paper, an integrated approach for the perception and assessment of the vehicle environment has been described. This technique allows the unified handling of uncertainties from the sensors to the decision stage. As a practical example, the automatic decision making of lane-change maneuver decisions

[9]However, such a confidence definition requires that there be only one optimal decision alternative for each possible situation.

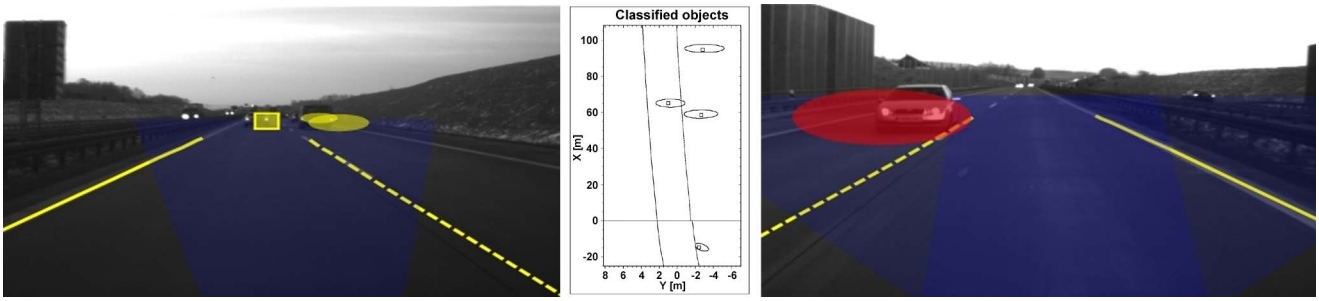

Fig. 8. Front view (left), bird's eye view (center), and rear view (right) of an example traffic situation.

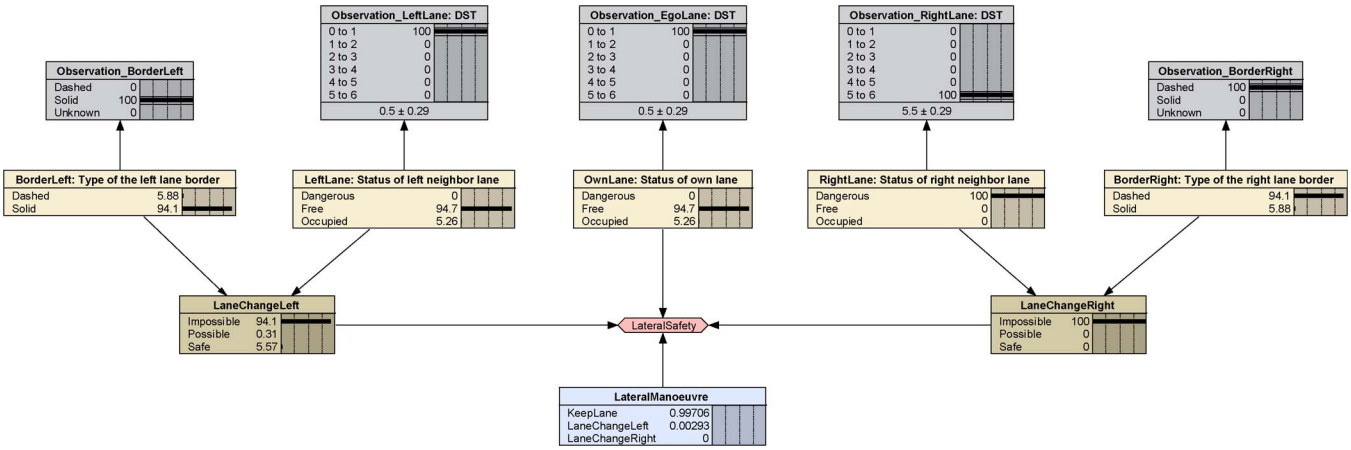

Fig. 9. Result of the probabilistic reasoning after entering the evidence given by the perception modules.

has been presented. Finally, preliminary results from real-world tests have been presented and discussed.

