2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Learning to Predict Lane Changes in Highway Scenarios Using Dynamic Filters On a Generic Traffic Representation

Joonatan Mänttäri[1], John Folkesson[1], Erik Ward[1]

Abstract— **In highway driving scenarios it is important for highly automated driving systems to be able to recognize and predict the intended maneuvers of other drivers in order to make robust and informed decisions. Many methods utilize the current kinematics of vehicles to make these predictions, but it is possible to examine the relations between vehicles as well to gain more information about the traffic scene and make more accurate predictions. The work presented in this paper proposes a novel method of predicting lane change maneuvers in highway scenarios using deep learning and a generic visual representation of the traffic scene. Experimental results suggest that by operating on the visual representation, the spacial relations between arbitrary vehicles can be captured by our method and used for more informed predictions without the need for explicit dynamic or driver interaction models. The proposed method is evaluated on highway driving scenarios using the Interstate-80 dataset and compared to a kinematics based prediction model, with results showing that the proposed method produces more robust predictions across the prediction horizon than the comparison model.**

I. INTRODUCTION

The emergence of Advanced Driver Assistance Systems (ADAS) and development of autonomous driving has led to significant steps in improving the safety of traffic participants by reducing human error and relying on intelligent systems to handle dangerous driving situations. In order for these systems to be pro-active and make informed decisions, it is necessary for their planners to be able to recognize and predict the future intentions and maneuvers of other agents in traffic [1]. One situation where this need arises is the planning of safe overtake maneuvers in highway driving scenarios. When planning a trajectory to overtake a vehicle, knowing if and when other vehicles intend to change lanes during the planning horizon is of great importance when calculating a safe trajectory. Some methods for addressing this problem utilize the kinematics of the other traffic agents together with information about road and lane geometry in order to recognize and predict lane change maneuvers using a probabilistic model [2], [3], [4], [5], [6], [7]. However these methods often do not take into account the scene information available in the form of the relative position of vehicles to each other. Although methods which aim to capture this information and utilize it for more robust predictions exist, they often place restrictions on the number of relations considered and the configuration of the vehicles [1], [8], or encounter difficulties with scalability or accuracy when increasing the number of traffic participants in the scene [9], [10]. More advanced methods utilizing Recurrent Neural Networks (RNNs) [11], deep learning, and reinforcement learning exist as well [12], but often aim to replicate a human driver's ability to control the ego vehicle in highway scenarios.

In this work we instead propose a novel deep learning-based method for predicting vehicle behavior in highway scenarios that is straightforward while simultaneously utilizing the established strengths of deep learning. By transforming the vehicle tracking information available into a generic image representation and learning how these image sequences evolve by applying an image prediction method based on RCNNs [13], our method is able to predict lane change maneuvers using not only the poses of the vehicles over time but also their positional relation to other vehicles. Convolutional nets capture increasingly global relations in image space by a hierarchy of convolutional operations, therefore we can examine spacial relationships between an increasingly large number of vehicles without an increase in computational complexity or the need to build dynamic models, driver interaction models, or driver reaction models over all possible interacting agents. Although our proposed method could be applicable to a range of traffic scenarios, we begin by exploring our novel method's capability for predicting lane change maneuvers in highway scenarios. As the experimental results for this less complex scenario are promising, it suggests that our novel method could be extended for other types of traffic scenes and participants.

A. *Related work*

The earlier work in recognizing and predicting driver intent most often relied on vehicle kinematics. Most methods incorporate uncertainty in these settings, where more direct approaches train a Naive Bayesian Classifier to recognize lane change maneuvers (LCMs) based on the vehicle position and velocities [3] relative to lane markers. Similar work was done in [2], where the current means and covariance matrices for these metrics were used to calculate a chi-squared distributed probability for lane changes that was then thresholded to determine if the vehicle was about to perform an LCM. After the system recognizes an LCM, it predicts the path that the vehicle would take during the maneuver as a combination of a quintic polynomial path and a Constant Yaw Rate and Acceleration (CYRA) based trajectory. While these methods can perform well for short prediction horizons, they often fail to predict LCMs further in the future due to lack of kinematic clues. In [1] Weidl et al. utilize Dynamic Bayesian Networks to implement a

[1]Robotics, Perception and Learning Lab, KTH, Stockholm, Sweden (e-mail: manttari@kth.se, johnf@kth.se, eward@kth.se)

probabilistic classifier for lane change maneuvers and extend their previous work to include information about the relative distance and velocity to the vehicle in front and not just vehicle kinematics, thereby increasing the horizon in which they can predict LCMs due to the relative difference in speed between vehicles. However, this approach ignores much of the spatial relationships present and could misinterpret certain situations, for example a vehicle not being able to execute an overtake due to car approaching from behind in the left lane. The quantile regression based prediction method by Wissing et al. addresses this problem by including the position and relative velocity of vehicles both behind and in front of the vehicle of interest not only in the same lane but the adjacent lanes as well, taking the consideration of the traffic scene one step further [7].

