2019 IEEE Intelligent Vehicles Symposium (IV)
Paris, France. June 9-12, 2019

Non-local Social Pooling for Vehicle Trajectory Prediction

Kaouther Messaoud[1], Itheri Yahiaoui[2], Anne Verroust-Blondet[1] and Fawzi Nashashibi[1]

Abstract— **For an efficient integration of autonomous vehicles on roads, human-like reasoning and decision making in complex traffic situations are needed. One of the key factors to achieve this goal is the estimation of the future behavior of the vehicles present in the scene. In this work, we propose a new approach to predict the motion of vehicles surrounding a target vehicle in a highway environment. Our approach is based on an LSTM encoder-decoder that uses a social pooling mechanism to model the interactions between all the neighboring vehicles. The originality of our social pooling module is that it combines both local and non-local operations. The non-local multi-head attention mechanism captures the relative importance of each vehicle despite the inter-vehicle distances to the target vehicle, while the local blocks represent nearby interactions between vehicles. This paper compares the proposed approach with the state-of-the-art using two naturalistic driving datasets: Next Generation Simulation (NGSIM) and the new highD Dataset. The proposed method outperforms existing ones in terms of RMS values of prediction error, which shows the effectiveness of combining local and non-local operations in such a context.**

I. INTRODUCTION

In autonomous driving, most of the challenging tasks are related to understanding, analyzing the driving situations and the interactions between traffic participants and making a reasonable and safe navigation decisions accordingly. Human drivers make decisions based on their implicit reasoning about how surrounding drivers will move in the future. In this work, we aim to predict the motion of drivers surrounding a target vehicle on a highway. This is a challenging task because drivers have different decision-making strategies and adequate reactions depending on different traffic situations: their behaviors are highly correlated among each other and they depend on traffic density and the road structure.

Previous studies have tackled some aspects of the above challenges. In order to model the driver behavior, traditional data-driven techniques [1], [2], [3] as well as deep learning models based on Long Short Term Memories (LSTMs) [4], [5], [6], [7], [8], [9], [10] have been used. LSTM based encoder-decoder architectures have shown great success in modeling the non-linear temporal dependency between the input sequence elements. However, they show poor performance at capturing spatial interactions.

As a remedy, this paper proposes the use of an LSTM encoder decoder architecture with an additional module modeling the spatial interactions between neighboring vehicles. The proposed module is a new variant of the social pooling approach, which was introduced by Alahi et al. [11] to model the interactions between nearby pedestrians and extended by Deo and Trivedi [12] to learn the vehicle motion interdependencies. In our approach, we attempt to mimic human reasoning, which focuses attention selectively on a subset of surrounding vehicles to extract the details that most influence the target vehicle's future trajectories while considering other vehicles less. For example, a vehicle intending to make a lane change focuses more on the vehicles in the target lane than those in the other lanes. Therefore, its future trajectory could be more influenced by distant vehicles in the target lane than the close ones in the other lanes. Thus, we propose to adapt the non-local multi head attention mechanism, which was introduced by Vaswani et al. [13] for natural language processing purposes, and to combine it with convolution blocks to model both distant and local influences of vehicles, based on their relation with the autonomous vehicle.

Our pooling mechanism builds a context vector which encodes the cues of surrounding vehicles that most influence the future trajectory. Our model combines the advantages of two individual architectures:

- Convolution Layer captures the interactions of nearby vehicles and produces the local context over which we use attention operations.
- Multiple head attentions learn different local and distant relationships and combine them according to their importance.

We get competitive results with the state-of-the-art on the publicly available NGSIM US-101 [14], NGSIM I-80 [15] and highD dataset[16].

II. RELATED RESEARCH

Numerous vehicle motion prediction models have recently been proposed. Lefèvre et al. [17] and Zhan et al. [18] consider that one of the main distinctions between vehicle behavior forecasting methods is whether or not they take into account interactions between the surrounding vehicles.