APPENDIX
DECELERATION TO SATISFY TIME

Under the assumption that the velocity of the object remains constant, the time for decelerating from the ego velocity v_e to the object velocity v_o is given by

$$t_d = \frac{v_e - v_o}{a}. \tag{23}$$

The required distance for this deceleration can be calculated using

$$s = \frac{a}{2} t_d^2 + v_e t_d. \tag{24}$$

The core idea for deriving the DST equation is that this distance needs to equal the required safety margin $v_o \cdot t_s$, which is corrected by the initial distance to the object x and the movement of the object itself, i.e.,

$$s \triangleq x + v_o \cdot t_d - v_o \cdot t_s. \tag{25}$$

Combining (24) and (25) and inserting (23) gives

$$\frac{a}{2}\left(\frac{v_e - v_o}{a}\right)^2 + v_e\left(\frac{v_e - v_o}{a}\right) = x + v_o\left(\frac{v_e - v_o}{a}\right) - v_o t_s \tag{26}$$

which can further be simplified to

$$\frac{(v_e - v_o)^2}{2a} + \frac{(v_e - v_o)^2}{a} = x - v_o t_s. \tag{27}$$

Rearranging this equation finally gives the (negative) acceleration, which is necessary to comply with the velocity of the object just at the safety margin—i.e., the *DST*. We have

$$a = \frac{3(v_e - v_o)^2}{2(x - v_o \cdot t_s)}. \tag{28}$$

ACKNOWLEDGMENT

The authors would like to thank N. Mattern and E. Richter for their helpful discussions and the anonymous reviewers for their detailed comments and suggestions.

REFERENCES

[1] T. Vaa, M. Penttinen, and I. Spyropoulou, "Intelligent transport systems and effects on road traffic accidents: State of the art," *IET Intell. Transp. Syst.*, vol. 1, no. 2, pp. 81–88, Jun. 2007.

[2] R. Isermann, U. Stählin, and M. Schorn, "Collision avoidance system proreta—Strategies trajectory control and test drives," in *Proc. 5th Int. Conf. ICINCO*, 2008, pp. 35–42.

[3] M. Buehler, K. Iagnemma, and S. Singh, Eds., *The DARPA Urban Challenge—Autonomous Vehicles in City Traffic*. Berlin, Germany: Springer-Verlag, 2009, ser. Springer Tracts in Advanced Robotics.

[4] M. Parent, "From drivers assistance to full automation for improved efficiency and better safety," in *Proc. IEEE 59th VTC—Spring*, May 17–19, 2004, vol. 5, pp. 2931–2934.

