Situation Analysis and Adaptive Risk Assessment for Intersection Safety Systems in Advanced Assisted Driving

Prof. Dr. Jianwei Zhang[1] and Bernd Roessler[2]

1: University of Hamburg, Faculty of Mathematics, Informatics and Natural Sciences, Department Informatics, Group TAMS, Vogt-Kölln-Straße 30, D - 22527 Hamburg,
2: Ibeo Automobile Sensor GmbH, Merkurring 20, 22143 Hamburg

Abstract. Intersection Safety Systems (ISS) are a relative new but an important research topic in the field of Advanced Driver Assistance Systems as accident statistics show. Unfortunately, intersections are one of the most complex scenarios out of all traffic related scenarios which complicates the development of such ISS. This paper presents situation analysis and risk assessment algorithms for Intersection Safety Systems which are suitable for online implementation. The demonstrator system is able to observe the intersection environment with several onboard sensors and to build an appropriate scene model including behaviors, intentions and interrelations of all vehicles in the scene. The subsequent risk assessment judges possible individual risks for the vehicle that is equipped with the safety system.

1 Introduction

Detailed personal car accident analysis shows that there is an urgent need to assist drivers at intersections. From [ROS$^+$05] three main scenarios can be extracted which cover a crucial number of intersection accidents. They define the basis for the ISS and the pertinent interpretations which are required in this paper. These scenarios are the "left turn across path scenario", the "turn into/straight crossing path scenario" and the "red light crossing scenario". Each situation is covered by a separate assistance function in this work. At first the scenario is analyzed by a behavior modeling and prediction with a Dynamic Bayesian Network approach. In a next step, the risk of the current situation is computed by an adaptive fuzzy logic based approach. Finally, the risk is presented to the driver of the equipped vehicle through an appropriate HMI.

The demonstrator vehicle used for this work has two laser scanners integrated in its two front corners. An additional video camera is mounted inside of the vehicle behind the windscreen (see Fig. 1). The specifications of the sensors can be found in [HST$^+$05]. The laser scanners are used for host vehicle landmark localization, object detection classification and tracking [RF06]. The video camera is used for lane detection. A time stamp based data fusion combines both localization outputs in order to gain a precise position and orientation of the vehicle within the intersection. Fig. 1 shows the test vehicle with the mounting positions of each sensor.

Fig. 1. Sensor integration in the test vehicle

2 Intersection Modeling

Intersections can include different lanes for different tasks (e.g. turning lanes), boundaries (e.g. curbstones, refuges, lane-markings) and last but not least complex right of way regulations (e.g. traffic-signs, traffic lights). Therefore, intersections are very complex since all traffic participants can move in nearly arbitrary directions and with different aims. This is the reason why intersections were neglected in the past and many situation analysis tasks for driver assistance applications were built upon freeway scenarios. In this work, this complex structure of intersections is reduced to a model that allows the formulation of many assistance functions for intersection scenarios with only few restrictions. The modeling developed in this work is a modification of the GDF (Geographic Data Files) description [fS02] and is named Lane Model Description (LMD). The idea is to introduce a lane as the most basic feature of its description. A road element of GDF turns into a lane in LMD and a junction into a lane link respectively. The road of GDF is interpreted as a road element in LMD. Fig. 2(a) shows the LMD in a GDF like format in order to express the different approach. For an advanced driver assistance system like the one developed in this work, it is crucial to have exact information on the driving lanes of a road. The developed ISS uses lane information amongst others in order to recognize possible conflicts and to identify the road users aims. In addition to *real* lanes with markings at each side, lanes of the LMD can also express *virtual* lanes that describe the possible paths a vehicle can take within the intersection. They provide an abstraction of the complex structure of an intersection by reducing it to the minimum required information. We will see later that this is highly suitable for high level assistance functions at intersections.