A. Independent prediction

The early work on vehicle motion modeling and prediction focused on studying only one single vehicle at a time. The future trajectory is predicted by applying physics-based evolution models [19], [20], [21]. Since these approaches mainly rely on the low level properties of motion, they are limited to short-term motion prediction. For longer-term motion prediction, Eidehall et al.[22] adopt a dynamic driver model with stochastic variables while Wiest et al.[23] use variational Gaussian mixture modeling. More recent methods decompose the motion of a vehicle into a finite set of typical patterns called maneuvers. Schlechtriemen et al. [1] use a

[1]Inria Paris, 2 rue Simone Iff 75012 Paris FRANCE {kaouther.messaoud,anne.verroust,fawzi.nashashibi}@inria.fr
[2] CReSTIC, Université de Reims Champagne-Ardenne, Reims, FRANCE itheri.yahiaoui@univ-reims.fr

Authorized licensed use limited to: University of Houston. Downloaded on July 10,2020 at 20:17:58 UTC from IEEE Xplore. Restrictions apply.

Hidden Markov Model (HMM) on the top of a Naive Bayes Classifier, where each state of the HMM represents one of the maneuvers inferred given the driving data. Houenou et al. [2] use the maneuvers recognition module to make predictions of future trajectories as realizations of the predicted maneuver. They predict the realization of the maneuver by minimizing a cost function that guaranties the safety and the comfort of the driver. Yoon et al. [24] use Multi-Layer Perceptron MLP for lateral motion prediction. They propose three representative trajectories per lane based on how fast the vehicle reaches each target lane. The MLP model provides probabilities that a vehicle will choose each lane and have each possible trajectory. The main limitation of these models is that they do not take into account the influence of the surrounding vehicles on the future trajectory.

B. Interaction aware models

Different approaches have been introduced to model interactions between vehicles. Sierra González et al. [25] use Markov Decision Process (MDPs) to model the driver decision-making approach. They consider a trajectory as a sequence of states of a vehicle and the cost function as a linear combination of static and dynamic features that parameterize each state. They use an Inverse Reinforcement Learning (IRL) algorithm to learn the cost function parameters while considering risk aversive interactions between vehicles. In [26] they propose using Dynamic Bayesian Networks to model interactions between vehicles. Deo et al. [3] introduce a framework for holistic surrounding vehicle trajectory prediction where the vehicle interaction module considers the global context of neighboring vehicles and assigns final predictions by minimizing an energy function.

C. Deep-Learning Based Methods

Motion prediction can be considered as a time series classification or generation task. RNNs play a crucial role in recent advancements in sequence modeling and generation. They have shown promising results in diverse domains such as natural language processing and speech recognition. Therefore, RNN based approaches have been solid candidates to model maneuver and trajectory prediction.

Long Short Term Memories (LSTMs) are a particular implementation of recurrent neural networks able to learn long-term relations between features. In contrast to other neural networks, they don't assume that inputs are independent of each other. They treat sequential information and model the dependence between inputs. They perform the same operations for every element of a sequence given the previous computation of the input sequence.

Recently LSTMs have been used for predicting the driver intention and different LSTM-based architectures have been adopted; A simple LSTM with one or more layers is used in [5], [6], [7], [10]. Xin et al. [8] deploy a dual LSTM. The first one for high-level driver intention recognition followed by a second for future trajectory prediction. Others [9], [12], [27] use an LSTM encoder decoder architecture. The entries to the LSTM are also different. While Lenz et al. [6] feed the LSTM with only the current state of a set of vehicles in order to follow the Markov Property, other studies [5], [7], [9], [12] consider the history sequence of features in order to add extra temporal information and help in the prediction task. They accord to the LSTM the task of capturing the important events and remembering them using the hidden state.

The interactions between surrounding vehicles are also modeled differently. Some existing models [5], [6], [7], [9] implicitly infer the dependencies between vehicles. They let the LSTM implicitly learn the influence of surrounding vehicles on the target vehicle's motion by introducing a sequence of surrounding vehicles features as inputs to the LSTMs. Other approaches explicitly model the vehicles' interactions by using multiple networks. One pioneering work is the social LSTM of Alahi et al. [11]. They model the interactions between pedestrians by sharing the LSTMs hidden states that encode the motion characteristics, between all the present agents. Deo et al. [12] extend the social pooling approach by generating a compact representation of the social interactions combining the outputs from the encoders of the surrounding vehicles in a context vector. They use convolutional layers that focus on successive local interactions followed by a maxpool layer. But local information is not always sufficient. Furthermore, the generated context vector is independent of the target vehicle state. We extend existing approaches by an attention based non-local vehicles dependencies modeling that represents vehicles' interactions based on their importance to the target vehicle. The attention mechanism reduces the number of local operations by directly relating distant elements. Our motion prediction results are compared with those reported in [9], [12].

