Object-Oriented Bayesian Networks for Detection of Lane Change Maneuvers

Dietmar Kasper, Galia Weidl, Thao Dang, Gabi Breuel, Andreas Tamke, Andreas Wedel* and Wolfgang Rosenstiel**

***Abstract*—This article introduces a novel approach towards the recognition of typical driving maneuvers in structured highway scenarios and shows some key benefits of traffic scene modeling with object-oriented Bayesian networks (*OOBNs*). The approach exploits the advantages of an introduced lane-related coordinate system together with individual occupancy schedule grids for all modeled vehicles. This combination allows an efficient classification of the existing vehicle-lane and vehicle-vehicle relations in traffic scenes and thus substantially improves the understanding of complex traffic scenes. Probabilities and variances within the network are propagated systematically which results in probabilistic sets of the modeled driving maneuvers. Using this generic approach, the network is able to classify a total of 27 driving maneuvers including merging and object following.**

***Index Terms*—Bayesian Network, Lane Change, Maneuver Recognition**

I. Introduction

THE identification and understanding of traffic scenarios are important key elements of modern driver assistance systems. The challenges for these issues are incomplete knowledge, scene complexity and sensor uncertainties. This requires reasoning under uncertainties. A number of probabilistic approaches can be found in the literature that address this field of research, namely Dempster-Shafer-Theory (*DST*), Hidden Markov models (*HMM*), or Bayesian networks (*BN*).

Our work deals with the understanding and classification of driving maneuvers. The recognition of driving maneuvers (e.g. lane change, object follow, overtake) and the degree of driver's attention have been studied by use of the *DST* [1], [2]. Similar problems (including driver intentions, vigilance, turn maneuver, etc.) have been treated with *HMM* [3], [4], [5]. Bayesian networks have been utilized for the recognition of driving maneuvers like lane change, overtake or left turn maneuvers [6], [7]. They have also been used to recognize turning maneuvers as well as cross over at red traffic lights [8], [9]. Other applications of Bayesian networks involve cut-in maneuvers [13] and emergency braking [10], or the prediction of driver behaviors (e.g. driver intention, driver stress) [11], [12].

A comparison between the different probabilistic methods can be found in the field of research in multisensor data fusion [**?**], [**?**], [**?**]. The *DST* deals with measures of "belief" and is based on the non classical idea of "mass" as opposed to probability used in the Bayesian approach. In general though both methods are found to be robust over the entire sensor information domain, and show comparable performance. *HMM* can be viewed as an extension of the Bayesian network containing three time slices of the same *BN* at three consequent time points (a.k.a. dynamic *BN*).

Our pursued approach for the recognition of driving maneuvers profits from the advantages of *OOBNs* to deal with interrelated objects [15]. They offer a natural framework to handle vehicle-lane and/or vehicle-vehicle relations. These advantages are additionally boosted due to the exploitation of the left/right symmetry of a lane-change-course. The causal probabilistic treatment of situation features allows exploiting heterogeneous sources of information and the quantitative incorporation of uncertainties in the measured signals. Furthermore, they provide a framework for systematically structured "a priory" knowledge representation of the problem domain. Last, the object-orientated modeling is efficient for exploring repetitive structure patterns. This allows reusability through the building of model libraries containing generic fragments of object-oriented Bayesian networks (*OOBN*-classes).

In this paper we focus on the theoretical background of our approach as well as its application for the modeling and the recognition of driving maneuvers in longitudinal traffic scenarios. Section II is dealing with the properties of object-oriented Bayesian networks (*OOBNs*). In section III we outline the features of a driving situation for the modeling of lane change maneuvers, their calculation and our approach to handle uncertainties. In addition, we describe the developed *OOBN* for the recognition of driving maneuvers. The corresponding results and outlook on future work are summarized and discussed in section IV.

II. Object-oriented Bayesian Networks (OOBNs)

Generally Bayesian networks are utilized for the representation of non-observable events, inference on possible conclusions, and they represent specific characteristics of probabilistic models. The advantage of such a probabilistic approach lies in the capability to handle all uncertainties implicitly. Uncertainties can originate from measurement noise in the sensor data as well as from incomplete knowledge of complex dependencies in real world situations. For example, due to complexity reasons, not all maneuver variants can be modeled. Moreover changes in the driving- and environmental conditions can lead to uncertainties in the models.

A *BN* represents a directed, acyclic graph, consisting of nodes and links connecting the nodes. The nodes represent

*Daimler AG, Group Research and Advanced Engineering Böblingen (e-mail: dietmar.kasper, galia.weidl, thao.dang, gabi.breuel, andreas.tamke, andreas.wedel@daimler.com)

**Eberhard Karls Universität Tübingen, Wilhelm-Schickard-Institut für Informatik, (e-mail: rosenstiel@informatik.uni-tuebingen.de)

