

# Velocity-based Driver Intent Inference at Urban Intersections in the Presence of Preceding Vehicles

Martin Liebner, Felix Klanner, Michael Baumann, Christian Ruhhammer and Christoph Stiller

**Abstract**—Predicting turn and stop maneuvers of potentially errant drivers is a basic requirement for advanced driver assistance systems for urban intersections. Previous work has shown that an early estimate of the driver's intent can be inferred by evaluating the vehicle's speed during the intersection approach. In the presence of a preceding vehicle, however, the velocity profile might be dictated by car-following behavior rather than by the need to slow down before doing a left or right turn. To infer the driver's intent under such circumstances, a simple, real-time capable approach using a parametric model to represent both car-following and turning behavior is proposed. The performance of two alternative parameterizations based on observations at an individual intersection and a generic curvature-based model is evaluated in combination with two different Bayes net classification algorithms. In addition, the driver model is shown to be capable of predicting the future trajectory of the vehicle.

**Index Terms**—Driver Intent Inference, Intelligent Driver Model, Path Curvature, Velocity Profile, Intersection Approach

## I. INTRODUCTION

Urban intersections have long been known to be a hotspot for accidents resulting in major injuries [1]. While passive safety systems can help to prevent such injuries for passengers inside a car, pedestrians and cyclists – so called vulnerable road users (VRU) – are left with limited protection.

Active safety systems aim to prevent or mitigate the effects of accidents before they happen. A typical application would be to warn the driver if he seems to have overlooked a relevant VRU. This poses two major challenges to the system: First, the early detection of VRUs which cannot satisfactorily be achieved by state-of-the-art onboard sensors, and second, to determine whether there is a need to warn the driver of that particular VRU.

The research initiative Ko-FAS [2] addresses the issue of detecting VRUs both by research on cooperative sensor technology [3] and stationary high resolution camera systems [4]. The general idea of Ko-PER [5], one of the projects associated with Ko-FAS, is to share the local perception of each individual car or infrastructure unit using a novel data fusion architecture [6] based on a manufacturer independent Car2X communication solution. Also, several approaches to lane level self localization are investigated [5], [7].

To resolve the second issue of whether or not the driver should be warned, the trajectory on which he intends to cross the intersection needs to be determined. This and the

underlying task of estimating the probability for each possible intent are addressed by this paper.

## A. Related Work

Driver intent inference for urban intersections has been an important research topic for the past couple of years. Approaches reported in literature include Bayesian networks [8], Monte Carlo Simulation [9], Hidden Markov Models (HMM) [10], [11], [12], Support Vector Machines [13] and prototype based methods [14].

The most obvious feature for predicting a driver's intent to do a left or right turn would be the turn indicator. However, according to a study of the *Auto Club Europa*, the indicator is set for only 80% of all turn maneuvers [15]. Also, it might be set although the driver actually intends to go straight, which is often the case when there are multiple possibilities to turn right that are close to each other. Finally, the indicator signal might just not be available, which is the case when we try to reason about surrounding vehicles that are observed only by RADAR or LIDAR sensors.

An alternative feature for estimating the driver's intent has been identified to be the velocity profile of a driver approaching the intersection [16]. In the presence of a preceding vehicle, however, the velocity profile might be dictated by car-following behavior rather than by the need to slow down before doing a left or right turn. For prototype based methods like [14], this is very difficult to account for due to the large number of possible situations. While HMM and Monte Carlo based methods are capable of dealing with interaction between vehicles in principle, doing so is computationally rather expensive.

In [17], the Gipps model [18] was used to represent car-following behavior directly. The deceleration of a driver in preparation of a turn maneuver was modeled deterministically by a set of fuzzy-logic rules based on the geometry of the path lying ahead. The driver's intent was estimated using a computationally expensive particle filter that limited the approach to non-realtime use.

## B. Problem addressed

Similar to the approach taken in [17], this paper introduces a method to infer the driver's intent based on an explicit model for the vehicle's velocity profile. Explicit models are more transparent than machine learning methods and more robust when generalized to situations for which there is no training data available.

In contrast to [17], the *Intelligent Driver Model* [19] is used to represent both car-following and turning behavior. Alternative parameter sets are extracted from real world data

Martin Liebner, Felix Klanner, Michael Baumann and Christian Ruhhammer are with BMW Group, Research and Technology, D-80788 Munich, Germany. {martin.liebner, felix.klanner}@bmw.de

Christoph Stiller is with Karlsruhe Institute of Technology, Institute for Measurement and Control, D-76131 Karlsruhe, Germany. stiller@kit.edu



Fig. 1. Pedestrian crossing and intersection approach. The rightmost lane has been used for measurements and evaluation.

to account for different driving styles. The general idea is to compare the trajectory of the past few seconds with the simulated driver behavior obtained from different parameter sets and driver intents to estimate the posterior probability of each such hypothesis. The resulting probability distribution can be used to make predictions both about the driver's intent and the future trajectory of the vehicle. Based on a Bayes net instead of a particle filter, our approach is real-time capable and can be easily extended to include additional features such as the turn indicator or the lateral lane center deviation.

The capability of our approach is evaluated for the rightmost lane of the intersection approach shown in Figure 1. In principle, there are four possible driver intents to be recognized:

- 1) Go straight.
- 2) Stop at red light.
- 3) Turn right.
- 4) Turn right, but stop at pedestrian crossing.

In the paper at hand, evaluation is based on distinguishing between going straight and turning right only. This is in line with a possible application of the system: To warn the driver of a cyclist that is moving parallel to the street if and only if the driver actually intends to do a right turn.

This paper extends preliminary work presented in [20] by a more elaborate analysis of the comparison-based approach and the introduction of a curvature-based parameterization method that generalizes velocity-based driver intent inference to arbitrary intersections. Different driving styles are modeled by the maximum acceleration parameter  $a$  and the desired velocity model  $M$ , both of which depend on the actual driver intent  $I$ . A brief summary of our investigations is given in Figure 2.

The remainder of this paper is organized as follows: In Section II, the general data collection and trajectory mapping framework is described. Section III introduces the IDM car-following model and demonstrates how it can be used to model turning behavior. Two different parameterizations are obtained from real world data, one by clustering actual intersection crossings and one by evaluating the path curvature. In Section IV, two alternative approaches to the classification task

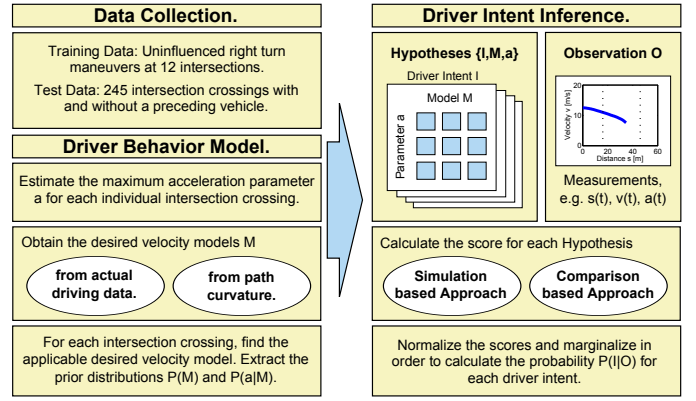


Fig. 2. Brief summary of our approach to velocity-based driver intent inference. The investigations include two alternative approaches both for the modeling and the classification task.

are discussed. Section V serves to evaluate the performance of each parameterization and classification method and to demonstrate the ability of our approach to predict the future trajectory of the vehicle. Finally, Section VI concludes this paper.

