2018 26th Mediterranean Conference on Control and Automation (MED)
Zadar, Croatia, June 19-22, 2018

Infrastructure-based vehicle maneuver estimation with intersection-specific models*

Christian-Eike Framing[1], Frank-Josef Heßeler[1] and Dirk Abel[1]

***Abstract*— Advanced Driver Assistance Systems have the potential of making traffic safer, more efficient and convenient. At urban intersections the conflicting intentions of different traffic participants frequently lead to accidents. In order to avoid dangerous situations in such a scenario an ADAS requires an accurate estimation of future driving maneuvers executed by other vehicles. In this work we present a framework that estimates the driving maneuver of vehicles approaching at intersections where the vehicle state can only be observed externally. The high variance in driver behavior that arises from different intersection layouts is met with intersection-specific models that are computed locally. The performance of the estimator and the potential of intersection-specific models are validated with data recorded from real intersections.**

I. Introduction

Major research has been conducted in the last years on automated driving which is expected to improve the safety, efficiency and comfort of today's traffic. For full automation of the driving task a respective system needs to follow a control strategy that, considering the constraints, is optimal with respect to a given objective. One important external constraint is the vehicle's surrounding traffic. In order to avoid collisions with other objects the current traffic situation needs to be perceived and predicted for a suitable time horizon into the future. With that information the system can assess potential risks and take appropriate action.

The traffic situation prediction yields future trajectories of each perceived traffic participant and is therefore also called *motion prediction*. One approach to predict the motion of a vehicle specifically is a maneuver-based motion model that follows a two-step procedure: Given the known sequence of past and current vehicle states a classification algorithm first determines the most probably intended driving maneuver. It selects from a predefined discrete set of driving maneuvers that each describe a characteristic movement sequence of the vehicle such as a right turn. Assuming that the vehicle will follow through with the estimated maneuver a specialized motion model will then generate a matching predicted trajectory.

Innercity traffic and intersections in particular still are accident-prone and thus pose a potential application for maneuver estimation. However intersections show a wide variety of intersection layouts, lane arrangements and traffic participant types. This variance as well as the higher density of heterogeneous traffic influence the traffic participants' behavior. Additionally sensors experience a higher rate of misdetections and the traffic environment may not be fully perceivable from the vehicle's point of view. This introduces a higher uncertainty in the environment perception of individual vehicles and complicates the formulation of a general maneuver estimation framework.

A. Related Work

Vehicle maneuver estimation has been investigated both for predicting the ego-vehicle and other vehicles at intersections. [1] and [2] use on-board available vehicle information such as the yaw rate and sequentially predict the driving maneuver by evaluating Hidden Markov Models. [3] employ a similar strategy but instead use Gaussian Process models. An advantage is the simultaneous use of the models for subsequent trajectory prediction. Instead of a generative model for each driving maneuver [4] use pair-wise coupling of individual binary classifiers to form detectors for different driving maneuvers. This approach allows to consider the interaction between vehicles. [5] use another discriminative approach in the form of Long Short Term Memory networks (LSTM). The classifier is trained with a large dataset and a combination of kinematic, interaction, rule-based and intersection features. The results indicate a generalization across intersections. In a V2X scenario [6] propose an intersection-specific approach that estimates the maneuver with the help of a probabilistic lane matching algorithm and on-board vehicle data. However the concept could only be validated in test scenarios.

B. Problem Statement

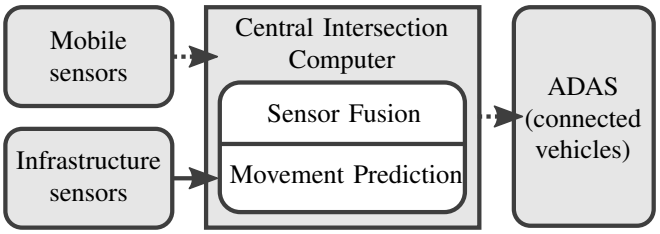


Fig. 1: Signal flow between sensors, users and the Central Intersection Computer at the connected intersection

In our scenario we consider an intersection that is both equipped with infrastructure sensors and wirelessly connected with traffic participants who act as mobile sensors (see Figure 1). All sensor information in this V2X scenario is locally processed on a *Central Intersection Computer* (CIC).

