2015 IEEE Intelligent Vehicles Symposium (IV)
June 28 - July 1, 2015. COEX, Seoul, Korea

When will it change the lane?A probabilistic regression approach for rarely occurring events

Julian Schlechtriemen[1], Florian Wirthmueller[2], Andreas Wedel[3], Gabi Breuel[4], Klaus-Dieter Kuhnert[5]

***Abstract*— Understanding traffic situations in dynamic traffic environments is an essential requirement for autonomous driving. The prediction of the current traffic scene into the future is one of the main problems in this context. In this publication we focus on highway scenarios, where the maneuver space for traffic participants is limited to a small number of possible behavior classes. Even though there are many publications in the field of maneuver prediction, most of them set the focus on the classification problem, whether a certain maneuver is executed or not. We extend approaches which solve the classification problem of lane-change behavior by introducing the novel aspect of estimating a continuous distribution of possible trajectories.**

Our novel approach uses the probabilities which are assigned by a Random Decision Forest to each of the maneuvers lane following, lane change left and lane change right. Using measured data of a vehicle and the knowledge of the typical lateral movement of vehicles over time taken from realworld-data, we derive a Gaussian Mixture Regression method. For the final result we combine the predicted probability density functions of the regression method and the computed maneuver probabilities using a Mixture of Experts approach.

In a large scale experiment on real world data collected on multiple test drives we trained and validated our prediction model and show the gained high prediction accuracy of the proposed method.

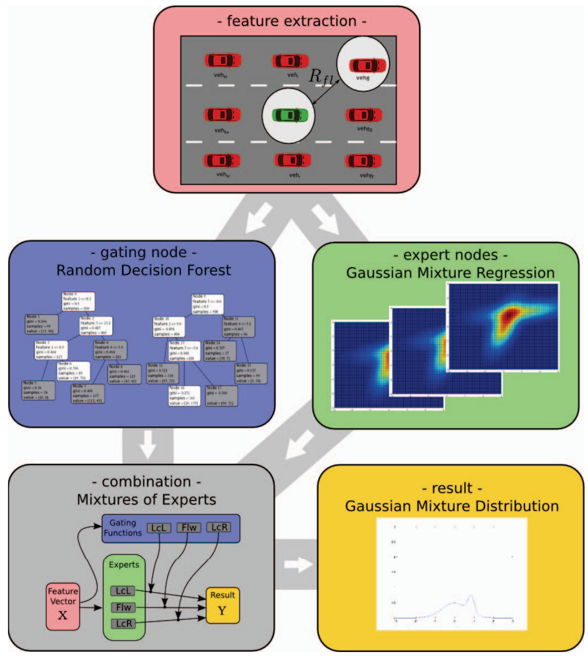


Fig. 1. Data flow of our proposed approach: Classification results of a maneuver recognition algorithm in the *gating node* and a probabilistic trajectory prediction in the *expert nodes* are combined using a *Mixture of Experts* approach. The result of the computation is a density of the future position conditioned over time.

I. INTRODUCTION

In order to avoid accidents and find a safe way to their destination, human drivers use their intuition and knowledge about formerly experienced situations when driving a vehicle. Due to their driving experience they are often able to predict the behavior of other drivers even in complex traffic scenarios. Based on the observation for example, that a vehicle on a neighboring lane will perform a cut-in maneuver, human drivers can estimate the dynamic behavior of other vehicles. For autonomous driving the evaluation of complex traffic scenarios has to be performed by algorithms: especially the task of risk assessment has to be thoroughly performed by an autonomous car because the driver is out of the loop. In this paper we introduce a probabilistic formulation in the space of trajectories, needed for such risk assessment algorithms.

This novel approach extends former approaches which only focus on determining the maneuver class of trajectories. In a follow-up step our derived probabilistic trajectory information can be used for robust decision making. This extends current state of the art methods for probabilistic risk assessment: by providing probabilities in the trajectory space, strategies for the planning of trajectories obtain the capability to take into account the uncertainty of the evolution of traffic situations.