## II. DATA COLLECTION AND MAP USAGE

Measurement data was collected from 4 different drivers at an inner-city intersection. The data set contains 165 right turns and 80 straight intersection crossings, including situations with and without a preceding vehicle. Special care was taken to capture both defensive and sporty driving styles. Previous works have shown that a single driver exhibits a much larger variance in his intersection approaches than can be found between the average behavior of different drivers [16]. Therefore, the number of different drivers used for data acquisition should not be that important. The corresponding GPS traces are shown in Figure 3.

In industrialized countries, road traffic takes place in a highly structured environment where, most of the time, cars and even cyclists follow their designated lanes. In such an environment, accidents occur mostly due to right-of-way violations, i.e. in situations where one of the conflicting traffic participants should have waited for the other but did not. Therefore, the challenge of risk assessment and collision avoidance lies more in predicting discrete decisions like the planned path through the intersection, right-of-way interactions with other traffic participants and the longitudinal behavior of the driver rather than the fully fledged 2D-trajectory. Reducing the options of each driver to a discrete set of one-dimensional trajectories lowers the computational burden and allows to spend more effort on the question of how the driver is going to interact with other traffic participants.

A common way to map two-dimensional trajectories onto one-dimensional lanes is to use lane markings obtained from digital maps or satellite pictures. Beside the disadvantage of having to deal with an additional map error, this approach does not work well at the inside of intersections that often lack lane markings. Instead, we chose to use the GPS traces obtained by a high-precision differential GPS/INS platform to generate

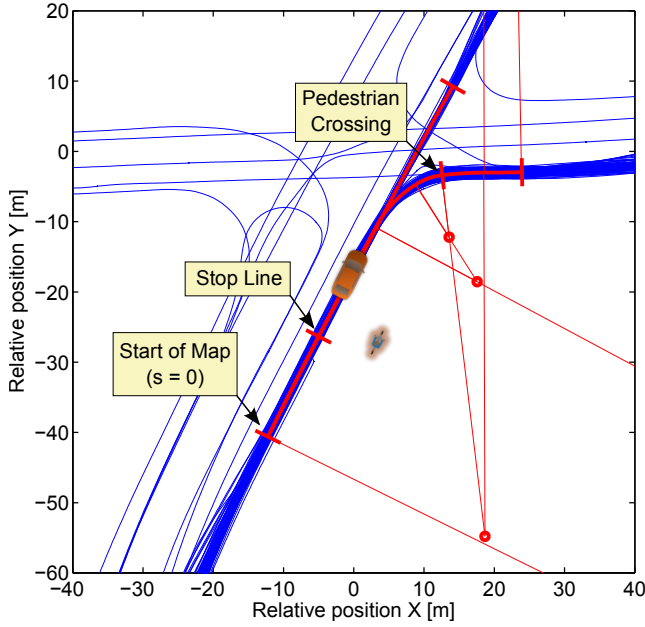


Fig. 3. GPS traces of intersection crossings and map consisting of two smooth circular arc splines. Auxiliary lines and arc centers are shown for the right turn branch only.

TABLE I  
DISTANCE ALONG CIRCULAR ARC SPLINE

Spline segment	0	1	2	3	4	5
Total distance [m]	0	33.52	42.00	45.92	51.98	57.36

a map that represents the mean path of vehicles crossing the intersection rather than the lane markings themselves.

Internally, the map uses smooth circular arc splines to represent the paths from one intersection entry to all possible exits. For the right turn path, the traveled distance at the end of each individual circular arc is summarized in Table I. An efficient algorithm to generate smooth circular arc splines is given in [21].

Compared to the standard polyline representation, smooth circular arc splines have a couple of advantages:

- Finding the correct mapping of an arbitrary point is computationally very efficient.
- Compact representation: Only few parameters are needed to model the individual arcs.
- The mapping of a smooth two-dimensional trajectory is continuous in the distance coordinate  $s$ , the orthogonal deviation  $y$  and their derivatives.
- The path curvature is directly represented in the map.

The velocity profiles of the intersection crossings are shown in Figure 4. Apparently, a significant difference between the profiles for turning right and going straight can be observed starting from 20 to 30 meters onwards. In contrast, the drivers intent to stop at the pedestrian crossing before turning right seems to be predictable only for  $s > 35$  m. Many trajectories include a stop at  $s \approx 13$  m, which corresponds to the position of the stop line at the traffic light.

For the mean path, the lateral deviations of the individual trajectories sum up to zero by definition. However, this is only

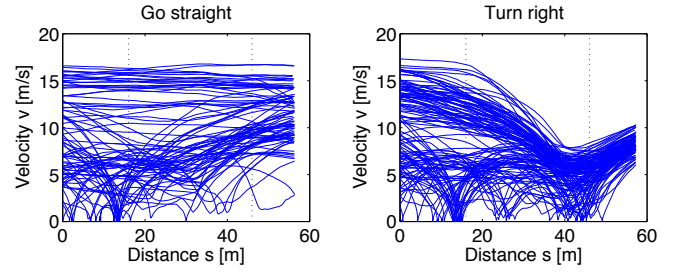


Fig. 4. Velocity profiles for going straight and turning right. The position of the stop line and that of the pedestrian crossing are indicated by dotted lines.

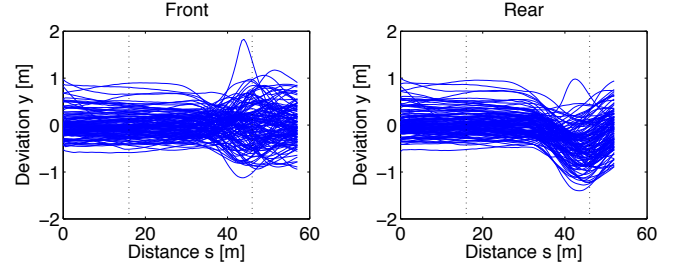


Fig. 5. Lateral deviation for turning right.

the case for the front of the vehicle for which the mean path was calculated. The rear of the vehicle is following a slightly different path throughout the right turn, which leads to high lateral deviations as shown in Figure 5. In order to obtain a correct mapping of preceding vehicles for which only the rear is observed, this fact has been accounted for by a model based approach.

### III. DRIVER BEHAVIOR MODEL

Microscopic car-following models have long been used to simulate individual driver behavior in the presence of preceding vehicles. A continuous, accurate but simple model for the acceleration  $\dot{v}$  is given by the Intelligent Driver Model (IDM):

$$\dot{v} = a \left[ 1 - \left( \frac{v}{u} \right)^\delta - \left( \frac{d^*(v, \Delta v)}{d} \right)^2 \right], \quad (1)$$

$$d^*(v, \Delta v) = d_0 + T v + \frac{v \Delta v}{2\sqrt{a} b}. \quad (2)$$

TABLE II  
PARAMETERS OF THE INTELLIGENT DRIVER MODEL

Parameter	Value
max. acceleration $a$	0.5 m/s <sup>2</sup>
acceleration exponent $\delta$	4
desired velocity $u$	0.60 km/h
comf. deceleration $b$	3 m/s <sup>2</sup>
min. gap to leading vehicle $d_0$	2.0 m
time gap to leading vehicle $T$	0.8 s

The corresponding parameter values are given in Table II. In the absence of a preceding vehicle, the calculated acceleration  $\dot{v}$  is determined only by the maximum acceleration parameter  $a$ , the current actual velocity  $v$ , the desired velocity  $u$  and a fixed acceleration exponent  $\delta$ . The influence of a preceding vehicle is represented by the ratio between the effective desired