*This work is part of the I2EASE project funded by the German Federal Ministry of Education and Research

[1]Christian-Eike Framing, Frank-Josef Heßeler and Dirk Abel are with the Institute of Automatic Control, RWTH Aachen University, 52074 Aachen, Germany C.Framing, F.Hesseler, D.Abel@irt.rwth-aachen.de

A sensor fusion algorithm combines the data to yield a complete representation of the traffic situation. Subsequently the future trajectories of all traffic participants are predicted. The perceived and predicted traffic situation then serve as input for ADAS to assess potential risks.

As part of the motion prediction the driving maneuvers of vehicles at the intersection need to be estimated *accurately* and *early*. Because not all vehicles at the intersection provide on-board data the estimation can only rely on *externally observable information*. Finally the estimator must be able to *adapt to local driver behavior*.

In the remainder of this paper Section II will detail our estimator framework. In Section III the intersection model and an appropriate lane matching algorithm are detailed. The recording and analysis of a dataset are described in Section IV before the estimator is validated in Section V. At last in Section VI we draw a conclusion of our work and identify potential future extensions.

II. Estimator Framework

From the literature it is apparent that the problem of vehicle maneuver estimation is suitable to be approached with machine learning methods. Since much of the information about a driving maneuver is represented in the characteristic sequence of observed vehicle states the model should represent this dynamic. Both generative and discriminative models are used in the literature. While discriminative approaches have the potential advantage of a better generalization to many scenarios they usually require a high amount of training data and features. Generative approaches are better suited for specialized scenarios where less training data and features are available [7]. Since for our problem we need an estimator specialized to the respective intersection and no on-board vehicle information is available we choose a generative approach in the form of Hidden Markov Models (HMM). The HMM as class model and its combination as multi-class estimator are described in the following.

A. Class model

A HMM is a stochastic model for dynamic processes. The core of the model is a hidden (*unobserved*) variable x_k that in each timestep k randomly transitions from one state to another. The transitions follow a probability distribution A that only depends on the state of the previous timestep x_{k-1}. This property is called *markov*, hence the sequence's name *markov chain*. Additionally there is a second *observed* random variable y_k. In each timestep k this variable assumes values from a probability distribution ϕ that only depends on the respective hidden state x_k. The observed values of y_k are also referred to as model *emissions*. The probability distribution A describing the state transitions is categorical so that x_k assumes discrete values. The probability distribution ϕ governing the model emissions is chosen specific to the modeled process. Our choice is detailed in Section IV-C.

Following [8] the formal definition of a HMM with $s_1, \ldots, s_n$ states and m-dimensional continuous emissions is the tuple $\lambda = (\pi, A, \phi)$ with:

- $\pi \in \mathbb{R}^n$ the initial probability distribution. $\pi_i = p(x_1 = s_i)$ is the probability that the initial state is s_i.
- $A \in \mathbb{R}^{n \times n}$ the transition matrix between hidden states. $a_{ij} = p(x_k = s_j | x_{k-1} = s_i)$ is the probability that the hidden variable changes from state s_i to s_j in one step
- $\phi_x \in \mathbb{R}^m$ the emission probability distribution where ϕ_1 describes the observed variable y if the current hidden state is s_1

