2017 IEEE Intelligent Vehicles Symposium (IV)
June 11-14, 2017, Redondo Beach, CA, USA

Long Short Term Memory for Driver Intent Prediction

Alex Zyner, Stewart Worrall, James Ward, and Eduardo Nebot[1]

***Abstract*— Advanced Driver Assistance Systems have been shown to greatly improve road safety. However, existing systems are typically reactive with an inability to understand complex traffic scenarios. We present a method to predict driver intention as the vehicle enters an intersection using a Long Short Term Memory (LSTM) based Recurrent Neural Network (RNN). The model is learnt using the position, heading and velocity fused from GPS, IMU and odometry data collected by the ego-vehicle. In this paper we focus on determining the earliest possible moment in which we can classify the driver's intention at an intersection. We consider the outcome of this work an essential component for all levels of road vehicle automation.**

I. INTRODUCTION

Advanced driver assistance systems (ADAS) are increasingly seen as a mechanism to improve the safety and efficiency of transportation by understanding and reacting to potential vehicle safety threats using state-of-the-art sensing and algorithms. A major component of these systems is the ability to infer the future intentions of drivers to predict the likelihood of potential collisions. This is a challenging task, particularly in intersections where complex traffic scenarios result in a proportionally high number of accidents. Human drivers are able to estimate the future trajectory of other vehicles from a combination of potentially subtle cues - the combination of the various kinematic properties - and the position of the vehicle on the road relative to the lane. Being able to reproduce this driver intuition in a computer model is still an open area of research.

An ADAS equipped vehicle has access to a number of important features: speed, heading, and position in the lane. These features, among others, are able to be broadcast to surrounding vehicles using Dedicated Short Range Communication (DSRC) in a transmission known as the Basic Safety Message [1]. Higher level analysis of this data could allow for an ADAS equipped vehicle to infer the intentions of manned vehicles, allowing these vehicles to interact in a safe manner. During the last two years a number of manufacturers have started to introduce urban vehicles with Level 2 autonomy. This level of autonomy is expected to increase to Level 4 by 2030 [2]. Nevertheless, even the most optimistic predictions have a majority of vehicles under human control over the next 30 years. This emphasises how the inference of intent is an essential component to facilitate the safe interaction of autonomous and manned vehicles.

Authors are with the Australian Centre for Field Robotics (ACFR) at the University of Sydney (NSW, Australia). E-mails: {a.zyner, s.worrall, j.ward, e.nebot} (at) acfr.usyd.edu.au.

Fig. 1. Vehicle fitted with GPS that allows for filtered GPS position, heading and speed to be recorded for experiments. It is also fitted with a 360 degree coverage lidar based perception system from Ibeo Automotive Systems GmbH [3], allowing for real-time detection and tracking of neighbouring vehicles. The vehicle is fitted with a DSRC radio allowing for communication between vehicles.

A. Contributions

This work aims to classify which manoeuvre a driver may take through an intersection, given the particular approach path of the vehicle. The dataset used in this work consist of recordings of speed, heading and position measurements taken via GPS and odometry, using a system installed in the vehicle in Figure 1. The method used in this paper is a Long Short Term Memory (LSTM) based Recurrent Neural Network (RNN). Models are evaluated based on how early it achieves a 100% classification rate for all sequences in the test set. Instead of analysing the observation at each time step individually, our model considers a short sequence of data to allow the model to make reliable predictions.

The paper is organised as follows. Related work is presented in section II. We outline the problem, and describe the network architecture in section III. We describe the testing method in section IV. In section V we demonstrate improvements over previous approaches. We present conclusions in section VI.

