Attention Based Vehicle Trajectory Prediction

Kaouther Messaoud, Itheri Yahiaoui, Anne Verroust-Blondet, and Fawzi Nashashibi

***Abstract*—Self-driving vehicles need to continuously analyse the driving scene, understand the behavior of other road users and predict their future trajectories in order to plan a safe motion and reduce their reaction time. Motivated by this idea, this paper addresses the problem of vehicle trajectory prediction over an extended horizon. On highways, human drivers continuously adapt their speed and paths according to the behavior of their neighboring vehicles. Therefore, vehicles' trajectories are very correlated and considering vehicle interactions makes motion prediction possible even before the start of a clear maneuver pattern. To this end, we introduce and analyze trajectory prediction methods based on how they model the vehicles interactions. Inspired by human reasoning, we use an attention mechanism that explicitly highlights the importance of neighboring vehicles with respect to their future states. We go beyond pairwise vehicle interactions and model higher order interactions. Moreover, the existence of different goals and driving behaviors induces multiple potential futures. We exploit a combination of global and partial attention paid to surrounding vehicles to generate different possible trajectory. Experiments on highway datasets show that the proposed model outperforms the state-of-the-art performances.**

***Index Terms*—Trajectory prediction, vehicles interactions, recurrent networks, multi-head attention, multi-modality.**

I. Introduction

In ORDER to navigate, self-driving vehicles need to understand the behavior of other traffic participants. As communications are not always possible, self-driving vehicles must perceive and anticipate the intentions of surrounding vehicles in order to plan comfortable proactive motions and avoid urgent reactive decisions and conflicts with others. In fact, motion prediction helps self-driving vehicles understand possible future situations and decide about a future behavior that minimizes the possible risks accordingly.

Motion behavior may be inferred by considering the features that characterises it. Vehicles' past states give relevant information about the dynamics, the direction and the speed of the performed maneuver. However, the trajectory taken by each vehicle in the future is not only dependent on its own state history: even the vehicle class impacts the motion pattern. In addition, the presence and actions of the neighboring vehicles have a great influence on a vehicle's behavior as well. Therefore, in this work, we propose to model the interactions between all the neighboring vehicles to represent the most relevant information about the social context with a focus on learning to capture long-range relations. In our approach, we attempt to mimic human reasoning, which pays a selective attention to a subset of surrounding vehicles in order to extract the elements that most influence the target vehicle's future trajectories while paying less attention to other vehicles. For example, a vehicle performing a lane change maneuver will pay more attention to the vehicles in the target lane than those in the other lanes. Consequently, its future behavior could be more dependent on distant vehicles in the target lane than the close ones in the other lanes.

This study is an extension of our previous work [1], which focuses on deploying multi-head attention in the task of trajectory prediction. We adopt the attention mechanism to derive the relative importance of surrounding vehicles with respect to their future motion: it selectively aggregates the features that model the interaction between the vehicles by a weighted sum of the features representing all the surrounding vehicles' trajectories and thus directly relates vehicles based on their correlation without regard to their distance. We also use multi-head attention in order to extract different types of interactions and combine them to capture higher order relationships. This provides a better understanding of the scene.

Drivers' behaviors are not deterministic. In similar driving situations, they can perform different maneuvers or even when doing the same maneuver, the execution can be different in terms of speed and pattern. Therefore, we propose a method that is able to predict a multi-modal finite set of trajectories that correspond to predicted trajectories conditioned on the degree of attention paid to the surrounding vehicles.

Quantitative and qualitative experiments are conducted to show the contribution of the model, and quantitative comparisons with recent approaches show that the proposed approach outperforms state-of-the-art accuracy in highway driving trajectory prediction.

II. Related Research

The task of vehicle motion forecasting has been addressed in the literature from different perspectives. Therefore, numerous vehicle motion prediction methods have recently been proposed. Here, we give an overview of the deployed methods, focusing on deep learning pattern based methods.

A. *Overall Motion Prediction Module*

We follow Rudenko *et al.* [2] who divide the motion prediction problem into three main components.