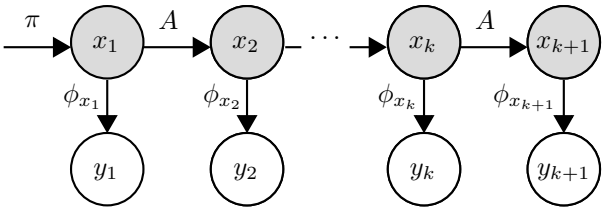


Fig. 2: Example for a temporal sequence in the HMM

In Figure 2 the state space model with the markov chain of hidden states and the observed emissions is depicted. Conditional dependence is indicated by an arrow pointing to the dependent variable. If a dynamic process is to be identified using this model structure a sequence of emissions $\{y_1, \ldots, y_o\}$ from the measurable process output is recorded. The model parameters λ_{ML} that best describe the emission sequence are then found by maximizing the likelihood on the observed data given the model:

$$\lambda_{ML} = \arg\max_{\lambda} p(y_1, \ldots, y_o | \lambda) \tag{1}$$

In general this *training problem* cannot be solved analytically. Instead an estimate of the maximum likelihood model parameters $\hat{\lambda}_{ML}$ can iteratively be determined by using the Baum-Welch algorithm. Since it is a variant of the Expectation-Maximization Algorithm (EM) it is guaranteed to find a set of parameters with higher likelihood on the data after each iteration. As this method will converge to a local optimum only repeated training with randomized initial parameters will lead to a model that describes the observed dynamic well.

Once a model is trained through Equation 1 we can determine how well it describes a new set of emissions $\{y_1^*, \ldots, y_o^*\}$. This *evaluation problem* corresponds to calculating the likelihood of the new emissions being produced by the trained model:

$$p(y_1^*, \ldots, y_o^* | \hat{\lambda}_{ML}) = \sum_X p(y_1^*, \ldots, y_o^*, x_1^*, \ldots, x_o^* | \hat{\lambda}_{ML}) \tag{2}$$

Depending on the model the emission sequence can be explained by different hidden state sequences $\{x_1^*, \ldots, x_o^*\}$. The total likelihood in Equation 2 is the sum of all possible combinations. This marginalization step is efficiently carried out by the Forward algorithm. For further details on the Baum-Welch and the Forward algorithm refer to [8].

B. Multi-class estimator

Given a vehicle trajectory the multi-class estimator determines the most probably driven maneuver. Since the

focus of this work lies on driving maneuvers carried out at intersections we distinguish between the three different maneuvers:

$$M = \{\text{Go straight, Turn left, Turn right}\} \tag{3}$$

The multi-class estimator is composed of a set of class models for individual driving maneuvers. Each class model is trained with trajectories of the corresponding driving maneuver that were extracted from a dataset. The result are the models:

$$\Lambda = \{\lambda_{straight}, \lambda_{left}, \lambda_{right}\} \tag{4}$$

In the CIC use case the estimator will determine the driving maneuver of an approaching vehicle in each time step by evaluating Equation 2 for each class model. The emissions that are used to determine the likelihood each consist of N features $y = (f_1, f_2, \ldots, f_N)$ that are derived from the observed vehicle state (see Section IV-B for the features selected in this work). As the trajectory gets longer each model is more and more unlikely to produce exactly the corresponding emission sequence. The likelihoods will thus decrease for all three models. Still a conclusion about the most probable driving maneuver can be drawn from the relative magnitude of the likelihoods.

III. Intersection Model

A. Digital Intersection Map

For post-processing and feature extraction of the vehicle trajectories a digital intersection map was created for each examined intersection. Based on the Openstreetmap standard it provides information about driving lane geometry, stop lines and permitted driving maneuvers at the intersection.

Each lane's centerline is modeled by a set of nodes connected through directional linear segments. Lanes are grouped in *entry*, *transit* and *exit* lanes. An entry lane marks the approach up to the stop line at intersection entry. A transit lane is the path a vehicle will follow inside the intersection area to execute a given driving maneuver. Its centerline is often not explicitly indicated with lane markings and had to be deduced from recorded trajectories. Upon exiting the intersection area a vehicle will follow an exit lane. All permitted driving maneuvers at the intersection are definable as a combination of one entry, transit and exit lane each. In a next step the piecewise linear representation of the lane centerlines is interpolated with cubic splines. This avoids discontinuities in the lane matching process.

Additionally to the lanes the entry and exit areas of the intersection are separately marked. If given a complete vehicle trajectory the corresponding driving maneuver label can be determined by the crossed entry and exit areas.