3 Behavior Modeling

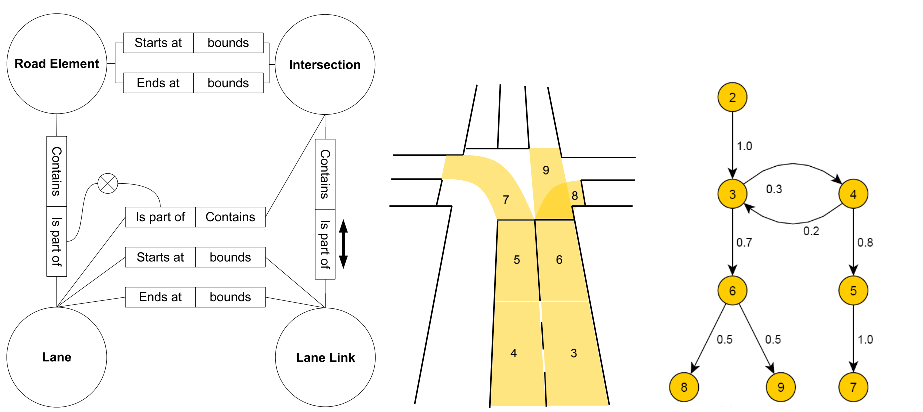
The driving behavior in this work is represented in terms of a few geometric entities which facilitates the calculation of possible trajectories and therefore the determination of possible conflicts within an intersection. This drastically reduces the computational complexity of the whole system and makes it suitable for online analysis tasks. The entities for the behavior modeling are the driving lanes (real and virtual lanes) of the intersection. The idea is that each traffic participant only drives in predefined areas within the intersection, i.e. the lanes. This form of modeling can be interpreted as the definition of one common trajectory with a tolerance area on both sides which corresponds to the width of a lane. Accident analysis [ROS+05] shows that accidents resulting from wrong lane changes are of no relevance in intersection accidents. Thus, the proposed behavior modeling is suitable for the proposed ISS.

In order to represent the behavior of all traffic participants, a common structure is needed. For this representation the LMD (see Section 2) is transferred into a probabilistic graph model for each object.

A **directed** graph is a tuple $G = (V, E)$ where $V(G)$ is the set of vertices and $E(G) \subseteq V(G) \times V(G)$ the set of edges in the graph G, respectively. The graph G is called **undirected** if $E(G) = E(G)^{-1}$ with $E(G)^{-1} = \{(u,v) | (v,u) \in E(G)\}$.

The k-th path $pa(v_s, v_e)$ of a directed graph G from vertex v_s to vertex v_e is a tuple $(v_1, \ldots, v_{n_k})$ where $v_1 = v_s, v_{n_k} = v_e$ and $(v_i, v_{i+1}) \in E(G) \forall i \in (1, \ldots, n_k - 1)$.

The so-called Microscopic Behavior (MiB) of a driver forms a probabilistic behavior network which is a directed graph with lanes as vertices and transition probabilities at the edges. Those transition probabilities from vertex v_1 to vertex v_2 are denoted by $prob(v_1, v_2)$. The MiB is build out of the LMD starting at the initial position of the appropriate traffic participant. The possible next lanes from the driving lane at initialization time are traversed and inserted into the graph. The possible next lanes are *the lanes connected to the same link as the considered lane* and *the neighboring lanes at the left and right side of the considered lane*. The transition probabilities describe the likelihood of a traffic participant moving from one lane to another. Normally, the likelihoods at the edges of the MiB are uniformly distributed over the number of outgoing edges for each vertex but they can also express terms like the most probable path at an intersection by assigning different probabilities to the outgoing edges. Nevertheless, the probabilities at the outgoing edges of one node must add up to 1.



(a) (b) (c)

Fig. 2. LMD and Microscopic Behavior

Fig. 2 shows a sample intersection and the corresponding MiB for one approaching direction. In Fig. 2(b) the lanes which are used in the MiB of Fig. 2(c) are shaded and numbered in order to state the relationship. Nevertheless, the shown MiB has additional nodes whose lanes are not shown in the intersection figure. A new lane, i.e. a new node, is used if there are changes in the attributes of the corresponding lane. The shown MiB also uses the mentioned deflection from the uniformly distributed probabilities at the edges, i.e. here it is more likely for a traffic participant to be going straight ahead than to be turning left. The initial global probabilities for a maneuver at an intersection (e.g. straight, left, right) can be computed from the MiB by summing the multiplied probabilities of all possible paths that lead to the destination starting from a specific vertex. Therefore the following has to be calculated: $p(v_s \xrightarrow{*} v_e) = \sum_{k=0}^{n} p(v_s \xrightarrow{k} v_e)$, where n is the number of possible paths from v_s to v_e and $p(v_s \xrightarrow{k} v_e)$ is the probability for reaching v_e beginning at v_s in the k-th path $pa_k(v_1 = v_s, v_{n_k} = v_e)$ which is computed by $p(v_s \xrightarrow{k} v_e) = \prod_{i=1,\dots,n_k-1} prob(v_i, v_{i+1})$.