Manuscript received January 22, 2020; accepted March 30, 2020. Date of publication May 6, 2020; date of current version February 24, 2021. The work was supported by PIA French project CAMPUS (Connected Automated Mobility Platform for Urban Sustainability). *(Corresponding author: Kaouther Messaoud.)*

Kaouther Messaoud, Anne Verroust-Blondet, and Fawzi Nashashibi are with Inria Paris, 75012 Paris, France (e-mail: kaouther.messaoud@inria.fr; anne.verroust@inria.fr; fawzi.nashashibi@inria.fr).

Itheri Yahiaoui is with the CReSTIC, Université de Reims Champagne-Ardenne, 51100 Reims, France (e-mail: itheri.yahiaoui@univ-reims.fr).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIV.2020.2991952

1) Stimuli: The features that influence and determine the future intention of the target vehicle are mainly composed of target vehicle cues and environment information.

Target vehicle features: They enclose target vehicle past state observations (positions, velocities, etc.). Lenz *et al.* [3] use as input to their model only the current state of a set of neighboring vehicles in order to achieve the Markov Property. Other existing studies [1], [4]–[8] use a sequence of past features to benefit from extra temporal information in the prediction task.

Environment features: These are composed of:

- Static elements including static obstacles and environment geometry.

- Dynamic elements representing the other traffic participants.

2) Modeling Approach: Different representations of the motion model are used, which can be classified into:

Physics-based methods, where the future trajectory is predicted by applying explicit, hand-crafted, physics-based dynamical models [9]–[11]. These approaches basically build upon the motion's low level properties. Consequently, they are restricted to short-term motion prediction.

Pattern-based methods that learn the motion and behaviors of vehicles from data of observed trajectories. Aoude *et al.* [12] combine a physics-based approach with Gaussian Processes based motion patterns to generate probabilistically weighted feasible motions of the surrounding vehicles. Other methods divide the vehicle trajectory into a finite set of typical patterns named maneuvers. Tran and Firl [13] identify the vehicle maneuvers by comparing the likelihoods of the observed track for the constructed non-parametric regression models. Hermes *et al.* [14] cluster the motion patterns with a rotationally-invariant distance metric into maneuvers and predict vehicles trajectories by matching the observation data to the maneuvers. Schlechtriemen *et al.* [15] deploy a Naive Bayes Classifier followed by a Hidden Markov Model (HMM), where each state of the HMM corresponds to one of the maneuvers extracted from the naturalistic driving data. Houenou *et al.* [16] conceive a maneuvers recognition module, then, generate different continuous realizations of the predicted maneuver. The main limitation of these approaches is that they do not model the interactions between the neighboring vehicles on the future trajectory. Kafer *et al.* [17] tackle the task of joint pairwise vehicle trajectory prediction at intersections. They compare the observed motion pattern to the database and extract, for each vehicle, possible predicted trajectories independently. Then, they jointly compute, for each pair, the probability of possible trajectories.

Most recent studies deploy deep learning based methods. They will be detailed in the Section II-B.

Planning-based methods reason on the motion intent of rational agents. Sierra González *et al.* [18] deploy Markov Decision Process (MDPs) to represent the driver decision-making strategy. They model a vehicle's trajectory by a sequence of states. Then, they build a cost function using a linear combination of static and dynamic features parameterizing each state. Inverse Reinforcement Learning (IRL), accounting for risk-aversive vehicles' interactions, operates to learn the cost function parameters from demonstrations. They use Dynamic Bayesian Networks, in [19], to model vehicles' interactions.

Li *et al.* [20] extend Generative Adversarial Imitation Learning (GAIL) [21] and deploy it to predict the driver's future actions given an image and past states. The proposed method is able to imitate different types of human driving behavior in a simulated highway scenario. Rhinehart *et al.* [22] use a deep imitative model to learn and predict desirable future autonomous behavior. They train their model with an expert human behaviors dataset, and use it to generate expert-like paths to each of the precomputed goals.