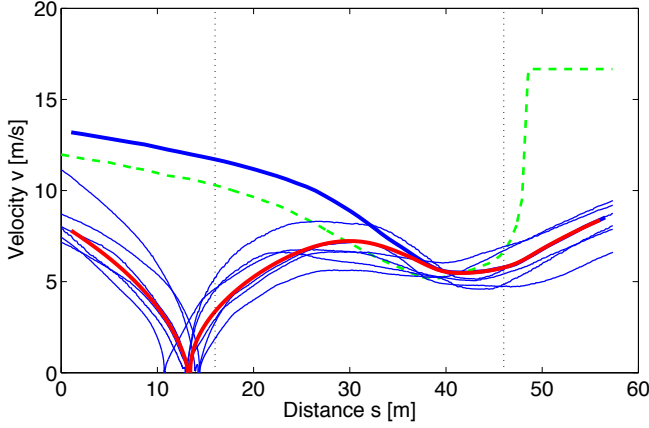


Fig. 6. Driver behavior during right turn maneuver. Original trajectory (thick blue), synthesized desired velocity curve (dashed green), trajectories for stop at red traffic light and consecutive right turn (thin blue), simulated trajectory (thick red).

gap  $d^*$  and the actual gap  $d$ . For the case without a preceding vehicle, we define the relative velocity  $\Delta v$  to be zero and  $d$  to be infinite which leads to  $d^*/d = 0$ . Note that the model is collision free, i.e. the calculated deceleration will always be as high as necessary to prevent a collision. Afterwards, it approaches the so-called comfortable deceleration  $b$  asymptotically.

In a typical instantiation of the model, the desired velocity  $u$  is a fixed value depending on the legal speed limit and the individual driver. For right turn modeling, however, it needs to be set dynamically as drivers will slow down when they approach the intersection. For a given velocity profile  $v_i(s)$  that is uninfluenced by a preceding vehicle, Equation (1) can be rewritten to obtain the desired velocities

$$u_i(s) = \min \left( \frac{v_i(s)}{\sqrt{s} \sqrt{1 - \dot{v}_i(s)/a_i}}, \bar{u} \right), \quad (3)$$

assuming a maximum desired velocity  $\bar{u} = 60 \text{ km/h}$ . Note that the car following term in Equation (1) has been omitted. To calculate the desired velocity profile, the maximum acceleration parameter  $a_i$  for the individual intersection crossing  $i$  needs to be known. Assuming  $u_i(s) = \bar{u}$  for  $s \geq 48 \text{ m}$ ,  $a_i$  is first estimated based on the acceleration after turning and then used to calculate the desired velocity profile for  $s < 48 \text{ m}$ .

An example velocity profile and the corresponding desired velocity profile are shown in Figure 6. The advantage of representing observations by  $u(s)$  rather than  $v(s)$  is that the former generalizes very well to arbitrary situations. This is demonstrated for a couple of velocity profiles that include a stop at  $s \approx 13 \text{ m}$ , which are reproduced by an IDM simulation using the original desired velocity profile and a virtual zero-velocity obstacle at the stop line at  $s = 16 \text{ m}$ , which is removed when  $v$  reaches zero. As can be seen, the simulated trajectory matches the observed trajectories very well even though the desired velocity profile has been obtained from a completely different trajectory.

Nevertheless, drivers can cross the intersection at different speeds and with varying acceleration parameters  $a$ . For the task of driver intent inference, it is necessary to obtain a small

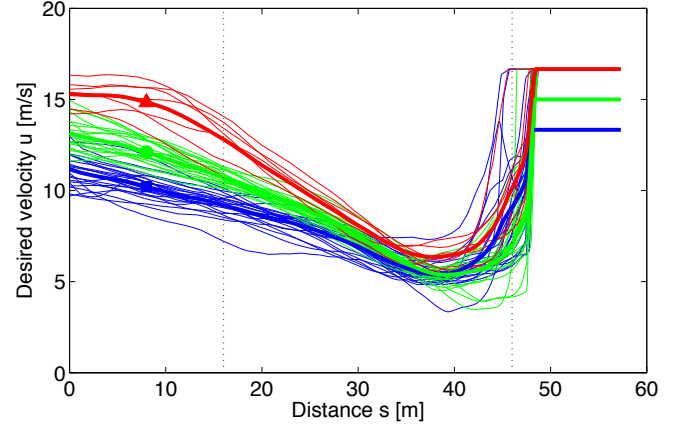


Fig. 7. Result of the k-means clustering algorithm. Cluster centers (thick lines) for Model 1 (blue square), Model 2 (green circle) and Model 3 (red triangle) and their associated desired velocity curves.

number of parameter sets covering a wide range of possible driver behaviors. In the following, two different approaches are investigated: One that creates desired velocity models based on observed driver behavior at this particular intersection, and one more general approach that calculates these models based on the curvature of the path lying ahead.

#### A. Clustering of desired velocity profiles

One way to obtain characteristic driver models is to use the *k-means* clustering algorithm to group  $N$  non-stop right turns without a preceding vehicle into  $K$  different models. In our experiments, we used  $N = 50$  and  $K = 3$  with results as shown in Figure 7. The cost function for the cluster assignment step has been chosen as

$$J_{ij} = \int_{0 \text{ m}}^{48 \text{ m}} (u_i(s) - c_j(s))^2 ds \quad (4)$$

where  $c_j(s)$  represents the  $j$ -th cluster center and  $u_i(s)$  the desired velocity profile of the  $i$ -th intersection crossing. For  $s > 48 \text{ m}$ , each cluster has been assigned a different value  $c_j(s) = \bar{u}_j$  so as to be consistent with the approach of the following section. Note that although the parameter  $a_i$  of the individual intersection crossing has been estimated for a different  $\bar{u}$ , the error in  $a_i$  is very small as the velocity during the acceleration period used for parameter estimation is considerably smaller than each  $\bar{u}_j$ .

The probability for an arbitrary crossing belonging to one of the three desired velocity models is obtained by counting the trajectories that have been assigned to each cluster center. In order to represent different driving styles, we also distinguish between three values of the maximum acceleration parameter  $a$ :  $1.5 \text{ m/s}^2$ ,  $2.0 \text{ m/s}^2$  and  $2.5 \text{ m/s}^2$ . The corresponding probability distributions given a specific desired velocity model  $M$  can again be obtained by counting. The results visualized in the upper row of Figure 8 show that on average, Model 1 is associated with higher values for the maximum acceleration parameter  $a$  than, judged by the driver's deceleration during the intersection approach, the more sporty Model 2. It seems that on this particular intersection, drivers tend to accelerate stronger after the turn if they approach the intersection more