The method by Jordan et al. [9] aims to retain all of the relational data by implementing a grid-based approach where detected objects are modeled as obstacle pixels in the grid, removing the restriction of defining specific spatial vehicle-to-vehicle relations. They utilize non-negative matrix factorization with $l1$-norm regularization to obtain a small number of representative basis vectors to describe a scene. To model temporal dependencies and to make predictions on the evolution of a traffic scene a second order auto-regressive model is used, applied to the sparse code representation. Although capable of achieving prediction horizons of more than three seconds, the accuracy of the approach was not remarkably higher than that of a constant velocity model. While sparse coding and auto-regressive models have clear computational advantages to recurrent deep networks, parallel processing hardware allows us to efficiently use vastly more expressive models. The work presented by Lawitsky et al. predicts maneuvers in highway scenarios by modeling them probabilistically as a result of both driver intent and risk evaluation [10]. Each evaluation takes into account all possible maneuvers taking place in the current and adjacent lanes by the other traffic participants to ensure robust predictions. While this approach retains much of the relational information of the vehicles in the scene, the method discretizes the possible action space into 9 maneuvers and its computational complexity scales quadratically with the number of considered traffic participants.

Deep learning has received a revival within recent years due to emergence of powerful parallel-architecture hardware and the availability of massive amounts of training data. Although deep learning has its greatest successes in the computer vision field, it has been successfully applied to other fields as well. An example of this is the work done by Kuefler et al. [12] which uses a combination of the traditional vehicle kinematics as well as the vehicle's relation to neighboring vehicles to train a deep neural network (DNN) that functions as the policy approximator in their reinforcement learning approach. This method is primarily meant to mimic the behavior of a human driver controlling the ego vehicle. While the method does take into account the distance to neighboring vehicles, portions of the scene information is lost as they only take into account the sensory output of 20 LIDAR-like beams around the ego vehicle. Further work utilizing a recurrent DNN has also been explored in [14] to predict the intended maneuver of a human controlled ego vehicle in intersections. Recurrent DNNs together with occupancy grids and shared latent representations have been used in [15] to predict how pedestrians move in crowded areas. Each pedestrian's future trajectory is predicted by a recurrent network using long short term memory (LSTM) to learn a bivariate Gaussian distribution where future trajectories are sampled from. By sharing the latent representation between pedestrians which are in each others occupancy grids, the model is able to account for the influence that different positional configurations have to the predicted path.

The work in this paper instead aims to use deep learning in the setting of one of its greatest strengths and have it operate on image prediction. A multitude of deep learning architectures exist for image sequence prediction [16], [17], but an especially compact and effective architecture is the dynamic filter architecture [13]. In this work we will use an architecture derived from the dynamic filter network and train it to predict the behavior of vehicles in highway traffic scenarios. The current traffic scene is converted to a generic visual representation described in Section II and then fed to the network for prediction, afterwards the vehicle positional data is extracted from the resulting predicted images. This method of prediction differs from the kinematics based approaches as we do not have an explicit model for the vehicle dynamics or interactions, but instead learn how images representing the traffic scene evolve statistically. The strength of operating in the space of the visual representation is the possibility to take into account arbitrary spacial relationships between the vehicles, independent of the exact number vehicles in the vicinity of a vehicle of interest, or their exact spacial configuration. It also provides an additional advantage; although the method presented is evaluated in the setting of LCMs in highway scenarios, it could be trained to predict vehicle behavior in other scenarios due to the representation used by the solution. Our main contributions are:

- Construction of generic visual representation of traffic scenes.
- Novel method for prediction of future vehicle positions using the generic visual representation.
- Evaluation of proposed method and comparison to kinematics based solution using the Interstate-80 dataset.

The remainder of the paper is structured as follows: Section II presents an overview of the network architecture used, as well as how vehicle tracking data is converted to be used with the network. Experiments to evaluate the proposed method are described in Section III, with their results shown in Section IV. Finally, Section V presents the conclusions and future work.

II. MANEUVER PREDICTION WITH DEEP NEURAL NETWORK

The proposed method utilizes a deep learning approach to predict the trajectories of vehicles in a traffic scenario by converting vehicle tracking data to a generic image representation. The following subsection explains the representation used and how to convert vehicle track data to it, while subsection *B* explains how the generic representation is used to generate predictions.