B. Lane Matching

The trajectory of a vehicle as output from the sensor fusion is described in a local intersection-specific East-North-Up (ENU) coordinate system $\boldsymbol{p}(t) = (x, y)$. For the estimator this representation has the disadvantage of being variant in rotation and translation depending on the approaching direction. In order to assess the similarity of trajectories from different approaches it is convenient to transform them into a lane-dependent *Frenet* frame [3]. For our problem a point (s, d) in the Frenet frame (see Figure 3) is defined by:

- s: Projected distance to stop line $[m]$
- d: Lateral offset d along lane normal vector $[m]$

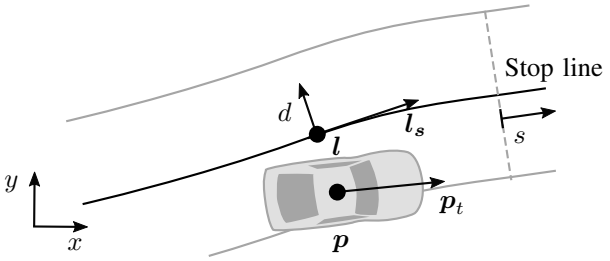


Fig. 3: Frenet frame moving along lane centerline and example vehicle with lateral offset to the right.

A requirement to the description of a vehicle position in the Frenet frame is the knowledge of its currently driven lane. For this *Lane matching* problem we apply a Nearest-Neighbor approach as follows: At time t^* the longitudinal distance of the vehicle along all L lane centerlines $\boldsymbol{l}_i(s)$ is determined as:

$$s_i^* = \arg\min_s \left\| \boldsymbol{p}(t^*) - \boldsymbol{l}_i(s) \right\|, \forall i \in \{1, \ldots, L\} \tag{5}$$

Now only lanes whose tangent offset angle is less than $30°$ from the trajectory are regarded:

$$\angle \frac{\partial \boldsymbol{p}(t^*)}{\partial t} \frac{\partial \boldsymbol{l}_i(s_i^*)}{\partial s} < 30° \tag{6}$$

The lane match (s^*, d^*) then lies on the i-th lane with minimum distance to the vehicle:

$$(s^*, d^*) = (s_i^*, \left\| \boldsymbol{p}(t^*) - \boldsymbol{l}_i(s_i^*) \right\|) \tag{7}$$

$$\text{with } i = \arg\min_i \left\| \boldsymbol{p}(t^*) - \boldsymbol{l}_i(s_i^*) \right\| \tag{8}$$

In the use case of the CIC object positions are acquired with an accuracy well below the lane width. As a consequence this simple approach achieves good results. Within the intersection area mismatches with overlapping lanes are effectively avoided by correlating the tangent angles.

IV. Data Analysis

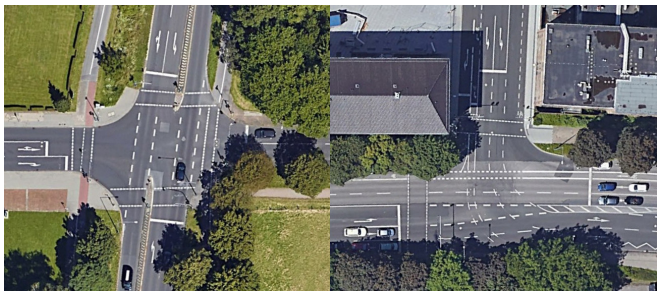
In order to parametrize and validate the proposed methodology a set of real vehicle trajectories is required. Since no public dataset was found suitable to evaluate vehicle movements at one intersection a new dataset was recorded. To capture both a high traffic volume as well as realistic driver behavior the recordings were conducted in real inner-city traffic. Two intersections with three and four incoming roads respectively were selected (see Figure 4). At both intersections a speed limit of $50\,\mathrm{km\,h^{-1}}$ is imposed. The traffic is controlled by traffic lights and on at least one incoming road per intersection a vehicle is permitted to go straight, turn left or turn right. As real traffic was recorded

Driving maneuver	Intersection A	Intersection B
Go straight	459	118
Turn left	37	63
Turn right	55	84

TABLE I: Recorded trajectories per intersection and driving maneuver

the trajectories originated from many different drivers and vehicles with varying size.