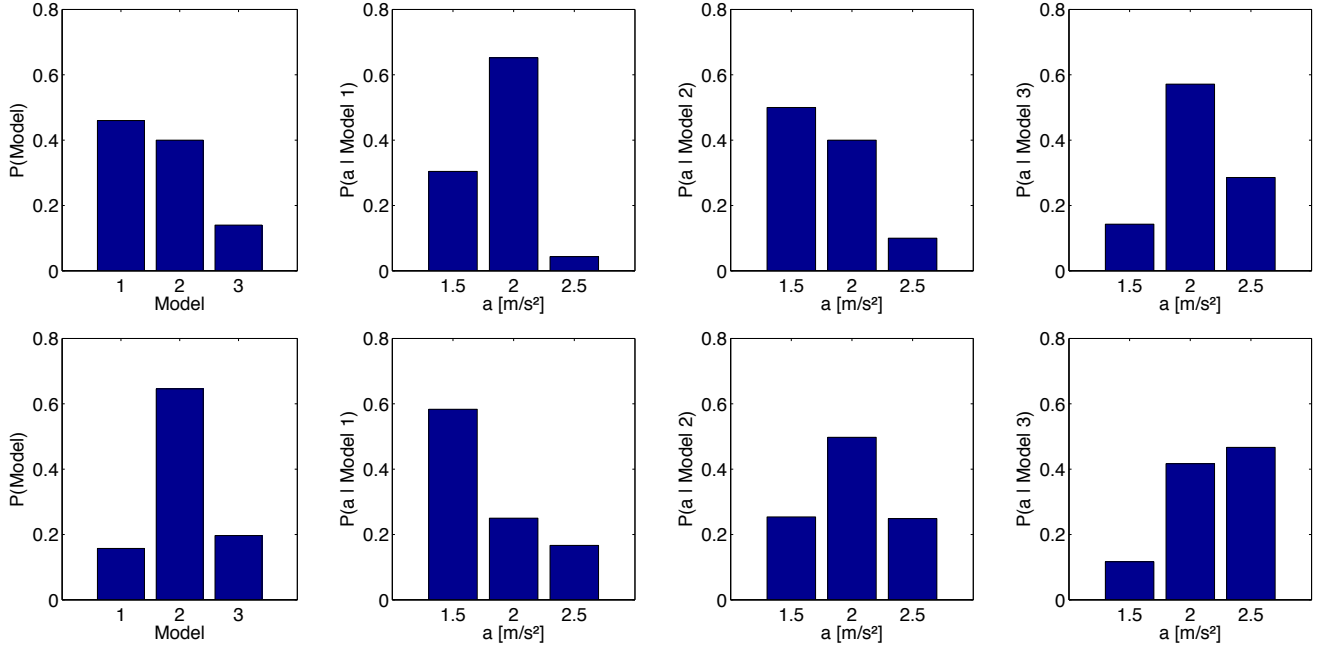


Fig. 8. Probability distributions for each model  $M$  and acceleration parameter  $a$  given that the parameterization was obtained by clustering (upper row) or from path curvature (lower row).

slowly. A higher number of desired velocity models might improve modeling in this point, but it would also increase the risk of overfitting and require more training data to obtain a meaningful distribution for the parameter  $a$ .

While the driver profiles above have been obtained from non-stop right turn maneuvers without a preceding vehicle, they have been shown to generalize well to arbitrary situations. In particular, preceding vehicles are taken into account by the brake term

$$\dot{v}_{\text{Brake}} = -a \left( \frac{d^*(v, \Delta v)}{d} \right)^2$$

of Equation (1), whereas stops can be modeled by placing a virtual, zero-velocity obstacle. For straight intersection crossings, we assume the same driver profiles as obtained for right turn maneuvers. The only difference is that there is no turn-related deceleration, so we have  $c_j(s) = \bar{u}_j$ .

### B. Obtaining desired velocity models from path curvature

In the previous section, the desired velocity models have been obtained from observed driver behavior at one particular intersection. For a practical application, this would require a database of typical driver behavior at each individual intersection. Although it is possible to create such a database using low-precision GPS onboard units, a less costly approach would be to generate the desired velocity models from information that is already contained in the map.

Based on circular arc splines, our map directly represents the path curvature that, as it turns out, is well suited to automatically generate desired velocity models. The process is illustrated in Figure 9. First, the path curvature  $k_1(s)$  is smoothed using a moving average filter. Roads are normally designed with continuous curvatures, so the result is a more realistic curvature profile that, to some degree, may also include

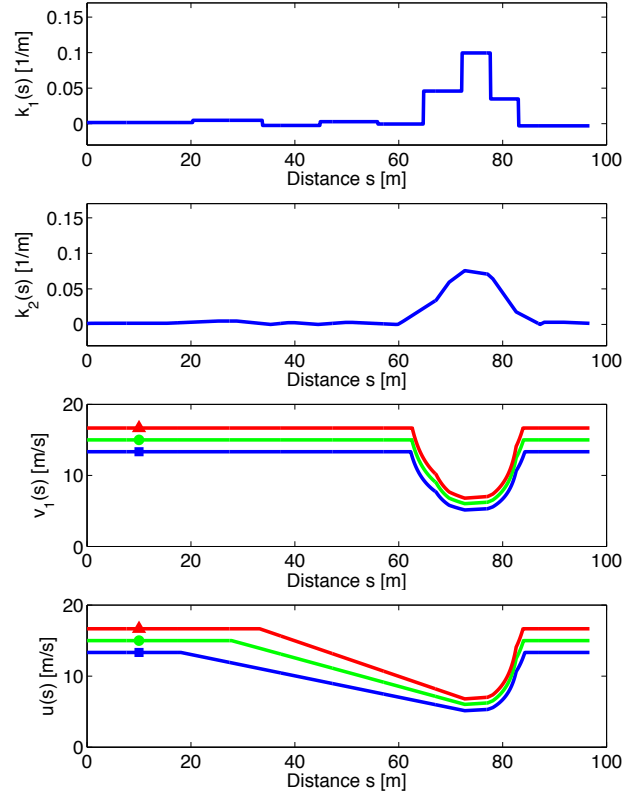


Fig. 9. Construction process for the desired velocity models based on path curvature, see text.

TABLE III  
PARAMETERS THE CURVATURE BASED APPROACH

	Model 1	Model 2	Model 3
lateral acceleration $a_{\text{lat}}$	2.00 m/s <sup>2</sup>	2.75 m/s <sup>2</sup>	3.50 m/s <sup>2</sup>
maximal velocity $\bar{v}$	48 km/h	54 km/h	60 km/h
minimal gradient $\underline{g}$	0.15 s <sup>-1</sup>	0.20 s <sup>-1</sup>	0.25 s <sup>-1</sup>

cutting corner behavior of the driver. Based on different values for the maximum lateral acceleration  $\bar{a}_{lat,j}$ , the smoothed curvature profile  $k_2(s)$  is then converted into velocities

$$v_j(s) = \min \left( \sqrt{\frac{\bar{a}_{lat,j}}{k_2(s)}}, \bar{v}_j \right), \quad (5)$$

with  $\bar{v}_j$  being the maximum velocity for each profile  $j$ . In a final step, the desired velocity gradient  $\frac{dv_j}{ds}$  is limited to different values of  $\underline{g}_j$  so as to model stronger and weaker deceleration before the turn.

The parameter values for  $\bar{a}_{lat,j}$ ,  $\bar{v}_j$  and  $\underline{g}_j$  are given in Table III. In order to parameterize the process shown in Figure 9, velocity profiles have been collected for a total of 12 intersections as shown in Figure 10. For each individual intersection crossing, the corresponding desired velocity model and acceleration parameter  $a_i$  have been determined. Then, the parameter values for  $a_{lat}$ ,  $\bar{v}$  and  $\underline{g}$  have been manually optimized to capture a broad range of actual driver behaviors, assuming that drivers who accept high lateral accelerations will also show a more sporty intersection approach and higher maximum desired velocities.

The resulting probability distributions for the desired velocity model and the maximum longitudinal acceleration parameter  $a$  are given in the lower row of Figure 8. Based on all 12 intersections, the distributions for  $a$  now also reflect the general driving style of their associated desired velocity model, ranging from rather defensive given Model 1 to sporty given Model 3.

#### IV. DRIVER INTENT INFERENCE

In the last section, we introduced the Intelligent Driver Model that allows us to model the longitudinal behavior of a driver for a given path or maneuver. Now, we would like to estimate the probability for each available maneuver based on how well the driver's current velocity profile fits the longitudinal behavior that we would expect if the driver was to perform that particular maneuver.

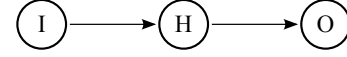
For the intersection shown in Figure 1, we defined four possible driver intents as shown in Table IV. For each intent, we would expect a different set of desired velocity profiles, and for each type of desired velocity profile a different probability distribution for the maximum acceleration parameter  $a$ . Stops are modeled using a virtual zero-velocity obstacle.