III. PROBLEM DEFINITION

Our goal is to predict the probability distribution of the future positions of a target vehicle T while considering its track history and the track history of all the surrounding vehicles at current time t_{obs}.

A. Inputs and Outputs

We assume that we have as input the track history of the target and n surrounding vehicles as $\mathbf{X} = \mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_n$. The input trajectory of a vehicle i is defined as $\mathbf{X}_i = [\mathbf{x}_i^1, \ldots, \mathbf{x}_i^{t_{obs}}]$ where $\mathbf{x}_i^t = (x_i^t, y_i^t)$. The coordinates are expressed in a stationary frame of reference where the origin is the position of the target vehicle at time t_{obs}. The $y-axis$ and $x-axis$ point respectively to the direction of motion of the freeway and to the direction perpendicular to it.

The output of the model is a probability distribution over the target vehicle's future positions.

$$\mathbf{Y} = [\mathbf{y}^{t_{obs}+1}, \ldots, \mathbf{y}^{t_{obs}+t_f}]$$

Where $\mathbf{y}^t = (x^t, y^t)$ is the target vehicle's predicted coordinates.

The model estimates the conditional probability distribution $\mathbf{P}(\mathbf{Y}|\mathbf{X})$. The position distribution at time $t \in \{t_{obs} + 1, \ldots, t_{obs} + t_f\}$ can be modeled as a bivariate Gaussian

Authorized licensed use limited to: University of Houston. Downloaded on July 10,2020 at 20:17:58 UTC from IEEE Xplore. Restrictions apply.

distribution with a set of parameters $\Theta^t = (\mu^t, \Sigma^t)$ of the form:

$$\mathbf{y}^t \sim \mathcal{N}(\mu^t, \Sigma^t)$$

Where μ^t is the mean vector and Σ^t the covariance matrix:

$$\mu^t = \begin{pmatrix} \mu_x^t \\ \mu_y^t \end{pmatrix}, \Sigma^t = \begin{pmatrix} (\sigma_x^t)^2 & \sigma_x^t \sigma_y^t \rho^t \\ \sigma_x^t \sigma_y^t \rho^t & (\sigma_y^t)^2 \end{pmatrix}$$

The deep neural networks output the values of the parameters of the Gaussian distributions presented above, while considering the dependencies between the interacting vehicles.

B. Loss Function

We train the model by minimizing the negative log-likelihood as a loss function:

$$L_{nll} = -\sum_{t_{obs+1} \leq t \leq t_{obs}+t_f} \left\{ log(P_{\Theta^t}(\mathbf{y}^t|\mathbf{X})) \right\}$$

IV. MODEL ARCHITECTURE

Our model extends the social pooling approaches [11], [12] that take into account the information of surrounding agents to generate the output. We use a similar encoder-decoder architecture [28] to Deo and Trivedi [12] as it improves the social pooling performance. However, instead of using convolutional layers to learn locally useful features to model the interactions between vehicles in social LSTM, we want to capture both local and non-local features. Thus, we use an improved social pooling module that represents the vehicles interactions based on their importance to the target vehicle. It combines two computational blocks capturing both local and non-local interactions using convolutions and attention mechanism. Therefore, our model consists of three main components, as illustrated in Figure 1:

- **LSTM Encoder**: models the temporal evolution of a vehicle trajectory and encodes its motion properties in an encoding vector.
- **Non-Local Social Pooling Module**: links the hidden states of the encoder and decoder. It models the relation and the spatial interaction between the target vehicle and its surrounding vehicles.
- **LSTM Decoder**: receives the context vector englobing the selected information about the surrounding and the target vehicles and outputs the distributions' parameters over the future positions of the target vehicle.