[5] A. Heide and K. Henning, "The "cognitive car": A roadmap for research issues in the automotive sector," *Annu. Rev. Control*, vol. 30, no. 2, pp. 197–203, 2006.
[6] R. Hoeger, A. Amditis, M. Kunert, A. Hoess, F. Flemish, H.-P. Krueger, A. Bartels, A. Beutner, and K. Pagle, "Highly automated vehicles for intelligent transport: HAVEit approach," in *Proc. 15th World Congr. ITS*, 2008, CD-ROM, 20186.
[7] F. Holzmann, M. BeHino, S. Kolskit, A. Sulzmann, G. Spiegelberg, and R. Siegwart, "Robots go automotive—The SPARC approach," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 6–8, 2005, pp. 478–483.
[8] R. Schubert, E. Richter, N. Mattern, P. Lindner, and G. Wanielik, "A concept vehicle for rapid prototyping of advanced driver assistance systems," in *Advanced Microsystems for Automotive Applications 2010: Smart Systems for Green Cars and Safe Mobility*, G. Meyer and J. Valldorfl, Eds. ISBN: 978-3-642-12647-5, pp. 211–219. [Online]. Available: http://www.springer.com/engineering/electronics/book/978-3-642-12647-5
[9] M. Bertozzi, A. Broggi, M. Cellario, A. Fascioli, P. Lombardi, and M. Porta, "Artificial vision in road vehicles," *Proc. IEEE*, vol. 90, no. 7, pp. 1258–1271, Jul. 2002.
[10] G. Alessandretti, A. Broggi, and P. Cerri, "Vehicle and guard rail detection using radar and vision data fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 95–105, Mar. 2007.
[11] R. Chapuis, F. Marmoiton, R. Aufrere, F. Collange, and J. Derutin, "Road detection and vehicles tracking by vision for an on-board ACC system in the VELAC vehicle," in *Proc. 3rd Int. Conf. FUSION*, Jul. 10–13, 2000, vol. 2, pp. 11–18.
[12] S. Kyo, T. Koga, K. Sakurai, and S. Okazaki, "A robust vehicle detecting and tracking system for wet weather conditions using the IMAP-VISION image processing board," in *Proc. IEEE/IEEJ/JSAI Int. Conf. Intell. Transp. Syst.*, Oct. 5–8, 1999, pp. 423–428.
[13] R. Schweiger, H. Neumann, and W. Ritter, "Multiple-cue data fusion with particle filters for vehicle detection in night view automotive applications," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 6–8, 2005, pp. 753–758.
[14] E. Richter, R. Schubert, and G. Wanielik, "Radar and vision based data fusion—Advanced filtering techniques for a multiobject vehicle tracking system," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 4–6, 2008, pp. 120–125.
[15] E. D. Dickmanns and A. Zapp, "A curvature-based scheme for improving road vehicle guidance by computer vision," in *Proc. SPIE Mobile Robots*, Cambridge, MA, 1986, vol. 727, pp. 161–168.
[16] S. Ishida and J. E. Gayko, "Development, evaluation, and introduction of a lane keeping assistance system," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 14–17, 2004, pp. 943–944.
[17] B. Southall and C. J. Taylor, "Stochastic road shape estimation," in *Proc. 8th IEEE ICCV*, Jul. 7–14, 2001, vol. 1, pp. 205–212.
[18] N. Apostoloff and A. Zelinsky, "Robust vision-based lane tracking using multiple cues and particle filtering," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 9–11, 2003, pp. 558–563.
[19] R. Chapuis, R. Aufrere, and F. Chausse, "Accurate road following and reconstruction by computer vision," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 4, pp. 261–270, Dec. 2002.
[20] N. Mattern, R. Schubert, and G. Wanielik, "Lane level positioning using line landmarks and high accurate maps," in *Proc. 16th World Congr. Intell. Transp. Syst.*, 2009, CD-ROM, 3499.
[21] R. Schubert, G. Wanielik, and K. Schulze, "An analysis of synergy effects in an omnidirectional modular perception system," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 3–5, 2009, pp. 54–59.
[22] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, 1960.
[23] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Norwood, MA: Artech House, 2004.
[24] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.
[25] S. M. Lavalle, *Planning Algorithms*, 1st ed. Cambridge, U.K.: Cambridge Univ. Press, Jun. 2006. [Online]. Available: http://msl.cs.uiuc.edu/planning/
[26] R. Schubert, E. Richter, and G. Wanielik, "Comparison and evaluation of advanced motion models for vehicle tracking," in *Proc. 11th Int. Conf. Inf. Fusion*, Jun. 30–Jul. 3, 2008, pp. 1–6.
[27] B. Fardi, U. Scheunert, H. Cramer, and G. Wanielik, "Multimodal detection and parameter-based tracking of road borders with a laser scanner," in *Proc. IEEE Intell. Veh. Symp.*, 2003, pp. 95–99.
[28] R. Schubert, G. Wanielik, M. Hopfeld, and K. Schulze, "Lane recognition using a high-resolution camera system," in *Advanced Mircosystems for Automotive Applications*, G. Meyer, J. Valldorf, and W. Gessner, Eds. Berlin, Germany: Springer-Verlag, 2009, pp. 209–227.
[29] H. Cramer, "Modelle zur multisensoriellen Erfassung des Fahrzeugumfeldes mit Hilfe von Schätzverfahren," Ph.D. dissertation, Tech. Univ. Chemnitz, Chemnitz, Germany, 2006 (in German).
[30] D. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proc. IEEE*, vol. 85, no. 1, pp. 6–23, Jan. 1997.
[31] S. Das, *High-Level Data Fusion*. Norwood, MA: Artech House, 2008.
[32] R. Mock-Hecker and M. Zeller, "Recognizing hazardous situations among vehicles in a traffic world," in *Proc. 4th Annu. Conf. Integr. Virtual Reality Model-Based Environ. AI, Simul., Plan. High Autonomy Syst.*, Sep. 20–22, 1993, pp. 205–210.
[33] M. Pellkofer and E. D. Dickmanns, "Behavior decision in autonomous vehicles," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 17–21, 2002, vol. 2, pp. 495–500.
[34] J. Hillenbrand, A. M. Spieker, and K. Kroschel, "A multilevel collision mitigation approach—Its situation assessment, decision making, and performance tradeoffs," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 528–540, Dec. 2006.
[35] N. Kaempchen, B. Schiele, and K. Dietmayer, "Situation assessment of an autonomous emergency brake for arbitrary vehicle-to-vehicle collision scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 678–687, Dec. 2009.
[36] J. Schneider, A. Wilde, and K. Naab, "Probabilistic approach for modeling and identifying driving situations," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 4–6, 2008, pp. 343–348.
[37] F. V. Jensen, *Bayesian Networks and Decision Graphs*. Berlin, Germany: Springer-Verlag, Jul. 2001, ser. Information Science and Statistics.
[38] Decision System Lab., Smile reasoning engine: Univ. Pittsburgh, Pittsburgh, PA. [Online]. Available: http://dsl.sis.pitt.edu
[39] C. Hydén, "The development of a method for traffic safety evaluation: The Swedish traffic conflicts technique," Ph.D. dissertation, Lund Inst. Technol., Lund, Sweden, 1987.
[40] C. Hupfer, "Deceleration to safety time (DST)—A useful figure to evaluate traffic safety," in *Proc. 3rd ICTCT*, 1997, Dept. Traffic Planning Eng., Lund Univ.