In this example, the initial global probability for a left turn is the limit of a geometric series $p(v_s = 2 \xrightarrow{*} v_e = 7) = \lim_{n\to\infty} \sum_{k=0}^{n} (0.3 \cdot 0.8) \cdot (0.3 \cdot 0.2)^k \approx 0.255$. Initially, the probabilities at the edges are uniformly distributed or assigned due to statistical background knowledge on the driving behavior at the intersection. Nevertheless, they can be modified by the application if the distribution changes due to some reason. Each MiB covers a specific maximum distance which can be seen as the range of vision for the appropriate traffic participant. Within this distance the behavior of the vehicle is described. It is mainly driven by the field of view of the sensors. The wider the field of view, the more extensive the maneuver decision as well as the behavior prediction can become.

4 Behavior Prediction

The prediction of human behavior is a very complex task, especially at intersections. In general, the behavior prediction can be classified into high-level and low-level behavior prediction. The high-level prediction considers global maneuver decisions like e.g. the intention to turn left or to stop at the intersection (e.g. due to a red traffic light). In the low-level prediction, trajectories or speed profiles are considered which describe the maneuvers from the high-level prediction. If thinking of a "simple" left turn maneuver at the low-level, the number of trajectories that can be used is very high (different trajectories and even different speed profiles). The accuracy of the prediction module developed in this work lies between this high-level and low-level prediction. It can be seen as the determination of the probability $P(v_1 \xrightarrow{*} v_2)$ of two arbitrary vertices of the MiB in each time step. Since the behavior prediction is based on the behavior network description which is defined on real and virtual lanes, the current lane information of a vehicle is of very high importance. The developed model is capable of solving two problems in parallel: 1) Computation of $P(x_t)$ for all lanes of the model (lane assignment for the current time step) and 2) Computation of $P(x_{t+\delta})$ for all lanes of the model (maneuver prediction for a vehicle).

For the realization of the behavior prediction, the theory of Dynamic Bayesian Networks (DBN) was used. Generally, the used network can be interpreted as an estimator for the lane a vehicle is driving on. Due to the description of the vehicle's behavior

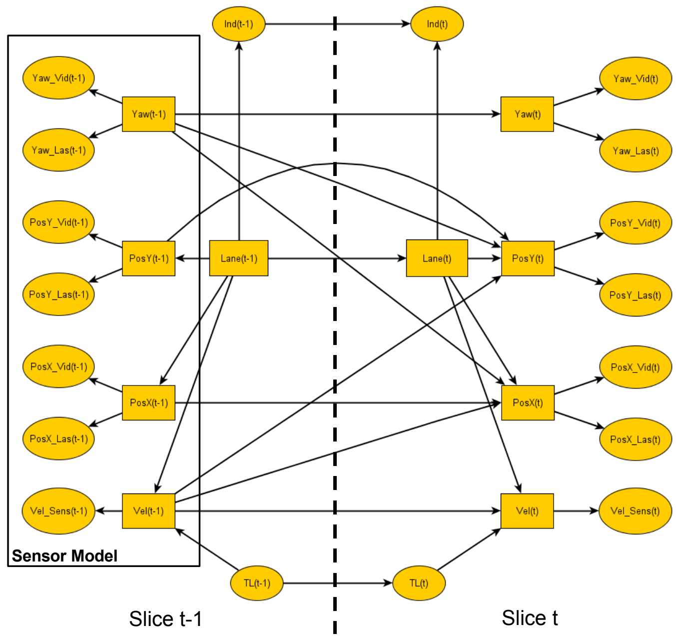


Fig. 3. 2 TBN Dynamic Bayesian Network for Behavior Prediction

solely by lanes, the intention of a driver can be inferred from the model. Fig. 3 shows the DBN that is used for current lane estimation as well as the behavior prediction. The observed states, i.e. the evidences in each slice, are the position of the vehicle, the indicator state (especially important for the approaching phase to an intersection and intersections with no separate turning lanes), the velocity and the status of the traffic light. The hidden state is the current lane. Through a filtering process the actual lane can be estimated for the current time t.

If the behavior of all traffic participants is analyzed, it has to be evaluated if the predicted plans are at odds with each other. In a first step, the possible conflicts that arise because of the construction of the intersection have to be identified and expressed by a so-called Conflict Graph (CG). In a second step, the conflicts that could arise because of the current combination of the objects and their predicted plans have to be analyzed. At the end a comprehensive view on the scene emerges.