3) Prediction: Vehicle intent prediction is divided into two main aspects: maneuver [4], [23] and trajectory prediction [5], [8], [24]. The former generates a high-level representation of the motion such as lane changing and lane keeping. The latter outputs the predicted state over time. Different forms of outputs are used in the motion prediction task. In [5], [8], the exact future positions are predicted. Others [6], [7], [25] deploy a multimodal solution using Gaussian mixture models over predicted states. Ridel *et al.* [26] generate the probability distributions over grids with multiple trajectory samples. Sampling generative models such as Generative Adversarial Networks (GANs) was used in [27]–[29]

B. Deep Learning Pattern-Based Motion Prediction

Motion prediction can be treated as a time series regression or classification problem. Recurrent Neural Networks (RNNs) are the main reason behind the significant advances 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 deployed as well in the tasks of maneuver and trajectory prediction.

Long Short Term Memories (LSTMs) are a particular implementation of RNNs. They are characterised by their ability to extract long-term relations between features. In other word, unlike other neural networks, they consider sequential information and model the dependency in inputs. They act by performing the same operations for every input item of a sequence while taking into consideration the computation of the previous input item.

LSTMs have been deployed, recently, for predicting driver future behaviors. Indeed, different LSTM-based models have been conceived going from simple LSTM with one or more layers in [3]–[5], [30] to different types of combinations and extensions: A dual LSTM architecture was adopted in [24]: the first LSTM extracts high-level driver behavior succeeded by a second for continuous trajectory generation. LSTM encoder decoder based architectures were deployed in [1], [6], [7], [31].

One of the most important parts in a driver intention prediction model is the surrounding vehicles' interaction extractor. It is also conceived differently in the state of the art. Some existing studies [3]–[6] implicitly infer the dependencies between vehicles. They feed a sequence of surrounding vehicles features as inputs to their model. Then, they accord to the LSTM the task of learning the influence of surrounding vehicles on the target vehicle's motion. Other approaches explicitly model the vehicles' interactions using several combinations of networks.

Alahi *et al.* [32] introduced the social LSTM concept for pedestrian trajectory prediction task. They encode the motion of each agent using an LSTM block. Then, they extract the interactions between agents by sharing the hidden states between all the LSTMs corresponding to a set of neighboring pedestrians. Hou *et al.* [33] use a structural-LSTM network to learn high-level dependencies between vehicles. Similar to social LSTM, they attribute one LSTM for each vehicle. Then, they use convolutional layers applying successive local operations followed by a maxpool layer. the spatial-neighboring LSTMs share their cell and hidden states by a radial connection. The output states of the LSTMs are treated recurrently in a deeper layer. The decoder generates all the predicted trajectories.

Deo *et al.* [7] extend the social pooling and deploy it for vehicle trajectory prediction task. They use an LSTM encoder to generate a representation of each vehicle trajectory. Then, they use convolutional layers applying successive local operations on the outputs from the encoders followed by a maxpool layer. Therefore, they generate a context vector that consists on a compact representation of the vehicles interactions. But successive local operations are not always sufficient. Furthermore, the generated context vector is independent of the target vehicle's state. Zhao *et al.* [27] extend the convolutional social pooling to simultaneous multi-agents trajectory prediction.

Multi-head attention mechanism was introduced by Vaswani *et al.* [34] for natural language processing purposes. A relational recurrent network based on attention mechanism was deployed in [8] for trajectory prediction. In [1], an attention-based non-local vehicle dependencies model that represents vehicles' interactions based on their importance to the target vehicle is introduced. The attention mechanism reduces the number of local operations by directly relating distant elements. The motion prediction results computed by this method on the NGSIM dataset [35], [36] improve those reported in [6], [7].

In this article, we extend our previous approach [1] to tackle the target vehicle trajectory prediction problem (cf. Section III) as follows:

- We focus on studying non-local social pooling using a multi-head attention mechanism. Therefore, we remove the convolution layer used to extract local interactions in our previous method [1].
- We expand our previous approach by exploiting additional information to boost our prediction. We follow [23] and, in order to take into account the social effect of the surrounding vehicles on the prediction target based on relative dynamics, we include additional information (velocity, acceleration) in the vehicle state vectors. We also integrate the vehicle class information since the type of the vehicle characterises its motion pattern.
- We investigate the interest of using multiple attention heads and we analyse the interactions extracted using each head. We also compare several ways of attention computation.
- We augment our architecture to generate a multi-modal solution based on a combination of partial and global attentions paid to the surrounding vehicles.