The recordings at each intersection were conducted with a system of four ibeo LUX 3 laserscanners. A mobile tripod was configured with the individual units and placed with a view on all incoming roads of the intersection. An internal sensor fusion of the system provided a list of detected objects at a rate of $25\,\mathrm{Hz}$. Similar to the use case of the estimator in the CIC the sensor fusion output consisted of an object ID, type, position and velocity. The object position was deduced as center point of an estimated two-dimensional bounding box around the object. By tracking objects of the vehicle type and identical ID the assembly of individual vehicle trajectories and their automatic labeling with corresponding driving maneuvers was possible.



(a) Four-way intersection (b) Three-way intersection

Fig. 4: The two examined intersections A and B of the recorded dataset both located in Aachen, Germany (Google Earth top-view)

A. Data Interpretation

The resulting number of trajectories is depicted in Table I. Depending on the placement of the laserscanner certain driving maneuvers were less often fully tracked than others. This is the case for turning maneuvers at Intersection A and straight maneuvers at Intersection B.

The velocity profile v of the recorded vehicles is displayed over their distance to the stop line s in Figure 5. For analysis the figure distinguishes between driving maneuvers and recording location. In order to visualize trends in the data a two-component gaussian mixture distribution was fitted for equidistant steps along the s-axis. The mean of each component is overlayed as dashed black line. The velocity profile of straight-crossing vehicles shows two trends: At both intersections vehicles travelling freely make a crossing with a constant velocity near the speed limit of $50\,\mathrm{km\,h^{-1}}$. Vehicles that need to make a stop or are following other vehicles will approach the intersection at a lower velocity with a mean of $20\,\mathrm{km\,h^{-1}}$ before accelerating upon crossing. It is notable that for Intersection B few examples of free travelling vehicles were tracked and in general the vehicles were detected only shortly before crossing the stop line. Apart from these sensor placement induced variations vehicles show a comparable behavior for a straight maneuver at both intersections.

For a left turn maneuver again two main trends are visible: Vehicles with a stop or car-following behavior are approaching at lower velocity before crossing the intersection with constant velocity. Free-travelling vehicles approach with higher velocity before showing constant deceleration. In this example there is a noticeable difference between the behavior at both intersections. Non-free travelling vehicles cross at a velocity about $7\,\mathrm{km\,h^{-1}}$ higher in Intersection B as compared to Intersection A. The deceleration of free-travelling vehicles at Intersection B continues further into the intersection area. Both effects may be explained by the different intersection geometry. In comparison Intersection B features a larger inner intersection area which leads to wider and thus faster left turns for vehicles approaching from the top end (visible in Figure 4b). The respective stop line is located far before the actual turn. The behavior for a right turn maneuver shows similar trends as that for a left turn maneuver and is comparable at both intersections.

B. Feature Selection

From the analysis of the data a suitable feature vector $\boldsymbol{y}_k = (f_1, f_2, \ldots, f_N)$ for the estimator has to be determined. A strong feature needs to sharply separate the data into at least two distinct sets. The combined use of a feature set should then lead to the separation of all three individual driving maneuver classes. Commonly found in the literature are features selected from on-board vehicle data such as the steering wheel angle or the yaw rate [1]. In the use case of the CIC these measurements are not available for most vehicles. Here a conclusion about a vehicle's driving maneuver can only be drawn from externally observed and further synthesized features.

The most commonly used features in the literature are the vehicle velocity v in combination with the distance to the intersection s. As seen in Section IV-A these are characteristic variables that intuitively separate a straight from a turn maneuver. Still depending on the intersection the velocity for a turn maneuver was found to change in magnitude and shift relative to the stop line. An attempt to synthesize a normalized distance $s_{norm} = \frac{s}{\text{length of transit lane}}$ independent of the intersection size did not show the desired generalization performance. The variations seem to be caused by a combination of intersection properties. Additional to the longitudinal distance the lateral distance d was found to significantly improve the overall estimation performance.

