Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Using Context Information and Probabilistic Classification for Making Extended Long-Term Trajectory Predictions

Stefan Klingelschmitt, Julian Eggert

2015

Preprint:

This is an accepted article published in 18th IEEE International Conference on Intelligent Transportation Systems 2015 (ITSC). The final authenticated version is available online at: https://doi.org/[DOI not available]

2015 IEEE 18th International Conference on Intelligent Transportation Systems

Using Context Information and Probabilistic Classification for Making Extended Long-Term Trajectory Predictions

Stefan Klingelschmitt[1] and Julian Eggert[2]

Abstract— **Intersections are among the most accident prone spots in traffic. Future Advanced Driver Assistance Systems (ADAS) are aiming to assist the driving task in these complex scenarios. This can be realized by assessing the criticality of possible occurring situations. For such criticality assessment techniques predicting the trajectories of the involved traffic participants several seconds in advance is necessary. In this paper we outline a method that makes exhaustive use of context information to reliably predict maneuver-specific trajectories up to 5 seconds into the future. Since the evolution of traffic scenes cannot be predicted with absolute certainty, approximating future states in form of probability density functions will be of great benefit in terms of robustness and reliability. We present an approach that is able to efficiently construct a discrete probability distribution by reformulating the problem as a probabilistic multiclass classification problem. The presented approach is evaluated on a real-world data set containing approaches to 85 different intersections. We show that we can make reliable maneuver-specific state estimations, even for a prediction horizon of up to 5 seconds.**

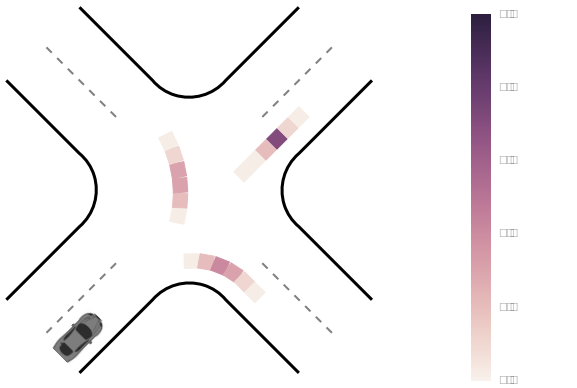


Fig. 1: Car approaching an intersection with three maneuver alternatives. The maneuver-specific position predictions in the form of a discrete probability distribution are highlighted.

I. INTRODUCTION

Assistance systems supporting the driving task in inner-city scenarios increasingly become a focus of attention for the scientific community as well as for car manufacturers. Especially assistance systems targeted for safely passing intersections are of great interest, as such systems will most likely reduce the number of accidents occurring in urban environments [1].

Elaborate criticality assessment techniques, incorporating surrounding traffic participants, can be used to determine the threat for the ego-vehicle planning to enter an intersection, as presented in [2], [3]. However, in order to assess the criticality of possible future events, long-term trajectory predictions are needed. Trajectories for different maneuver alternatives per traffic participant need to be considered which aggravates the problem. In general, the following challenges need to be considered:

1) real-time capabilities,
2) sufficiently long prediction horizon,
3) uncertain state estimations.

In regard to the **first** challenge, the computational power in currently available automobiles is still limited and will be kept as small as possible due to power consumption and production cost limitations. The problem of a sufficient prediction horizon is addressed by the **second** point. In order for assistance systems to reliably assist the driving task with a safe reaction, either performed by a human or an automated vehicle, the prediction horizon needs to be correspondingly long. The **third** point addresses the problem of dealing with uncertainties originating from both, the sensor measurements and the uncertain evolution of the assessed situation. Kinematic information obtained from sensor measurements are usually noisy and are afflicted with errors. In addition, predicting absolute kinematic states of traffic participants is prone to be incorrect as the evolution of a complete trajectory cannot be foreseen with certainty. One could imagine a repeatedly identical maneuver, e.g. the same driver approaching the same intersection performing the same maneuver with identically modeled initial conditions, however the recorded trajectories will most likely not be identical. This is due to the fact that traffic scenes do not evolve deterministically, from a subjective, observer-based point of view, for several reasons like unobservable human factors (e.g. detailed motion plan, skill level, inattentiveness, ...) and/or not modeled effects regarding context information (weather, daytime, ...). It is therefore essential to have the ability to quantify the uncertainties regarding the future predicted course of a trajectory.