Experimental evaluations presented in Section IV show the benefits of using attention mechanisms to solve this problem.

III. Target Vehicle Trajectory Prediction

A. Problem Definition

The goal of this part is to predict the future trajectory of a target vehicle T, knowing its past tracks and the past tracks of its neighboring vehicles at observation time t_{obs}.

We have as input the past tracks of the target and its n neighboring vehicles. The input tracks of a vehicle i are defined as $\mathbf{X}_i = [\mathbf{x}_i^1, \ldots, \mathbf{x}_i^{t_{obs}}]$ where $\mathbf{x}_i^t = (x_i^t, y_i^t, v_i^t, a_i^t, class)$ is the state vector. We note $\mathbf{X}_T$ the state of the target vehicle T.

The coordinates of all the considered vehicles, 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 one direction of motion on the highway and to the direction perpendicular to it.

Our model outputs the parameters characterizing a probability distribution over the predicted positions of the target vehicle.

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

Where $\mathbf{y}^t = (x^t, y^t)$ is the predicted coordinates of the target vehicle.

Our model infers the conditional probability distribution $\mathbf{P}(\mathbf{Y}|\mathbf{X})$. The distribution over the possible positions at time $t \in \{t_{obs}+1, \ldots, t_{obs}+t_f\}$ can be presented as a bivariate Gaussian distribution with the 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 is the covariance matrix:

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

We evaluate our model by considering the mean μ^t values as the predicted positions $\mathbf{y}^t$.

B. Overall Model

It is crucial to understand the relationships and interactions that occur on the road to make realistic predictions about vehicle motions. Therefore, our model architecture is made up of three main components (cf. Fig. 1):

- *Encoding Layer* where the temporal evolution of the vehicle's trajectories and their motion properties are encoded by an LSTM encoder.
- *Attention module* which links the hidden states of the encoder and decoder. It explicitly extracts the importance of the surrounding vehicles based on their spatio-temporal encoding in determining the future motion of the target vehicle using different operations. Then, it forms a vector representing the context influence.
- *Decoding Layer* which receives the context vector containing the selected information about the neighboring vehicles and the target vehicle motion encoding and generates parameters of the distribution over the target vehicle's predicted future positions.