TABLE IV  
HYPOTHESES FOR DRIVER INTENT INFERENCE

Intention $I$	$H$ : Model $\times a$ [m/s <sup>2</sup> ]	Obstacle
$I_1$ : Go Straight	$\{1, 2, 3\} \times \{1.5, 2.0, 2.5\}$	–
$I_2$ : Stop at stop line	$\{1, 2, 3\} \times \{1.5, 2.0, 2.5\}$	16 m
$I_3$ : Turn right	$\{1, 2, 3\} \times \{1.5, 2.0, 2.5\}$	–
$I_4$ : Turn right but stop	$\{1, 2, 3\} \times \{1.5, 2.0, 2.5\}$	46 m

Each combination of driver intent and expected longitudinal behavior is represented by a separate hypothesis  $H$ . Taking into account four different driver intents, three desired velocity models and three values for  $a$  each, we have a total of 36 hypotheses.

The probability distribution for the actual driver intent  $I$ , the applicable hypothesis  $H$  and a not yet defined observation  $O$  can be modeled by a simple Bayes net:



The probability for a particular intent  $I_j$  given the observation  $O$  can thus be written as

$$P(I_j|O) = \sum_i P(I_j|H_i) P(H_i|O) \quad (6)$$

where  $P(I_j|H_i)$  is either 0 or 1 depending on  $H_i$ , i.e. each expected longitudinal behavior is uniquely assigned to a single intent. The probabilities for the individual hypotheses are given by

$$P(H_i|O) = \frac{P(O|H_i) P(H_i)}{\sum_j P(O|H_j) P(H_j)}. \quad (7)$$

The prior probabilities  $P(H_i)$  can be obtained from

$$P(H_i) = P(I_j) P(M_k) P(a_l|M_k) \quad (8)$$

where  $I_j$  is the intent,  $M_k$  the desired velocity model and  $a_l$  the maximum longitudinal acceleration parameter associated with hypothesis  $H_i$ . The distributions for  $P(M)$  and  $P(a|M)$  are given in Figure 8. The prior distribution of the intent  $I$  can be set heuristically or from statistical data. For our experiments, we assume a uniform distribution  $P(I_j) = 0.25 \forall j$ .

The remaining term  $P(O|H_i)$  represents the likelihood of our current observation given the longitudinal behavior of the driver could be modeled by the parameter set associated with hypothesis  $H_i$ .

##### A. Simulation-based approach

One possibility to estimate  $P(O|H_i)$  at time  $t$  is to simulate the driver behavior corresponding to hypothesis  $H_i$  for the time interval  $[t - T_S, t]$  with starting values

$$\hat{s}_i(t - T_S) = s(t - T_S) \quad \text{and} \quad \hat{v}_i(t - T_S) = v(t - T_S)$$

where  $\hat{s}$  and  $\hat{v}$  denote the simulated values for distance and velocity. For a typical right turn maneuver, the simulation results at two different times  $t$  are shown in Figure 11. There are no hypothesis curves for  $I_2$ , as the stop line of the traffic light has already been passed.

The velocity profiles of the remaining hypotheses terminate at different velocities  $\hat{v}_i(t)$  and distances  $\hat{s}_i(t)$ . These are the expected values given that the current driver could be modeled by the parameter set of the corresponding hypothesis. One possible way to define the observation  $O$  is to use only the final values of the simulation and compare them to the actual values  $v(t)$  and  $s(t)$ . Assuming normal distributed noise for both  $s(t)$  and  $v(t)$ , the probability density function  $f_{AS}$  for hypothesis  $H_i$  is given by

$$f_{AS}(a(t), s(t)|H_i) = \frac{1}{2\pi\sigma_s\sigma_v} \exp\left(-\frac{1}{2}e^2\right) \quad \text{with} \quad (9)$$

$$e = \sqrt{\left(\frac{s(t) - \hat{s}(t)}{\sigma_s}\right)^2 + \left(\frac{v(t) - \hat{v}(t)}{\sigma_v}\right)^2}. \quad (10)$$

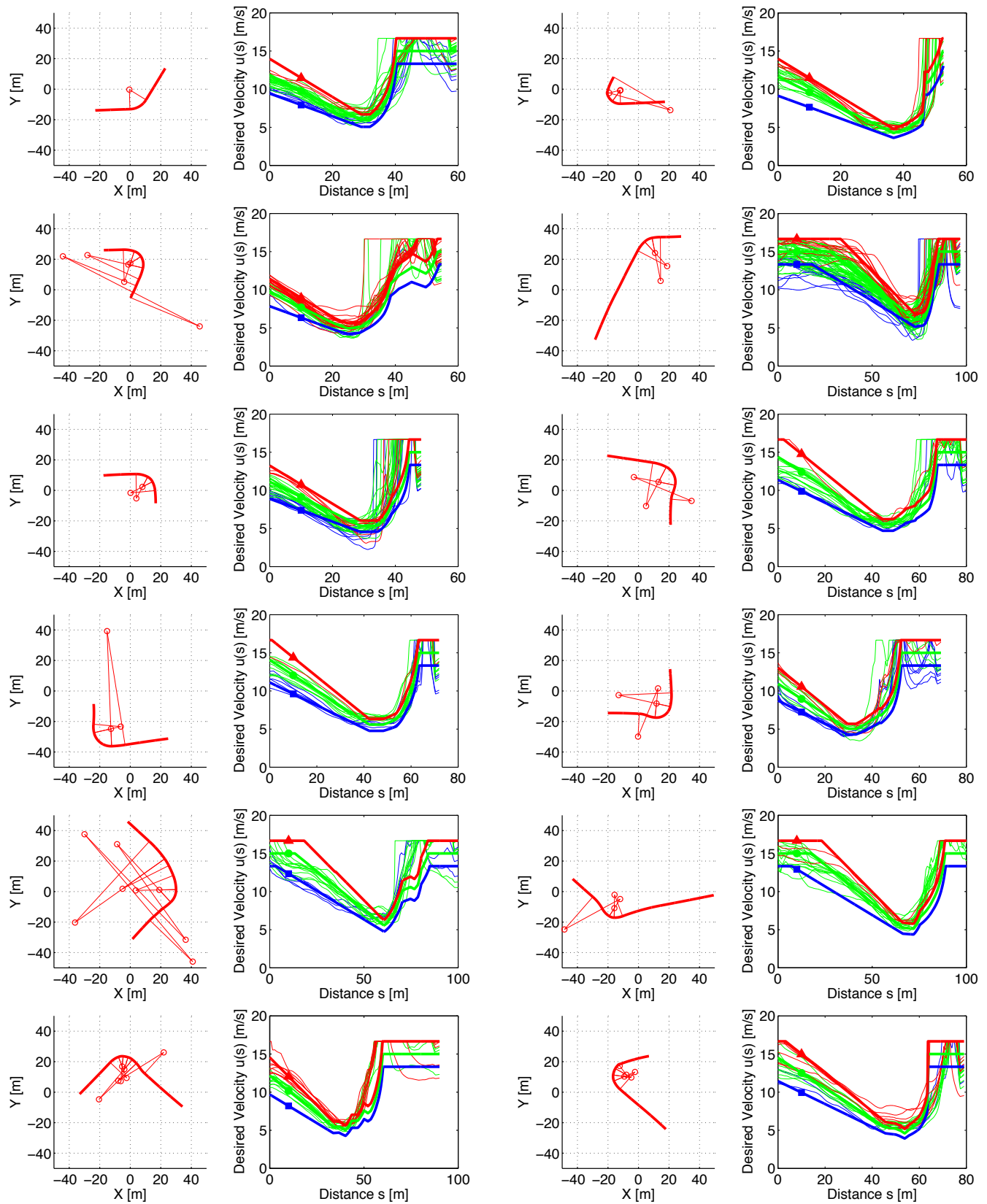


Fig. 10. Geometry and desired velocity profiles for 12 different intersections. Thin lines represent desired velocity profiles for individual intersection crossings, thick lines represent desired velocity profiles generated using path curvature.