One of the main challenges that one has to deal with in the context of lane-change recognition is, that the problem is highly unbalanced. As we showed in [1], the proportion of lane-change samples to lane-following samples in real-world traffic Scenarios is close to 1 % vs. 99 %. Conventional learning methods will run into a problem with such kind of unbalanced data [2]. To overcome this, we propose the use of a Mixture of Experts approach see Fig. 1. A Mixture of Experts Approach aims at solving such balancing problems

[1]Julian Schlechtriemen is with Daimler AG, 71034 Böblingen, Germany julian.schlechtriemen@daimler.com
[2]Florian Wirthmueller is with the Illmenau University of Technology, flwirthm@googlemail.com
[3]Andreas Wedel is with Daimler AG, 71034 Böblingen, Germany andreas.wedel@daimler.com
[4]Gabi Breuel is with Daimler AG, 71034 Böblingen, Germany gabi.breuel@daimler.com
[5]Klaus-Dieter Kuhnert is with the Institute for Real-Time Learning Systems, University of Siegen, Germany kuhnert@fb12.uni-siegen.de

978-1-4673-7266-4/15/$31.00 ©2015 IEEE

by splitting the data into different areas corresponding to different classes [3] [4]. This kind of splitting allows us to provide precise estimates of the position of a vehicle for each maneuver class and for every vehicle, for which the required information is measurable.

Our paper is structured as follows: in section II we review related work in the field of trajectory prediction and discuss the necessary extensions in order to derive a probabilistic formulation in the space of trajectories. Then we provide an overview on the feature-set that is used for the prediction in the gating and regression nodes of our Mixture of Expert algorithm.

Section IV introduces the gating nodes of the Mixture of Experts approach. In Section V we show in detail how we solve the regression problem in order to get probabilistic results. Our algorithm avoids the problem with rare events when dealing with the regression problem. This is achieved by fusing the information of the gating and the experts nodes using the Mixture of Experts approach, which results in a unique distribution of the future lateral position of the vehicle. In section VI we describe this fusion algorithm. Finally, we give an overview of our experimental results in section VII and conclude with section VIII.

II. Related Work

In the research field of probabilistic vehicle trajectory prediction there exist numerous publications. In general these publications can be divided into two parts. The first part attacks the correct classification of a discrete driving-maneuver set, while the second part is focused on the prediction of trajectories. To the best of our knowledge: When looking at publications which focus on the classification of maneuvers, we can distinguish between probabilistic and deterministic approaches.

In former work we derived a simple classification approach using a Naive Bayesian Classifier with Gaussian Mixture as distribution model. The main focus was the definition of an environment model and the systematic reduction of variables. We showed the superior prediction performance of this straightforward technique in comparison to former approaches, while using only three features [1]. In comparison [5] proposes the use of a Support Vector Machine and an additional Bayes Filter with an expert chosen feature-set. In [6] the recognition problem was solved by implementing a Case-Based-Reasoning approach which enables online-learning. As an extension to former maneuver recognition approaches [7] proposes the use of a hierarchical model to combine the results of multiple classifiers tailored for specific situations. For a more comprehensive review of the state of the art approaches for the recognition of driving maneuvers, see also [5].

When it comes to trajectory prediction algorithms, one example is the map based prediction approach described in [8]. For every hypothesis the uncertainty estimates are generated by a Kalman-Filter using pseudo update steps based on the road geometry. This generative modeling provides the desired probabilistic output, but it is not guaranteed

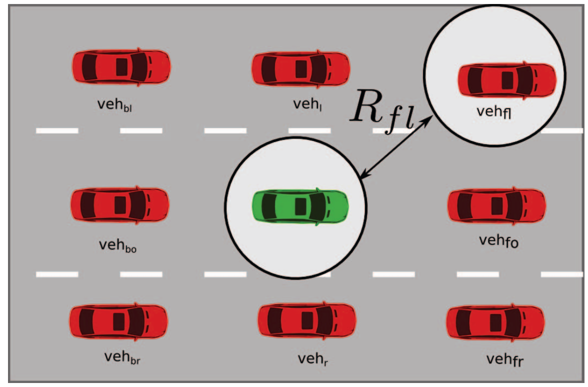


Fig. 2. For the prediction of an arbitrary vehicle o (green) we define relations to the surrounding vehicles. Each of these relations contains multiple features, for example the relation R_{fl} contains the relative velocity to the vehicle in front on the left lane $v^{rel}_{x,fl}$, the time-gap, etc.

that the uncertainty estimates correspond to the uncertainty in real-world. Furthermore uncertainties are assumed to be Gaussian-distributed and the assigned probabilities of the hypotheses only depend on the geometric configuration of the vehicle within the road geometry. In comparison [9] estimates a distribution over all possible trajectories. The approach uses a Variational Gaussian Mixture Model, where the density function of future trajectories is determined by computing the conditional distribution of future trajectory snippets given past trajectory snippets. The evaluation of this algorithm showed promising prediction results. A more general approach for scene understanding using trajectory clustering can be found in [10], where a 3-stage hierarchical learning process is implemented. By using video data, motion patterns are predicted and abnormalities of behavior are detected. For a more extensive survey see [11] [12].