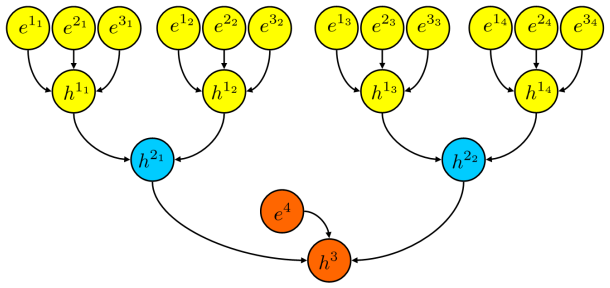


Fig. 1. Example for an not object-oriented Bayesian Network

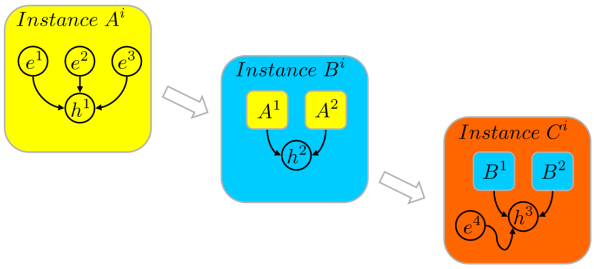


Fig. 2. Object-oriented Bayesian Network

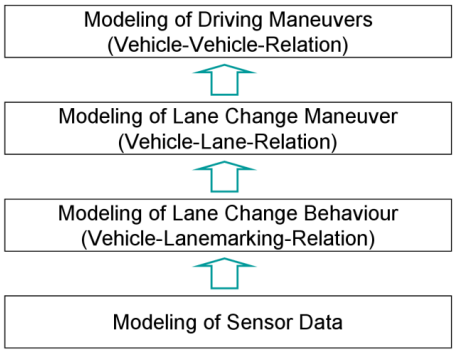


Fig. 3. Hierarchical *OOBN*-layers for the recognition of driving maneuvers

discrete random variables X with n states $x_1,...,x_n$. The links induce a set of conditional in/dependence relations between the nodes (conditional probability distributions *CPD*). Thus the *CPD* expresses the strength of the (causal) dependency relations between the nodes in this model. These relations change when the states of a subset of the nodes are known or observed events. Therefore evidence on a variable provides information on its states and its conditional probability distribution. As already mentioned the graphical structure of a *BN* expresses the knowledge on causal in/dependencies between the variables and thus requires expert knowledge to build the network. This has the advantage that a-priory knowledge of the problem domain can be taken into account. The explicit exploitation of causality allows a compact representation of the joint probability distribution for all the variables involved in the model. The object-oriented Bayesian network possess all advantages of classic Bayesian networks [14] and contains instance nodes in addition to the classical nodes. An instance node is an abstraction of a network fragment into a single unit (network class) [15]. Therefore instance nodes can be used to represent different network classes within other nets, and they transmit all properties of the net fragment (encapsulation). Thus an object-oriented network can be viewed as a hierarchical description/model of a problem domain. Every layer in this hierarchy expresses another level of abstraction in the *OOBN* model. This simplifies the modeling since the *OOBN*-fragments at different levels of abstraction are easier to discuss and to review. An intuitive understanding of the *OOBN* modeling can be provided, if one considers first a usual (non object-oriented) Bayesian network (Fig. **??**) and transforms it into an *OOBN* (Fig. **??**). The usual *BN* structure is build as follows. The evidence e^{1x},e^{2x},e^{3x} support the corresponding hypothesis h^{1x} of same type. These evidence are computed from different input data characterising each object. The various hypotheses from type h^{1x} combine to a new hypothesis of type h^{2x}. Finally the combination of h^{2x} leads to a new hypothesis h^3. Moreover, if an extra evidence is essential, it can also be considered for h^3. As one can see, although a usual *BN* is very good structured by nature, this has not been exploited in their construction, making them to appear not ergonomic, if consistent *BN* change is necessary. For example, if a new evidence e^{ix} should be added to a hypothesis h^{1x}, it has to be added at four different sub-networks of the same type in the non object-oriented version of the *BN*. Similar difficulties appear, if changes in the structure of a certain hypothesis type are needed. This is much easier to realize with an *OOBN*, which represents models a specific combination of variables as a level of abstraction. The above described usual *BN* is then transformed into an *OOBN* with three levels of abstraction A^i, B^i , C^i, see Fig. **??**. The first level absorbs the evidence combined into the basic hypothesis h^1. Thus, each hypothesis of the same type corresponds to an instant node A^i and can be viewed as a representation of a certain feature. One abstraction level higher B^i combines the objects A^1,A^2 of an instance A^i into hypothesis h^2. These are then combined to an instance C^i with hypothesis h^3, representing qualitative relations between the involved objects B^1,B^2 for each instance B^i considering also a new evidence e^i. Thus, one obtains a network, which is modular, structured, easily extendable and modifiable. If one needs an extension of the structure, it is then sufficient to edit just the affected instance and this modification will be inherited in all instances of the same structure type on all levels of the *OOBN*.

III. Modeling and Recognition of Driving Maneuvers by the use of OOBNs

In this approach the driving maneuvers are modeled as object-to-object relations, such as vehicle-lane relations and vehicle-vehicle relations. These relations are modeled on four different hierarchical levels of abstraction (Fig. 1). The first level promotes to model the properties of sensor measurements and delivers the evidence input data for the developed Bayesian network. On the second level the vehicle-lane-marking relations represent the likelihood that a vehicle crosses a specific lane marking. These vehicle-lane-marking relations are modeled by exploiting the assumed left/right symmetry of lane change maneuvers. The next level takes

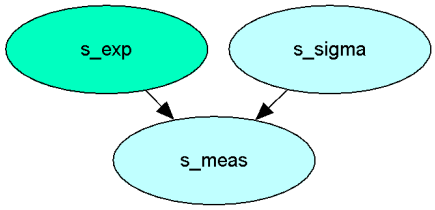


Fig. 4. Model for handling of uncertainties in measured data

all vehicle-lane relations into account and evaluates potential lane-change maneuvers. The combination of vehicle-lane-relations for any two vehicles leads to nine movement classes at this level of abstraction. This represents the pairwise relative movement of any two vehicles with respect to their associated lanes. All existing vehicle-vehicle relations are considered within the fourth level. Regarding all possible relative positions between the vehicle pairs: to the left, to the right or in front, one obtains 27 driving maneuvers, which are recognizable by our *OOBN*. The "cut-in maneuver" is just one from these 27 feasible maneuvers. The hierarchical layers allow a structured overview and represent an easily extendable design for the recognition of driving maneuvers. In the following section the four levels of abstraction are described in more details.

A. Modeling of Sensor Data

At the first level of our *OOBN* model we are using a generic network class for the handling of uncertainties in the sensor data. This data is used as direct evidence in the *OOBN* model and as well as soft evidence for our defined situation features, which are estimated by physical models.

In general, the measured signal $s_{measured}$ is composed of its real value under measurement $s_{expected}$ and its disturbance (sensor noise) s_{err} around the real value, i.e. $s_{measured} = s_{expected} + s_{err}$. In many applications, the sensor noise is simplified modeled as a zero-mean Gaussian random process. In that case, the disturbance is fully described by the variance $s_{err} = s_{\sigma^2}$. Thus the measured signal is conditionally dependent on the random changes in these two variables:

$$p(s_{measured}|s_{expected}, s_{\sigma^2}) = N(s_{expected}, s_{\sigma^2})$$

where $N(s_{expected}, s_{\sigma^2})$ denotes the assumed Gaussian distribution [**?**]. Fig. 2 shows the structure of our network class for handling of uncertainties due to measurement noise within a sensor device. To deal with the uncertainties in the sensor signals and to be able to distinguish between the states of deduced features, the measured signals are discretized in predefined partitions. Through the observation (evidence) of the variables $s_{measured}$ and their variances s_{σ^2} the probability distribution of their real values $s_{expected}$ can be obtained.

B. Modeling of the Driving Behavior During a Lane Change Maneuver

This section constitutes the background, which prepares the *OOBN* model to mimic the human reasoning for maneuver recognition. It starts with the observation and evaluation of characteristic features, which are used to build the vehicle-lane and/or vehicle-vehicle relations at the higher levels of the *OOBN*. A model simplification is achieved due to the observation, that a lane change towards the right or left is symmetric from the point of view of a vehicle positioned in its associated lane. To incorporate the symmetry into the model, we have introduced for all n observable vehicles $\{veh_1, ..., veh_n\}$ the corresponding curvilinear coordinate systems:

$$K_i^s, \quad s \in \{L, R\}, \quad i \in \{veh_1, ..., veh_n\}$$

which are attached to the left/right (L/R) lane marking (or road boundaries) (Fig. 3A).

Some suitable characteristic features were published earlier [13], [**?**]. This approach represents a certain extension of this work. The extension includes

- modeling and computation of the features for both left and right lane-coordinate-system, (Fig. 3A)
- the new introduction of occupancy schedule grids around a vehicle, taking into account the occupancy time of other objects in the grid,
- and consideration of the relative position of its neighbor objects.

These are performed for all observable vehicles, recognized lane markings or road boundaries. The following situation features are used:

- the lateral offset between a vehicle and a lane marking o_{lat}^s and the lateral speed v_{lat}^s, $s \in \{L, R\}$ (Fig. 3B),
- the time to lane marking crossing T_{lcr}^s, the lateral acceleration $a_{lat_{max}}^s$ as well as the head angle relative to the lane course ϕ_{lane}^s (Fig. 3C),
- the relative position of neighboring vehicles Pos^p, $p \in \{left, right, infront\}$
- the occupancy schedule grid which identifies if a certain cell of the grid is occupied or free. It is characterized by the following variables: The time $T_{TE_{occ}}^k$ a vehicle needs until it appears into a certain cell of the grid $k \in \{l, f, r, b, fl, fr, bl, br\}$, the time $T_{TD_{occ}}^k$ a vehicle needs until it disappears from a cell, as well as the corresponding distances from the vehicle to the considered cell k, i.e. $S_{TE_{occ}}^k$ and $S_{TD_{occ}}^k$ (Fig. 3D).