II. RELATED WORK

Although the prediction of driver behaviour on the road has been widely studied, it is still an open problem. There are several approaches used in making predictions on driver behaviour. Physics based models use kinematic or dynamic models to represent and predict vehicle movement. Popular approaches include a Constant Turn Rate and Velocity model (CTRV) [4], Switching Kalman Filters [5], or Monte Carlo

978-1-5090-4804-5/17/$31.00 ©2017 IEEE

Simulation [6] methods. These types of models are generally limited to very short term prediction (under one second) [7]. Manoeuvre based models make the assumption that the actions of the driver is limited to a set of manoeuvres. If the manoeuvre can be identified early, the rest of the vehicle motion can be predicted, as it should match the manoeuvre. A variety of algorithms have been tested for classifying driver behaviour into manoeuvres, including Multi-Layer Perceptrons [8], Support Vector Machines [9], Relevance Vector Machines [10], Conditional Random Fields [11], and Hidden Markov Models [12] [13]. Jain et. al. [14] use a similar network to fuse data from a driver facing camera, a road facing camera, and GPS. In this work we avoid the use of driver facing cameras as they are not widely available. A more in-depth review of behaviour prediction at intersections is presented by Shirazi et. al. [15].

This work presents an innovative manoeuvre based model that is able to analyse correlation between time steps, does not use a physical based model for prediction, and is easily expandable to accept more data about the chosen vehicle. This allows it to be trained on any data that is available, and it is fast enough to work in real-time.

III. APPROACH

A. The Intersection

The data set used in this paper, known as the Naturalistic Intersection Driving Dataset [16], consists of 198 paths travelled through an unmarked T shaped intersection, as driven by three different drivers. This set consists of 6 possible manoeuvres, and each driver was instructed to perform each manoeuvre 10 times. A driver will behave differently depending on their intended manoeuvre - this can manifest in a different trajectory, velocity profile, or changes in heading. We try to model the potentially subtle cues based on the way a driver approaches an intersection. The aim of the model is to analyse a combination of kinematic features and location context to model the intended destination, as executed by a manoeuvre.

Each manoeuvre a driver can take has different characteristics leading up to the intersection. For instance, a driver turning across traffic will slow down and move to the centre of the intersection. The driver may have to come to a complete stop to give way to other traffic. A driver making a close turn will generally move to the outside of the road and slow down in preparation for the turn. A driver continuing straight will do none of these things, and maintain speed and heading. These subtle differences in driving behaviour are the features that are exploited in this classification algorithm, so that an autonomous vehicle can predict the intentions of other nearby vehicles.

B. Data Preparation

The position data for the vehicle was recorded at 10Hz by a global navigation satellite system (GNSS). Speed was recorded using the vehicle's wheel encoder, and inertial data was recorded via an inertial measurement unit (IMU). All this data was fused via a extended Kalman filter, and used as input into the model.

Fig. 2. Satellite view of the intersection showing the route the vehicle took in blue, and the outer bounds of the intersection in red.

Each recorded vehicle path is bounded by a 60m box relative to the centre of the intersection. This distance is sufficiently far enough to consider that the driver has not committed to a particular manoeuvre through an intersection. Within the intersection, a reference line is used. This reference line represents the place at which a driver has committed to a particular manoeuvre. As the chosen intersection has no road markings, this line does not represent any mark on the road, but it is used as the nominated start of the intersection. This line is located at as a distance of 20 metres from the centre of the intersection.

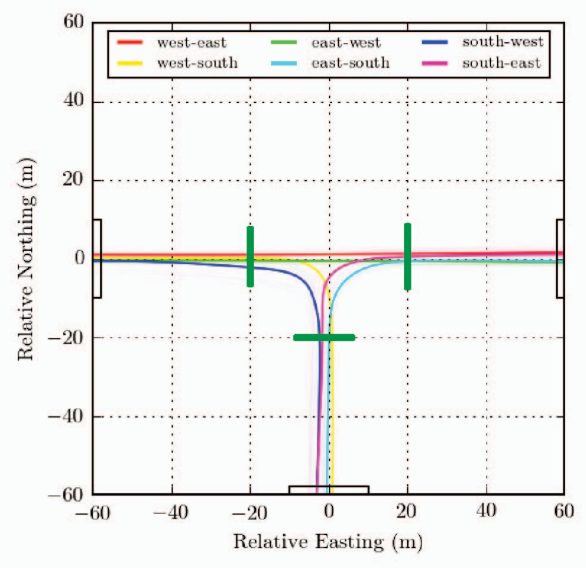

Fig. 3. The isolated intersection, showing the mean trajectory for all 6 manoeuvres. The legend at the top is in the format [origin-destination]. The reference line, at the 20m mark, is shown in green.