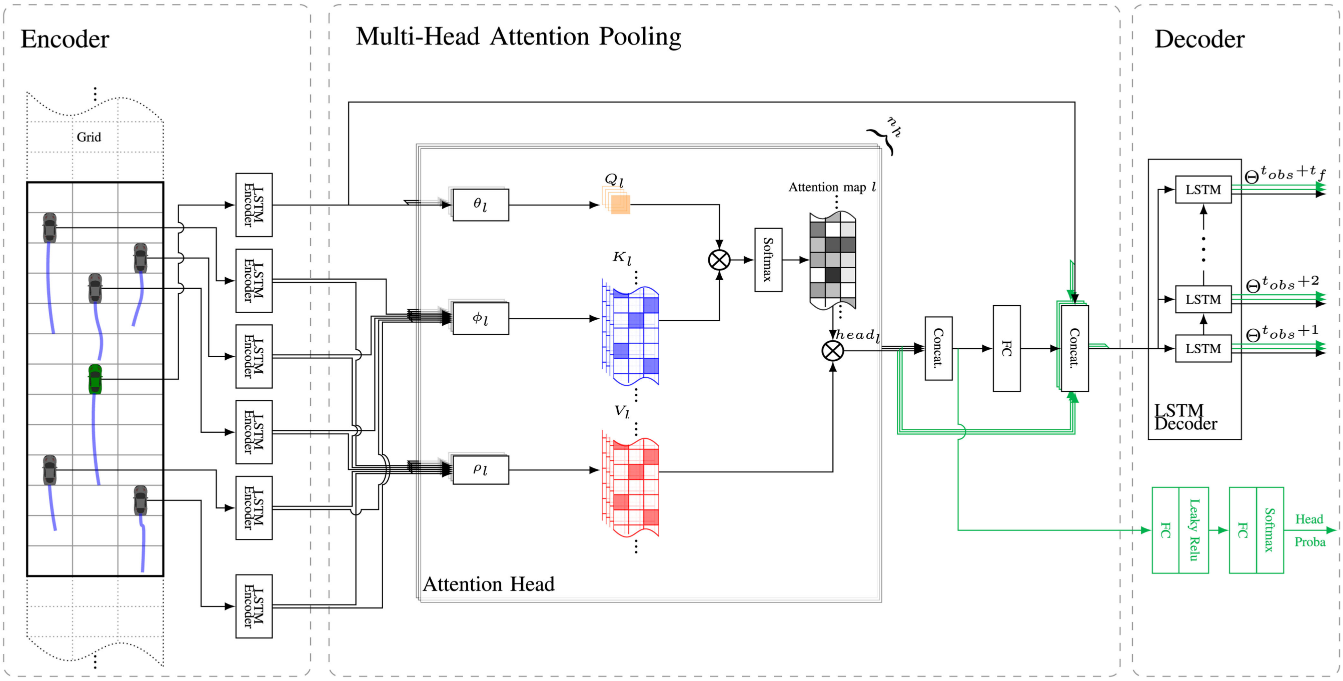


Fig. 1. Proposed Model: The LSTM encoders, with shared weights, generate a vector encoding of each vehicle motion. The multi-head attention module models the interactions between the target (green car) and the neighboring vehicles based on their importance. The decoder receives the interaction vector and the target vehicle encoding and generates a distribution for the predicted trajectory. The blocks added in green are the extension of the multi-head attention method to Multi-Modal Trajectory Prediction.

C. Trajectory Encoder

This encoding layer encodes the trajectories of the vehicles belonging to a neighborhood of the target vehicle at time $t = t_{obs}$. Unlike most of the previous studies that consider a restricted number of vehicles immediately around the target vehicle, we compute a grid over the surrounding area. This representation of the context has the following advantages:

- It represents the drivable areas.
- It enables us to consider all the vehicles present in the neighboring area without restriction.

Each state vector $\mathbf{x}_i^t$ of each vehicle i of the neighboring area is embedded using a fully connected layer to form an embedding vector e_i^t.

$$e_i^t = \Psi(\mathbf{x}_i^t; W_{emb})$$

where $\Psi()$ is a fully connected function with LeakyReLU non linearity, W_{emb} is the learnt the embedding weights.

The LSTM encoder is fed by the embedding vectors of each vehicle i for time steps $t = 1, \ldots, t_{obs}$:

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

h_i^t is the hidden state vector of the ith neighboring vehicle at time t. We note h_T^t the hidden state vector of the target vehicle at time t. $W_{encoder}$ are the LSTM encoders weights. Each LSTM encoder share the same weights $W_{encoder}$.

We built a $3D$ spatial grid H composed of the neighboring vehicles' hidden states at time t_{obs} based on their positions at time t_{obs}.

$$H(n, m, :) = \delta_{nm}(x_i^{t_{obs}}, y_i^{t_{obs}}) h_i^{t_{obs}} \quad \forall i \in \mathcal{A}_T$$

$\delta_{nm}(x, y)$ is an indicator function that equals 1 if (x, y) is in the cell (n, m) and 0 otherwise. $\mathcal{A}_T$ consists of the set of surrounding vehicles present in the considered area.

The columns correspond to the three lanes $(M = 3)$. The considered spacial area corresponding to the grid is centered on the target vehicle position and sized of (N, M). It covers a longitudinal distance of 90 m with a grid cell size of 4.5 m. We note C the dimension of the trajectory encoding vectors $h_i^{t_{obs}}$ and we reshape the grid H to (NM, C).