Thus, the following novel definition for the feature vector F of a traffic situation is obtained:

$$F_{s,i,k,p} = \begin{pmatrix} o_{lat}^{s,i}, & v_{lat}^{s,i}, & a_{lat_{max}}^{s,i}, & \phi_{lane}^{s,i}, & T_{lcr}^{s,i}, & Pos^{p,i} \\ & T_{TE_{occ}}^{k,i}, & S_{TE_{occ}}^{k,i}, & T_{TD_{occ}}^{k,i}, & S_{TD_{occ}}^{k,i} & \end{pmatrix}$$

Thus, the feature vector of the entire observable traffic situation has the dimension $n \cdot 45$.

C. Calculation of the Feature Vector

We base our computation on two different models representing the vehicles behavior: a lane follow model and a lane change model. Here it is assumed that the driving states of all observable vehicles as well as the lane course are detected by the on-board sensors and consequently can be considered as known.

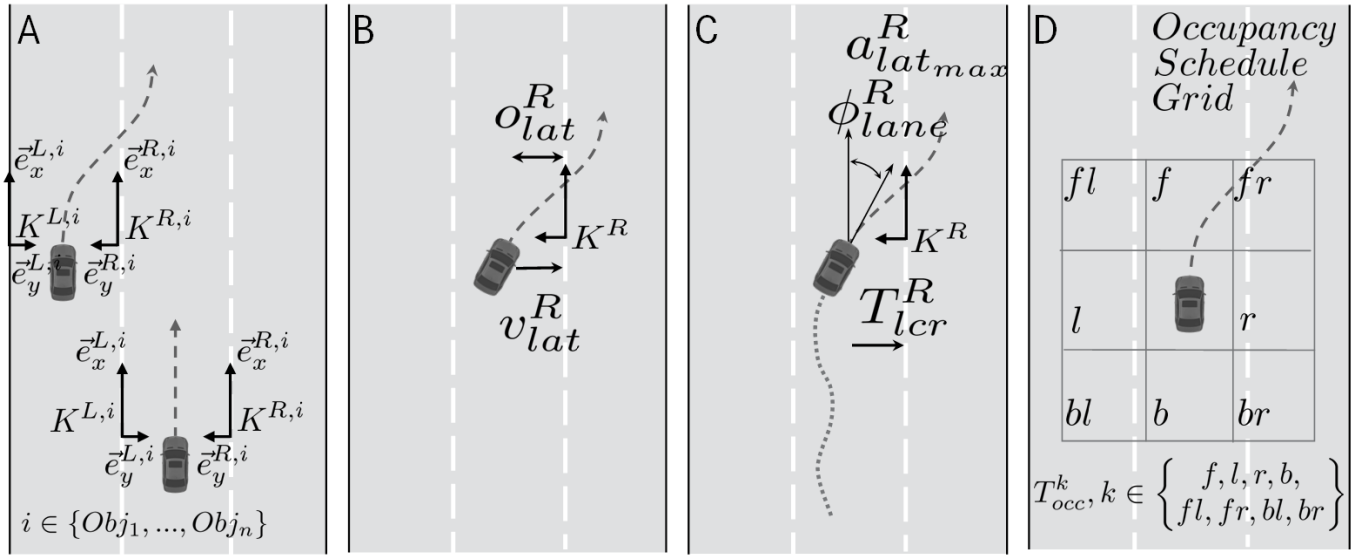


Fig. 5. A: Symmetric lane-coordinate-system for each vehicle; B,C,D: Situation features for a lane change maneuver, in the right coordinate system (B: Lateral Evidence, C: Trajectory and D: Occupancy Schedule Grid)

1) Lane Follow Model: The lateral velocity of the considered vehicle v^s_{lat} within the lane and the distance to the left/right lane marking o^s_{lat} are calculated according to the lane follow model [16]. The lane marking parameters L are given by: $L = (c_0, c_1, \phi_{lane}, dy_{lane})$. Hereby c_0, c_1 are the curvature and the curvature derivation, ϕ_{lane} is the lane angle error and dy_{lane} is the distance to the lane marking. The lane marking course is defined by the following equation:

$$\begin{aligned} \vec{r}(x)_{K_{lane}} &= (f_{LF}(x), g_{LF}(x))^T \\ f_{LF}(x) &= x \\ g_{LF}(x) &= dy_{lane} + \frac{1}{2} \cdot c_0 \cdot x^2 + \frac{1}{6} \cdot c_1 \cdot x^3 \end{aligned}$$

Fig. 4 shows an example for the right coordinate system K^R spanned by the vectors $\vec{T}$ and $\vec{N}$. In this coordinate system, the position of the object is given by $P_{obj_{K^R}} = (0, y_\perp)_{K^R}$. To determine the distance $y_\perp$ between the lane marking and the object, the following equation system has to be solved:

$$\begin{aligned} \vec{P}_{obj_{K_{lane}}} &= \vec{r}(x_\perp)_{K_{lane}} + y_\perp \cdot \vec{N}(x_\perp)_{K_{lane}}, \\ \vec{P}_{obj_{K_{lane}}} &= R(\phi_{lane}) \cdot \begin{pmatrix} x_{obj} \\ y_{obj} \end{pmatrix}_{K_{ego}} . \end{aligned}$$

The matrix

$$R(\phi_{lane}) = \begin{pmatrix} \cos(\phi_{lane}) & -\sin(\phi_{lane}) \\ \sin(\phi_{lane}) & \cos(\phi_{lane}) \end{pmatrix}$$

defines a 2d rotation about the angle ϕ_{lane}. Solving the equation we get the unknown values $(x_\perp, y_\perp)$. The distance from the vehicle to the lane marking is then calculated as follows

$$o_{lat_{K^R}} = y_\perp - \frac{1}{2} \cdot B_{obj},$$

where B_{obj} stands for the object width. The velocity of the object is measured in the EGO coordinate system K_{ego} and

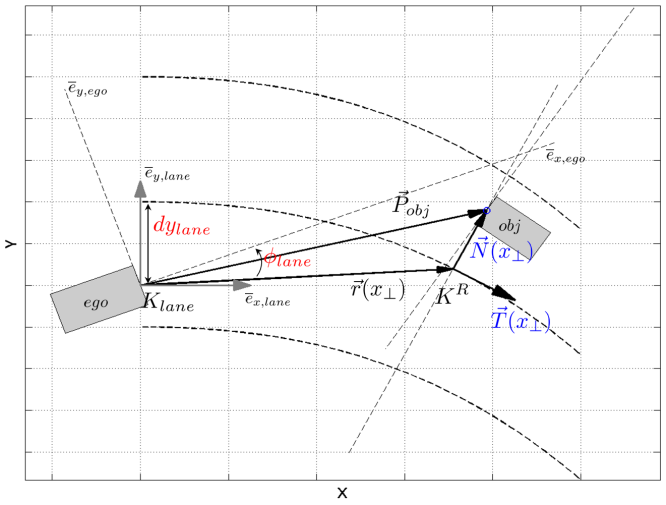


Fig. 6. Calculation of the lateral distance to the lane marking

is given by the vector $\vec{v}_{obj_{K_{ego}}} = (v_x, v_y)_{K_{ego}}$. The object velocity in the K^R coordinate system is calculated by correcting the velocity $\vec{v}_{obj_{K_{ego}}}$ about the angle ϕ_{lane} between the lane coordinate system and the EGO vehicle minus the angle $\phi_\perp$ between the lane coordinate system and the unit tangent vector $\vec{T}$. The lateral velocity $\vec{v}_{lat_{K^R}}$ can be calculated by:

$$\begin{aligned} \vec{v}_{lat_{K^R}} &= R(\phi_{lane} - \phi_\perp) \cdot \begin{pmatrix} v_{x_{obj}} \\ v_{y_{obj}} \end{pmatrix}_{K_{ego}} \\ \phi_\perp &= \angle \vec{T}(x_\perp) \end{aligned}$$