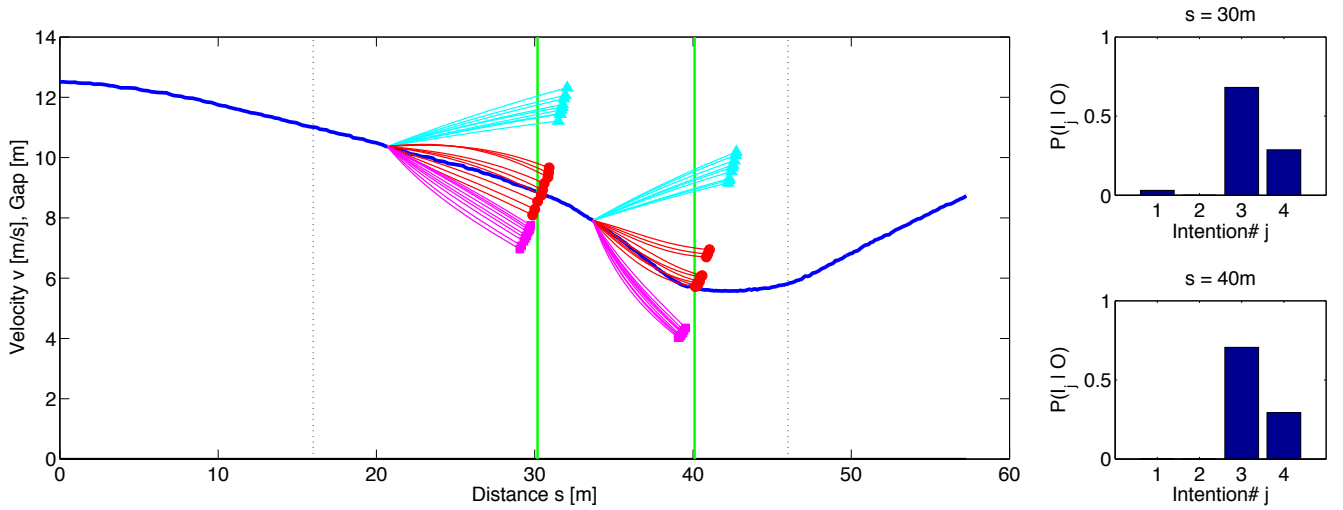


Fig. 11. Typical right turn maneuver in absence of a preceding vehicle. Simulation is carried out at  $s = 30$  m and  $s = 40$  m, starting 1 s in the past. The predicted trajectories are colored and marked by their corresponding driver intention: Cyan triangle for  $I_1$  "Go straight", red circle for  $I_3$  "Turn right" and magenta square for  $I_4$  "Turn but stop first". The resulting probabilities for the driver intents are shown on the right.

Using Bayes Theorem for probability density functions, the probability  $P(H_i|a(t))$  is then determined by

$$P(H_i|a(t), s(t)) = \frac{f_{AS}(a(t), s(t)|H_i) P(H_i)}{\sum_j f_{AS}(a(t), s(t)|H_j) P(H_j)} \quad (11)$$

which, basically, evaluates the likelihood of observation  $[v(t), s(t)]$  for  $H_i$  relative to all other hypotheses.

For empirical  $\sigma_s = 1.2$  m and  $\sigma_v = 1.2$  m/s, the posterior distribution of the driver intent is shown on the right in Figure 11. Using appropriate values for  $\sigma_s$  and  $\sigma_v$  is important as they have a major influence on how easily a hypothesis will be favored above others.

One advantage of using explicit models to infer the driver's intent is that arbitrary environmental conditions such as the presence of a leading vehicle can be easily taken into account. Figure 12 shows the simulation results for a straight intersection crossing which is influenced by a preceding vehicle that is slowing down to do a right turn. For  $s = 28$  m, all hypotheses  $H$  lead to similar results due to the car-following situation. Only after the leading vehicle has turned, the curves corresponding to intention  $I_1$  for going straight show a significantly different behavior which is reflected in the calculated probabilities  $P(I_j|O)$  as shown on the right in Figure 12. This demonstrates that the approach is well suited to express ignorance in situations for which the driver's intent cannot be determined by the velocity profile alone.

### B. Comparison-based approach

As an alternative to the simulation-based approach, the current driver intent can also be estimated by comparing the actual acceleration  $\dot{v}(t)$  with that calculated by the IDM equation for each individual hypothesis  $H_i$  directly. Similar to Equation (9), the probability density function  $f_{A(t_j)}$  for  $a(t_j)$  and Hypothesis  $H_i$  is given by

$$f_{A(t_j)}(a(t_j)|H_i) = \frac{1}{\sigma_a \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{a_j - \hat{a}_{ij}}{\sigma_a} \right)^2 \right]. \quad (12)$$

For robustness against errant detections of preceding vehicles and fluctuations in the longitudinal behavior of the driver, it is still necessary to take observations over the full time interval  $[t - T_S, t]$ . The individual observations are not independent given  $H_i$ , so a naive Bayesian approach would not model the underlying probability distribution correctly. Instead, we chose to average the individual scores from (12) to obtain

$$f_A(a|H_i) = \frac{1}{N} \sum_{j=1}^N f_{A(t_j)}(a(t_j)|H_i) \quad (13)$$

which is not a real probability density function anymore as

$$\int_{-\infty}^{\infty} f_A(a|H_i) da = \frac{1}{N} \sum_{j=1}^N \prod_{k \in \{1..N \setminus j\}} \int_{-\infty}^{\infty} da_k \quad (14)$$

but can still be used to calculate

$$P(H_i|a) = \frac{f_A(a|H_i) P(H_i)}{\sum_j f_A(a|H_j) P(H_j)} \quad (15)$$

because (14) is the same for each hypothesis  $H_i$ . Compared to the simulation-based approach, the comparison-based approach has two major advantages:

- It is more robust in the presence of sporadic false positive detections of a preceding vehicle directly in front of our own, as each time step influences the final score only by  $1/N$  whereas for the simulation based approach, all hypotheses will become unlikely as the expected driver behavior will be a full stop.
- It is much faster, because at each time step, only the current acceleration needs to be evaluated while all other scores can be reused.