The CG is a graph that results from an interconnection of the different MiBs of each traffic participant currently in the scene. An interconnection between two vertices of two MiBs exists if there is a potential conflict with the lanes belonging to these vertices. A potential conflict exists if two lanes have an overlapping area called conflict zone. The conflict zone can be reduced to the Conflict Point (CP) which is the intersection of the middle line of both lanes. The edges that represent a conflict are undirected edges in order to differentiate them from the transition edges of the initial MiBs. The initial CG does not take the traffic control at the intersection into account, i.e. it cannot be seen which lane of the connection is preferred in terms of right-of-way regulations.

If the computed CG is reduced to the conflicts where the host vehicle has to give right-of-way to other vehicles, the Right-of-Way Graph (RoWG) is obtained. Therefore, the RoWG is only valid for the host vehicle. While the CG is a *static* graph, the RoWG is a *dynamic* graph that can change continuously during system operation. Every time a new vehicle is detected at the intersection or the categories of the lanes change (e.g. due to a traffic light), the RoWG has to be updated.

5 Adaptive Risk Assessment

Calculating the risk for a driver is a complex task as well. The situation analysis provides all possible conflicts and the assigned probabilities of the driving behavior of the vehicles. With an analytical approach it would be very hard to try to compute the occurrence of the conflicts, to cope with the fact of imperfect sensor data and to include aspects like the subjective assessment of the current risk for a specific driver. Here, a fuzzy rule-base is built to reproduce "human thinking" for risk assessment so that an adequate strategy can be formulated. Input variables are speed, acceleration and distance to the conflict. In addition, inputs such as the type of the opponent could also be used as a severity index for a possible crash [SUBW04]. Visibility can also easily be modeled in such a fuzzy rule-base system since this also effects risk assessment. E.g., the risk for an available time-gap in an intersection could be assessed higher for poor visibility than in very good weather conditions. The risk assessment takes the probabilities of the behavior prediction as weighting factors and computes a risk level for each possible conflict in the time and space domain. The results of the risk assessment module are different risk levels for all possible conflicts in the scene.

A fuzzy-logic [Zad65] approach was used in this work to judge the risk of a possible conflict for the intersection scenarios. Such a fuzzy-logic approach allows the formulation of the rules for risk assessment in a very natural and easy way. No widespread expert knowledge is needed in order to define the rules so that the resulting system can work properly. The concept of the inference system is based on B-spline membership functions and is similar to that presented in [ZK99]. Each input is uniformly covered by five membership functions. This is due to a natural human partitioning of a specific linguistic variable. The linguistic variable *speed* would therefore be represented by the linguistic terms *very slow*, *slow*, *medium*, *high* and *very high*. The risk computation is shown in Equation (1).

$$risk = \frac{\sum_{i_1=1}^{m_1} \cdots \sum_{i_n=1}^{m_n} (Y_{i_1,\ldots,i_n} \prod_{j=1}^{n} X_{i_j,k_j}^{j}(x_j))}{\sum_{i_1=1}^{m_1} \cdots \sum_{i_n=1}^{m_n} \prod_{j=1}^{n} X_{i_j,k_j}^{j}(x_j)} = \sum_{i_1=1}^{m_1} \cdots \sum_{i_n=1}^{m_n} (Y_{i_1,\ldots,i_n} \prod_{j=1}^{n} X_{i_j,k_j}^{j}(x_j)) \quad (1)$$

where x_j is the j-th input, k_j the order of the B-Spline basis function for x_j, X_{i_j,k_j}^{j} the i-th B-Spline membership function of x_j, $i_j = 1,\ldots,m_j$ the number of membership functions for the linguistic variable for x_j and $Y_{i_1 i_2 \ldots i_n}$ the fuzzy singleton for Rule$(i_1,i_2,\ldots,i_n)$.

This above risk assessment computation is performed for every conflict c which possibly can occur in the current situation. Since due to the number of the road users currently at the intersection and their predicted driving behavior which will be less than 100% most of the time, a lot of possible conflicts can occur. Therefore an additional weighting factor has to be added to the risk computation which influences the overall risk for a specific conflict. Assuming $\mathscr{C} = (c_1, \ldots, c_N)$ are the current conflicts for a specific situation, the most critical conflict is computed by:

$\theta = \mathrm{argmax}_{c_l \in \mathscr{C}} \left[\omega_{c_l} \sum_{i_1=1}^{m_1} \cdots \sum_{i_n=1}^{m_n} (Y_{i_1,\ldots,i_n} \prod_{j=1}^{n} X_{i_j,k_j}^{j} (x_{j,c_l})) \right],$

where x_{j,c_l} is the j-th input and ω_{c_l} the weight for conflict c_l. Several procedures for the computation of the weight ω_{c_l} can be used. All are based on the probabilities calculated for the current conflict. The easiest way is to weight the risk computation simply by the probability for the conflict: $\omega_{c_l} = p(c_l)$. This has the effect that the risk changes if the probability for the conflict changes. If the risk should not change proportionally to the probabilities of the conflicts, the following weighting factor can be used:

$\omega_{c_l} = 1$ if $p(c_l) > \frac{1}{N}, \omega_{c_l} = 0$ otherwise, where N is the number of calculated possible conflicts.