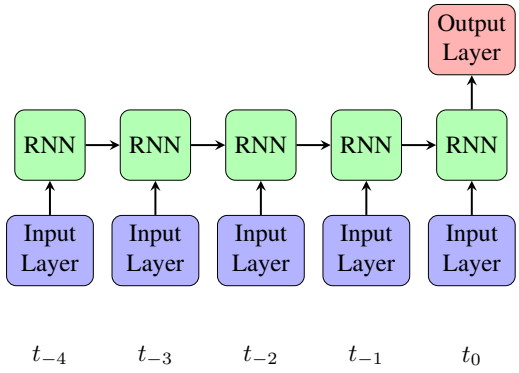

Fig. 4. Diagram representing a RNN of length 5, that is given data for time t. Here the model takes a sequence of length 5, that ends at the nominated time-step, and outputs a class, in the range [East, West, South].

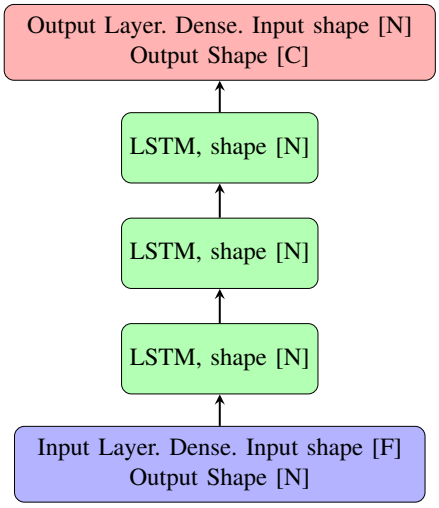

Fig. 5. Diagram that depicts the individual layers of the RNN. F = number of features. N = number of nodes in RNN. C = Number of classes (3 [east, west, south])

C. Long Short Term Memory based RNN

An RNN has a different structure to a typical neural network. An RNN is used in problems where there is dependence in the ordering of the data, i.e. the data has a sequential structure. This sequence could exist as the ordering of words in a sentence, which is an approach used in the model presented by Mikolov et. al. [17]. The sequence can also occur in time, where each iteration of the sequence represents a single time-step, such as frames in a video, or sensor data logged over time. We use this network structure to model the correlation between the consecutive samples of the sequence.

The unique structure in an RNN is that the network contains feed-back connections, such that network activations can flow cyclically in a loop. This allows an activation to persist over multiple input steps, giving the network a sense of 'memory' between time steps. The network proposed in this paper is a many-to-one style network, such that it produces a single output after it is presented with a full sequence of data. Figure 4 depicts this style of network for a sequence length of 5. For a better visual understanding, the network is 'unrolled' in this diagram, so that the activations at every step can be visualized. This unrolling of the sequence into space means that the weights of the network are repeated. There is only one set of input weights, and they are repeated on the figure for each of the 5 time steps. This common set of input weights allows the network to parse the input data in a common format. The output layer is only used during the final step of the RNN, and it will then output a prediction class after a soft-max function is applied.

The recurrent neural network cell chosen in this paper is known as a Long Short-Term Memory (LSTM) cell [18]. This cell has a number of properties that make it favourable over other cell types. An LSTM has the property of remembering a value for an arbitrary length of time, allowing it to overcome the vanishing gradient problem. In general RNNs, the input decays or increases exponentially over time, which causes problems in training. An LSTM uses a gating system to overcome this problem [19]. LSTMs are popular for text parsing, [17] but have had recent success in sequence generation problems, such as co-ordinates of pedestrian paths [20] and handwriting generation [21].