D. Vehicle Interaction Modules

As the behavior of vehicles on a highway could be highly correlated, it is important to consider the interactions between the vehicles when predicting their future motion. Attention is used to capture long-range spatio-temporal dependencies. The attention module explicitly models the interactions between the target vehicle and the other vehicles in the grid H and selects the surrounding vehicles to pay attention to when computing the future trajectory of the target vehicle.

Instead of computing vehicle relationships at each time step, which is computationally expensive, we use the hidden states of the encoder LSTM computed at the observation time as inputs to the attention module. These hidden states are projected into a high-dimensional space, if we consider all the attention heads. The vehicles interactions can be exploited as follows:

- The hidden state of the target vehicle is mapped to a query $Q_l = \theta_l(h_T^{t_{obs}}, W_{\theta_l})$
- The grid is mapped to form the keys $K_l = \phi_l(H, W_{\phi_l})$ and the values $V_l = \rho_l(H, W_{\rho_l})$.

W_{θ_l}, W_{ϕ_l} and W_{ρ_l} are the weight matrices that will be learned in each attention head l.

An attention feature $head_l$ is then calculated as a weighted sum of values v_{lj}, where the attention weights, α_{lj}, weight the effect of surrounding vehicles on the target vehicle future motions, based on their relative dynamics.

We investigate three possible ways to compute the attention weights α_l:

$$head_l = \sum_{j=1}^{NM} \alpha_{l_j} v_{l_j}$$

1) α-Attention: Attention weights are computed from the encoding vectors of the surrounding vehicles independently of the target vehicle state. They are computed using a $tanh$ function and a fully-connected layer.

$$\alpha_l = softmax(w_l^T tanh(K_l))$$

w_l is a learned weight, and $\alpha_l \in R^{1\times NM}$ is the lth attention.

2) Dot-Product Attention: The weights represent the effect of an interaction between a pair of vehicles based on their relative dynamics. They are the product of the query Q with keys K.

$$\alpha_l = softmax\left(\frac{Q_l K_l^T}{\sqrt{d}}\right)$$

$Q_l K_l^T$ is matrix multiplication used to calculate dot product similarities. d is a scaling factor that equals to the dimensionality of the projection space.

3) Concatenation Attention: The pairwise relation can be also represented by concatenation operation, as in [37], [38].

$$\alpha_l = softmax(w_l^T concat(repeat(Q_l), K_l))$$

One can notice that dot-product and concatenation attentions consider pairwise inter-relationships, whereas α-attention does not.

E. High Order Interaction

We deploy a higher order interaction extractor based on multi-head attention to retain different types of spatio-temporal relationships. The use of multi-head is inspired by the Transformer [34] architecture. In fact, a single learned attention feature mainly focuses on one inter-related subgroup of vehicles that may represent a single aspect of the possible spatio-temporal relationships occurring in the neighborhood of the target vehicle. In order to extend the attention to higher order interactions, different queries, keys and values are generated n_h times in parallel, in n_h attention heads, with different learned linear projections Q_l, K_l and V_l, $l \in [1, n_h]$.

The n_h generated attention features represent n_h subgroups of vehicles inter-related with the target vehicle. These representations are concatenated and dynamically weighted to extract complex interactions between the different subgroups.

$$z = Concat(head_1, \ldots, head_{nh}) W^O$$

z is the compact context vector that combines interaction information of all the vehicles.

F. Trajectory Prediction

LSTM Decoder is fed by the context vector z, which contains the selected information about the vehicles interactions, and the motion encoding of the target vehicle: $h_{dec} = Concat(h_T^{t_{obs}}, z)$. It generates the predicted parameters of the distributions over the target vehicle's estimated future positions for time steps $t = t_{obs} + 1, \ldots, t_{obs} + t_f$.

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

where Θ^t is the predicted parameters of the positions distribution at time t, $\Lambda()$ is a fully connected function followed by a LeakyReLU non linearity, W_{dec} are the learnt weights of the LSTM decoder and h_{dec}^{t-1} is the hidden state vector of the decoder at time $t-1$.