A. Problem Statement

In general the three mentioned challenges are difficult to tackle simultaneously and the following current issues need to be solved. Long-term motion predictions are often constructed by recursively predicting over short horizons. The effort needed for constructing these sequential predictions is

[1]S. Klingelschmitt is with Control Methods and Robotics Lab, Technical University of Darmstadt, D-64283 Darmstadt, Germany stefan.klingelschmitt@rmr.tu-darmstadt.de

[2]J. Eggert is with Honda Research Institute Europe GmbH, D-63073 Offenbach am Main, Germany jeggert@honda-ri.de

978-1-4673-6596-3/15 $31.00 © 2015 IEEE
DOI 10.1109/ITSC.2015.120

IEEE computer society

concomitantly high, particularly when combined with probability density estimates. Additionally, the prediction errors made for each prediction interval get accumulated. Directly estimating the states of interest at the desired prediction horizon might be beneficial in terms of needed computational resources and prediction accuracy.
Long-term maneuver prediction is usually associated with prediction horizons between 1 and 3 seconds. However, longer prediction horizons might be beneficial in terms of more foresighted reactions, especially in the context of automated vehicles. E.g. assuming a typical inner-city intersection approach with $13\frac{m}{s}$ and a comfortable deceleration of $-3\frac{m}{s^2}$ leads to a comfortable reaction time of $\sim 4.3s$. As a consequence the trajectory prediction horizons need to be at least in similar ranges.
The uncertainties evolving from inaccurate and noisy measurements can be dealt with e.g. stochastic filters and/or elaborate tracking methods. Quantifying uncertainties resulting from the uncertain evolution of traffic scenes is an additional challenge. For the case of model-based approaches, this results in sampling over large input spaces. In cases where the prediction results from a learned state regressor, quantifying the uncertainties is not trivial.

To overcome the addressed problems we propose a system that is able to make maneuver-specific long-term motion predictions (up to $5s$), providing a probability density for the target variable, while remaining computationally efficient. The contribution of this paper consists of two main parts. **Firstly**, we outline how adding more and more context information, like intersection crossing angles, priority rules, etc. can significantly extend prediction horizons. **Secondly**, we propose a system for efficiently approximating conditional probability densities for target variables. By discretizing the target space and treating the problem as a probabilistic multiclass classification problem, we obtain a discrete probability distribution over the target space. Fig. 1 shows exemplary discrete probability distribution estimations for available maneuver alternatives.

The reminder of this paper is organized as follows. In Section II we discuss related work. Section III formally introduces our proposed framework for extended long-term trajectory prediction. A detailed quantitative real-world evaluation is shown in Section IV. In Section V we summarize our results and give a brief outlook.

II. RELATED WORK

Literature addressing long-term inner-city trajectory prediction can be divided into two main groups: approaches trying to solve the problem by acquiring techniques from the field of machine learning and model-based approaches.

The **first** group is concerned with building prediction systems by learning from existing data or relying on motion databases.
An early publication on long-term vehicle maneuver prediction is proposed by the authors of [4]. They compare the history of the current trajectory with a reference motion database in order to assess the similarity between driven trajectories. A subsequent particle filter is used to generate motion hypotheses utilizing the probability determined by the preceding trajectory comparison. At last the motion prediction is obtained by clustering the motion hypotheses and choosing the cluster center with the highest likelihood. In a subsequent work [5] the authors extend their system by two additional prediction models, targeted at newly detected objects for which the motion history is unknown. The first model makes a prediction of the future states based on optical flow; the second model uses a kinematic extrapolation technique assuming constant acceleration (CA). In [6] a probabilistic search framework is adapted in order to speed up the search in motion databases.
The authors of [7] infer a probability distribution for the future motion model based on the history of the current trajectory. The probability distribution is obtained by a Gaussian Mixture Model (GMM), respectively Variational Gaussian Mixture Model (VGMM), trained on a recorded motion database. In [8] they extend their framework by incorporating context information in order to make more accurate long-term predictions.
The discussed approaches rely on complex features used for the prediction as well as for the comparison of the trajectory histories.

The **second group** is concerned with parametrized models capturing the vehicle dynamics and sequentially constructing trajectory predictions making assumptions on the evolution of the system's input variables, e.g. acceleration or steering angle.
In [9] a hierarchical system is introduced to tackle the shortcomings of constant input assumptions of any kind. They divide the turning process into three phases. Each phase has its own input assumption, e.g. CTAAV (constant turn acceleration and velocity acceleration), CTRA (constant turn rate acceleration) and CA. The switching logic is determined by hand. It relies, among other things, on using map information in form of the particular road crossing angle. The authors of [10] propose a system consisting of two prediction models. The first model is a kinematic extrapolation using the input assumptions CTRA, targeted at shorter prediction horizons. The second model generates possible future trajectories based on the upcoming road topology, associated with different maneuvers (e.g. keep lane, lane change, etc.) and chooses a generated trajectory based on a hand-designed cost function. They, however maintain the constant acceleration assumption for their long-term trajectory generation. Thus, it can be expected that the system can not cope with maneuvers having strongly varying acceleration profiles, e.g. turning at an intersection. An elaborate trajectory prediction approach based on stochastic filters and heavily used map information is presented in [11]. The motion hypotheses generation is obtained by using an extended Kalman filter with a bicycle model as vehicle dynamics. The control inputs in the form of longitudinal acceleration and steering angle are determined using digital map data. The longitudinal acceleration is calculated under the assumption that a driver tries to keep

its lateral acceleration at a comfortable level. However, when considering unsignalized inner-city traffic scenarios, driving at a maximum comfortable velocity might not be the most important concern of a driver. A similar approach for determining prototypical velocity profiles based on lateral acceleration is proposed in [12].
The assumptions made by the discussed approaches usually restrict prediction horizons as it gets more and more complicated to infer correct parametrized control input models in complex inner-city environments.