The main contribution of our paper is the realization of a lean and computationally efficient method which provides a distribution of trajectories instead of a countable set of hypotheses even for highly unbalanced data. This extends former approaches like [5] ,[13], [6] where only the prediction of the behavior class was in the focus of interest. Unlike [9] the distribution of the future position does not only depend on the past trajectory but on the actual situation the vehicle is faced with. This also distinguishes from [8], where the hypotheses space was based on the geometric configuration of the vehicle within the road.

III. Environment Model

Our employed features are divided into two groups. One part consists of the lateral state of the vehicle, e.g. the distance to the left and right marking and the lateral velocity. The second part is defined by relations to the surrounding vehicles like the relative velocity of the preceding vehicle or the distance of the vehicle back on the left lane. In our approach the number of vehicles which can influence the behavior of another arbitrary vehicle is limited to a maximum of 8 due to the geometric configuration, see Fig. 2. All features are defined in a curvilinear coordinate system along

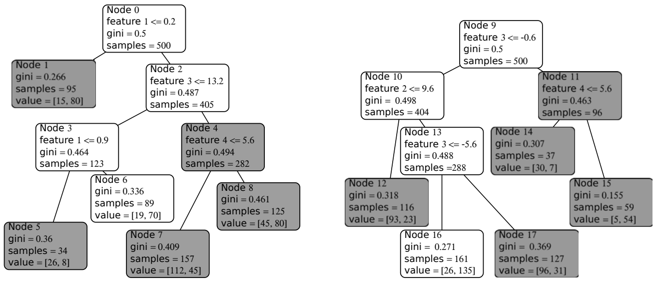


Fig. 3. Simple example of a Random Decision Forest, which is an ensemble of multiple decision trees. The decision process for an example dataset $d = [3.2, 9.7, -7.1, 0.2]$ is highlighted in white.

the curvature of the road. For a more detailed description of the model see [1].

IV. Gating Functions

In a Mixture of Experts approach the Gating Nodes are responsible for computing the weight of each of the results of the experts to the final result. The problem which has to be solved in this case is a simple classification problem, where the probability of each class corresponds to the weight of each expert. Due to the desired prediction horizon of 5s and our large feature set [1] we cannot assume independence between the features. In addition, our dataset is highly affected by noise and outliers. In order to solve this classification problem we propose the use of a Random Forest [14], which have gained popularity because of their good classification performance even for high dimensional data. The method is an ensemble of several Decision Tree Classifiers. Each Decision Tree of a Random Forest (as shown in Fig. 3) splits at each stage all presented samples into two groups based on a threshold of a specific feature. We have chosen Random Forests in this paper because of their fast learning and prediction process, their easy parallelizability, their high accuracy, and the property that they are white box models, meaning that each result can be retraced [15].

A. Training of Random Forests

In the usual training process Decision Trees are stepwise generated as follows: In each step a subset of the training data is randomly selected. For n randomly selected features the best threshold to split all samples in node q into two parts, defined by the lowest Gini impurity $H(X_q^g)$ is calculated:

$$H(X_q^g) = \sum_m P_{q,m}(1 - P_{q,m}) \tag{1}$$

where n is a parameter which is constant over the whole Random Forest and has to be lower than the number of all features. where $P_{q,m}$ is the probability of maneuver m in node q, see (2). Out of these thresholds the split with the lowest $H(X_q^g)$ is selected as a new split. This splitting step has to be repeated until the maximum tree depth is reached or there are not enough remaining samples for a further split, see Fig. 3 for an example consisting of two trees. For a more detailed discussion, see [14].

B. Classification using Random Forests

For each single tree of the Random Forest the probabilistic prediction result for a maneuver class m can be found as the probability for this maneuver in the corresponding leaf node q as $P_{q,m}$, see (2). The final probability estimates P_m of the Random Forest are computed by averaging over all trees [15]. This stands in contrast with the original publication of Breiman [14], where the predicted class is chosen by the maximum number of votes of the single trees.

$$P_{q,m} = \frac{N_{q,m}}{N_q} = \frac{1}{N_q} * \sum_{x_i \in R_q} I(y_i = m) \tag{2}$$

The probability $P_{q,m}$ of maneuver m in a node q can be calculated as the proportion of $N_{q,m}$ to N_q. Whereby $N_{q,m}$ is calculated as the sum over all samples $x_i \in R_q$ with the label y_i belonging to maneuver m. A sample is an element of a data-region R_q if it fulfills all split criterions leading to node q. N_q in this context describes the overall number of samples in node q,see [15]. I denotes the identity function in this context. The probability P_m for a current maneuver m can be straightforward calculated by:

$$P_m = \frac{1}{|Q|} \sum_{q \in Q} P_{q,m} \tag{3}$$

whereby Q denotes the set of the leaf nodes for this sample.