Our model is trained by minimizing the following negative log-likelihood loss function:

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

G. Multi-Modal Trajectory Prediction

Given the history of a vehicle's motion, there are many plausible future trajectories. Generating one trajectory for motion forecasting tends to be the average of the possible motions. When a driver decides to perform a specific motion, he directs his attention to a set of neighboring vehicles. For example, a driver exerting a lane change maneuver will mainly pay attention to the vehicles in the target lane. Therefore, from each considered set of neighboring vehicles, we may derive a plausible future trajectory. To do so, we deploy a muti-head attention as described before and, we proceed as following (Figure 1):

The decoder receives n_h encodings of the scene based on different attention heads.

$$h_{dec}^l = Concat(h_T^{t_{obs}}, head_l, z) \;\; l \in [1, n_h]$$

Then, using each encoding, the decoder generates a plausible trajectory $\mathbf{Y}_{pred}^l$.

During the training, we compute only the loss $L_{nll}(\mathbf{Y}_{pred}^{l^*})$ corresponding to closest predicted trajectory to the ground-truth $\mathbf{Y}_{pred}^{l^*}$. Therefore, the position outputs are updated only for the minimum error.

We augment the proposed architecture by a network composed of two fully connected layers separated by a non-linear function. It receives the outputs of all the attention heads and decides about the probability (p_l, $l \in [1, n_h]$) of each produced trajectory being the closest to the real one. This network outputs the likelihood of the n_h predicted trajectories. For this purpose, we add to the loss function a second term, which is the classification cross-entropy loss with n_h classes [25].

$$L_{Class} = -\sum_{l=1}^{n_h} \delta_{l^*}(l) \log(p_l)$$

where δ is function equal to 1 if $l = l^*$ and 0 otherwise.

Therefore, the probability of the best matching trajectory p_{l^*} is trained to become closer to 1, and the probabilities of the

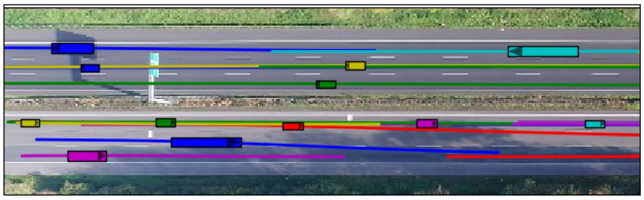


Fig. 2. Highway drone dataset highD [39]: recordings cover about 420 m of German highways.

others to 0. This makes the probability outputs updated for all the attention heads.

During the evaluation, we compute the loss function by taking the selected trajectory $\mathbf{Y}^s_{pred}$ having the maximum probability p_s (Note that $\mathbf{Y}^s_{pred}$ may be different from $\mathbf{Y}^{l^*}_{pred}$).

The proposed network causes each attention head to specialize in extracting interaction features characterizing a distinct class of driver behavior without requiring explicit labels.

IV. Experimental Evaluations

Evaluations have been performed on public driving datasets that are described in Section IV-A. The approach proposed in Section III is compared with state-of-the-art quantitatively. Qualitative results are also presented for further analysis.

A. Datasets

1) highD [39]: captured in 2017 and 2018. It was recorded by camera-equipped drones from an aerial perspective of six different German highways at 25 Hz. It is composed of 60 recordings of about 17 minutes each, covering a segment of about 420 m of two-way 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 [35], [36]: a publicly available large dataset captured in 2005 at 10 Hz, widely studied and used in the literature, especially in the task of future intention prediction of vehicles [3]–[7]. 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 8 s of the trajectories composed of a track history of 3 s and a prediction horizon of 5 s. We downsample each segment to get only 5 fps to reduce the complexity of the model.

B. Training and Implementation Details

We deploy LSTM encoder with 64 units (C = 64) and decoder with 128 units. The dimension of the embedding space is 32. We use different number of parallel attention operations applied on the projected vectors of size d = 32. The batch size is 128 and the adopted optimizer is Adam [40]. The model is implemented using PyTorch [41].

C. Evaluation Metric

In our evaluation, we use Root of the Mean Squared Error (RMSE) metric since it averages the distance between predicted trajectories and the ground truth.

$$L_{RMSE} = \sqrt{\frac{1}{t_f}\sum_{t=t_{obs+1}}^{t_{obs}+t_f}(x^t_T - x^t_{pred})^2 + (y^t_T - y^t_{pred})^2}$$

We use the means of the predicted distributions over the future trajectories to calculate the RMSE.