The fuzzy-system approach introduced above can be extended with the ability for optimizing and generating rules by a machine learning approach. The practical suitability of this method was among others shown by the authors of this paper in the field of robotic grasp learning [ZR03]. The goal is to minimize the following squared error function: $E = \frac{1}{2}(risk_r - risk_d)^2$, where $risk_r$ and $risk_d$ are the current calculated risk and the desired outcome, respectively. The momentous risk level $risk_r$ is thereby calculated by Equation (1). In order to minimize the error function, the parameters $Y_{i_1,\ldots,i_n}$ of Equation (1) have to be adapted. For this purpose the gradient descent method is used: $\Delta Y_{i_1,\ldots,i_n} = \varepsilon \frac{\delta E}{\delta Y_{i_1,\ldots,i_n}} = \varepsilon (risk_r - risk_d) \prod_{j=1}^{n} X_{i_j,k_j}^{j} (x_j)$.

The automatic adaption of the proposed risk assessment by machine learning techniques offers additional enhancement. It is either done online in the vehicle or offline in a developed simulation. In both cases the adaption is used in order to adjust the parameters of the membership functions for the fuzzy rule base.

In the offline risk adaption, the parameters for the fuzzy rule base are generated automatically from scratch without designing the membership functions with expert knowledge. The idea is to simulate a lot of different situations which do or do not result in accidents. From the conflict situation (e.g. accident, near accident, no accident) corresponding feedback can be calculated and thus the parameters can be adopted. The measure for evaluating the risk is called Conflict Time Gap (CTG). For the definition of this measure the Time-to-Collision-Point (TTCP) is used as differentiation to the well-known term Time-to-Collision (TTC) (e.g. see [vdHH93]). For the TTC the time trajectories of two vehicles $(\mathscr{T}_h(x,y), \mathscr{T}_o(x,y))$ are intersecting exactly in one point $P = (x_c, y_c, t_c)$ in the time-space domain. Thus, the TTC is the same for both vehicles. This results in: $\mathsf{TTC} = \mathscr{T}_h(x_c, y_c) = \mathscr{T}_o(x_c, y_c)$ The TTCP is defined as the time to a conflict point (within an intersection) where a collision would occur if the last equation would be true. For the TTCP both trajectories are not necessarily intersecting in the time-space domain but at least in the space domain, i.e. at $P = (x_c, y_c)$. This means that the TTCP is simply the time value of the time-space trajectory of the host vehicle h at a defined conflict point: $\mathsf{TTCP}_h = \mathscr{T}_h(x_c, y_c)$. The TTCP for the host vehicle h

becomes the TTC if another vehicle o exists where $\mathsf{TTCP}_o = \mathsf{TTCP}_h$. Considering these two vehicles, the CTG is defined as follows: $\mathsf{CTG} = abs(\mathsf{TTCP}_h - \mathsf{TTCP}_o)$. The CTG defines the time for a vehicle V_1 to reach a predefined point P after another vehicle V_2 has already crossed it. It is used to compute the risk of the situation and its conflict point: $risk_d = max(0., risk_{max} - \frac{CTG \cdot risk_{max}}{CTG_{max}})$, where $risk_{max}$ is the maximum number of the risk level and CTG_{max} the maximum CTG up to which a risk should be assigned.

The online risk adaption is performed in order to adjust the warning system to a specific driving behavior or driver's skill. Here, an initial parameter set is already given and the system is performing well in most situations. Just for some specific situations, where a driver reacts noticeably against the warning or recommendations of the assistance system, the risk assessment is adapted to the driver. Therefore, it is necessary to compare the computed risk of the system with the risk felt by the driver. So $risk_d$ becomes the felt risk of the current driver. The strategy introduced is as follows: if a driver always disregards the suggestion/warning of the assistance system in the car because from his point of view it is not suitable for his driving behavior, the automatic risk assessment adaption can adjust to his skills. This strategy can be explained by rules like "if the computed risk is high, but the driver passes the conflict point without stopping anyhow, then the risk is changed to a lower risk". Vice versa, "if the computed risk is low, but the driver stops in front of the conflict point anyhow, then the risk is changed to a high risk".