A. Generic Visual Representation

At a rate of 10 Hz, we consider all the vehicle tracks present in a 50m x 50m square with the ego vehicle position as the center. This square area operates as the sensor range of the vehicle, which is assumed to be at least this large. The vehicle positions are then rendered relative to the ego vehicle on a 128x128 single-channel image as seen in Fig. 3, using their assumed sizes and translating the image so that the leftmost lane starts at the same position in each image. The specified image size of 128x128 was chosen in order to maximize the positional accuracy of the predictions, while keeping the memory requirements of the network within the available resources. Additional images are also rendered that show only the position of each individual vehicle, which simplifies analysis of individual agents later. Utilizing this generic image set as a representation of the traffic scene provides us with several advantages:

- We have no need to place restrictions on the exact spacial configuration of vehicles we take into consideration.
- The computational complexity does not change with the number of traffic agents.
- Representation can be extended to include arbitrary traffic agent classes.

Models that take into consideration only adjacent vehicles or specific relationships often need to define the maximum number of agents to look for, which spacial segment they reside in, and their correct order for input into the model. As our model operates on recurrent and convolutional operations on an image, we have no need to define beforehand which configurations are possible or how many vehicles to look for, they are simply present in the visual representation. Also, the same operations are executed each prediction cycle since the visual representation fed to the network does not change in complexity with the number of traffic agents, meaning our method does not scale poorly with increasing traffic participants. Lastly, although we currently only take into consideration passenger vehicles in the current representation, it would be possible to incorporate other traffic agents such as cyclists and pedestrians as different shapes and colors in the representation and evaluate the possibility of learning interactions between these agents as well. Using icons to represent traffic agent positions instead an occupancy grid not only gives us the ability to represent different classes of agents with different types of icons, but also avoids the problem of reliably discerning exactly which cells an agent occupies. While this representation does provide us with the necessary information to conduct predictions, it also entails some loss of information:

- The accuracy of the generated paths is dependent on the resolution used in the images.
- Explicit modeling of the dynamics of the vehicles is lost.
- Performance is dependent on training data.
- Uncertainty of the generated predictions is not modeled in the current representation.

The conversion to a discrete pixel representation means we are not able to output positional values as strictly continuous variables. While the resulting accuracy seems to be enough for predicting LCMs, it could be insufficient depending on the desired use-case. Although pose information over time is represented in the images, the explicit modeling of for example acceleration, yaw-rate or forces on the vehicle is not present, which are useful in predicting trajectories for short time horizons. The current method is also dependent on sufficient training data for being able to predict various interactions, which means that gathering large amounts of data and incorporating it through continuous learning is required for deployment of the proposed method on real systems. Finally, there is no modeling of uncertainty in the current representation, which is a useful metric to take into consideration for many autonomous planners. After conversion to this representation, the traffic scene is sent to the deep neural network to produce the predictions. The specific architecture presented in this work is based on the Dynamic Filter architecture described in [13].

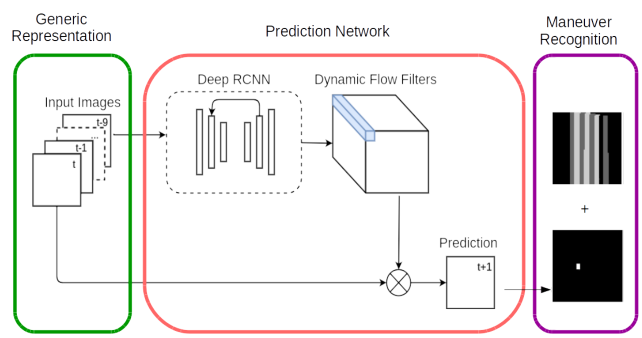

Fig. 1: System overview of prediction method.

B. Prediction Generation

An overview of the prediction system can be seen in Fig. 1. First, the network receives a sequence of $k = 10$ input images corresponding to the data received up until 1 second ago from the current time, t. To produce the prediction for frame F_{t+1}, frames $F_t, F_{t-1}, F_{t-2}, ..., F_{t-(k-1)}$ are sent to the deep Recurrent Convolutional Neural Network (RCNN). The feed-forward operation outputs a dynamic filter, see top right of Fig. 1, which acts as a flow map that is position specific, meaning it contains one flow filter for each position in the input image. This flow filter is then applied to the most recent input frame, F_t, to produce the prediction for timestep $t + 1$. Once the prediction for frame F_{t+1} is complete it is added as the latest image in the input sequence, the oldest input image $F_{t-(k-1)}$ is discarded, and the process is repeated until the desired amount of frames are predicted. Since the least recent input image is discarded at each iteration, after k predictions the network is no long operating on any original input images, only its own predictions. Predictions are executed on the input images that portray all vehicles in the scene, while saving each of the dynamic filter flow maps in between predicted frames. After all flow maps are produced for the desired prediction horizon, they are then re-applied to the images portraying each individual vehicle, producing images that show their predicted trajectory. By looking at the position of these vehicles in the predicted images and comparing it to lane positions, it is possible to extract information such when an LCM is performed. The architecture of the RCNN creating the dynamic filters can be seen in Fig. 2. We modified the original architecture described in [13] by increasing the receptive field in the first two layers and increasing the input sequence length, as our use case had more complex structures to consider and longer desired prediction horizons.