2) Lane Change Model: Since the previously presented situation features ($a_{lat_{max}}$, ϕ_{lane}, T_{lcr}) are not directly measurable, these quantities must be estimated. For this purpose we introduce in this chapter a model for planning a lane change trajectory. After estimating the parameters of the lane change trajectory, the desired situation features can be calculated.

Fig. 5 shows a model for the trajectory of a lane change maneuver, which can be modeled by a third order polynom introduced in [17].

$$\begin{aligned} y = f_{LC}(x) &= a_3 \cdot (x - x_{start})^3 + a_2 \cdot (x - x_{start})^2 \\ &+ a_1 \cdot (x - x_{start}) + a_0 \\ & x_{start} < x < x_{start} + D \end{aligned}$$

where D denotes the transition length of the lane change in x direction from the maneuver beginning until its end. To determine the unknown coefficients a_3, a_2, a_1, a_0, the following conditions on the lane change trajectory have been formulated:

$$\begin{aligned} f(0) &= y_{start}, \quad f(D) = y_{end} \\ \frac{df}{dx}(0) &= 0, \quad \frac{df}{dx}(D) = 0. \end{aligned}$$

The boundary conditions determine, that the y offset at the beginning, respectively at the end of the modeled trajectory, take the values y_{start} or y_{end} correspondingly. Moreover, at the beginning, respectively at the end, of the lane change maneuver, the tangent to the state transition curve is equal to the tangent to the corresponding lane. The solution of the

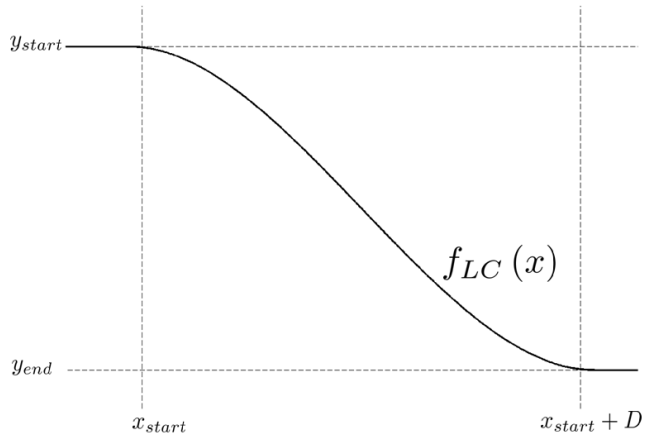


Fig. 7. Lane change trajectory

system of equations yields the coefficients:

$$\begin{aligned} a_0 &= y_{start} \\ a_1 &= 0 \\ a_2 &= \frac{3 \cdot (y_{end} - y_{start})}{D^2} \\ a_3 &= \frac{-2 \cdot (y_{end} - y_{start})}{D^3} \end{aligned}$$

If the vehicle should not exceed a certain maximal lateral acceleration $a_{lat_{max}}$ during the entire lane change maneuver, then the system should satisfy the following condition:

$$\frac{d^2 f}{dt^2} \leq a_{lat_{max}}$$

Considering the simplifying assumptions, that the longitudinal velocity $v_{x_{obj}}$ is approximately equal to the velocity v_{obj} and that the longitudinal acceleration $a_{x_{obj}}$ of the vehicle can be neglected, i.e.

$$\begin{aligned} \dot{x} = v_{x_{obj}} &\approx v_{obj} \\ \ddot{x} = a_{x_{obj}} &\approx 0 \end{aligned}$$

the second temporal derivative of the function $f_{LC}(x)$ reads:

$$\frac{d^2 f}{dt^2} = v_{obj}^2 \cdot (6 \cdot a_3 \cdot x + 2 \cdot a_2)$$

The above mentioned requirement on the maximal lateral acceleration $a_{lat_{max}}$ leads to the following condition

$$\begin{aligned} \left| \frac{d^2 f}{dt^2} \right| &= \left| \frac{d^2 f}{dt^2}(0) \right| = \left| \frac{d^2 f}{dt^2}(D) \right| \\ &= \left| v_{obj}^2 \cdot \frac{6 \cdot (y_{end} - y_{start})}{D^2} \right| \leq a_{lat_{max}} \end{aligned}$$

Thus, the transition length D of the lane change maneuver reads:

$$D \geq v_{obj} \cdot \sqrt{\frac{6 \cdot |y_{end} - y_{start}|}{|a_{lat_{max}}|}}$$

Since the modeled lane change trajectory shall be defined only on the intervall $[x_{start}, x_{start} + D]$ the entire lane change model is defined as follows:

$$y = f_{LC}(x) = \begin{cases} y_{start}, & x < x_{start} \\ a_3 \cdot (x\text{-}x_{start})^3 + \\ \quad a_2 \cdot (x\text{-}x_{start})^2 + a_0 & , \quad x_{start} < x < x_{start} + D \\ y_{end}, & x > x_{start} + D. \end{cases}$$

The modeled lane change trajectory is determined using a segment of the actually driven trajectory, where we take the last N positions $(\hat{x}_i, \hat{y}_i)$, $i \in \{0, ..., N\}$ of the vehicle into account (Fig. 6).

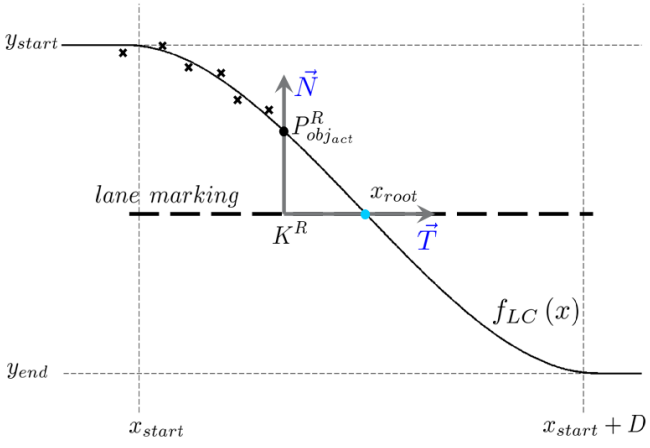


Fig. 8. Lane change trajectory estimation

A complete description of the lane change model requires the estimation of the trajectories parameters from the history. These parameters are y_{start}, y_{end}, x_{start}, $a_{lat_{max}}$. Since the

parameter y_{end} is known, the estimated parameter vector z is defined as follows:

$$z = \begin{pmatrix} x_{start} \\ y_{start} \\ a_{lat_{max}} \end{pmatrix}$$

The function $\hat{y}_i = f_{LC}(z, \hat{x}_i)$ defines the polynomial model. Under the assumption, that only the $\hat{y}$ components of the trajectory are error-prone, i.e.

$$\hat{y}_i = f_{LC}(z, \hat{x}_i) + e_i, \quad i \in \{0, ..., N\}$$

an optimal parameter vector z has to be found, which ensures a minimal error:

$$z = \arg\min_z \sum_{i=0}^{N} |f_{LC}(z, \hat{x}_i) - \hat{y}_i|^2$$

The latter is achieved by computing a non-linear fit of a lane change trajectory in the points of the vehicles trajectory history using the Levenberg−Marquardt optimization [**?**], [**?**], [**?**].