III. Long-term Probabilistic Trajectory Prediction Framework

For assessing the criticality of intersection-related traffic scenes, making a prediction of future states of involved traffic participants is indispensable. The sooner a reliable assessment of the current scene can be performed the better. Therefore, the system we introduce makes heavy usage of context information like digital map information and traffic rules. Additionally, we focus on making *maneuver-specific* trajectory predictions, meaning that we are making predictions for one maneuver option (e. g. turning right, going straight, etc.) at a time. If multiple competing maneuver alternatives are possible maneuver estimation systems are needed, which are not in the scope of this work. In previous works, we introduced a systems for estimating upcoming maneuvers ([13]) and handling ambiguity associated with such systems ([14]). However, combining longterm trajectory predictions of multiple, probable maneuver options on prediction level might not be very reasonable and should be considered separately by subsequent systems (e.g. risk assessment or behavior planning, like proposed in [15]).
Furthermore, when treating the problem probabilistically, elaborate criticality assessment techniques can strongly benefit from this information. For this purpose, we introduce the probability distribution

$$p(\mathbf{y}|\mathbf{x}, \tau), \tag{1}$$

where $\mathbf{y}$ represents the predicted target states vector of interest, $\mathbf{x}$ the current initial scene information, further discussed in III-C, and τ the prediction horizon.

A. Trajectory Definition

In general, trajectories are defined as sets of tuples containing the temporal and spatial positions (x, y) in the cartesian 2D plane. In order to obtain general, normalized trajectory prediction, we transform the absolute (x, y)-positions into an intersection- and prototypical-path-dependent coordinate system. The prototypical path can be generated by calculating centerlines using successive waypoints (ν_i), like presented in [11]. The origin of the coordinate frame is set to waypoint ν_i, corresponding to the intersection entry. A constructed prototypical path is exemplarily shown in Fig. 2. The spatial position features are transformed into the longitudinal component s and the lateral deviation n, using perpendicular projection of the cartesian (x, y)-coordinates onto the prototypical path. By using this transformation

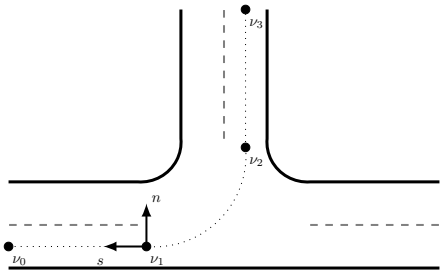


Fig. 2: Example prototypical path for a left turn maneuver. The prototypical path (dotted line) is calculated using the waypoints $\nu_0, ..., \nu_3$. The origin of the intersection- and path-dependent coordinate system (s, n) is set to the intersection entry waypoint ν_1.

our trajectory representation becomes invariant to differing starting and end points, as well as varying intersection layouts. Each trajectory $\mathcal{T}_j$ consists of N-tuple representing the temporal information t_q and spatial information (s_q, n_q) with $q = 1, ..., N_j$, where N_j corresponds to the samples of this particular trajectory $\mathcal{T}_j$. The trajectory representation is finally given by

$$\mathcal{T}_j = \{((s_q, n_q), t_q)_{q=1,...,N_j}\}. \tag{2}$$

Corresponding additional information, like velocities, yaw rates or accelerations can be easily calculated, if needed.

B. Probability Density Estimation

Quantifying the prediction uncertainty of estimated future states is important as the evolution of traffic scenes can in general not be foreseen with certainty. Therefore, we estimate the conditional probability density of (1).
Treating the estimation of future states under a machine learning perspective corresponds to a regression problem. However, regression methods simultaneously quantifying their prediction quality are rare and rather complex. Gaussian Processes with variational heteroscedastic variance adaptation [16] are able to estimate the prediction uncertainties. Similar results can be obtained using Neural Networks with weights learned by a fully Bayesian treatment [17]. However, these approaches are not well suited for the application in intelligent vehicles due to their high complexity and computational requirements.

Therefore, we propose a computationally cheap probabilistic trajectory prediction, based on quantifying the target space $\mathbf{y}$, similar to [18], into K bins $\omega \in \Omega$, with $\Omega = \{\omega_1, ..., \omega_K\}$ as sample space of possible bins. We define a random variable

$$\mathbf{Y}(\omega) = \Gamma(\omega) = \hat{\mathbf{y}}. \tag{3}$$

Γ is a bijective transformation mapping each bin to its discrete target space centers $\hat{\mathbf{y}}$. The discretization can be from arbitrary granularity. For each dimension b of the target space $\mathbf{y}$, the number of chosen bins is specified by K_b. The complete number of bins is given by $K = \prod_{b=1}^{dim(\mathbf{y})} K_b$.