C. Training

The network was trained with pixel-wise cross-entropy loss against the ground truth images, using the ADAM algorithm for gradient descent with a batch size of 10 and an initial learning rate of $7 * 10^{-4}$. At the end of training the validation loss was 0.037 and the training loss was 0.019.

III. EVALUATION

The Interstate-80 dataset [18] was used to create the training and test data for evaluating the proposed method's ability to predict lane changes. The dataset contains three different 15 minute subsets of vehicle tracks in a highway segment of approximately 500 meters recorded by roof mounted cameras, giving positional data of all vehicles at 10 Hz. For these tests, the subset with the least traffic was used (4.00 to 4.15 pm). The data was then split to into scenarios consisting of 20 frames of input data and 20 frames of ground truth data which was used for testing the prediction, resulting in 40 frames for each scenario. The scenarios included the tracks of the current chosen ego vehicle as well as the tracks of the vehicles within its simulated sensor range, which was set to be a 50 meter by 50 meter square around the ego vehicle. This was done as follows:

1) The trajectory data was split into 4-second segments, with the segments starting at all possible offsets of 0.1 seconds.
2) All vehicles in a trajectory data segment were cycled through and chosen as the ego vehicle. If during the last 2 seconds of the scenario a lane change was executed by at least one vehicle within the sensor range of the ego vehicle, it was added to the bank of test scenarios. The IDs of the vehicles performing a lane change were also saved.
3) All test scenarios including more than 14 vehicles within the ego vehicles sensor range were filtered out.

This resulted in 16400 traffic scenarios which after construction were split into training, validation and test sets consisting of 70%, 10%, and 20% of the original set respectively. The split was made before shuffling the scenarios and in such a way that each set would include tracks from a proportional time slice from the first, second and last 5 minutes of the dataset. This way all sets would include the different traffic congestion levels while avoiding test scenarios that are the same as training scenarios but seen from a different ego vehicle perspective. This constructed dataset is available at https://strands.pdc.kth.se/public/I80_RCNN/. The images corresponding to the first 20 frames of the scenario were then fed to the network, which performed the prediction for the following 20 frames. After producing the dynamic filters associated with each prediction, the predicted path of each individual car was produced by applying the dynamic filters to their corresponding individual images. The predicted positions were then compared to the lane geometry to see if and when the vehicle would cross a lane marking, thereby performing an LCM. Three different definitions were evaluated regarding how much of the vehicle width had to cross the lane marking to be defined as a lane change; 10%, 25% and 50%. The results presented in Section IV use the

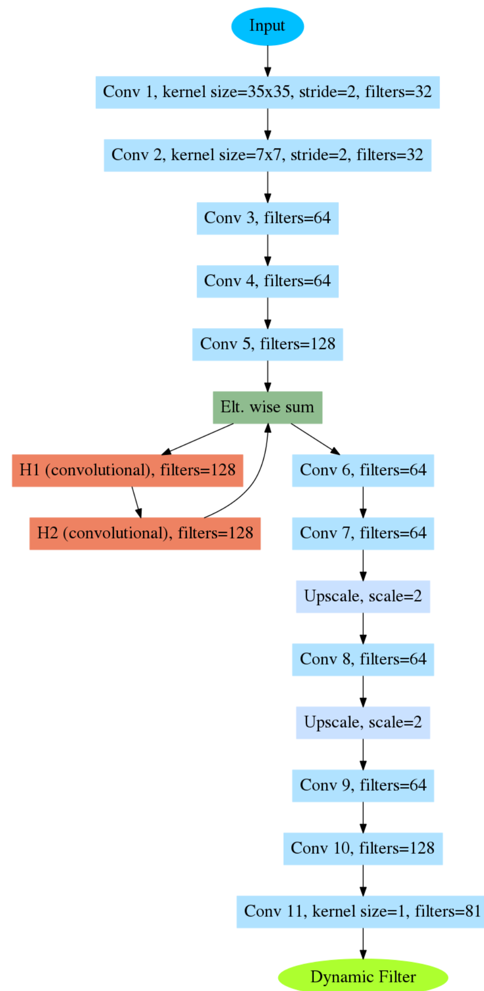

Fig. 2: Network architecture employed to create the dynamic filters for image sequence prediction. All layers have padding set to *"same"*, stride set to (1,1), kernel size set to 5x5, and *Leaky ReLu* as non-linearity unless otherwise stated.

25% definition as it showed the best performance through experimental results.