D. Model Architecture

We now present the architecture of our LSTM based RNN model used in this paper. The goal of this system is to predict the manoeuvre of the driver as early as possible. The destination can be one of three locations: East, West, or South. The training data consists of 198 tracks of sequential data, where each track contains a number of observations, outlined in (1). Here $\boldsymbol{M}$ is defined as the set of all manoeuvre sequences, T is the number of observations in sequence j, N is the number of sequences, $\boldsymbol{x}_t$ is the observation at time t, and $\boldsymbol{y}_j$ is the destination in the set [East, West, South] for the j^{th} sequence. The model is of a fixed length, k, so the training data must be sampled in such a way to fit this constraint. As such, the training input set, $\boldsymbol{S}$ is defined as every possible consecutive sequence of length k that exists for all sequences in the data set, shown in (2). This sampling can be thought of as a 'sliding window' across the data set. The data was recorded at a rate of 10Hz, allowing for a time between observations of 0.1 seconds. By not sub-sampling the input data, we can ensure that the data fed into the model is consistent between physical recording time and network time-steps, which maintains temporal consistency across training samples. The network consists of an input layer, of input shape F (the number of features) and output shape N (the number of nodes in the LSTM layers). Three LSTM layers are then used, and a final output layer of output shape C (the number of manoeuvre classes) is applied. This output is then passed through a soft-max classifier to select the predicted manoeuvre.

$$\boldsymbol{M} = \{(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, ..., \boldsymbol{x}_T)_j, \boldsymbol{y}_j\}_{j=1}^{N} \quad (1)$$

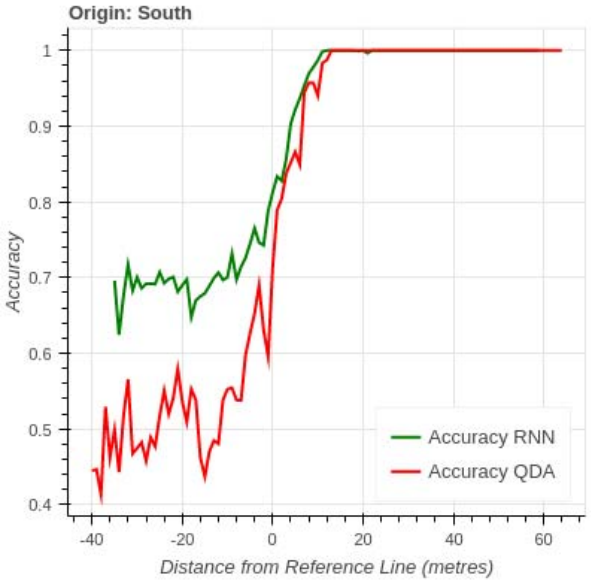

Fig. 6. Accuracy of both the QDA classifier and the LSTM based RNN, when vehicles approach from the south. Here the LSTM method outperforms the QDA method quite early in the approach, however both converge to 100% at approximately the same distance.

$$\boldsymbol{S} = \{\{(\boldsymbol{x}_w, \boldsymbol{x}_{w+1}, \boldsymbol{x}_{w+2}, ..., \boldsymbol{x}_{w+k}), \boldsymbol{y}_j\}_{w=1}^{T-k}\}_{j=1}^{N} \quad (2)$$

E. Features

The features that make up an observation $\boldsymbol{x}_t$ are as follows: vehicle co-ordinates in easting and northing, speed of the vehicle, and heading, relative to north. The vehicle co-ordinates are measured in metres, and the origin is defined as the centre of the intersection. Speed is measured in metres per second, and is collected by the vehicle's wheel encoder. Finally, the heading of the vehicle is input directly as a float that exists in the range [0,360] degrees.