Robin Schubert (S'07) received the Diploma degree in electrical engineering in 2006 from Chemnitz University of Technology, Germany.

He is currently a Research Assistant with Chemnitz University of Technology, Chemnitz, Germany, where he coordinates the Data Fusion Research Group of the Professorship of Communications Engineering. His research interests include data fusion, probabilistic tracking, and situation assessment for advanced driver-assistance systems.

Karsten Schulze received the Diploma degree in electrical engineering in 1999.

In 1999, he was an Engineer and Team Manager of camera-based driver-assistance systems with the IAV GmbH, Chemnitz, Germany. Since 2007, he has been the Head of the Department of Safety Electronics and Surrounding Field Sensors, IAV GmbH. His research and engineering interests include image processing and functional design and function development for advanced driver-assistance systems and active safety.

Gerd Wanielik (SM'90) received the Diploma degree in electrical engineering from the Technical University of Darmstadt, Darmstadt, Germany, in 1979 and the Ph.D. degree in electrical engineering from the Technical University of Karlsruhe, Karlsruhe, Germany, in 1988.

In 1979, he was a Scientist with the AEG-Telefunken Research Institute and later with the Daimler Chrysler Research Center. Since 1999, he has been a Professor with the Professorship of Communications Engineering, Chemnitz University of Technology, Chemnitz, Germany. His research interests include multisensor processing and systems, polarimetry, communications, and navigation systems.

Prof. Wanielik is a member of the Information Technology Society (ITG) and the German Institute for Navigation (DGON). He received the ITG Award in 1991 and the Daimler Chrysler Research Award in 1997.