The results of the proposed method were also compared to a replication of the method described in [2] using the same test scenario data. The method combines two modules, a maneuver recognition module and a trajectory prediction module to predict LCMs. The maneuver recognition module utilizes the means and covariances of the vehicle kinematics and lane geometry to create a Chi-squared distributed value which is thresholded to determine if a vehicle is about to perform an LCM. If an LCM is recognized, the trajectory prediction module then generates the path that the vehicle will take as a combination of a Constant Yaw Rate and Acceleration (CYRA) trajectory and a quintic polynomial using a cost function. Since no values for the feature covariances are given in the Interstate-80 dataset, the tracks were pre-processed by smoothing them with an additive unscented Kalman filter which also provided the covariance values. The method utilizes a parameter that must be experimentally tuned, τ, which is the threshold value for the maneuver recognition. In order to give the comparison method the best possible performance, a search using different values for these parameters was conducted on the test scenario set with the results shown in Section IV. The maneuver recognition module was given the first 20 frames as input and if it classified an LCM within those frames, the trajectory prediction module was executed to evaluate when the vehicle would cross the lane marking. The false positives (FPs) and false negatives (FNs) for this method and the proposed method were also recorded with the results shown in Section IV.

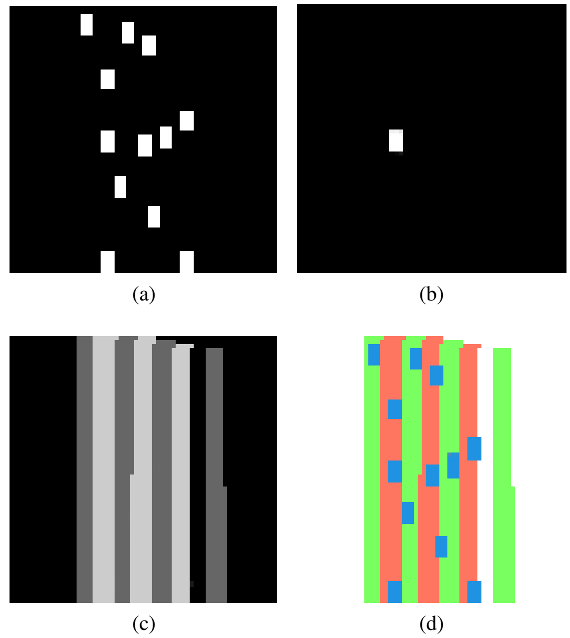

Fig. 3: The visual representations used by the proposed method. Figure a) shows the input used by the prediction network, b) shows an example of an image used for analysis of an individual car, c) shows the lane geometry image used for discerning LCMs, and d) shows a composite image for visualization with all vehicles superimposed on the lanes.

IV. RESULTS

Both the proposed method and the probabilistic method from [2] were evaluated on the test set of the created highway scenarios. The set consisted of 3280 scenes where 4 to 15 vehicles were present and at least one vehicle performed an LCM anywhere from 0.1 to 2 seconds in the future. An example scenario with the predictions of the proposed method can be seen in Fig. 4. Statistical evaluations were performed on both methods by recording the precision and recall, as well as the average error in LCM prediction times along the prediction horizon according to the following equation:

$$AvgError(t) = \frac{\sum_{l \in L_t} |t_{l_{GT}} - t_{l_{pred}}|}{|L_t|} \quad (1)$$

where L_t is the set of all LCMs that occur at time t in the prediction horizon, and $t_{l_{GT}}$, $t_{l_{pred}}$ are the times the LCM happened according to the ground truth and the prediction respectively. As the performance of the comparison method depends on the value for the recognition threshold, τ, the experiments were repeated using different values for this threshold. The precision, recall and F1 scores are summarized in Table I which also shows the corresponding scores for our proposed method. The further in-depth results regarding the comparison method are shown only for $\tau = 2.5$ which gave the highest F1 score, and $\tau = 1.25$ which gave sufficient true positives at later prediction horizons to make a statistical comparison meaningful. The results from this table point to our method generally outperforming the kinematics-based method at predicting lane changes, achieving a higher F1 score when considering all LCMs as well those specifically at 1 and 2 seconds in the future. By choosing a sufficiently high or low value for τ the kinematics-based method is able to achieve higher performance for the short prediction horizon of 0.3 seconds, but is unable to uphold this performance for the remaining span. In order to more closely examine where the strengths of each method are, we also present graphs plotting these scores on the prediction horizon. In figures 6-9 we present the precision, recall, F1 scores, as well as the error in prediction time of when a vehicle has changed lanes, plotted against how far ahead the lane change occurs according to the ground truth. The results in Fig. 8 show that the probabilistic method based on vehicle kinematics excels at predicting the time of the LCM at shorter prediction horizons, if a low enough threshold is used. However, when the LCM occurs further in the future the kinematics based method is no longer able to perform an as accurate prediction. Also, as we can see in Fig. 7, if the threshold τ is set too high, the method is often unable to recognize the maneuver, and no prediction occurs. If the threshold is set low enough, many more LCMs are recognized but at the cost of lowered prediction time accuracy and significantly higher false positives due to vehicles swaying

TABLE I: The precision, recall, and F1 scores for the comparison and proposed method while predicting lane changes made at 0.3, 1.0, and 2.0 seconds in to the future. The scores while taking into account all LCMs are shown in the *All* column.