The continuous probability density function (1) can be approximated using the discrete probability distribution

$$p(\{\omega \in \Omega : Y(\omega) = \hat{\mathbf{y}}\}|\mathbf{x}, \tau) \equiv p(\hat{\mathbf{y}}|\mathbf{x}, \tau). \tag{4}$$

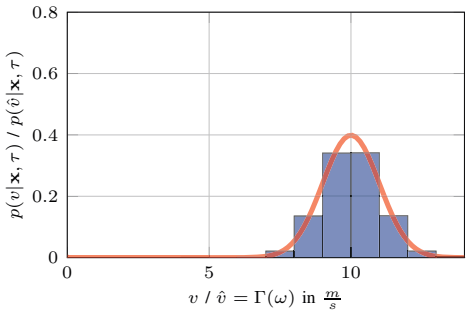


Fig. 3: The red line indicates an exemplary continuous distribution for an estimated target variable, the velocity v in the range of $[0, 14]\frac{m}{s}$. The blue histogram indicates the discrete probability distribution, using a $1\frac{m}{s}$ discretization resulting in $k = 1, ..., 14$ bins ω and corresponding discretized target space centers $\hat{v} = \Gamma(\omega)$.

A comparison of a continuous and discrete estimated target variable can be seen in Fig. 3.

Estimating this discrete probability distribution is rather simple and can be performed by any probabilistic multiclass classification method. Additionally, we are not forced to make assumptions on the distribution of the target variable in advance, as the obtained discrete distribution is nonparametric. Probabilistic classification models approximate the conditional class probabilities $p(\mathcal{C}_k|\mathbf{x})$, for k discrete classes $\mathcal{C}_k$, where $k = 1, ..., K$. Assuming that class $\mathcal{C}_k$ corresponds to a bin ω, these methods can be used to approximate (4).

For this problem well studied classification methods can be used, e.g. Neural Networks or Random Forest Classifiers. A more complete overview can be found in [19]. In Section IV we evaluate an example system using a Neural Network. Based on the obtained discrete probability distribution a continuous probability density function could be fitted, like presented in [20], where a similar approach for estimating conditional probability densities is outlined. However, we omitted this step, as an adequate descritization provides a sufficient accuracy. Moreover, using the proposed discrete distribution for criticality assessments can be computationally beneficial, e.g. for calculating common occupied spatial areas by different traffic participants, marginalizing over crash probabilities for a given time period, etc.

C. Feature Space Definition

In order to estimate (4) the evidence $\mathbf{x}$ and τ needs to be further defined and discussed. The random variable τ is used as the prediction horizon, it can also be interpreted as a control variable. For the case of motion prediction τ would usually correspond the the temporal prediction horizon t. However, cases where the future kinematic states of vehicles at a given *spatial location* s is of interest can also be handled.

As stated, extended long-term motion predictions heavily rely on the usage of context information. This is incorporated into the evidence feature vector $\mathbf{x}$. We separate the used features into three groups: (i) kinematic features $\mathbf{x}_{kin}$, representing the vehicle's current state, (ii) kinematic-specific context information $\mathbf{x}_{cx}$ and (iii) purely context related information $\mathbf{c}$. Equation (4) then becomes

$$p(\hat{\mathbf{y}}|\mathbf{x}_{kin}, \mathbf{x}_{cx}, \mathbf{c}, \tau).$$

The kinematic information $\mathbf{x}_{kin} = \begin{pmatrix} v & a \end{pmatrix}$ used for our model is composed of the vehicle's longitudinal velocity v and acceleration a. The features used for the kinematic-specific-context information are $\mathbf{x}_{cx} = \begin{pmatrix} s & d_{ex,l} \end{pmatrix}$, where s corresponds to the longitudinal distance to the intersection entry defined in III-A, and the longitudinal distance to the intersection exit waypoint $d_{ex,l}$, l indicating a specific maneuver path. This implicitly incorporates the information on the length of the path within the intersection, when executing maneuver $\mathcal{M}_l$. Purely contextual information is incorporated by $\mathbf{c} = \begin{pmatrix} \phi_l & r_l \end{pmatrix}$. The intersection crossing angle is given by ϕ_l. It measures the angle between the center line of the vehicle's current lane and the lane the vehicle is exiting the intersection, if performing a specific maneuver l. For possibly competing maneuver options, maneuver-specific predictions can be obtained by adapting the features $\mathbf{c}$ and $\mathbf{x}_{cx}$. The crossing angle provides indispensable information on the kind of maneuver and the corresponding future path layout. By incorporating ϕ_l, we do not need to learn maneuver-dependent regression models. r_l is a binary variable that indicates whether the vehicle has right of way, assuming maneuver l.