The above mentioned situation features T^s_{lcr} and ϕ^s_{lane} (Fig. 3C) are derived from the characteristics of the matched lane change trajectory. The object's head angle relative to the lane course is given by the derivative of the function:

$$\phi_{lane} = \frac{df}{dx}(P_{obj_{act}})$$

The time to lane marking crossing is given as

$$T_{lcr} = \frac{x_{start} + x_{root}}{v_{obj}}$$

where x_{root} is the intersection of the modeled trajectory with the x axis (i.e. the lane marking, see (Fig. 6)).

3) Occupancy Schedule Grid: An occupancy schedule grid is not to be confused with the classical occupancy grid known from robotics. Here, the cells are not just marked as *occupied* or *not occupied*. Instead they carry the speed-dependent information *when* a cell will be occupied or will become free. The environment perception supplies the position of neighbor vehicles relative to the EGO vehicle. The considered occupancy schedule grid is dynamically centered around the vehicle. Here, the positions of its dynamic grid cells are computed. For each cell and each vehicle in a scene, the time to enter or disappear $T_{TE,TD}$ together with the corresponding distances $S_{TE,TD}$ are evaluated. Thus a measure of when a cell will become free or occupied is established. Since all computations are performed in a curvilinear coordinate system, a simple equation of motion is sufficient for the computation of the occupancy time in the occupancy schedule grid (Fig. 3D):

$$\frac{1}{2} \cdot a_{rel} \cdot T^2_{TE,TD_{occ}} + v_{rel} \cdot T_{TE,TD_{occ}} - S_{TE,TD_{occ}} = 0$$

The values $S_{TE,TD_{occ}}$ can be directly determined from the position of the objects.

4) Relative Vehicle Position: The relative position of the vehicles (Pos^p, $p \in \{left, right, infront\}$) is calculated based on their lane position. If a vehicle is situated in a certain (predefined) distance in the lane, then the vehicle is positioned on this lane. Thereby, there are three lanes: *Left-Lane, Right-Lane, EGO-Lane*. These lanes are computed in a preprocessing step.

D. Lane Marking Crossing Hypothesis

The situation features described in III-B are used to model the lane change behavior as a relation between the considered vehicle and its associated left (or right) lane marking. They are combined into the three nodes: *Trajectory*, *Lateral Evidence* and *Occupancy Schedule Grid*. We associate the same weights to these basis hypotheses in oder to create redundancy (therefore to achieve more robustness if one sensor fails). Their combination express the probability for the hypothesis *Lane Marking Crossing* (Fig. 7).

The node *Lateral Evidence* describes the driving state of the vehicle in the lane (Fig. 7). The node *Trajectory* collects situation features, estimated from the trajectory's history. The node *Occupancy Schedule Grid* describes the occupancy of the grid-cells around the considered object. If a cell is free for a predefined time, we assume that the vehicle can move unobstructed towards this cell.

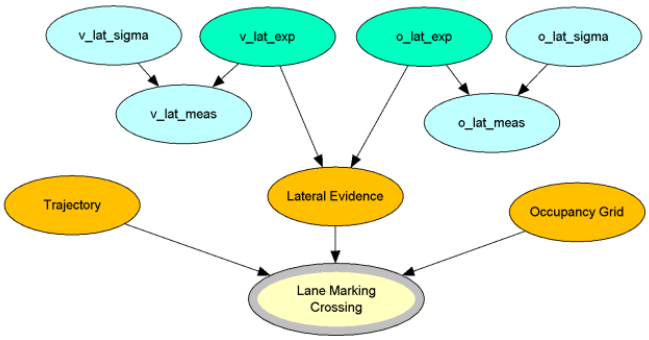


Fig. 9. *OOBN* instance modeling the hypothesis *Lane Marking Crossing*

In order to recognize a lane change maneuver, one has to parametrize the nodes expressing the above mentioned basic hypotheses. The parametrization of these nodes is shown in Fig. 7 by means of the node *Lateral Evidence*. The parametrization of the conditional probabilities for the other nodes is analogous.

The lateral velocity of a vehicle in its associated lane $v_{lat\,expected}$ and its lateral offset to the lane marking $o_{lat\,expected}$ are extended by uncertainties as presented in III-A (Fig. 2, Fig. 7). These two values are summarized in the node *Lateral Evidence*.

The conditional probability of the node *Lateral Evidence* $P(LE|v_{lat\,expected}, o_{lat\,expected})$ is modeled as product of the two independent conditional probabilities $P(LE|v_{lat\,expected})$ and $P(LE|o_{lat\,expected})$. The last are sigmoid functions with scaling parameters $a_{vlat}, b_{vlat}, a_{olat}, b_{olat}$, which are defined as follows:

$$P(LE|v_{lat\,expected}) = \eta \cdot \frac{1}{a_{vlat} + exp(b_{vlat} \cdot v_{lat\,expected})}$$
$$\forall\, v_{lat\,expected} \in \{-1.0, \{0.1\}, 0.5\}$$
$$\text{and } P(LE|o_{lat\,expected}) = \eta \cdot \frac{1}{a_{olat} + exp(b_{olat} \cdot o_{lat\,expected})},$$
$$\forall\, o_{lat\,expected} \in \{-1.0, \{0.1\}, 1.0\}$$

where η is a constant which normalizes the probability. For simplification the indices s, i, k, p were omit-

ted. Fig. 8 shows the shape of the modeled conditional probabilities $P(LE|v_{lat\,expected})$, $P(LE|o_{lat\,expected})$, and $P(LE|v_{lat\,expected}, o_{lat\,expected})$.

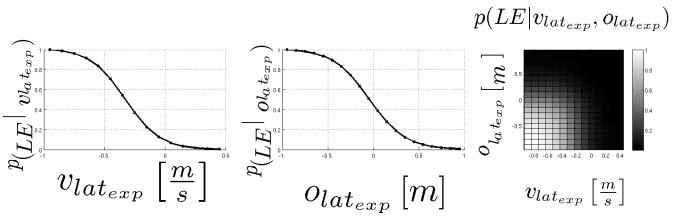


Fig. 10. Conditional probabilities of the node *Lateral Evidence*

E. Modeling of Lane Change Maneuvers

For the recognition of lane change maneuvers, the hypothesis *Lane Marking Crossing* (*LMC*) towards right or left are modeled as vehicle-lane relations. These are shown by the two instance nodes in the BN fragment (Fig. 9). Instance nodes are represented as rounded squares, while the output-interface-nodes are represented as ellipses with shaded line borders. To model lane change maneuvers, we consider elementary actions, which a vehicle can perform. It can follow the lane (f), leave the lane towards the left (l) or towards the right (r). Since the hypothesis *LMC* is used for both left and right lane marking crossing, one can identify three movement classes (l,r,f) of a vehicle, which are logically combined in a node *Lane Change*. In the case of lane change towards the right, the logical parametrization of the node *Lane Change* reads:

$$LC = r \quad \Longleftrightarrow \quad \left(LMC_{left} = false \ \wedge \ LMC_{right} = true\right)$$

Besides the single states (l,r,f) of the node *LMC*, their combination $(LMC_{left} = true \wedge LMC_{right} = true)$ is also possible (e.g. because of wrong input data) and has to be also taken into account. In that case, the states of *Lane Change* are characterized by a uniform probability distribution.