		t=0.3s			t=1s			t=2s			All		
Method	**Params**	Prec.	Recall	F1	Prec.	Recall	F1	Prec.	Recall	F1	Prec.	Recall	F1
Kinematic	$\tau = 3.75$	0.96	0.55	0.70	0.17	0.13	0.15	0.22	0.04	0.06	0.49	0.22	0.30
	$\tau = 3.00$	0.71	0.82	<u>0.76</u>	0.12	0.26	0.16	0.17	0.08	0.11	0.38	0.41	0.40
	$\tau = 2.50$	0.49	0.87	0.62	0.16	0.42	0.23	0.19	0.15	0.17	0.30	0.52	0.38
	$\tau = 2.00$	0.21	0.87	0.34	0.15	0.62	0.25	0.20	0.24	0.22	0.23	0.62	0.33
	$\tau = 1.25$	0.06	0.91	0.11	0.20	0.75	0.32	0.23	0.51	0.32	0.14	0.76	0.24
Proposed	—	0.59	0.91	0.71	0.66	0.76	<u>0.71</u>	0.32	0.49	<u>0.38</u>	0.66	0.83	<u>0.74</u>

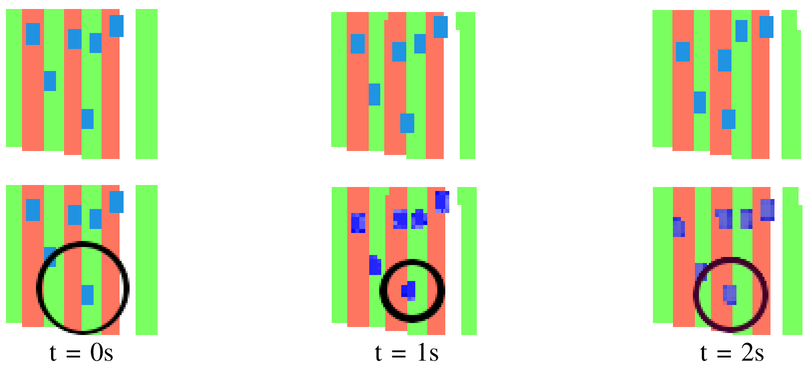

Fig. 4: An example of the predictions made by the proposed method for a scenario. The first row shows the ground truth with the predictions made by the system shown on the row below it. The images are shown at 0, 1 and 2 seconds in to the prediction horizon with a black circle showing the vehicle performing a lane change.

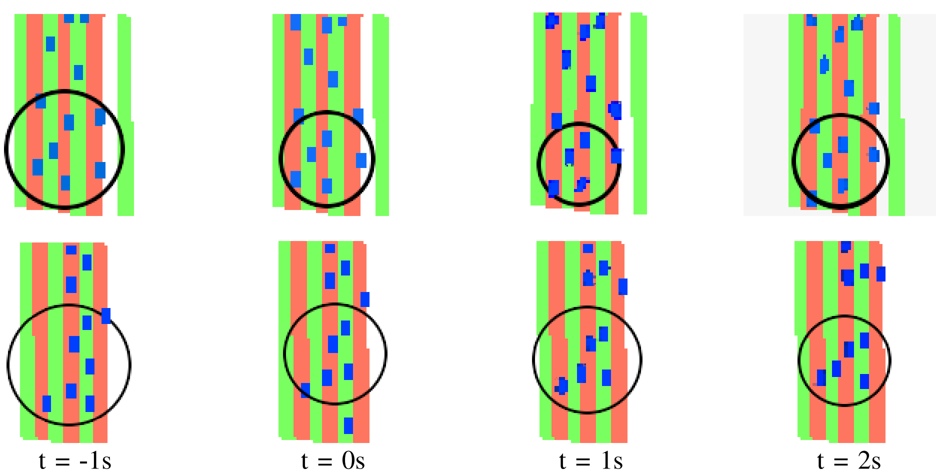

Fig. 5: Two example predictions by the proposed method in scenarios where a vehicle is approaching another from behind. The bottom vehicle has a higher velocity than the one in front effectively closing the distance between them during the second before the start of prediction, making a lane change predictable as it is likely to attempt to overtake the slower car. In the first scenario shown in the top row, the vehicle in the middle bottom of the circle is however not predicted to change lanes due to the car in the left lane blocking its path. In the scenario shown in the second row the vehicle is predicted to change lanes as the lane to the left is free.