V. Expert Functions

The task of the Expert Functions is the computation of lateral trajectory distributions. By having a set of maneuver classes $M = \{Flw, LcL, LcR\}$, we have to train one expert for each maneuver class m. Each expert computes a distribution of possible trajectories for one maneuver class $m \in M$, which are then weighted using the output of the gating functions. To solve the problem of the estimation of a distribution of future positions we use conditional distributions of Gaussian Mixtures. We formerly presented the algorithmic foundations of this regression method more detailed in [12]. The future positions of a vehicle in the presented approach are conditioned by the measured lateral state. The corresponding density function $p_{m,\supset}$ is defined as:

$$p_{m,\supset}(X^e) = \sum_{k=1}^{n} w_{k,m} \mathcal{N}(\mu_{k,m}, \Sigma_{k,m}, X^e) \tag{4}$$

where $w_{k,m}$ is the weight, $\mu_{k,m} \in \mathbb{R}^5$ the mean, and $\Sigma_{k,m} \in \mathbb{R}^{5\times5}$ the covariance matrix of each of the n Gaussian components of the Mixture. These parameters will be denoted as $\Theta_{k,m}$ in the following:

$$\Theta_{k,m} = \quad (w_{k,m}, \quad , \mu_{k,m}, \quad \Sigma_{k,m}) \tag{5}$$

with the constraint:

$$\sum_{k=1}^{n} w_{k,m} = 1. \tag{6}$$

The vector X^e denoted above consists of the following features:

$$X^e = \quad (pos_y^t \quad t \quad d_{cl}^0 \quad d_{req}^0 \quad v_y^0) \tag{7}$$

where the future lateral position pos_y^t is the output dimension o. The input dimensions i are the prediction time-step t and the distance d_{cl}^0 to the center of the lane a vehicle is assigned to. On top we use the distance to the center of the lane which is the destination of the maneuver d_{req}^0, f.e. the distance to center of the right neighbor lane for the maneuver class LcR. The measured lateral velocity v_y^0 is the last remaining input.

To compute conditional distributions, the n estimated covariance matrices have to be decomposed into their i input- and o output-dimension. We can compute the parameters of the output Gaussian Mixture distribution, conditioned by an input data vector I as follows:

$$\Sigma_{k,m,o|i} = \Sigma_{k,m,o} - \Sigma_{k,m,o,i}\Sigma_{k,m,i}^{-1}\Sigma_{k,m,i,o} \tag{8}$$

$$\mu_{k,m,o|i} = \mu_{k,m,o} + \Sigma_{k,m,o,i}\Sigma_{k,m,i}^{-1}(I - \mu_{k,m,i}) \tag{9}$$

$$w_{k,m|i} = \frac{w_{k,m}p(I|\mathcal{N}(\mu_{k,m}, \Sigma_{k,m}))}{\sum_{n=1}^{k} w_n p(I|\mathcal{N}(\mu_n, \Sigma_n))} \tag{10}$$

In this paper we used only pos_y as output for the evaluation, and the remaining dimensions as input

$$p_m(pos_y) = p_{m,\supset|i}. \tag{11}$$

The last remaining problem in this context is the estimation of the distribution, and therefore the selection of the *correct* number of components for the Expectation Maximization process. We propose the use of the estimate of the structural risk [16], which has proven to be robust in a former publication [12].

VI. Mixture of Experts

Multiple variants of the Mixture of Experts approach are popular, f.e. Hierarchical Mixture of Experts [17], Mixture of Experts for adaptive Kalman-Filters [18], or straightforward Mixtures of Experts of classification or regression models [19]. All of those methods have one main idea in common: They want to ensure that local information in the data is not *optimized out* by a global optimization process. Therefore they are a solution for highly unbalanced classification problems and regression applications where local information in different parts of the dataset should be maintained. We split the dataset along the maneuver classes $M = \{Flw, LcL, LcR\}$, where Flw denotes lane-following, LcL lane-change to the left and LcR lane-change to the right situations. Using the probability estimates P_m for each class (see (3)), which are computed by the algorithm presented in Sec.IV and the results of the experts, which are n parameters of a Gaussian Mixture distribution Θ_t^m (see (5)) at a future time-step t for a specific maneuver m, the combination to the overall result is straightforward, see Fig. 4. We compute the weight vectors for each maneuver m:

$$\mathbf{w_m} = P_m * \begin{pmatrix} w_{1,m} \\ ... \\ w_{n,m} \end{pmatrix} \tag{12}$$

and concetenate them to the overall weight $\mathbf{w}$. The mean vector $\boldsymbol{\mu}$ of the final distribution is computed in the same manner:

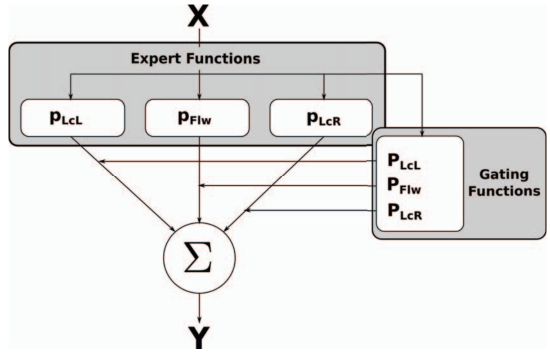


Fig. 4. Schematic diagram of the implemented Mixture of Expert approach. The input for the expert and gating nodes is the feature vector X. The results of each of the experts is combined using the probability estimates of the gating functions into the output Y

$$\mathbf{w} = \begin{pmatrix} \mathbf{w_{LcL}} \\ \mathbf{w_{Flw}} \\ \mathbf{w_{LcL}} \end{pmatrix}, \ \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_{LcL} \\ \boldsymbol{\mu}_{Flw} \\ \boldsymbol{\mu}_{LcR} \end{pmatrix} \tag{13}$$

where $\boldsymbol{\mu_m}$ denotes the vector of all means $\mu_{k,m}$ of one maneuver class m. The overall vector of all covariance matrices is Σ is aggregated in the same manner as $\boldsymbol{\mu}$.

VII. Experiments

We base our investigations on a dataset which is collected in real traffic by our system vehicle equipped with a 360° field of view sensor setup. The vehicles which are used for prediction are measured using an automotive stereo-camera, which provides a high-quality lateral resolution. In comparison the features which are derived from the relations to the preceding and following vehicle are measured using radar sensors. When it comes to the representation of the road, we used an accurate digital map. All sensors which are used in our experiments are standard automotive sensors. We split our investigations into three parts. The classification performance of the gating functions in VII-A, the prediction accuracy of the experts in VII-B given a perfect gating function and the overall performance of the proposed algorithm, see VII-C.

For training and the evaluation, each sample of our database was assigned to a maneuver-class $m \in \{Flw, LcL, LcR\}$. Within this process we divided the dataset into situations. Each situation consists of only one maneuver-class and one vehicle but multiple samples. The time-horizon, which we used for the labeling was defined for both lane-change classes as the time-interval $0 - 5s$ before the vehicle is assigned to the neighbor lane.

A. Evaluation of the gating nodes

For the training and evaluation process, a class-balanced training-database with 4197 samples was generated using the method of random undersampling. For the evaluation we used a *leave one out cross-validation* strategy, where in each fold one situation was left out for training. This technique ensures that all samples from one situation are either in the training or in the test-dataset, which otherwise may cause too optimistic results. To ensure the best possible performance of a Random Forest, its parameters $num_e = 28$ which denotes the number of trees in the Random Forest and the parameter $max_d = 5$ which denotes the maximum number of splits have to be chosen carefully. We calculated them using a 2D-Grid Search. The remaining parameter f_r which denotes the number of features, taken into account for a new split in the training process, is chosen according to [14], as the square-root of the total number of features. The features which are used for the classification problem are chosen in the training process, see Tab. I. The ROC-Curve (Receiver Operating Characteristic) for each class was computed based on the probability estimates of the decision tree, for the result see Fig. 7a. This result does not achieve the theoretical optimum of an AUC(Area Under the Curve) of 1. Nevertheless it is still considerable, that round about 65% of the lane-change maneuvers to the left side can be detected in a time-interval 5s before the lane-assignment changes with a false alarm rate of only 10%. To get a deeper insight into the properties of the approach, we visualized the prediction time via a threshold for a one against all evaluation. By increasing the threshold, the frequency of the true positive decreases. However we are not only interested in the number of true positives, but also in the time-horizon when a lanechange can be detected.