These features are measured by the ego vehicle, using an Extended Kalman Filter fed with GNSS, IMU and wheel odometry data. The vehicle is also fitted with a DSRC radio, allowing for transmission of this data to nearby vehicles. This would allow the model to take features from nearby vehicles and make predictions about them, given enough data is available.

IV. METHOD

The code used to generate the model was written in TensorFlow [22], and trained on a GPU. The system was validated using 5 fold stratified cross validation, leading to an average training time of 30 minutes per fold, on a single Nvidia GTX 1080. This model is compared to the results of Bender et. al. [16], which uses a Quadratic Discriminate Analysis model to predict the manoeuvre on the same dataset.

The distance metric used in the plots and analysis is the distance travelled (in metres) with respect to the reference line. Negative values indicate the vehicle is still approaching the intersection. A value of zero indicates the vehicle has begun entering the intersection, and the driver has committed to a particular manoeuvre at this point. A large value indicates the vehicle is leaving, or has left the intersection. This distance metric allows the identification of the earliest possible moment the model can predict the driver's manoeuvre.

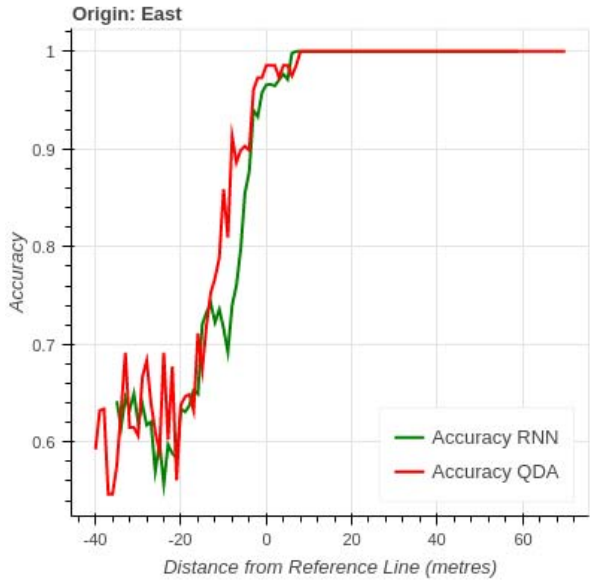

Fig. 7. Accuracy of both the QDA classifier and the LSTM based RNN, when vehicles approach from the east. Vehicles approaching from the east do not have to give way to any other traffic, making this approach difficult to classify.

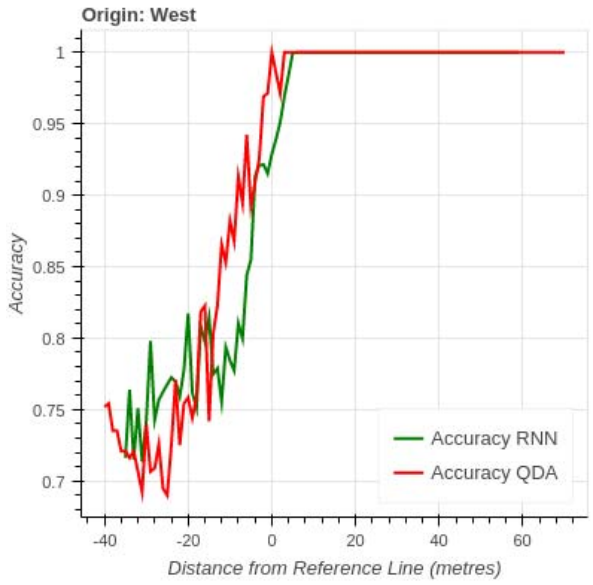

Fig. 8. Accuracy of both the QDA classifier and the LSTM based RNN, when vehicles approach from the west. Vehicles approaching from the west may have to give way to oncoming traffic when making a right turn. This decrease in speed is an early indicator, which allows both models to classify relatively early.