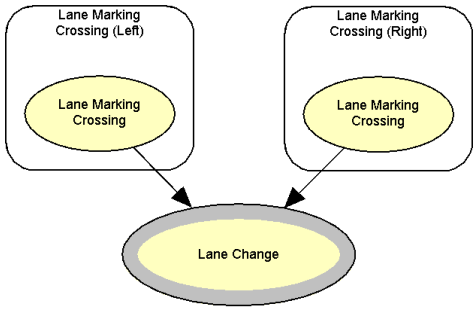


Fig. 11. *OOBN* fragment for the modeling of lane change maneuvers

F. Modeling of Driving Maneuvers

For the recognition of driving maneuvers one has to consider also vehicle-vehicle relations $R = (vehicle_i, vehicle_j)$, $i,j \in \{1,...,n\}, i \neq j$. These are expressed by elementary driving maneuvers, as outlined below.

For the purpose of modeling the elementary driving maneuvers, we introduce the two nodes *Movement-* and *Position classes* (Fig. 10). The node *Movement classes* is classifying

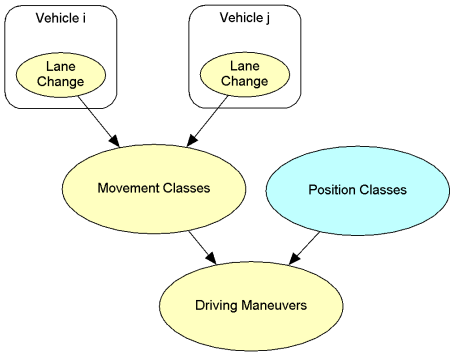


Fig. 12. *OOBN* for modeling of elementary driving maneuvers. It contains the hypothesis *Lane Change* as instance and the movement-, position classes.

the relative movement of a pair of vehicles towards their associated lanes. These results in nine movement classes $(ll, lr, lf, rl, rr, rf, fl, fr, ff)$. The combination of all states given by the nodes *Movement-* and *Position classes* results in 27 possible elementary driving maneuvers, which two vehicles can perform in relation to each other, as explained in section III. In this framework, a cut-in maneuver is therefore one case out of the 27 possibilities. It is defined as follows: The preceding neighbor vehicle is moving onto the considered vehicle's current lane from the left or right side i.e. cutting in. From here, one can deduce the logical parametrization for the recognition of selected driving maneuvers. In the following we list, as an example, the cut-in and cut-out maneuvers for the considered $vehicle_i$ and its neighbor $vehicle_j$.

$$\begin{aligned} cutin_{veh_2} &\Longleftrightarrow (LC_{veh1} = f \wedge LC_{veh2} = r \wedge Pos_{veh2} = left) \\ cutin_{veh_1} &\Longleftrightarrow (LC_{veh1} = r \wedge LC_{veh2} = f \wedge Pos_{veh2} = right) \\ cutout_{veh_2} &\Longleftrightarrow (LC_{veh1} = f \wedge LC_{veh2} = r \wedge Pos_{veh2} = infront) \\ cutout_{veh_1} &\Longleftrightarrow (LC_{veh1} = r \wedge LC_{veh2} = f \wedge Pos_{veh2} = infront) \end{aligned}$$

IV. Results

A typical traffic scene with different driving maneuvers is shown in Fig. 11. The results in this section are based on the relation between the EGO vehicle (veh_1) and the vehicle marked with a white rectangle (veh_2). The frame numbers are depicted at the top of the frames on the right hand side. These involve the maneuvers: *LaneFollow*, *ObjCutIn*, *ObjFollow*, *ObjCutOut*.

The probability of the nodes *Trajectory, Lateral Evidence, Occupancy Schedule Grid*, as well as the probability for the hypothesis on crossing of a lane marking *Lane Marking Crossing* are shown in Fig. 12. As described in the sections above, these nodes contribute to the classification of the three movement classes (*l, r, f*) of node *Lane Change*. Figure 13 shows the probability for each movement class. The probabilities, which were deduced as results for the recognition of the driving maneuver, are depicted in the Fig. 14 and Fig. 15 for the four showcases.

Fig. 13. Showcases of a vehicle-vehicle relation $veh_1 - veh_2$ ($LaneFollow_{veh_{1,2}}$, $ObjCutIn_{veh_2}$, $ObjFollow_{veh_{1,2}}$, $ObjCutOut_{veh_2}$).

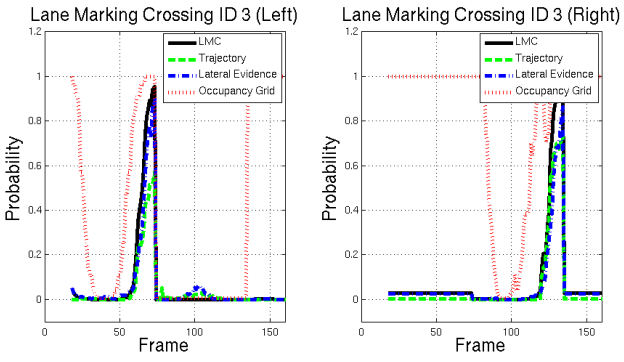


Fig. 14. Probability of the nodes *Trajectory, Lateral Evidence, Occupancy Schedule Grid* and *Lane Marking Crossing* for the left and right lane coordinate system respectively

The final classification result of the *OOBN* can be seen in Fig. 16. The classification result of a common ACC system has two states *"is relevant"* and *"not relevant"*. These are shown on the right axes and depicted by the dashed line. As shown in Fig. 11, the considered vehicle is first on the right side. The class *LaneFollow* (Frame 37) is classified. Next, the vehicle is going to cut in, the class *ObjCutIn* (Frame 67) wins. From the ACC System point of view the vehicle is going to be relevant. After the maneuver *ObjCutIn*, the EGO vehicle follows the considered object, so the class *ObjFollow* (Frame 100) is classified. Next, the object is going to cut out (class *ObjCutOut* (Frame 132)). After the maneuver *ObjCutOut*, the EGO vehicle once again follows the lane (class *LaneFollow* (Frame 151)). From the ACC system point of view the considered vehicle is not relevant anymore.