B. Evaluation of the experts

We chose the training data for the individual experts by selecting the data according to a winner-takes-all strategy by the probability estimates of the gating-nodes. For the evaluation of each of the maneuvers we calculated the prediction performance by measuring the Mean Squared Error between prediction and a pseudo ground-truth. To evaluate only the Expert Nodes, we did not take information into account which is generated by the Gating Nodes for evaluation purposes. Instead we used the label information to choose the right expert. Again, as in the evaluation of the gating nodes, we were interested in the prediction performance vs. the prediction horizon, therefore we visualized the absolute error vs. the time horizon using a boxplot for every expert. Please see Fig. 8a,8b,8c for the results. As can be seen, the results for the lane-change classes LcR and LcL are more noisy compared to the Flw class. Reasons for this can be seen in the fact, that only 1% of the data contains lane-change trajectories. Even though it is remarkable in this context, that the median error in none of the experts, at no time-horizon, exceeds $0.2m$.

C. Overall Performance

When it comes to the overall performance, which is evaluated in the same way as in VII-B, one can see that the error of the Mixture of Experts approach (see Fig. 5) is strongly dominated by the results of the Flw class, which containts 99% of the data (see Fig. 8). However the overall prediction results are quite precise. At no time-step of the prediction horizon, the median error exceeds $0.2m$. This result seems to be quite promising for the application in long-term risk assessment methods.

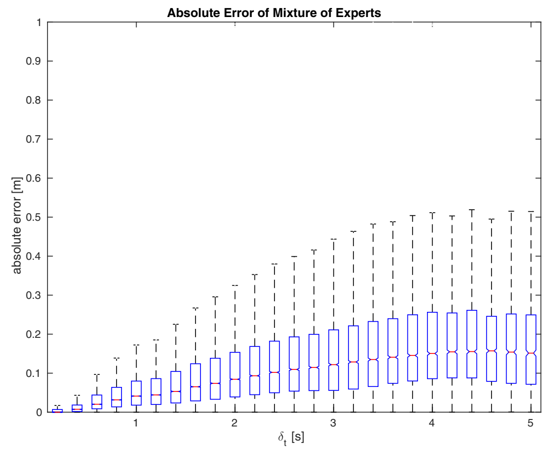


Fig. 5. Absolute error of the predicted mean of the Mixture of Experts approach vs. the ground-truth. The whiskers denote approximately 99.3% coverage of the data, and the box approximately 50% coverage, assuming normal distributed errors.

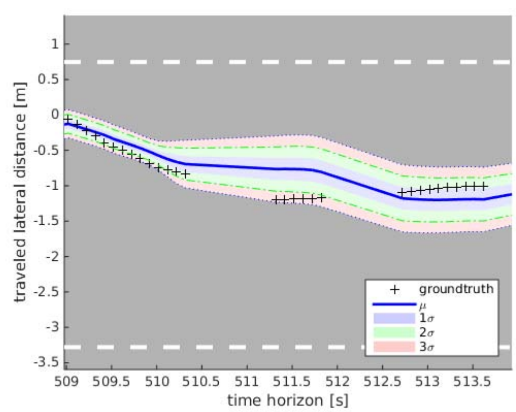


Fig. 6. Example trajectory predicted by our system. The blue line corresponds to the predicted mean, the black symbols to the measured ground-truth and the shaded areas denote the confidence regions as described in the legend. For the visualization, we fused the Gaussian-Mixture distribution to an unimodal Gaussian.

VIII. Conclusions

This paper presents a novel approach for the probabilistic prediction of lane-change maneuvers. We showed that a large part of lane-changing vehicles can be detected seconds before the lane assignment changes. In combination with the probabilistic prediction of future lateral positions, this method provides data which is needed to give an autonomous vehicle more foresight compared to current approaches. We showed, that the estimates of future position provided by the Mixture of Expert approach have an median error of less than 0.2m for prediction horizons up to 5 seconds. Further work will be about making the classification method in the gating node more robust. On top a calibration of the estimated probabilities [20] could help to improve the performance of our algorithm.

References

[1] J. Schlechtriemen, A. Wedel, J. Hillenbrand, G. Breuel, and K.-D. Kuhnert. A lane change detection approach using feature ranking with maximized predictive power. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 108–114. IEEE, 2014.

[2] G. King and L. Zeng. Logistic regression in rare events data. *Political analysis*, 9(2):137–163, 2001.

[3] A. Estabrooks and N. Japkowicz. A mixture-of-experts framework for learning from imbalanced data sets. In *Advances in Intelligent Data Analysis*, pages 34–43. Springer, 2001.