As the sensor did not measure the vehicle acceleration a we obtain it through a discrete derivative of the velocity. Effects of sensor noise on the derivative are mitigated by filtering the velocity with a moving average and a flat window of 25 samples. Instead of using the acceleration directly we

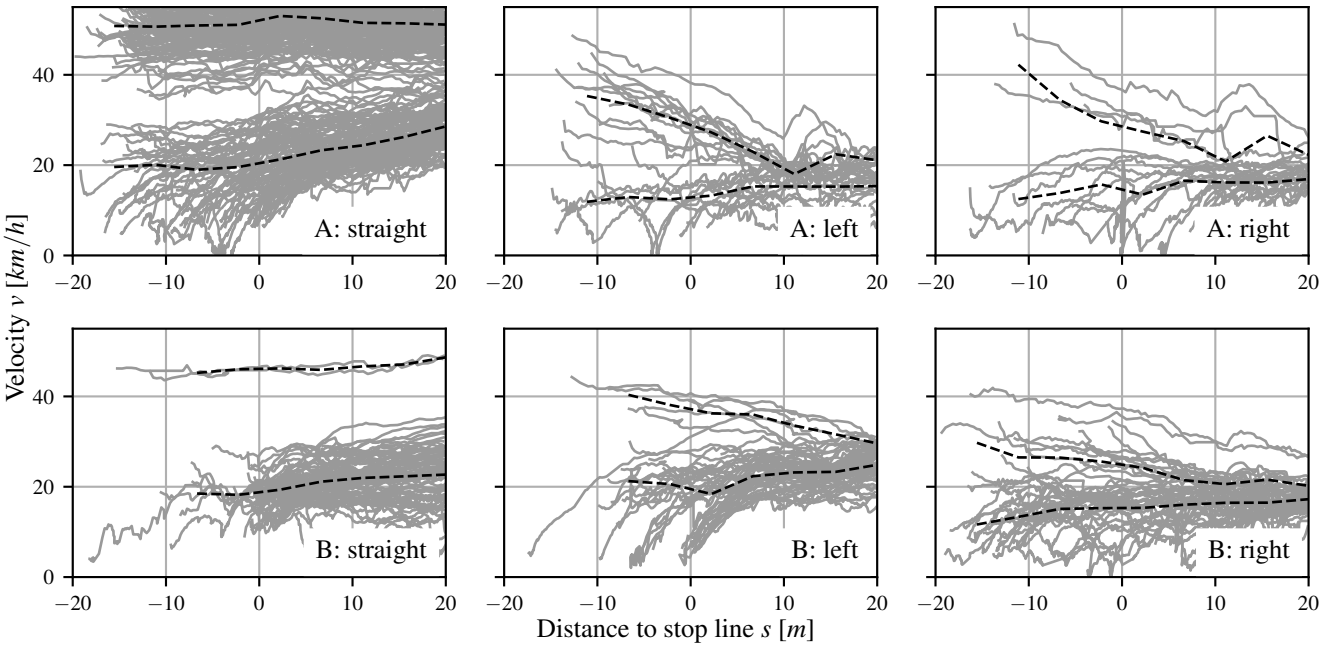


Fig. 5: Vehicle velocity (solid grey) before and during intersection crossing plotted over distance to stop line. The plots are separated into the respective driving maneuver (left to right) and the two intersections (top and bottom). Velocity trends are indicated as dashed black lines.

employ the anticipated velocity at the stop line avs proposed by [9]. In each time step this feature is calculated from the current distance to the stop line, the velocity and the acceleration by assuming constant acceleration behavior. A feature that was intentionally omitted is the set of permitted driving maneuvers on the current lane. Even though the estimator in the CIC use case would benefit from this feature we choose to evaluate all intersection approaches equally. With the relatively small dataset in mind we put the focus on determining the performance of kinematic features. The selected features together form the four-dimensional model emission:

$$y = (s, d, v, avs) \tag{9}$$

C. Estimator Parametrization

In the class models used by the estimator the process dynamic is captured in the transition matrix A governing the hidden state sequence. Each state can be interpreted as the description of one specific phase out of the respective driving maneuver. As opposed to [10] we found that fixed left-to-right state topologies generally were not able to reproduce the dynamic of all driving maneuvers equally well. Similar to [1] we thus use a *dense transition matrix*. A grid search revealed the number of 5 *hidden states* to be sufficient.