One drawback, however, is that the comparison-based approach cannot average car-following behavior as good as the simulation based approach. If a driver is keeping an untypically large clearance to his preceding vehicle, the comparison-based approach will deduct scores for the acceleration mismatch throughout the whole interval whereas for



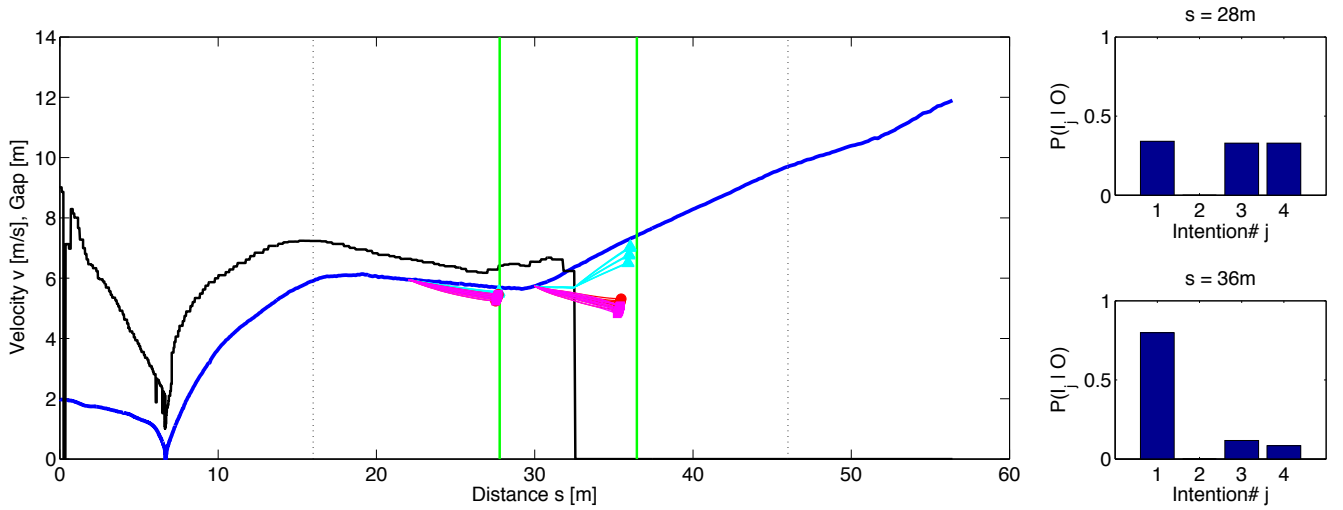


Fig. 12. Classification results for going straight in the presence of a leading vehicle. The gap  $d$  between the two vehicles is represented by the black line.

the simulation-based approach, the simulated driver will close the gap to the preceding vehicle but then assume the same velocity as the actual driver, so scores will be deducted for the mismatch of the distance variable  $s(t)$  at the end of the simulation only.

## V. EXPERIMENTAL RESULTS

Evaluations have been carried out for the intersection shown in Figure 1. The counts for the different cases contained in the dataset are given in Table V.

### A. TTC Estimation

As mentioned before, a possible application of driver intent inference at intersections is to warn the driver of cyclists moving parallel to the road if and only if he intends to do a right turn. Assuming a reaction time of up to 1.5s and a turning velocity of around 6 m/s,  $TTC = 2$  s would be an appropriate value for the time-to-collision threshold at which the corresponding driver assistance system issues the warning. However, estimating the remaining  $TTC$  is non-trivial since its conventional constant velocity definition does not hold due to the turn-related deceleration. Much better results have been obtained by simulating the driver behavior according to the desired velocity profiles of the hypotheses  $H$ . A comparison between the two approaches is shown in Figure 13. The desired velocity-based  $TTC$  estimation overestimates the remaining  $TTC$  by a maximum of 0.5 s, which still allows for a reaction time of 1 s. The underestimates are mainly due to slow preceding vehicles.

TABLE V  
CASES INCLUDED IN THE TEST SET (PV: PRECEDING VEHICLE)

	No PV	PV turns right	PV goes straight
Turning right	119	21	25
Going straight	33	28	19

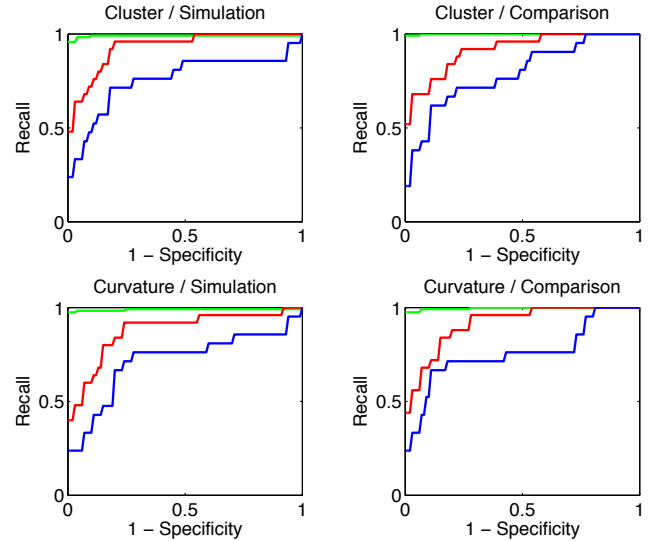


Fig. 14. Right turn prediction results for the dataset containing cases as in Table V. The best results are obtained for the case of no preceding vehicle, the worst for the case when the preceding vehicle is doing a right-turn itself.

### B. Classification Performance

Based on the dataset shown in Table V, the classification performance has been evaluated at a predicted  $TTC$  of 2 seconds. Parameter sets obtained either by clustering driver behavior at that particular intersection or by the more general path curvature approach have been evaluated in combination with each of the two classification methods. The results are shown in Figure 14.

Apparently, the cluster-based method obtains somewhat better results than the curvature-based method. This is not surprising, as the curvature-based approach has been optimized for all 12 intersections while the cluster-based approach is only suited for the one that is evaluated. Both simulation and comparison-based approach deliver equally good results.

As expected, the comparison-based approach is computationally much more efficient than the simulation-based approach, as it has to compute only one score at each time

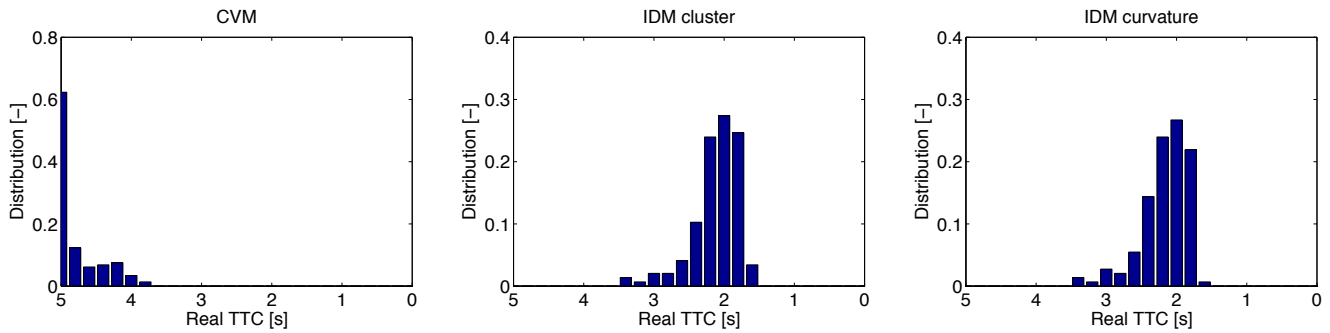


Fig. 13. Comparison between the constant velocity (CVM) and the desired velocity based  $TTC$  estimation. Real  $TTC$  refers to the time span between the first  $TTC \leq 2$  s estimate and the actual time at which the pedestrian crossing is reached. Values higher than 5 s are represented by  $TTC = 5$  s.

step whereas the simulation-based approach needs to run a simulation for the whole time interval. In MatLab 2010, evaluating the velocity profile for the past one second with a discretization of 0.01 s took 17 ms using the comparison-based and 207 ms using the simulation-based approach. As a C-coded MEX-function, the comparison-based approach took less than 1 ms.

## VI. CONCLUSIONS AND FUTURE WORK

In this paper, a simple, transparent and real-time capable approach to driver intent estimation for urban intersections has been presented. It is based on an explicit parametric model for the longitudinal velocity and considers a desired velocity that is approached in the absence of a preceding vehicle. Two different methods have been proposed to extract characteristic desired velocity profiles from real world data that allow the IDM to account for turn-related deceleration. For the example of a right turn assistance system, it has been shown that good classification performance can be achieved both with the simulation-based and the comparison-based classification method even in the presence of a preceding vehicle. If the driver intent cannot be inferred in a particular situation, the approach is able to express ignorance which is an advantage if combined with additional features like the lateral displacement, the driver's head pose or the turn indicator.