A. LSTM Encoder

Each position of each vehicle is embedded by a fully connected layer to compose an embedding vector. The vectors embedding the positions of a vehicle i for time steps $t = 1, \ldots, t_{obs}$ are fed to the LSTM encoder:

$$e_i^t = \Psi(x_i^t, y_i^t; W_{emb})$$

$$h_i^t = LSTM(h_i^{t-1}, e_i^t; W_{encoder})$$

where $\Psi()$ is a fully connected function with LeakyReLU non linearity, W_{emb} and $W_{encoder}$ are the embedding and the encoder weights respectively. h_i^t and h_T^t are the encoder hidden state vector at time t of the ith and the target vehicle respectively. The encoders of each vehicle have the same weights $W_{encoder}$.

B. Non-Local Social Pooling Module

Interaction between traffic participants can be complex. It is composed not only of local but also non-local influences. In the proposed pooling module, convolutions and attention complement each other to model local and distant dependencies. We define a spatial grid H_α composed of the transformation of the encoders last hidden state of each of the surrounding vehicles by a function α (defined later) based on their positions at time t_{obs}.

$$H_\alpha(m, n, :) = \sum_{\forall i \in \mathcal{A}_T} \delta_{mn}(x_i^{t_{obs}}, y_i^{t_{obs}}) \alpha(h_i^{t_{obs}}) \qquad (1)$$

$\delta_{mn}(x, y)$ is an indicator function equal to 1 if and only if (x, y) is in the cell (m, n), $\mathcal{A}_T$ is the set of neighboring vehicles.

This representation preserves the spatial relationships between vehicles and the lane structure. The columns correspond to the three lanes. Depending on the used dataset, we consider two different grid sizes $(13, 3)$ and $(41, 3)$ covering a longitudinal distance of respectively 58.5 and 184.5 meters. The greater size of the second grid is justified by the relatively high inter-vehicles distances caused by high velocities in the HighD dataset.

*1) **Convolution Layer**:* Convolutions enable the model to learn local dependencies. We use a (3×3) depthwise convolution kernel [29] to capture the dependencies between nearby vehicles. We use horizontal zero padding to maintain the three-lane road structure.

*2) **Non-local Multi-head Attention Mechanism**:* The attention mechanism enables us to get a compact representation which combines information from all the surrounding vehicles based on their relationship with the target vehicle. We build N_{head} interaction vectors a_j, $j \in \{1, \ldots, N_{head}\}$ inspired by the generic non-local operation defined in [30]:

$$a_j = \frac{1}{\mathcal{C}_j(h_T^{t_{obs}})} \sum_{\forall (m,n) \in grid} f_j(h_T^{t_{obs}}, h_i^{t_{obs}}) \otimes Conv(H_{g_j}, W_L^j)$$

$\otimes$ represents an elementwise multiplication operation applied to the two grids $f_j(h_T^{t_{obs}}, h_i^{t_{obs}})$ and $Conv(H_{g_j}, W_L^j)$.

As defined in equation (1), H_{g_j} is the grid composed of the projections of the encoders last hidden states $h_i^{t_{obs}}$ by the linear transformation $g_j(h_i^{t_{obs}}) = W_g^j h_i^{t_{obs}}$. W_L^j is the convolution kernel and $\mathcal{C}_j(h_T^{t_{obs}})$ is a normalization factor.

The vector describing the target vehicle motion $h_T^{t_{obs}}$ is transformed into a new feature space by a linear function $\theta_j(h_T^{t_{obs}}) = W_\theta^j h_T^{t_{obs}}$. We build a second grid H_{Φ_j} composed of the projections of the encoders last hidden states $h_i^{t_{obs}}$ by the linear transformation $\Phi_j(h_i^{t_{obs}}) = W_\Phi^j h_i^{t_{obs}}$.

The Gaussian function computes the influence of the position (m, n) in the grid H_{Φ_j} on the target vehicle's motion in the projection space.

$$f_j(h_T^{t_{obs}}, h_i^{t_{obs}}) = e^{transpose(\theta_j(h_T^{t_{obs}}))\ Conv(H_{\Phi_j}, W_L^j)}$$

Authorized licensed use limited to: University of Houston. Downloaded on July 10,2020 at 20:17:58 UTC from IEEE Xplore. Restrictions apply.