Associated to each state is an emission probability distribution ϕ that describes the observed features in that specific phase. From Figure 5 it becomes apparent that a multimodal distribution is needed to reproduce the separate velocity trends. For our purpose we use a *mixture* of *multivariate gaussian distributions*

$$p(y) = \sum_{i=1}^{D} p(y|\mu_i, \Sigma_i) \cdot P_j \tag{10}$$

where each component i is described by a mean $\mu_i \in \mathbb{R}^m$, a covariance matrix $\Sigma_i \in \mathbb{R}^{m \times m}$ and a weight P_j. Through further grid search we determined $D = 3$ *mixture components* to be sufficient. Upon initialization the component means μ_i are for all states equally distributed across the feature space before the model is fitted on the dataset.

V. Validation

The framework was validated with the task of predicting the driving maneuver of an approaching vehicle before crossing the stop line ($s < 0$). In a realistic scenario this would leave enough time for an ADAS to take action in case of an impending conflict. After preparation of the datasets the performance of the trained estimator was determined as the average of a five-fold stratified cross-validation. For each fold the estimator was trained on $80\,\%$ of the data and validated on the remaining unseen $20\,\%$.

A. Overall performance

The overall performance of the estimator was examined with the dataset from Intersection A. All three driving maneuvers were considered. Result Table II lists the confusion matrix, the accuracy, F_1-score[1] and the mean prediction time before intersection entry. For this multi-class problem the true positive (TP), true negative (TN), false positive

[1] $ACC = \frac{TP+TN}{TP+TN+FP+FN}$, $F_1 = \frac{2 \cdot TP}{2 \cdot TP+FP+FN}$ (see [11])

Actual → Predicted ↓	Straight	Left	Right	ACC	F_1	$\overline{TTI}_{pred}$
Straight	74	0	0	87 %	91 %	3.7 s
Left	12	6	1	88 %	48 %	4.9 s
Right	2	0	9	97 %	86 %	9.4 s

TABLE II: Confusion matrix, individual class performance and mean prediction time for Intersection A

Training/Test data	Actual → Predicted ↓	Left	Right	ACC
A	Left	6	1	94 %
	Right	0	9	
B	Left	7	1	96 %
	Right	0	15	
A+B	Left	14	13	68 %
	Right	0	13	

TABLE III: Comparison of estimator performance for intersection-specific or general training data

(FP) and false negative (FN) predictions were determined in a one-vs-all fashion. $\overline{TTI}_{pred}$ is the average time before intersection entry when a correct prediction was achieved and held until the entry.

All individual class performances exceed a threshold of 87 %. This performance is slightly lower than in [1] but can still be rated good considering a higher variance in the data and the fact that only externally observable features were used. However it is notable that 12 straight-crossings and 1 right turn maneuver were falsely predicted as left turns. An individual inspection revealed that the estimator shows rather poor performance for vehicles coming to a halt before the intersection. This behavior is frequently misinterpreted as a left turn maneuver. A vehicle starting shortly before the stop line will clearly make very little way to enter the intersection. This short time span seems not characteristic enough to distinguish the maneuvers. An additional class for a stop-maneuver could be introduced to avoid misinterpretation. In case of correct predictions the $\overline{TTI}_{pred}$ leaves enough time for correctional measures for all three maneuvers.

B. Potential of Intersection-specific models

The CIC use case potentially allows estimator models specifically fitted for one intersection. In a comparison it was examined how the estimator performs if trained with a dataset from individual or both intersections. Because only few intersection approaches ($s < 0$) for straight-crossing vehicles are available from Intersection B this comparison was conducted with left and right turn maneuvers only. The results are shown in Table III.