in their lanes, as can be seen in Fig. 6. Our proposed method however presents higher precision throughout almost the entire prediction horizon, slightly improved recall w.r.t the kinematics based method using the lowest $\tau = 1.25$, and only moderate increase in prediction time error. We believe these results are a property of the generic representation used by the proposed method. While the kinematics based method relies solely on the vehicles kinematics in relation to the lane, the example scenarios presented in Fig. 5 suggest that the proposed method in addition to this incorporates the current configuration of vehicles into its visual representation. The two rows in the figure show separate scenarios where one vehicle is approaching another from behind due to differences in velocity. In the second scenario shown in the bottom row, we can see how the vehicle shown in the bottom middle decreases its distance to the vehicle in front in the second before the start of prediction. Halfway through the prediction it starts performing a lane change despite no significant indication of lateral movement in the second before the prediction, and two seconds into the prediction it has almost completed the maneuver. A similar scenario is shown in the top row, but here an additional vehicle is positioned in the lane to the left of the vehicle at the bottom middle. The difference in velocity is noted again, and at t=1s the vehicle is close to being predicted to change lanes although this time the slower car in the lane to the left is noticed and the prediction ends with the vehicle slowing down and staying in its lane. The scenario presented in the bottom row also shows a vehicle in the far right lane keeping its velocity due to the its lane being free in front of it. These scenarios suggest that the proposed method is not only capable of learning to predict an LCM that will occur further in the future due to velocity differences between vehicles, but also notice when such a maneuver is not possible due to an occupied lane, giving it an advantage over the kinematics based method that must wait on lateral movement of the vehicle to recognize the LCM. Although the comparison method excels at predictions when this lateral movement is apparent, both methods are still capable of making this prediction while the proposed method provides more robust predictions of LCMs when considering the entire prediction horizon.

V. CONCLUSIONS AND FUTURE WORK

In this work we present a novel approach to predicting lane change maneuvers in highway scenarios using image prediction performed by deep RCNNs. We have shown that a new representation and learning method can be combined to perform as well or better than conventional kinematics based methods. Our method outperforms the baseline probabilistic method at a larger scope of time horizons, with results suggesting it does so by capturing more implicit information about when a lane change will occur from the relationships between vehicles. As the proposed method operates on a generic visual representation, the promise is that with more work the method could be extended to learn how to execute predictions in more complex driving scenarios. As the dataset

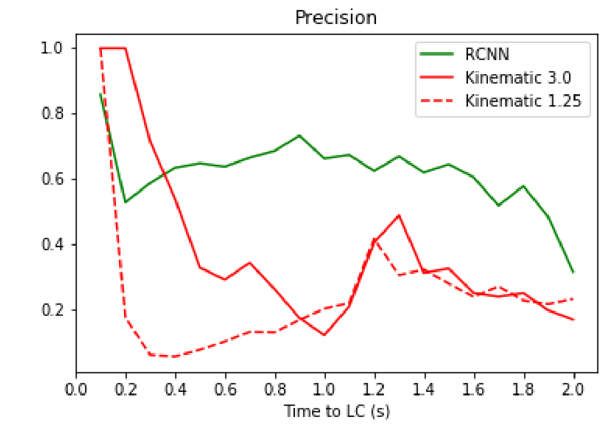

Fig. 6: Precision achieved by the proposed method and the comparison method with two different values for τ.

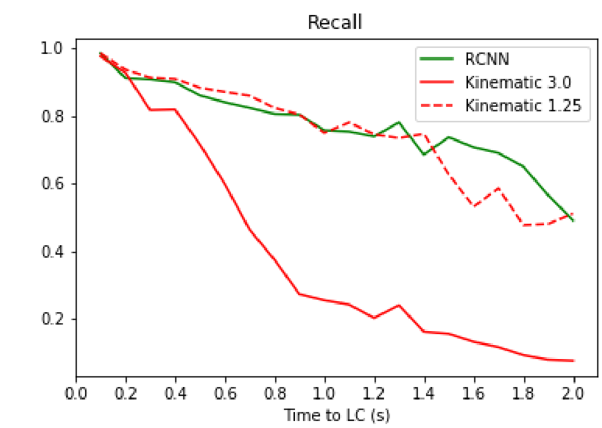

Fig. 7: Recall achieved by the proposed method and the comparison method with two different values for τ.

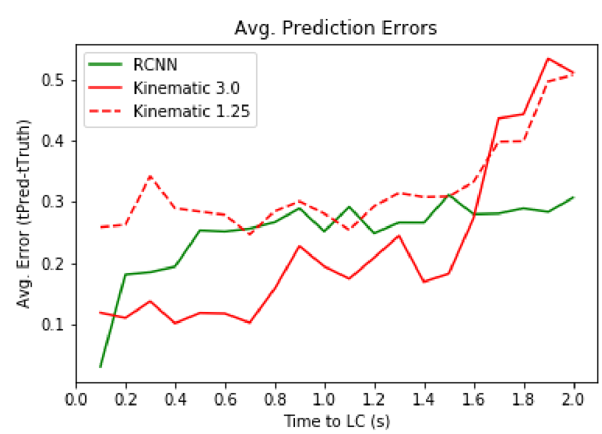

Fig. 8: Average error between ground truth and prediction time for LCMs plotted against time to LCM in prediction horizon.