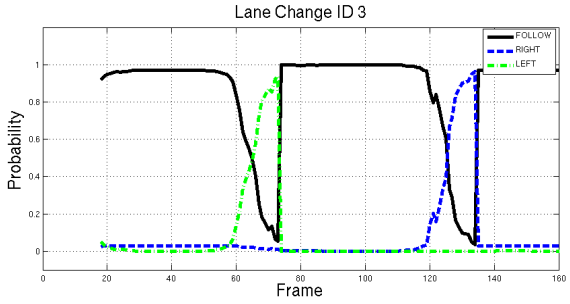


Fig. 15. Probability of the node Lane Change (*l, r, f*)

Fig. 16. Probability of the driving maneuvers *ObjFollow, LaneFollow*

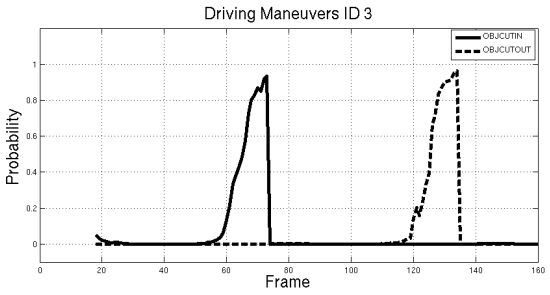


Fig. 17. Probability of the driving maneuvers *ObjCutIn, ObjCutOut*

We performed the same evaluation online during the test drive. The duration of the entire test drive was about twelve hours. In total 968 maneuvers were recorded by the evaluation module. In order to evaluate online the results of the classification, a state machine has been developed, where the state transitions has been marked as *False Positive*, *False Negativ* or *No Error*. A few examples for these state transitions are shown below:

- *False Positive*: *ObjFollow* → *ObjCutOut* → *ObjFollow*
- *False Negative*: *ObjFollow* → *LaneFollow*
- *No Errror*: *ObjFollow* → *ObjCutOut* → *LaneFollow*

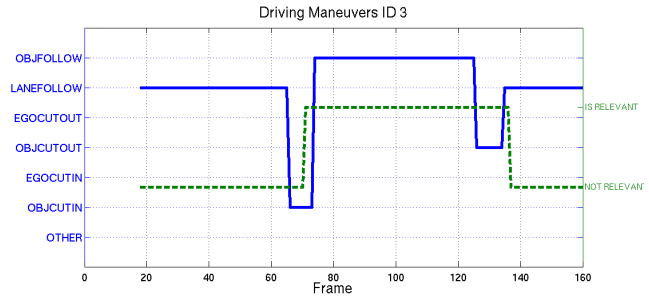


Fig. 18. Result of the final classification

Fig. 17 shows the error statistic for the respective maneuvers. After the evaluation 812 maneuvers are classified correctly, 42 maneuvers exhibit false positives, which means that the classification is not correct. Moreover, 114 are false negatives, which means that those are not recognized. Fig. 18 depicts the

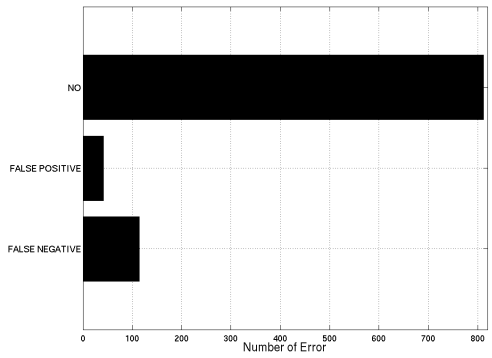


Fig. 19. Error statistic

812 correctly classified maneuvers and their categories.

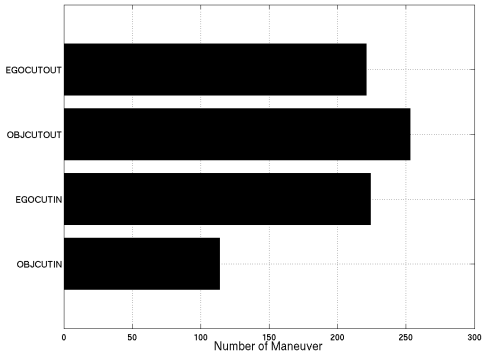


Fig. 20. Maneuver statistic

The maximum probability p_{max} and the time $\triangle t$ have been measured and logged for each maneuver execution. The time $\triangle t$ denotes the difference, between the point in time the maneuver has been recognized by the Bayesian network, and the point in time the ACC system classified the vehicle to be *"relevant"* or *"not relevant"*.

The average value for the probability is $E(p_{max}) = 0.916$. Larger likelihood is an evidence that the classification result is more reliable. Therefore we believe that our approach is well suited to discriminate driving maneuvers. Using our novel approach, we are able to recognize a maneuver about $E(\triangle t) = -0.605[s]$ earlier compared to the classification result of the ACC System.

V. Conclusion

In this paper, we presented an object-oriented approach for the recognition of driving maneuvers with *OOBNs*. The modularity and reusability of Bayesian network fragments were simplified through the consideration of two model properties: symmetric lane-coordinate-system for each vehicle and pairwise defined object-object relations. The hierarchical modeling allows the construction of various model libraries with generic *OOBN*-fragments. This leads to models with good overview and an easily extendable design. Our approach allows to handle uncertainties in the model and in the measurements. The probability distribution of certain maneuvers between two vehicles can be read out from the node *Driving Maneuver*.

Future work will focus on the analysis of the performance for the developed network and on possible improvements. In order to detect lane change maneuvers at an earlier stage, one needs situation features, such as *indicator* or *shoulder check*, for earlier maneuver indication. These features can then contribute to the Bayesian network and help to recognize lane change maneuvers with a high reliability at an earlier stage. Moreover one can adjust the network parameters online depending on the available free space around the vehicle. If for example the vehicle moves towards a solid boundary, one could adjust the parameters such that a collision with the solid boundary can be detected at an earlier stage and possibly be prevented. However, there will always be a trade-off between an earlier detection and the false positive rate of our algorithm.

Moreover, we will study if the performance can be boosted by the extension of the static network to a dynamic *OOBN*. Due to the fact that the parametrization of the network is time consuming, but also for the reason of improving the system performance, learning algorithms can be employed. In addition, the recognition of the criticality of an observed situation or maneuver can be extended by physical models exploiting the relative states (distance, speed, acceleration) between the vehicles.

References

[1] M. Tsogas, X. Dai, G. Thomaidis, P. Lytrivis, and A. Amditis, Detection of maneuvers using evidence theory, IEEE Intelligent Vehicles Symposium, Eindhoven University of Technology, Eindhoven, The Netherlands, 2008.

[2] D. Gruyer, A. Rakotonirainy, and J. Vrignon, The use of belief theory to assess driver's vigilance. In: Australasian Road Safety Research, Policing and Education Conference, Wellington NZ, 2005.

[3] Christoph Stiller, Georg Färber, and Sören Kammel, Cooperative Cognitive Automobiles, Proceedings of the IEEE Intelligent Vehicles Symposium, Istanbul, Turkey, 2007.

[4] H. Berndt, J. Emmert, and K. Dietmayer, Continuous Driver Intention Recognition with Hidden Markov Models, Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems, Beijing, China, 2008.

[5] P. Boyraz, M. Acar, and D. Kerr, Signal Modelling and Hidden Markov Models for Driving Manoeuvre Recognition and Driver Fault Diagnosis in an urban road scenario, Proceedings of the IEEE Intelligent Vehicles Symposium, Istanbul, Turkey, 2007.