D. Training Samples Generation

We strive for a system that can make predictions for every initial state $\mathbf{x}$, which means that we are not limited to make predictions only at a certain position s before an intersection, and a fixed prediction horizon τ. Therefore, our classifier to be trained needs to generalize well and be flexible. To meet these requirements we make extensive use of the available trajectories $\mathcal{T}_j$, $j = 1, ..., J$, J is the number of recorded training trajectories, to create our training data set $\mathcal{D}$.
Starting from each sample q, $m = q, ..., N_j$ training cases are created, for the remaining samples in $\mathcal{T}_j$. Each training case contains the target bin ω_m (serving as class label) obtained by discretizing the target vector to $\hat{\mathbf{y}}_m$ and transforming it to the bins

$$\omega_m = \Gamma^{-1}(\hat{\mathbf{y}}_m), \tag{5}$$

as well as the extracted feature vector $\mathbf{x}_q$ and the corresponding relative prediction horizons $\tau_m = t_m - t_q$. This is repeated for each sample $q = 1, ..., N_j - 1$ in trajectory j. The obtained data set from one trajectory is given by

$$\mathcal{D}_j = \{((\omega_m, \begin{pmatrix} \mathbf{x}_q & \tau_m \end{pmatrix})_{m=q,...,N_j})_{q=1,...,N_j-1}\}.$$

In other words, in order to be invariant to varying initial feature vectors $\mathbf{x}_q$ and differing prediction horizons τ a training sub-trajectory is created starting at each sample q reaching till the end of trajectory j, with relative prediction horizons $\tau_m = t_m - t_q$. Hence, $\mathcal{D}_j$ contains $\sum_{q=1}^{N_j-1}(N_j - q)$ training samples.
The complete data set, containing all trajectories J, is given by

$$\mathcal{D} = \{(\mathcal{D}_j)_{j=1,...,J}\}.$$

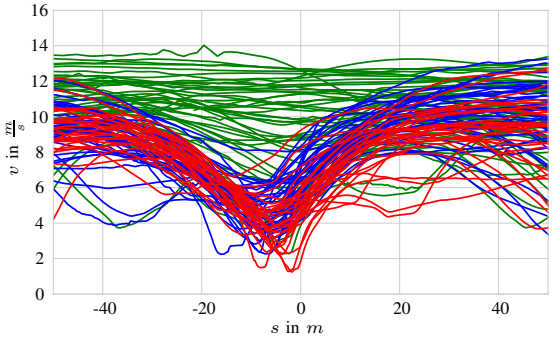


Fig. 4: Velocity profiles for all 112 intersection approaches. The green lines indicate going straight maneuvers, the red lines right turn maneuver and the blue lines left turn maneuvers.

At this point we would like to emphasize that by training our model with the obtained data set we can directly estimate the probability distribution $p(\hat{\mathbf{y}}|\mathbf{x}, \tau)$ (e.g. shown by the histogram in Fig. 3) of future states at concrete prediction horizons τ, without the need for recursively constructing the complete trajectory prediction up to τ. [1]

IV. EVALUATION

The proposed extended long-term trajectory prediction is evaluated using real-world data. The data set consists of recordings performed with two different test vehicles and several different drivers, altogether approaching 85 different intersections. We show that our proposed approach performs well on this highly heterogeneous recorded data which emphasizes its ability for generalization.

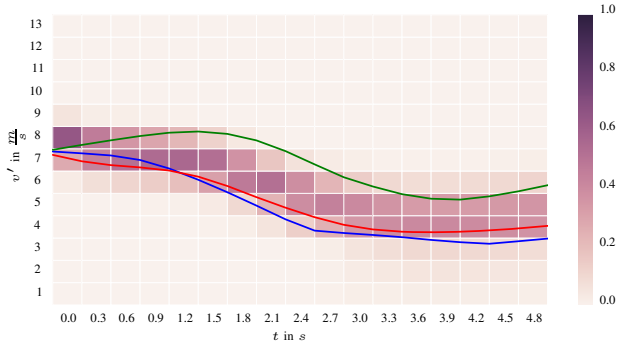
A. Data Acquisition

The test data is obtained from two different sources. One part of the data set is obtained from using the raw data from the KITTI Vision Benchmark [21]. 84 intersection approaches in residential areas are extracted. The second part of 28 intersection approaches was generated by two drivers using a test car equipped with a consumer-grade GPS. Information about the vehicle's velocity is obtained by tapping the CAN-Bus. Acceleration information is filtered using a simple moving average filter. To provide a consistent sampling rate, the data is resampled at $10Hz$. To have the ability of a subsequent qualitative analysis of the encountered scenarios and the possibility to annotate missing information, the test drives were recorded with a stereo-camera.
As a result we obtain 112 trajectories $\mathcal{T}_j$ approaching 85 different, unsignalized intersections with no inferences by other traffic participants. The data set contains 48 straight, 37 right turn and 27 left turn maneuvers. The velocity profiles for all recorded trajectories can be seen in Fig. 4.

Waypoints ν_i for generating the path information are labeled by hand using satellite images of the particular crossings. Intersection entry, intersection exit as well as road crossing angles can be obtained from the waypoint information. Priority information is inferred from available maneuver options or present road signposts.