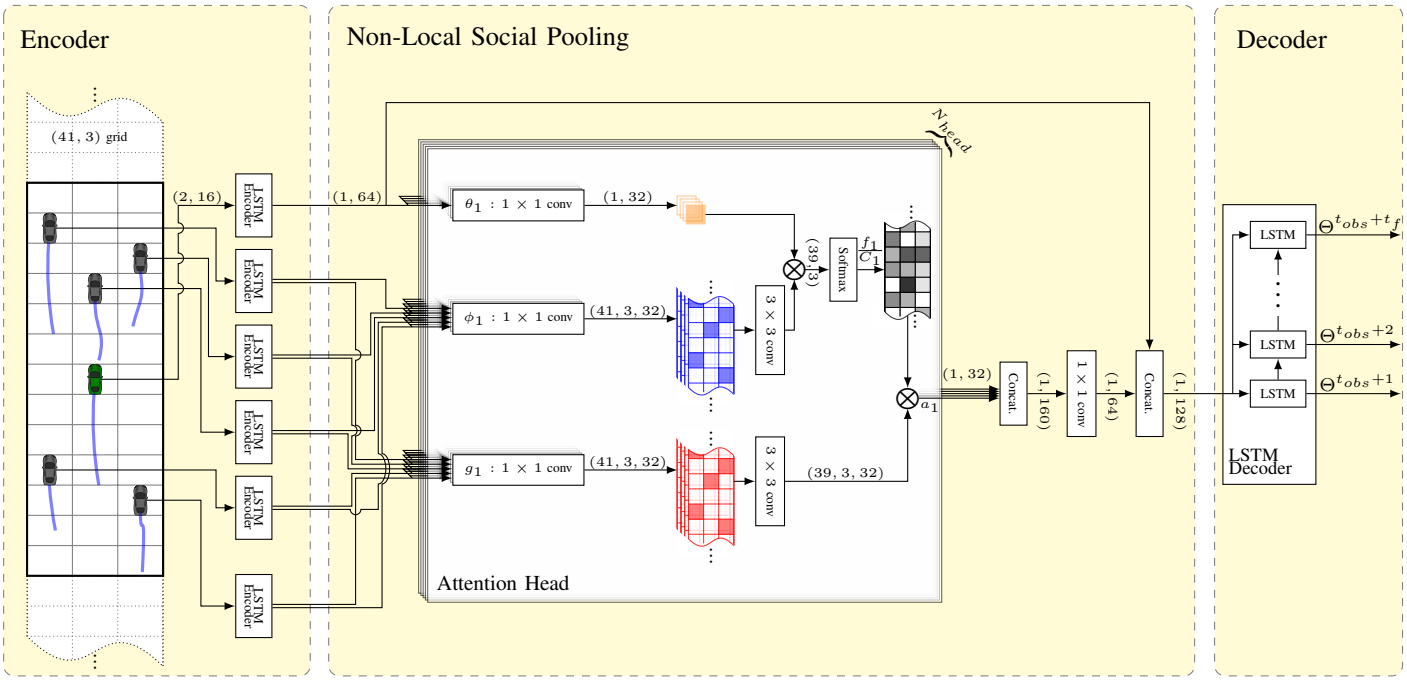

Fig. 1. **Proposed Model**:The LSTM encoders, with shared weights, learn each vehicle motion. The non-local social pooling module models the spatial interdependencies between the target (green car) and the other tracks based on their importance. Finally, the decoder outputs a distribution for the future trajectory of the target vehicle (the marked dimensions correspond to the application on HighD dataset).

The normalization factor is

$$\mathcal{C}_j(h_T^{t_{obs}}) = \sum_{\forall(m,n)\in grid} f_j(h_T^{t_{obs}}, h_i^{t_{obs}})$$

The attention weight $\frac{1}{\mathcal{C}_j(h_T^{t_{obs}})} f_j(h_T^{t_{obs}}, h_i^{t_{obs}})$ computes how much attention is put on the (m, n) position in the grid when predicting the target vehicle's motion.