[4] R. Jacobs, M. Jordan, S. Nowlan, and G. Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[5] P. Kumar, M. Perrollaz, S. Lefèvre, and C. Laugier. Learning-based approach for online lane change intention prediction. In *IEEE Intelligent Vehicles Symposium*, Gold Coast, Australie, 2013.

[6] R. Graf, H. Deusch, M. Fritzsche, and K. Dietmayer. A learning concept for behavior prediction in traffic situations. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 672–677, 2013.

[7] S. Bonnin, T. H Weisswange, F. Kummert, and J. Schmuedderich. General behavior prediction by a combination of scenario-specific models. *IEEE transactions on intelligent transportation systems*, 15(4):1478–1488, 2014.

[8] D. Petrich, T. Dang, D. Kasper, G. Breuel, and C. Stiller. Map-based long term motion prediction for vehicles in traffic environments. In *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, pages 2166–2172, Oct 2013.

[9] J. Wiest, M. Hoffken, U. Kressel, and K. Dietmayer. Probabilistic trajectory prediction with gaussian mixture models. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 141–146, June 2012.

[10] B.T. Morris and M.M. Trivedi. Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(11):2287–2301, Nov 2011.

[11] S. Lefvre, D. Vasquez, and C. Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH Journal*, 1(1), 2014.

[12] J. Schlechtriemen, A. Wedel, G. Breuel, and K.-D. Kuhnert. A probabilistic long term prediction approach for highway scenarios. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 732–738. IEEE, 2014.

[13] D. Kasper, G. Weidl, Thao Dang, G. Breuel, A. Tamke, and W. Rosenstiel. Object-oriented bayesian networks for detection of lane change maneuvers. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 673–678, 2011.

[14] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

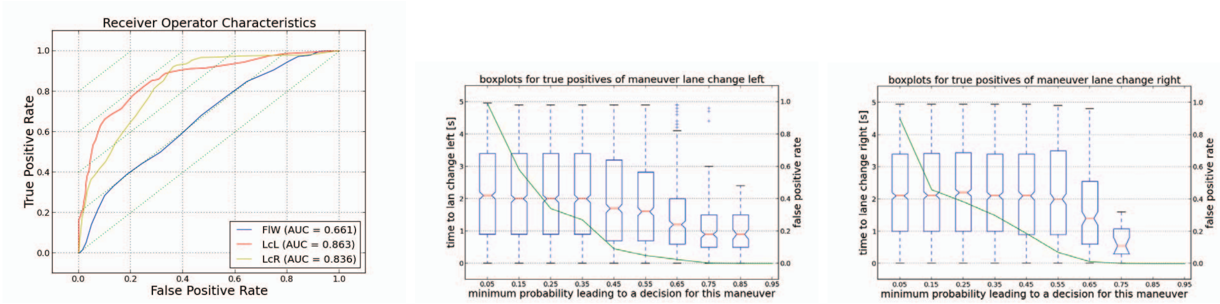
[16] L. Wang and J.Feng. Learning gaussian mixture models by structural risk minimization. In *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, volume 8, pages 4858–4863 Vol. 8, Aug 2005.

[17] Michael I Jordan and Robert A Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214, 1994.

[18] W. Chaer, R. Bishop, and J. Ghosh. A mixture-of-experts framework for adaptive kalman filtering. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 27(3):452–464, 1997.

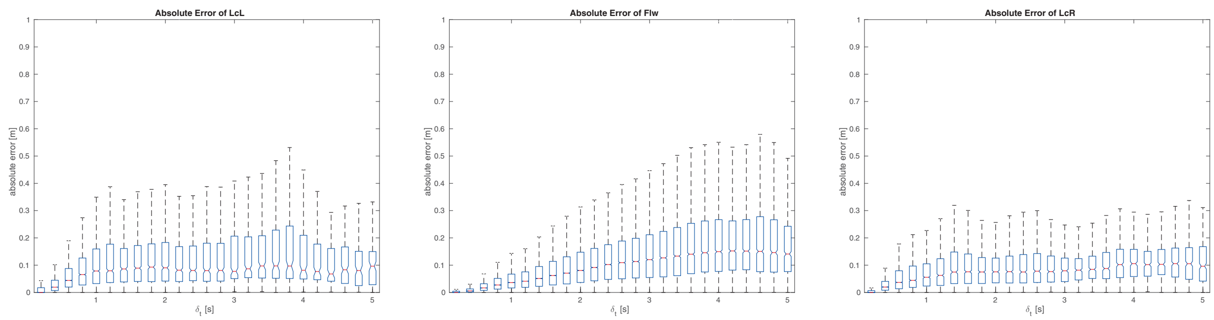
[19] J. Kwok. Support vector mixture for classification and regression problems. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, volume 1, pages 255–258. IEEE, 1998.