[1]Complete temporal courses of future states, up to τ, can be predicted by estimating probability distributions $\{p(\hat{\mathbf{y}}|\mathbf{x}, \tau')_{\tau' \in [0,\dots,\tau]}\}$, like shown in Fig. 5.

(a)



(b)

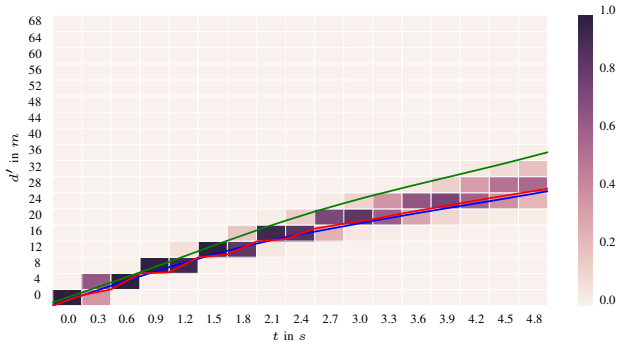


Fig. 5: Visualization of the estimated discrete probability distributions for a right turn trajectory, using model M. The prediction starts $s_q = 20m$ before entering the intersection and is computed every $0.3s$, up to a maximal prediction horizon of $4.8s$. The blue line indicates the ground truth (actual recorded trajectory), the red line shows the calculated mean (see (6)) for model M and the green line shows the prediction obtained from model M_{ref}. (a) shows the estimated velocities v', (b) the predicted traveled longitudinal distance d'.

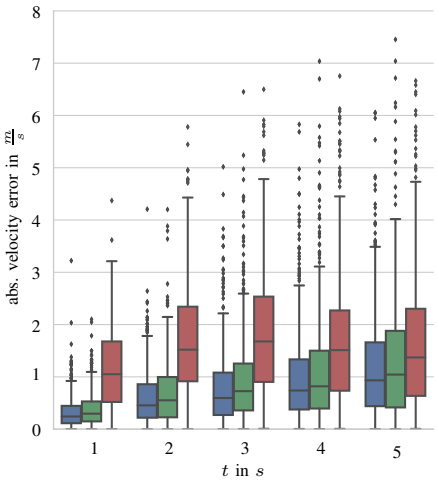
B. Experimental Results

Three different models are evaluated. The implemented model M_{ref} serves as reference system. It is based on a generic concept for creating prototypical velocity profiles, adapted from the well-known approach proposed in [12]. The authors calculate prototypical velocity profiles based on keeping the lateral acceleration at a comfortable level, while crossing an intersection. For calculating these velocity profiles detailed information of the road curvature needs to be provided. Here, we use the actually driven curvature of each trajectory for generating the prototypical velocity profiles. This information would not be available in a real-world application, as it relies on future state information. However, the system serves as reference and should not suffer from incorrectly approximated road curvatures (It should be noted that this information is not being used for models

(a)

t	model	*v-rmse*	*d-rmse*
$1s$	M	0.380 m/s	0.965 m
	$M_{\setminus r}$	0.351 m/s	0.971 m
	M_{ref}	0.582 m/s	0.941 m
$2s$	M	0.492 m/s	0.851 m
	$M_{\setminus r}$	0.519 m/s	0.876 m
	M_{ref}	1.166 m/s	1.364 m
$3s$	M	0.649 m/s	0.971 m
	$M_{\setminus r}$	0.714 m/s	1.049 m
	M_{ref}	1.549 m/s	2.260 m
$4s$	M	0.815 m/s	1.284 m
	$M_{\setminus r}$	0.909 m/s	1.433 m
	M_{ref}	1.701 m/s	3.325 m
$5s$	M	0.973 m/s	1.729 m
	$M_{\setminus r}$	1.089 m/s	1.979 m
	M_{ref}	1.794 m/s	4.388 m

(b)



(c)

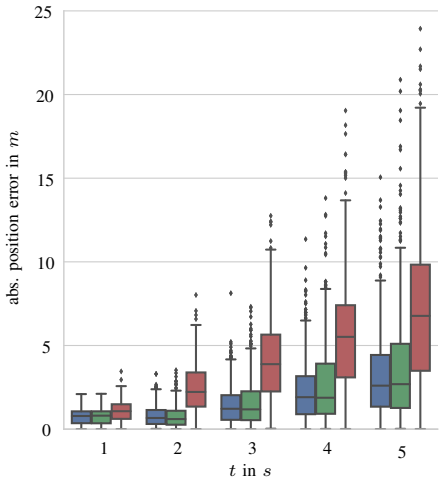


Fig. 6: (a) shows the *rmse* error metrics results. (b) shows the absolute position error (see (7)) distribution at concrete prediction horizons t over all tested trajectories. (c) shows the absolute velocity error (see (7)) distribution using boxplots. The blue boxplots correspond to model M, the green boxplots to model $M_{\setminus r}$ and the red boxplots to model M_{ref}. The bottom and the top of the boxplots represent the first and third quartile, separated by the median band. The whiskers represent data inside 2.689σ interval, the dots indicate errors which lie outside of the 2.689σ interval.