A. Training

The system is trained using mini-batches of size 128, with 3 LSTM layers of size 112, with a sequence length of 3. The LSTM layers are interconnected with peephole connections [23]. The optimizer used is RMSprop, with a decaying learning rate, starting at 0.001. To generate training data for the LSTM network, all possible consecutive sequences of length k time-steps were added to the training data, each with the appropriate class label of destination (East, West or South).

B. Testing

The testing of each algorithm occurs at regular intervals of 1 metre, as measured by distance travelled relative to the reference line. As the LSTM network requires sequential data, the sequence of length k that ends at the specified distance is used. For the QDA model, a single sample of data at the specified distance is used. This ensures that the LSTM network is not fed data that occurs later in the sequence than the data fed to the QDA, allowing for a fair comparison.

V. RESULTS

The graphs in Figures 6, 7, and 8 depict the average results over the 5 fold cross validation, split by the origin of the track. The x axis of the graph is metres from the reference line, which is set 20 metres from the centre of the intersection. The intersection is a T junction, so a driver may only make one of two manoeuvres when approaching the intersection. This means that the classifier should have at least a 50% accuracy at the start of the track, well before the driver has started executing the manoeuvre. It is important to note that this dataset was taken in Australia, where vehicles drive on the left side of the road.

A. Eastern Origin

The eastern origin allows the driver to go straight ahead, or make a close, left turn. If there are no cars in front, a driver may proceed straight through the intersection without slowing, potentially travelling at the speed limit. If a driver is turning, they are performing a left turn, and the driver is not required to give way to any other traffic in the intersection, so a driver has no reason to come to a complete stop. Here, the results between the QDA model and the LSTM network are very similar, as both models approach 100% accuracy at approximately the same distance from the reference line (7 metres).

B. Western Origin

A driver approaching from the western origin may turn right or continue straight ahead. Again, if the driver is travelling straight ahead they have no reason to slow the vehicle. As the turning manoeuvre is across traffic, the driver may have to slow down and give way to oncoming vehicles when turning right. In some cases, the driver has to come to a complete stop to let other traffic traverse the intersection first. This is very apparent in the speed profile during the approach. Both models pick up on this fact rather early, and approach 100% accuracy in manoeuvre classification at 6 metres past the reference line.

C. Southern Origin

Vehicles approaching from the south must turn either left or right, they cannot continue forward due to the nature of the T intersection. So, a vehicle approaching from this direction needs to slow down to make a turn. A driver turning either left or right may also have to give way to traffic, so the vehicle may come to a complete stop performing either manoeuvre. Here the LSTM network clearly outperforms the QDA in early predictions, as it has a 70% accuracy very early in the approach, at around -30 metres to the reference line, compared to the QDA's 50% accuracy. However, the LSTM network takes more distance to reach a completely accurate estimate than the other two origins, which is most likely due to the speed profile being very similar between left and right turns. Both the QDA and the LSTM techniques take 12 metres past the reference line achieve 100% accuracy.

D. Discussion

Classifiers are only able to make useful predictions when the data is notably separate. The earliest point at which each classifier can correctly predict the manoeuvre is identified, and used as a comparison. The QDA classifier is fed with a single time step of data, and the LSTM network is fed with multiple, consecutive time steps. While for the East and West origins the results appear similar, the LSTM network has a clear advantage for vehicles in the southern approach. In the southern approach, the driver is forced to slow the vehicle, and make a turn. This makes the classification significantly more difficult, as the speed profile of both turns are very similar. Having improved results in this area shows promise that the LSTM has advantage in more complicated scenarios.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we present a model that predicts the manoeuvre of a driver at an unsignalized intersection. This level of intuition is natural to an experienced driver, and is a necessary tool for safely negotiating busy roads. As such, it is invaluable for intelligent vehicles to be able to harness such a system and be able to broadcast intention via DSRC to surrounding vehicles. The LSTM network proposed allows for the analysis of a consecutive sequence of data, instead of analysing time steps individually. This method has led to improvements compared to previous approaches, namely when approaching the intersection from the south. There is still room for improvement, as earlier correct classification is favourable, to allow for a better understanding of drivers negotiating an intersection.