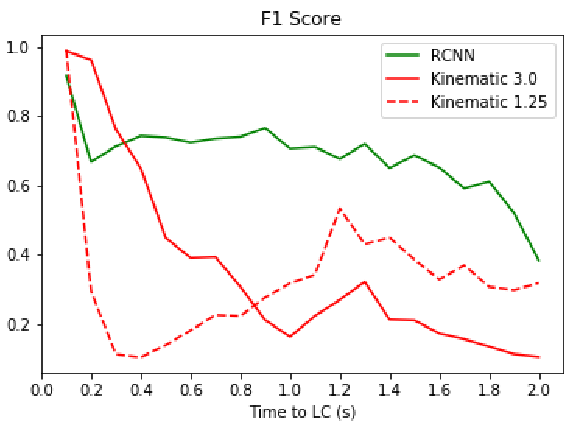

Fig. 9: F1 score achieved by the proposed method and the comparison method with two different values for τ, plotted against time to LCM in prediction horizon.

used in the experiments does not take into account occlusions caused by neighboring vehicles, it would be of interest to examine the effect that these occlusions have on the proposed method in future experiments. Furthermore, although the data used in the experiments is from a real world dataset, it is possible that the vehicle tracking data acquired from on-board sensors of an autonomous vehicle could include noisier readings. The robustness of the proposed method against this noise could be examined by implementation on a real platform. The subset of the dataset used in the experiments included few congested highway scenarios, meaning there is a fairly narrow distribution of speeds between the vehicles. The effects of highly varying speeds between traffic agents on the proposed method could also be explored. Additional future work includes increasing the prediction horizon, examining the use of LSTMs for recursion, and evaluating the effect of using a longer input sequence.

ACKNOWLEDGMENT

We are grateful to Vinnova project 2016-02547: IQPilot - City Transport with autonomous heavy vehicles for funding the work. The authors would also like to thank Tinne Tuytelaars and her team from KU Leuven for providing the code to the original network used in [13].

REFERENCES

[1] G. Weidl, A. L. Madsen, V. Tereshchenko, D. Kasper, and G. Breuel, *Early Recognition of Maneuvers in Highway Traffic*, pp. 529–540. Cham: Springer International Publishing, 2015.

[2] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 4363–4369, IEEE, 2013.

[3] M. Bahram, A. Wolf, M. Aeberhard, and D. Wollherr, "A prediction-based reactive driving strategy for highly automated driving function on freeways," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pp. 400–406, IEEE, 2014.

[4] T. Hulnhagen, I. Dengler, A. Tamke, T. Dang, and G. Breuel, "Maneuver recognition using probabilistic finite-state machines and fuzzy logic," in *IEEE Intelligent Vehicles Symposium, San Diego*, 2010.

[5] H. Berndt, J. Emmert, and K. Dietmayer, "Continuous driver intention recognition with hidden markov models," in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, IEEE, 2008.

[6] 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, Australia), June 2013.

[7] C. Wissing, T. Nattermann, K.-H. Glander, and T. Bertram, "Probabilistic time-to-lane-change prediction on highways," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*, pp. 1452–1457, IEEE, 2017.

[8] "Combining behavior and situation information for reliably estimating multiple intentions,"

[9] J. Jordan, C. Ruhhammer, H. Kloeden, and M. Kleinsteuber, "Learning driving scene prediction from environmental perception of vehicle fleet data," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, IEEE, 2015.

[10] A. Lawitzky, D. Althoff, C. F. Passenberg, G. Tanzmeister, D. Wollherr, and M. Buss, "Interactive scene prediction for automotive applications," in *Intelligent Vehicles Symposium (IV)*, IEEE, 2013.

[11] Ü. Dogan, J. Edelbrunner, and I. Iossifidis, "Autonomous driving: A comparison of machine learning techniques by means of the prediction of lane change behavior," in *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*, pp. 1837–1843, IEEE, 2011.

[12] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," *arXiv preprint arXiv:1701.06699*, 2017.

[13] B. De Brabandere, X. Jia, T. Tuytelaars, and L. Van Gool, "Dynamic filter networks," in *Neural Information Processing Systems (NIPS)*, 2016.

[14] A. Zyner, S. Worrall, J. Ward, and E. Nebot, "Long short term memory for driver intent prediction," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*, pp. 1484–1489, IEEE, 2017.

[15] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[16] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 843–852, PMLR, 07–09 Jul 2015.

[17] W. Lotter, G. Kreiman, and D. Cox, "Deep predictive coding networks for video prediction and unsupervised learning," *arXiv preprint arXiv:1605.08104*, 2016.

[18] J. Colyar and J. Halkias, "US highway 80 dataset," tech. rep., Federal Highway Admin. (FHWA), Washington, DC, USA, Tech. Rep. FHWA-HRT-06-137, 2006.