[20] B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699. ACM, 2002.



(a) Receiver Operating Characteristic of P_m (b) P_{LcL} vs maneuver time (c) P_{LcR} vs maneuver time

Fig. 7. Evaluation of the outputs P_{LcL} (red), P_{LcR} (yellow) and P_{Flw} (blue) of the Gating Node. In Fig. 7a you can see the Receiver operating characteristic of the Gating Nodes if used as classifier. The classification task is recognizing the intent of a vehicle 5s before its lane assignment changes. Because not only the value of the True Positive Rate is in the focus of interest, but also how early the lane change can be recognized, we visualized this property vs. a decision threshold in Fig. 7b and 7c. The corresponding value on the y-axis corresponds to the time at which this decision threshold is overrun for the first time and never undershot until the vehicle has changed its lane-assignment. The green line in this context visualizes the false positive rate.



(a) LcL (b) Flw (c) LcR

Fig. 8. Evaluation of the experts. As one can expect the precision of the position estimates decreases with an increasing prediction time horizon. The result of the noisy curves in Fig. 8c and Fig. 8a is due to having not enough samples in this data-region

TABLE I

FEATURESET WICH WAS DERIVED IN THE TRAINING PROCESS OF THE RANDOM DECISION FORREST

f	description	f	decsription
car_l	boolean if there is a car on the left side of the observed vehicle	$v_{b,l}^{rel}$	relative speed to the object on the back left lane
car_r	boolean if there is a car on the right side of the observed vehicle	$v_{b,r}^{rel}$	relative speed to the object on the back right lane
$nlane_l$	number of lanes on the left of the observed vehicle	a_f^{rel}	relative acceleration to the object in front
$nlane_r$	number of lanes on the right of the observed vehicle	$a_{f,l}^{rel}$	relative speed to the object on the front left lane
t_l^m	boolean if marking left is dashed	$a_{f,r}^{rel}$	relative acceleration to the object on the front right lane
t_r^m	boolean if marking right is dashed	a_y^{req}	required lateral acceleration to stay in the current lane
v_y	lateral velocity of the observed vehicle	$a_{pos,l}^{req}$	req. acc. for collision avoidance for a lanchange left
v_y^{smo}	smoothed v_y	$a_{pos,r}^{req}$	req. acc. for collision avoidance for a lanchange right
t_{LcL}	predicted time to a lanechange left using a c.a. assumption	$a_{neg,l}^{req}$	req. decc. for collision avoidance for a lanchange left
t_{LcR}	predicted time to a lanechange right using a c.a. assumption	$a_{neg,r}^{req}$	req. decc. for collision avoidance for a lanchange right
d_{cl}	distance between vehicle center and assigned centerline	ttc_f	time to collision with the object in front
d_y^{ml}	distance between vehicle center and the left marking	ttc_b	time to collision with the object in the back
d_y^{mr}	distance between vehicle center and the right marking	$ttc_{f,l}$	time to collision with the object on the front left lane
x_f^{rel}	relative distance to the object in front	$ttc_{f,r}$	time to collision with the object on the front right lane
x_b^{rel}	relative distance to the object in the back	$ttc_{b,l}$	time to collision with the object on the back left lane
$x_{f,l}^{rel}$	relative distance to the object on the front left lane	$ttc_{b,r}$	time to collision with the object on the back right lane
$x_{f,r}^{rel}$	relative distance to the object on the front right lane	τ_f	timegap to the object in front
$x_{b,l}^{rel}$	relative distance to the object on the back left lane	τ_b	timegap to the object in the back
$x_{b,r}^{rel}$	relative distance to the object on the back right lane	$\tau_{f,l}$	timegap to the object on the front left lane
x_f^{rel}	relative distance to the object in front	$\tau_{f,r}$	timegap to the object on the front right lane
$x_{f,l}^{rel}$	relative distance to the object on the front left	$\tau_{b,l}$	timegap to the object on the back left lane
v_f^{rel}	relative velocity to the object in front	$\tau_{b,r}$	timegap to the object on the back right lane
v_b^{rel}	relative velocity to the object in the back	v_f^y	lateral speed of the object in front in lane coords
$v_{f,l}^{rel}$	relative speed to the object on the front left lane	$v_{f,l}^y$	lateral speed of the object in front left in lane coords