The a_j is a weighted sum of the features at all positions of the grid in the projection space. Similar to [13], we consider a_j also as the jth attention head. We use a multi-head attention approach to enable the model to differently focus on different positions in different projection spaces.

The attention heads are concatenated and projected to form the output m of the attention block:

$$m = W_m Concat(a_1, .., a_{N_{head}})$$

where N_{head} is the number of the attention heads.

We employ a residual connection [31] around the attention block, followed by a normalization layer.

C. LSTM Decoder

LSTM Decoder receives the context vector, englobing the important information about the surrounding and the target vehicles: $h_{dec}^{t_{obs}} = Concat(h_T^{t_{obs}}, m)$. It outputs the predicted distributions' parameters over the future positions of the target vehicle for time steps $t = t_{obs} + 1, \ldots, t_f$.

$$\Theta^t = \Lambda(LSTM(h_{dec}^{t-1}; W_{dec}))$$

where Θ^t contains the output parameters of the motion distribution at time t, $\Lambda()$ is a fully connected function with LeakyReLU non linearity, W_{dec} are the LSTM decoder weights and h_{dec}^{t-1} is the decoder hidden state vector of the target vehicle at time $t-1$.

D. Training and Implementation Details

We use LSTMs with 64 units for the encoder and 128 units for the decoder. The embedding vector is 32 in size. We employ $N_{head} = 5$ parallel attention heads over projected vectors of size 32. We use a batch size of 128. We adopt the Adam optimizer [32] and the ReLU activation. The model is implemented using PyTorch [33].

V. EXPERIMENTAL EVALUATION

A. Datasets

We use two publicly available naturalistic vehicle trajectory datasets to train and evaluate our network:

1) HighD [16]: a new dataset captured in 2017 and 2018. It is recorded by camera-equipped drones from an aerial

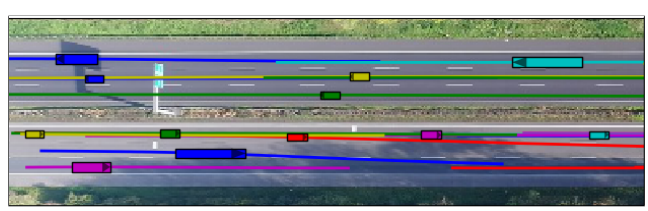

Fig. 2. Highway drone dataset highD [16]: recordings cover about 420 m of highway.

perspective of six different German highways at 25 Hz. It is composed of 60 recordings of about 17 minutes each,

Authorized licensed use limited to: University of Houston. Downloaded on July 10,2020 at 20:17:58 UTC from IEEE Xplore. Restrictions apply.

TABLE I

ROOT MEAN SQUARED PREDICTION ERROR (RMSE) IN METERS OVER A 5 SECOND PREDICTION HORIZON FOR THE MODELS.

Dataset	Prediction Horizon (s)	M-LSTM	S-LSTM	CS-LSTM	CS-LSTM(M)	NLS-LSTM
HighD	1	-	0.22	0.22	0.23	**0.20**
	2	-	0.62	0.61	0.65	**0.57**
	3	-	1.27	1.24	1.29	**1.14**
	4	-	2.15	2.10	2.18	**1.90**
	5	-	3.41	3.27	3.37	**2.91**
NGSIM	1	0.58	0.65	0.61	0.62	**0.56**
	2	1.26	1.31	1.27	1.29	**1.22**
	3	2.12	2.16	2.09	2.13	**2.02**
	4	3.24	3.25	3.10	3.20	**3.03**
	5	4.66	4.55	4.37	4.52	**4.30**

covering a segment of about 420m of two driving directions roads (Figure 2). It consists of vehicle position measurements from six different highways with 110 000 vehicles (about 12 times as many vehicles as NGSIM) and a total driven distance of 45 000 km. This dataset is of great importance since it has 5 600 recorded complete lane changes and presents recent driver behaviors.