M and $M_{\setminus r}$). Based on the prototypical velocity profiles the Intelligent Driver Model (IDM) is used to simulate the future course of each test trajectory. The IDM controls the vehicle's acceleration in order to adjust the velocity to the calculated prototypical velocity, if no other vehicles are ahead. The kinematic states determined by the performed simulation can be used as trajectory prediction. [2] The reference system constructs good state-of-the-art benchmarks for state predictions at intersections. It is compared to two models based on the approach proposed in Section III. The model M uses the suggested feature vector $\mathbf{x}$, whereas model $M_{\setminus r}$ is reduced by the priority context information r_l. The priority information is excluded in order to perform a fairer comparison with the reference system, as the reference approach does not consider priority information.

Both models are based on a Neural Network with linear input neurons, two hidden layers with 30 neurons each and sigmoidal activation functions. The size of the output layer results from the number of chosen discrete bins and is realized using a softmax-layer, which outputs normalized class probabilities. In order to obtain a good performance on unseen data and to prevent the net from over-fitting $L1$-regularization as well as *dropout* techniques are applied.

The systems are trained with a temporal prediction horizon $\tau = t$ for two target variables y: traveled longitudinal distance d' and velocity v'. The traveled longitudinal distance d' is predicted, in order to estimate $s' = s_q + d'$, where s_q corresponds to the longitudinal start position at prediction time t_q. Since the focus is on extended long-term *maneuver specific* trajectory prediction, we neglect the influence of the transversal deviation n from the maneuver dependent prototypical path. However, if it is of interest, for a specific application in mind, it can also be considered. The distance d' is discretized in the range $[0, 68]m$ using equidistant $4m$ steps, while the velocity v' is discretized in the range $[0, 14]\frac{m}{s}$ using $1\frac{m}{s}$ equidistant steps.

Two exemplary predicted distance and velocity profiles, with corresponding discrete probability distributions are shown in Fig. 5.

A quantitative evaluation is performed splitting the trajectories according to their corresponding maneuver classes using a 4-fold cross validation. Each training trajectory set is used to generate training data, as described in III-D, including discretizing the target space, like exemplary shown for the velocity in Fig. 3. It should be noted that each test set completely consists of unseen trajectories. In order to calculate the error metrics the mean of the discrete distribution is calculated using

$$\bar{y}' = \sum_{\Omega} \hat{\mathbf{y}}\, p(\hat{\mathbf{y}}|\mathbf{x}, \tau). \tag{6}$$

The *root mean square error* for a predicted trajectory j, a prediction horizon of $t \in T = \{1, 2, 3, 4, 5\}s$ and starting points $s_q \in S = \{30, 20, 10, 0\}m$ before entering the intersection is defined as follows

$$e_{s_q,u} = |y_{s_q,u} - \bar{y}'_{s_q,u}|, \tag{7}$$

$$rmse_{t,j} = \sqrt{\frac{1}{|U_t||S|} \sum_{s_q \in S} \sum_{u \in U_t} e^2_{s_q,u}}, \tag{8}$$

with $U_t = \{m \in \{q, ..., N_j\} | t_m \leq t\}$ and $y_{s_q,u}$ referring to the ground truth.

[2]The parameters used for the IDM simulations correspond to the parameters of the most sporty driver model, suggested in [12], as these seem to be appropriate for the driving styles in the recorded data.

Table 6a shows the result for the mean $rmse_t$ error metric evaluation over all tested trajectories. We can see that already model $M_{\setminus r}$ significantly reduces the position error as well as the velocity error compared to the reference implementation M_{ref}, even though the reference system is based on the real future driven curvature. This indicates that the used, easy obtainable intersection geometry information, like distance to the intersection entry and exit as well as crossing angles strongly influence the behavior while crossing intersections, leading to improved prediction results.

Including priority information, as done for model M, further improves the prediction accuracy. We obtain a *rmse* for the velocity prediction of less than $1\frac{m}{s}$ and considerably less than $2m$ for the position *rmse*, at a prediction horizon of $5s$.

In order to obtain a better understanding of the error distributions, Fig. 6b and 6c show the absolute prediction errors $e_{s_q,u}$, as defined in (7). The predictions are calculated $s_q = 20m$ before entering the intersection at the concrete prediction horizon t, $u = \{m \in \{q, ..., N_j\} | t_m = t\}$. The median errors as well as the variance can be considerably reduced in comparison to the reference model.

Calculating one discrete probability distribution for model M takes approximately $0.065ms$, using a Neural Network implementation based on *Theano* ([22], [23]). The computation times were measured on one core of an Intel Core $i5$ with a maximum clock speed of 3.1*Ghz*.