For both intersections the intersection-specific trained estimator achieves good performance on unseen data. Using the externally observable features the two driving maneuvers can be effectively distinguished. For the general case with training and test data from both intersections the accuracy drops to 68 %. This result conforms with remarks from Section IV-A where driver's behavior was observed to be specific to the intersection. In contrast the estimator trained for one intersection is able to learn this specific behavior and adapt to local characteristics not captured in a feature.

VI. Conclusion

In this work a framework for the infrastructure-based estimation of vehicle maneuvers at urban intersections was presented. Using a recorded dataset from two intersections observable features were selected and synthesized with the help of a lane matching algorithm. The framework could be validated to overall distinguish with good accuracy between a straight, a left and a right turn maneuver. Only vehicles with a stop before the intersection did lead to increased false predictions. With different sets of training data from two intersections we noted that the estimator performance decreases with added data. We assume that from externally observable features alone it is difficult to generalize driver behavior over different intersections. In the CIC use case the estimator can fit local driver behavior and effectively adapt to specific intersection layouts.

For future work the behavior of vehicles halting and departing shortly before intersection entry has to be examined for possible extension of the model. The traffic light state could serve as an appropriate feature. As the estimator is regularly provided with new measured vehicle trajectories from the sensor fusion a continuously learning model seems promising with regard to temporally changing behavior.

References

[1] Thomas Streubel and Karl Heinz Hoffmann. "Prediction of Driver Intended Path at Intersections". In: *IEEE Intelligent Vehicles Symp. (IV) Proc.* 2014, pp. 134–139.

[2] Seifemichael B. Amsalu et al. "Driver Intention Estimation via Discrete Hidden Markov Model". In: *IEEE Systems, Man, and Cybernetics (SMC) Proc.* 2017, pp. 2712–2717.

[3] Quan Tran and Jonas Firl. "Online Maneuver Recognition and Multimodal Trajectory Prediction for Intersection Assistance Using Non-Parametric Regression". In: *IEEE Intelligent Vehicles Symp. (IV) Proc.* 2014, pp. 918–923.

[4] Stefan Klingelschmitt, Volker Willert, and Julian Eggert. "Probabilistic, Discriminative Maneuver Estimation in Generic Traffic Scenes Using Pairwise Probability Coupling". In: *IEEE Intelligent Transportation Systems (ITSC) Proc.* 2016, pp. 1269–1276.

[5] Derek J. Phillips, Tim A. Wheeler, and Mykel J. Kochenderfer. "Generalizable Intention Prediction of Human Drivers at Intersections". In: *IEEE Intelligent Vehicles Symp. (IV) Proc.* 2017, pp. 1665–1670.

[6] Tobias Schendzielorz, Paul Mathias, and Fritz Busch. "Infrastructure-Based Vehicle Maneuver Estimation at Urban Intersections". In: *IEEE Intelligent Transportation Systems (ITSC) Proc.* 2013, pp. 1442–1447.

[7] Maximilian Panzner and Philipp Cimiano. "Comparing Hidden Markov Models and Long Short Term Memory Neural Networks for Learning Action Representations". In: *Machine Learning, Optimization and Big Data*. Springer, 2016, pp. 94–105.

[8] Christopher M Bishop. *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[9] Stefan Klingelschmitt et al. "Combining Behavior and Situation Information for Reliably Estimating Multiple Intentions". In: *IEEE Intelligent Vehicles Symp. (IV) Proc.* June 2014, pp. 388–393.

[10] Holger Berndt and Klaus Dietmayer. "Driver Intention Inference with Vehicle Onboard Sensors". In: *IEEE Vehicular Electronics and Safety (ICVES) Proc.* 2009, pp. 102–107.

[11] Marina Sokolova and Guy Lapalme. "A Systematic Analysis of Performance Measures for Classification Tasks". In: *Information Processing & Management* 45.4 (2009), pp. 427–437.