2) NGSIM [14], [15]: publicly available large dataset captured in 2005 at 10Hz, widely studied and used in the literature, especially in the task of future intention prediction of vehicles [5], [12], [6], [7], [9]. We use this dataset to compare our model with the state-of-the-art.

We split each of the datasets into train (75%) and test (25%) sets. We split the trajectories into segments of 8s of the trajectories composed of a track history of 3s and a prediction horizon of 5s. We downsample each segment to get only 5 fps to reduce the complexity of the model.

B. Evaluation Metric

Our model generates bivariate Gaussian distributions. Therefore, we use the predicted means for the Root of the Mean Squared Error (RMSE) calculation. The RMSE averages the distance between predicted trajectories and the ground truth.

C. Models Compared

We compare our proposed model with the following models which all consider the interactions between surrounding vehicles. They are fed with the track history of the target vehicle and surrounding vehicles and output distributions over the target future trajectory.

- **Maneuver-LSTM** (*M-LSTM*) [9]: an encoder decoder based model where the encoder encodes the trajectories of the target and surrounding vehicles. The encoding vector and maneuver encodings are fed to the decoder which generates multi-modal trajectory predictions.
- **Social LSTM** (*S-LSTM*) [11]: social encoder decoder using fully connected pooling.
- **Convolutional Social Pooling** (*CS-LSTM*) [12]: social encoder decoder using convolutional pooling.
  (*CS-LSTM(M)*) generates multi-modal trajectory predictions based on six maneuvers (2 longitudinal and 3 lateral).
- **Non-local Social Pooling** (*NLS-LSTM*): This is the model described in this paper.

D. Results

Table I shows the RMSE values for the models being compared. Previous studies [11], [12], [9] compare their results to independent prediction models to prove the importance of considering surrounding agents. In this work, we not only show that surrounding vehicles are key factors in the task of trajectory prediction but we also model their interactions in a more efficient way. We train and test our model on the NGSIM and HighD datasets separately and we notice the RMSE on the NGSIM dataset is higher than that of the HighD dataset. This may be due to the difference in the sizes of the two datasets: HighD contains about 12 times as many vehicles as NGSIM. It can be also caused by annotations inaccuracies resulting in physically unrealistic vehicle behaviors in the NGSIM dataset [34]. To compare our model, we consider the results reported in [12], [9] on the NGSIM dataset and we train S-LSTM and CS-LSTM on HighD dataset as well. Our model (NLS-LSTM) outperforms the existing models based on the RMSE metric. NLS-LSTM reduces the prediction error by about 10% compared to the CS-LSTM while having comparable execution time. Therefore, a non local pooling mechanism better models the interdependencies of vehicle motion compared to convolutional social pooling. This suggests that considering the relative importance of surrounding vehicles when encoding the context is better than focusing on local dependencies.

VI. CONCLUSIONS

In this work, we introduced a neural network architecture based on a new social pooling mechanism for vehicle trajectory prediction on highways. Our approach combines a non-local multi-head attention mechanism and convolution layers to capture the relative importance of each neighboring vehicle in predicting the future motion of the target vehicle, regardless of its proximity. The proposed model outperforms the reported state-of-the-art on two naturalistic driving large scale datasets based on the RMSE metric. The obtained results proved the effectiveness of combining local and non-local operations in modeling the interactions between vehicles to predict vehicle trajectories. Once a sufficiently larger real datasets are available, we believe that the proposed architecture can be extended and utilized to further improve

Authorized licensed use limited to: University of Houston. Downloaded on July 10,2020 at 20:17:58 UTC from IEEE Xplore. Restrictions apply.

vehicle motion prediction in various driving environments such as intersections and roundabouts. Moreover, and part of our future work, we plan to extend and validate the proposed approach to consider heterogeneous and mixed traffic scenarios with different road agents such as buses, trucks, cars, scooters, bicycles, or pedestrians.

ACKNOWLEDGMENT

The work presented in this paper has been financially supported by PIA French project CAMPUS (Connected Automated Mobility Platform for Urban Sustainability).

REFERENCES

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

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

[3] N. Deo, A. Rangesh, and M. M. Trivedi, "How would surround vehicles move? a unified framework for maneuver classification and motion prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 2, pp. 129–140, June 2018.