6 Results

The described behavior prediction and risk assessment algorithms were successfully implemented in the real demonstrator vehicle. A big challenge for humans while driving a car is to assess parameters like speed and distance of other approaching vehicles. Therefore, the developed HMI approach uses a warning interface that visualizes the computed risk level in a continuous manner for the time of a possible dangerous situation to the driver (see Fig. 4) In this way, the driver has a direct visual link to those parameters which are difficult to estimate.

Fig. 4. The used HMI for Risk Level Visualization in the Demonstrator Vehicle

6.1 System and User Test Results

System as well as user tests were carried out on a test intersection [FHO+07]. The system test evaluations were carried out based on the number and the rate of correct alarms, false alarms and missing alarms. The results of the test indicated that the system had a correct alarm rate of 93% in left turn scenarios and 100% in lateral traffic scenarios. For the user tests sixteen untrained subjects had been selected by taking their age, gender and driver experience into account. Each subject took around 2.5 hours to drive the demonstrator vehicles on the test intersection and assessed the performance of the systems. The subjects rated the ISS helpful and relieving. Further analysis showed that the subjects thought the ISS for left turn was more useful than for lateral traffic. They judged that such a system could have helped them in their daily driving and it was agreed that it would improve traffic safety.

7 Conclusion

Intersection Safety is a very hot research topic. Nearly every automobile manufacturer and also the suppliers have recognized that future advanced driver assistance systems have to deal to some extent with the topic of ISS. This is mainly due to the fact that intersections are a black spot in terms of road accidents. This work has shown a promising approach for all of the tasks that need to be solved in order to build a comprehensive ISS. A new approach was shown how to deal with the challenging topic of scenario interpretation and risk assessment in intersection safety systems. This approach was successfully implemented on a demonstrator vehicle and extensively tested in the European-funded project PReVENT-INTERSAFE.

References

[FHO+07] K. Fürstenberg, M. Hopstock, M. A. Obojski, B. Rössler, J. Chen, S. Deutschle, C. Benson, J. Weingart, and A. de La Fortelle. D40.75 final report. INTERSAFE Deliverable, 2007.

[fS02] International Organization for Standardization. *Intelligent transport systems - Geographic Data Files (GDF) - Overall dataspecification*. Draft International Standard ISO/DIS 14825. ISO, 2002.

[HST+05] A. Heenan, C. Shooter, M. Tucker, K. Fuerstenberg, and T. Kluge. Feature-Level Map Building and Object Recognition for Intersection Safety Applications. In *9th International Forum on AMAA 2005*, 2005.

[RF06] B. Rössler and K. Fürstenberg. Intersection Safety - the EC Project INTERSAFE. In *10th International Forum on Advanced Microsystems for Automotive Applications*, Berlin, April 2006.

[ROS+05] B. Rössler, M. A. Obojski, M. Stanzel, D. Ehmanns, T. Kosch, M. Hopstock, C. Shooter, A. Heenan, P. Woolley, C. Benson, J. F. Boissou, Y. Martail, L. Serezat, Y. Page, C. Chauvel, K. Fürstenberg, J. Mousain, A. de La Fortelle, M. Parent, and S. Deutschle. D40.4 requirements for intersection safety applications. INTERSAFE Deliverable, 2005.

[SUBW04] D. Sun, S. Ukkusuri, R. F. Benekohal, and S. T. Waller. Fuzzy logic based online collision prediction system for signalized intersections. In *Advances in Transportation Studies an International Journal*, July 2004.

[vdHH93] R. van der Horst and Jeroen Hogema. Time-to-Collision and Collision Avoidance Systems. In *6th ICTCT Workshop*, Salzburg, 1993.

[Zad65] L. A. Zadeh. Fuzzy sets. In *Information and Control*, pages 338–353, 1965.

[ZK99] J. Zhang and A. Knoll. Designing fuzzy controllers by rapid learning. In *Fuzzy Sets and Systems*, pages 287–301, January 1999.

[ZR03] J. Zhang and B. Rössler. *Grasp Learning by Active Experimentation Using Continuous B-Spline Model*, chapter 3, pages 353–372. In Autonomous Robotic Systems - Soft Computing and hard Computing Methodologiesand Application. Physica-Verlag, 2003.