V. CONCLUSION & FUTURE WORK

Elaborate criticality assessment techniques need to predict trajectories of present traffic participants several seconds in advance. However, the estimation of future states is strongly influenced by the encountered context. Therefore, we incorporate additional context information for enhancing the accuracy and prediction horizon of our trajectory prediction system. For the discussed intersection-related prediction framework, we include information describing the intersection layout, e.g. intersection entry and exit as well as the maneuver-specific crossing angle. By incorporating the crossing angle we do not need to train different systems for different maneuvers in order for our system to make accurate predictions. This is a strong indicator that the system generalizes well.

Moreover, we propose a system that makes efficient probabilistic state estimations by reformulating the probabilistic regression problem as a multiclass classification problem. Thereby, discrete probability distributions over the target variables are estimated.

The proposed system is flexible in terms of initial conditions (e.g. before entering, in or right after the intersection) and prediction horizons, while there is no need for recursively constructed predictions. The benefits of the system are validated on a real-world data set consisting of approaches to 85 different intersections.

In future, we are planning to increase the size of the data set for training the system with more complex and more extensive situations. Furthermore, we plan to incorporate bilateral interactions between traffic participants, as introduced in [24] to evaluate their effects on the prediction statistics.

REFERENCES

[1] K. C. Fuerstenberg and L. U., "New european approach for intersection safety - the ec-project intersafe," in *IV*, June 2005, pp. 177–180.

[2] F. Seeliger, G. Weidl, D. Petrich, F. Naujoks, G. Breuel, A. Neukum, and K. Dietmayer, "Advisory warnings based on cooperative perception," in *IV*, June 2014, pp. 246–252.

[3] F. Damerow and J. Eggert, "Predictive risk maps," in *ITSC*, Oct 2014, pp. 703–710.

[4] C. Hermes, C. Wohler, K. Schenk, and F. Kummert, "Long-term vehicle motion prediction," in *Intelligent Vehicles Symposium, 2009 IEEE*, June 2009, pp. 652–657.

[5] C. Hermes, J. Einhaus, M. Hahn, C. Wohler, and F. Kummert, "Vehicle tracking and motion prediction in complex urban scenarios," in *IV, 2010 IEEE*, June 2010, pp. 26–33.

[6] E. Käfer, C. Hermes, C. Wöhler, H. Ritter, and F. Kummert, "Recognition of situation classes at road intersections," in *ICRA*, May 2010, pp. 3960–3965.

[7] J. Wiest, M. Hoffken, U. Kresel, and K. Dietmayer, "Probabilistic trajectory prediction with gaussian mixture models," in *IV, 2012 IEEE*, June 2012, pp. 141–146.

[8] J. Wiest, F. Kunz, U. Kressel, and K. Dietmayer, "Incorporating categorical information for enhanced probabilistic trajectory prediction," in *ICMLA*, vol. 1, Dec 2013, pp. 402–407.

[9] M. Pandey and Y. Dalbah, "A method for reducing false warnings in collision warning system (cws) during turning maneuvers at road intersection," in *IV, 2014 IEEE*, June 2014, pp. 310–316.

[10] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *IROS, 2013 IEEE/RSJ*, Nov 2013, pp. 4363–4369.

[11] D. Petrich, T. Dang, D. Kasper, G. Breuel, and C. Stiller, "Map-based long term motion prediction for vehicles in traffic environments," in *ITSC, 2013 IEEE*, Oct 2013, pp. 2166–2172.

[12] M. Liebner, F. Klanner, M. Baumann, C. Ruhhammer, and C. Stiller, "Velocity-based driver intent inference at urban intersections in the presence of preceding vehicles," *Intelligent Transportation Systems Magazine, IEEE*, vol. 5, no. 2, pp. 10–21, Summer 2013.

[13] S. Klingelschmitt, M. Platho, H.-M. Gross, V. Willert, and J. Eggert, "Combining behavior and situation information for reliably estimating multiple intentions," in *IV*, June 2014, pp. 388–393.

[14] S. Klingelschmitt, F. Damerow, and J. Eggert, "Managing the complexity of inner-city scenes: An efficient situation hypotheses selection scheme," in *IV*, 2015.

[15] F. Damerow and J. Eggert, "Risk-aversive behavior planning under multiple situations with uncertainty," in *ITSC*. IEEE, in press, 2015.

[16] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard, "Most likely heteroscedastic gaussian process regression," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 393–400.

[17] R. M. Neal, "Bayesian learning for neural networks," Ph.D. dissertation, University of Toronto, 1995.

[18] M. Althoff, O. Stursberg, and M. Buss, "Model-based probabilistic collision detection in autonomous driving," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 10, no. 2, pp. 299–310, June 2009.

[19] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[20] M. Feindt, "A Neural Bayesian Estimator for Conditional Probability Densities," *ArXiv Physics e-prints*, Feb. 2004.

[21] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012.

[22] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio, "Theano: a CPU and GPU math expression compiler," in *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June 2010, oral Presentation.

[23] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio, "Theano: new features and speed improvements," Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[24] M. Platho, H.-M. Gros, and J. Eggert, "Predicting velocity profiles of road users at intersections using configurations," in *IV, 2013 IEEE*, June 2013, pp. 945–951.