[4] A. Khosroshahi, E. Ohn-Bar, and M. M. Trivedi, "Surround vehicles trajectory analysis with recurrent neural networks," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2016, pp. 2267–2272.

[5] D. J. Phillips, T. A. Wheeler, and M. J. Kochenderfer, "Generalizable intention prediction of human drivers at intersections," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, June 2017, pp. 1665–1670.

[6] D. Lenz, F. Diehl, M. T. Le, and A. Knoll, "Deep neural networks for markovian interactive scene prediction in highway scenarios," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, June 2017, pp. 685–692.

[7] F. Altché and A. de La Fortelle, "An lstm network for highway trajectory prediction," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Oct 2017, pp. 353–359.

[8] L. Xin, P. Wang, C. Chan, J. Chen, S. E. Li, and B. Cheng, "Intention-aware long horizon trajectory prediction of surrounding vehicles using dual lstm networks," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2018, pp. 1441–1446.

[9] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based lstms," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1179–1184.

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

[11] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 961–971.

[12] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2018, pp. 1549–15 498.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Neural Information Processing Systems (NIPS)*, 2017.

[14] J. Colyar and J. Halkias, "Us highway 101 dataset." in *Federal Highway Administration (FHWA), Tech. Rep. FHWA-HRT07-030*, 2007.

[15] J. Colyar and J. Halkias, "Us highway i-80 dataset." in *Federal Highway Administration (FHWA), Tech. Rep. FHWA-HRT-06-137*, 2006.

[16] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2018, pp. 2118–2125.

[17] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH Journal*, vol. 1, no. 1, pp. 1–14, 2014.

[18] W. Zhan, A. L. de Fortelle, Y. Chen, C. Chan, and M. Tomizuka, "Probabilistic prediction from planning perspective: Problem formulation, representation simplification and evaluation metric," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1150–1156.

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

[20] A. Barth and U. Franke, "Where will the oncoming vehicle be the next second?" in *2008 IEEE Intelligent Vehicles Symposium*, June 2008, pp. 1068–1073.

[21] R. Toledo-Moreo and M. A. Zamora-Izquierdo, "Imm-based lane-change prediction in highways with low-cost gps/ins," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 1, pp. 180–185, March 2009.

[22] A. Eidehall and L. Petersson, "Statistical threat assessment for general road scenes using monte carlo sampling," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 1, pp. 137–147, March 2008.

[23] J. Wiest, M. Höffken, U. Kreßel, and K. Dietmayer, "Probabilistic trajectory prediction with gaussian mixture models," in *2012 IEEE Intelligent Vehicles Symposium*, June 2012, pp. 141–146.

[24] S. Yoon and D. Kum, "The multilayer perceptron approach to lateral motion prediction of surrounding vehicles for autonomous vehicles," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 1307–1312.

[25] D. Sierra González, J. S. Dibangoye, and C. Laugier, "High-speed highway scene prediction based on driver models learned from demonstrations," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2016, pp. 149–155.

[26] D. Sierra González, V. Romero-Cano, J. S. Dibangoye, and C. Laugier, "Interaction-aware driver maneuver inference in highways using realistic driver models," in *2017 IEEE International Conference on Intelligent Transportation Systems, ITSC*, Oct. 2017, pp. 1–8.

[27] S. H. Park, B. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1672–1678.

[28] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *CoRR*, vol. abs/1406.1078, 2014. [Online]. Available: http://arxiv.org/abs/1406.1078

[29] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 1800–1807.

[30] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 7794–7803.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[33] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. D. L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS 2017 Autodiff Workshop: The Future of Gradient-based Machine Learning Software and Techniques*, Dec. 2017.

[34] B. Coifman and L. Li, "A critical evaluation of the next generation simulation (ngsim) vehicle trajectory dataset," *Transportation Research Part B: Methodological*, vol. 105, pp. 362–377, 11 2017.

Authorized licensed use limited to: University of Houston. Downloaded on July 10,2020 at 20:17:58 UTC from IEEE Xplore. Restrictions apply.