Ongoing work focuses on the development of a more sophisticated driver model. For situations in which the preceding vehicle is doing a right turn but has to wait at the pedestrian crossing, a typical misclassification cause with the current model is that some drivers will pass the preceding vehicle even though it is still blocking the straight lane to some degree, while others will not. Hence, this decision should be modeled explicitly. Also, in order to improve classification performance as well as to infer the driver's intent in situations in which this is not possible based on the velocity profile alone, we are working on a generic architecture that allows to infer the driver's intent by comparing expected and observed behavior for multiple features. Finally, another exciting field of research is posed by the need to take interactions with other traffic participants into account. Detecting driver's gaze direction might help to infer both his intent and his cognitive model of the environment, based on which he might or might not change his planned trajectory to allow for a safe intersection crossing.

## ACKNOWLEDGEMENTS

This work was funded in part by the Federal Ministry of Economics and Technology of the Federal Republic of Germany under grant no. 19 S 9022.

## REFERENCES

- [1] "GIDAS - German In-Depth Accident Study," 2011.
- [2] "Ko-FAS research initiative." [Online]. Available: <http://www.kofas.de>
- [3] D. Schwarz, H. Kloeden, and R. Raschofer, "Ko-TAG - Cooperative Sensor Technology for Traffic Safety Applications," in *8th International Workshop on Intelligent Transportation*, 2011.
- [4] D. Weimer, S. Kohler, C. Hellert, K. Doll, U. Brunsmann, and R. Krzikalla, "GPU architecture for stationary multisensor pedestrian detection at smart intersections," in *IEEE Intelligent Vehicles Symposium*, 2011, pp. 89–94.
- [5] R. Wertheimer and F. Klanner, "Cooperative Perception to Promote Driver Assistance and Preventive Safety," in *8th International Workshop on Intelligent Transportation*, 2011.
- [6] A. Rauch, F. Klanner, F. Mueller-Witt, and K. Dietmayer, "Analysis of V2X Communication Parameters for the Development of a Fusion Architecture for Cooperative Perception Systems," in *IEEE Intelligent Vehicles Symposium*, 2011.
- [7] H. Kloeden, D. Schwarz, E. Biebl, and R. Raschofer, "Vehicle localization using cooperative RF-based landmarks," in *IEEE Intelligent Vehicles Symposium*, 2011, pp. 387–392.
- [8] S. Lefevre and C. Laugier, "Exploiting Map Information for Driver Intention Estimation at Road Intersections," in *IEEE Intelligent Vehicles Symposium*, 2011, pp. 583–588.
- [9] S. Danielsson, L. Petersson, and A. Eidehall, "Monte Carlo based Threat Assessment: Analysis and Improvements," *IEEE Intelligent Vehicles Symposium*, pp. 233–238, June 2007.
- [10] H. Berndt and K. Dietmayer, "Driver intention inference with vehicle onboard sensors," in *Vehicular Electronics and Safety Conference*, 2009, pp. 102–107.
- [11] C. Tay, "Analysis of dynamic scenes: Application to driving assistance," Dissertation, L'Institut polytechnique de Grenoble, 2009.
- [12] J. Firl, H. Stübing, S. Huss, and C. Stiller, "Predictive maneuver evaluation for enhancement of car-to-x mobility data," in *IEEE Intelligent Vehicles Symposium*, Alcalá de Henares, Spain, June 2012, pp. 558–564.
- [13] G. S. Aoude, V. R. Desharaju, L. H. Stephens, and J. P. How, "Behavior Classification Algorithms at Intersections and Validation using Naturalistic Data," in *IEEE Intelligent Vehicles Symposium*, 2011, pp. 601–606.
- [14] C. Hermes, C. Wöhler, K. Schenk, and F. Kummert, "Long-term vehicle motion prediction," in *IEEE Intelligent Vehicles Symposium*, 2009, pp. 652–657.
- [15] Auto Club Europa, "Reviere der Blinkmuffel," 2008.
- [16] F. Klanner, "Entwicklung eines kommunikationsbasierten Querverkehrsassistenten im Fahrzeug," Dissertation, Technische Universität Darmstadt, 2008.
- [17] K. Lidström and T. Larsson, "Model-based Estimation of Driver Intentions Using Particle Filtering," in *IEEE Intelligent Transportation Systems Conference*, 2008, pp. 1177–1182.
- [18] P. G. Gipps, "A behavioural car-following model for computer simulation," *Transp. Res.-B.*, no. 15, pp. 105–111, 1981.

- [19] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E* 62, 2000.
- [20] M. Liebner, M. Baumann, F. Klanner, and C. Stiller, "Driver intent inference at urban intersections using the intelligent driver model," in *IEEE Intelligent Vehicles Symposium*, Alcalá de Henares, Spain, June 2012, pp. 1162–1167, (best paper award).
- [21] G. Maier, "Smooth Minimum Arc Paths - Contour Approximation with Smooth Arc Splines." Dissertation, Universität Passau, 2010.



**Martin Liebner** received the M.Sc. degree in mechatronics from Technische Universität Dresden, Dresden, Germany, in 2010.

He is currently working towards his Ph.D. degree at BMW Group Research and Technology. His research interests include driver intent inference, situation analysis and risk assessment for driver assistance systems.



**Felix Klanner** received the M.Sc. degree from Technische Universität München, Munich, Germany, in 2004, and the Ph.D. from Technische Universität Darmstadt, Darmstadt, Germany, in 2008.

Since 2007, he is with BMW Group Research and Technology. He has contributed to the German research initiative AKTIV and the integrated European research project PReVENT, and is project leader for the German research project Ko-PER. His research interests include cooperative perception systems and driver assistance systems for collision avoidance.



**Michael Baumann** received the B.Eng. degree in electrical engineering from Munich University of Applied Sciences, Munich, Germany, in 2012, and is currently working towards his M.Sc. degree.

As a student assistant with BMW Group Research and Technology, his research interests focus on machine learning in the context of advanced driver assistance systems.



**Christian Ruhhammer** received the M.Sc. degree in automotive software engineering from Technische Universität München, Munich, Germany, in 2012.

He is currently working towards his Ph.D. degree at BMW Group Research and Technology. His research focuses on the analysis and prediction of driver behavior and the extraction of knowledge from big data.



**Christoph Stiller** (S'93–M'95–SM'99) received the Diploma and Dr.-Ing. degree from Aachen University of Technology, Aachen, Germany, in 1988 and 1994, respectively. Afterwards, he was employed by INRS-Telecommunications, Montreal, QC, Canada, as a postdoctoral member of the scientific staff.

In 1995, he joined the Corporate Research and Advanced Development of Robert Bosch GmbH, Hildesheim, Germany, where he was responsible for computer vision for automotive applications. In 2001, he became chaired professor and director of the Institute for Measurement and Control Systems at Karlsruhe Institute of Technology, Karlsruhe, Germany. His current research interests include cognition of mobile systems, computer vision, visual inspection, and real-time applications thereof.

Dr. Stiller serves as President of the IEEE Intelligent Transportation Systems Society (2012-2013) and has been the Vice President for Publications (2009-2010) and for Member Activities (2006-2008). He was the Editor-in-Chief of the IEEE INTELLIGENT TRANSPORTATION SYSTEM MAGAZINE (2009-2011) and Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING (1999-2003) and the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (since 2004).