Outside of the four features used in this model: position (easting, northing), speed and heading, it may be worth investigating other measurable features such as driver gaze, or indicator status. We plan to test this model on a vehicle equipped with both a lidar perception system and DSRC. This would potentially allow for both the inference of intent of surrounding vehicles, and broadcasting this information

via DSRC to neighbouring vehicles. This could then be used by an autonomous vehicle to make a well informed decision about navigating the road ahead.

ACKNOWLEDGMENT

This research was funded partially by the Australian Government through the Australian Research Council Discovery Grant DP160104081.

REFERENCES

[1] D. Committee *et al.*, "SAE J2735 Dedicated Short Range Communications (DSRC) message set dictionary," *Society of Automotive Engineers, Warrendale, PA*, 2016.

[2] T. Litman, "Autonomous vehicle implementation predictions," *Victoria Transport Policy Institute*, 2014.

[3] Ibeo automotive systems gmbh. [Online]. Available: www.ibeo-as.de

[4] A. Polychronopoulos, M. Tsogas, A. J. Amditis, and L. Andreone, "Sensor fusion for predicting vehicles' path for collision avoidance systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 3, pp. 549–562, 2007.

[5] H. Veeraraghavan, N. Papanikolopoulos, and P. Schrater, "Deterministic sampling-based switching kalman filtering for vehicle tracking," in *2006 IEEE Intelligent Transportation Systems Conference*. IEEE, 2006, pp. 1340–1345.

[6] M. Althoff and A. Mergel, "Comparison of markov chain abstraction and monte carlo simulation for the safety assessment of autonomous cars," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1237–1247, 2011.

[7] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *Robomech Journal*, vol. 1, no. 1, p. 1, 2014.

[8] M. G. Ortiz, J. Fritsch, F. Kummert, and A. Gepperth, "Behavior prediction at multiple time-scales in inner-city scenarios," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 1068–1073.

[9] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 797–802.

[10] B. Morris, A. Doshi, and M. Trivedi, "Lane change intent prediction for driver assistance: On-road design and evaluation," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 895–901.

[11] E. Ohn-Bar, A. Tawari, S. Martin, and M. M. Trivedi, "On surveillance for safety critical events: In-vehicle video networks for predictive driver assistance systems," *Computer Vision and Image Understanding*, vol. 134, pp. 130–140, 2015.

[12] H. Berndt and K. Dietmayer, "Driver intention inference with vehicle onboard sensors," in *Vehicular Electronics and Safety (ICVES), 2009 IEEE International Conference on*. IEEE, 2009, pp. 102–107.

[13] T. Streubel and K. H. Hoffmann, "Prediction of driver intended path at intersections," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, 2014, pp. 134–139.

[14] A. Jain, A. Singh, H. S. Koppula, S. Soh, and A. Saxena, "Recurrent neural networks for driver activity anticipation via sensory-fusion architecture," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 3118–3125.

[15] M. S. Shirazi and B. Morris, "Observing behaviors at intersections: A review of recent studies amp; developments," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, June 2015, pp. 1258–1263.

[16] A. Bender, J. R. Ward, S. Worrall, and E. M. Nebot, "Predicting driver intent from models of naturalistic driving," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, 2015, pp. 1609–1615.

[17] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[19] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks (Studies in Computational Intelligence)*. Springer, 2012.

[20] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 961–971.

[21] A. Graves, "Generating sequences with recurrent neural networks," 2013.

[22] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[23] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling." in *INTERSPEECH*, 2014, pp. 338–342.