[6] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, Automatic Symbolic Traffic Scene Analysis Using Belief Networks, AAAI-94 Proceedings, 1994.
[7] D. Meyer-Delius, C. Plagemann, G. von Wichert, W. Feiten, G. Lawitzky, and W. Burgard, A Probabilistic Relational Model for Characterizing Situations in Dynamic Multi-Agent Systems, In Proc. of the 31th Annual Conference of the German Classification Society on Data Analysis, Machine Learning, and Applications (GFKL), Freiburg, Germany, 2007.
[8] F. Schroven, and T. Giebel, Fahrerintentionserkennung für Fahrerassistenzsysteme, VDI-Berichte, Band 2048: Proceedings der 24. VDI/VW-Gemeinschaftstagung - Integrierte Sicherheit und Fahrerassistenzsysteme, Wolfsburg, VDI Verlag, Düsseldorf, 2008.
[9] J. Zhang, and B. Roessler, Situation Analysis and Adaptive Risk Assessment for Intersection Safety Systems in Advanced Assisted Driving, Autonome Mobile Systeme, Informatik aktuell, Springer-Verlag Berlin Heidelberg, p. 249, 2009. doi: 10.1007/978-3-642-10284-4-32.
[10] J. Schneider, A. Wilde, and K. Naab, Probabilistic Approach for Modeling and Identifying Driving Situations, IEEE Intelligent Vehicles Symposium, Eindhoven University of Technology, Eindhoven, The Netherlands, 2008.
[11] H. Amata, C. Miyajima, T. Nishino, N. Kitaoka, and K. Takeda, Prediction Model of Driving Behavior Based on Traffic Conditions and Driver Types, Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems, St. Louis, MO, USA, 2009.
[12] G. Rigas, C.D. Katsis, P. Bougia, and D.I. Fotiadis, A reasoning-based framework for car driver's stress prediction Control and Automation, 16th Mediterranean Conference on Control and Automation, 2008, pp. 627 - 632; doi: 10.1109/MED.2008.4602162.
[13] S. Challa and D. Koks, Bayesian and Dempster-Shafer fusion, Sadhana Vol. 29, Part 2, 2004, pp. 145176.
[14] J. J. Braun, Dempster-Shafer theory and Bayesian reasoning in multisensor data fusion, Proc. SPIE 4051, 255 (2000); doi:10.1117/12.381638.
[15] F. Cremer, E. Breejen, K. Schutte. Sensor Data Fusion for Anti-Personnel Land-Mine Detection. Proc. EuroFusion 1998, pp. 5560.
[16] I. Dagli, Erkennung von Einscherer-Situationen für Abstandsregeltempomaten, Universität Tübingen, 2005.
[17] D. Kasper, G. Weidl, T. Dang, G. Breuel, A. Tamke, W. Rosenstiel. Object-Oriented Bayesian Networks for Detection of Lane Change Maneuvers. Intelligent Vehicles Symposium (IV), 2011 IEEE.
[18] F.V. Jensen, Bayesian Networks and Decision Graphs, Springer-Verlag, 2001.
[19] D. Koller, and A. Pfeffer, Object-Oriented Bayesian Networks (OOBN), In Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-97), pp. 302-313, Providence, Rhode Island, August 1-3, 1997.
[20] G. Weidl, A.L. Madsen, S. Israelsson, Object-Oriented Bayesian Networks for Condition Monitoring, Root Cause Analysis and Decision Support on Operation of Complex Continuous Processes: Methodology and Applications, 2005: http://www.ist.uni-stuttgart.de/reports/pdf/2005-1.pdf
[21] H.D. Dickmanns, A. Zapp. A curvature-based scheme for improved road vehicle guidance by computer vision. In Proc. SPIE Conference on Mobile robots, pp. 161-168, 1986.
[22] H. Fritz, Patentschrift, Vorschrift zur Durchführung eines Fahrspurwechsels durch ein Kraftfahrzeug, DaimlerChrysler AG, 2000.
[23] K. Levenberg, A Method for the Solution of Certain Problems in Least Squares, Quart. Appl. Math. 2, pp. 164-168, 1944.
[24] D. Marquardt, An Algorithm for Least-Squares Estimation of Nonlinear Parameters, SIAM J. Appl. Math. 11, pp. 431-441, 1963.
[25] J. J. Moré, The Levenberg-Marquardt algorithm: Implementation and theory, Lecture Notes in Mathematics, 1978, Volume 630/1978, pp. 105-116, doi: 10.1007/BFb0067700.

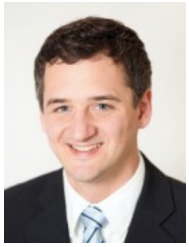
Dietmar Kasper received his diploma degree in technical computer sciences from the University of applied sciences in Esslingen in 2006. From 2006 to 2009 he worked as a developer in cooperation with the Daimler AG. At the same time he received his Masters degree in Distributed Computing Systems Engineering from the Brunel University in London. Since 2009 he is pursuing his Ph.D degree in the situation analysis research team at the Daimler AG. His dissertation title is "Driving Maneuver Recognition using object-oriented Bayesian Networks".

Galia Weidl has obtained her MSc. (Hons.) degree in physics and mathematics from the St. Petersburg State University, Russia, in 1993, her Fil.Lic. degree in theoretical physics from the Stockholm University, Sweden, in 1996, and a TeknD doctoral degree in process engineering from Mälardalen University, Sweden in 2002. Until 2006 she has been a postdoc at Stuttgart University. She was with the research teams of ABB Sweden (1997-2002), Bosch (2006-2008) and Daimler (since 2008). Her current research topic is Bayesian networks in the area of advanced driver assistance and safety systems.

Thao Dang studied electrical engineering at the University of Karlsruhe, Germany, the Massachusetts Institute of Technology, Cambridge, and the University of Massachusetts, Dartmouth. He received the Dr.-Ing. degree (Ph.D.) with distinction from the University of Karlsruhe, Germany, in 2007. Since September 2007, he has been a Research Engineer with Daimler Research and Development, Ulm, Germany. His research interests include driver-assistance systems, stereo vision, camera self calibration, and situation analysis for autonomous systems.

Gabi Breuel studied technical cybernetics (1976-82) at the University of Stuttgart. Subsequently, she attained a doctorate in 1989 from the University of Stuttgart at the faculty "Technical cybernetics and process engineering". Since 1996 she is with the Daimler AG research and technology. After activities in the areas of traffic engineering and telematics she joined the department "Active safety and Driver assistance systems" in 2002. Since 2008 she is manager of the research group "Situation analysis for Driver assistance systems".

Andreas Tamke studied Software Engineering at the University of Stuttgart (2000-2005). From 2006 to 2010 he worked at the Institute of Parallel and Distributed Systems of the University of Stuttgart on his Ph.D degree. Since 2010 he is with the Daimler AG research and technology as a member of the situation analysis research group. His current research topics are simulation and criticality assessment of traffic scenarios.



Andreas Wedel studied Mathematics and Information Technologies at the University of Bonn (2001-2006) and the University of Canterbury, New Zealand (2005). He received his Ph.D degree in computer vision from the University of Bonn (2009). Since 2006 he is with the Daimler AG research and technology and joined the situation analysis research group in 2009. His current interests are free space estimation and path planning for autonomeous vehicles, motion estimation and sensor fusion.

Wolfgang Rosenstiel received his Ph.D. in 1984 from Karlsruhe University. Since 1990 he is a Professor (Chair for Computer Engineering) at the Wilhelm-Schickard-Institute for Informatics at the University of Tübingen, as well as Director of the FZI Department "System Design in Microelectronics". He is chairman of eda-centrum, member of ITRS-Design-Committee, editor-in-chief of the journal Design Automation for Embedded Systems, and committee member of DFG senate for Collaborative Research Centers. His research topics are hardware/software design and design tools for embedded systems especially for driver assistance in automotive applications and machine learning algorithms.