D. Models Compared

Evaluations have been performed on the following models that all consider the interactions between surrounding vehicles. They are fed with the track history of the target and the surrounding vehicles and output distributions over the future trajectory of the target vehicle.

- *Maneuver-LSTM (M-LSTM) [6]:* 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) [32]:* social encoder-decoder using fully connected pooling.
- *Convolutional Social Pooling (CS-LSTM) [7]:* social encoder-decoder using convolutional pooling. (*CS-LSTM(M)*) generates multi-modal trajectory predictions based on six maneuvers (2 longitudinal and 3 lateral).
- *Multi-Agent Tensor Fusion (MATF GAN) [27]:* the model encodes the scene context and vehicles' past trajectories, then, deploys convolutional layers to capture interactions. Finally, the decoder generates the predicted trajectories, using adversarial loss.
- *Non-local Social Pooling (NLS-LSTM) [1]:* combines local and non local operations to generate an adapted context vector for social pooling. Five attention heads are used in this approach.
- *Multi-head Attention Social Pooling (MHA-LSTM):* This is the model described in this paper using multi-head dot product attention with $\mathbf{x}^t_i = (x^t_i, y^t_i)$, i.e. without using velocity, acceleration and class information for each vehicle and with four attention heads.
- *Multi-head Attention Social Pooling (MHA-LSTM(+f)): MHA-LSTM* with additional input features (velocity, acceleration and class) and with three attention heads.

E. Target Vehicle Trajectory Prediction: Quantitative Evaluation

1) Overall Evaluation: Table I shows the RMSE values for the models being compared on the NGSIM and highD datasets. Previous studies [6], [7], [32] compare their results with independent prediction models to put emphasis on the importance of considering surrounding agents. In this work, we not only show that considering surrounding vehicles is a key factor to perform

TABLE I
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)	MATF GAN	NLS-LSTM	MHA-LSTM	MHA-LSTM(+f)
highD	1	-	0.22	0.22	0.23	-	0.20	0.19	**0.06**
	2	-	0.62	0.61	0.65	-	0.57	0.55	**0.09**
	3	-	1.27	1.24	1.29	-	1.14	1.10	**0.24**
	4	-	2.15	2.10	2.18	-	1.90	1.84	**0.59**
	5	-	3.41	3.27	3.37	-	2.91	2.78	**1.18**
NGSIM	1	0.58	0.65	0.61	0.62	0.66	0.56	0.56	**0.41**
	2	1.26	1.31	1.27	1.29	1.34	1.22	1.22	**1.01**
	3	2.12	2.16	2.09	2.13	2.08	2.02	2.01	**1.74**
	4	3.24	3.25	3.10	3.20	2.97	3.03	3.00	**2.67**
	5	4.66	4.55	4.37	4.52	4.13	4.30	4.25	**3.83**

trajectory prediction but we also model their interactions in a more efficient way.

To compare our model, we consider the results reported in [6], [7] on the NGSIM dataset and we train S-LSTM and CS-LSTM on highD dataset as well. We train and test the approaches on the NGSIM and highD datasets separately and we notice that the RMSE values obtained on the NGSIM dataset are higher than the ones computed on the highD dataset. This may be due to the difference in size of the two datasets: highD contains about 12 times more vehicles than NGSIM. It can be also caused by annotation inaccuracies resulting in physically unrealistic vehicle behaviors in the NGSIM dataset, as observed by Coifman *et al.* [42].

Anyway, examining the RMSE values for either NGSIM or highD datasets leads to the same order for the proposed methods. Our attention-based approaches (*NLS-LSTM*, *MHA-LSTM* and *MHA-LSTM(+f)*) perform better than the others. *MHA-LSTM* reduces the prediction error by about 10% compared to the CS-LSTM while having comparable execution time. With *MHA-LSTM(+f)*, we investigate the use of additional features like the speed and acceleration. We notice that this leads to significant improvements in the motion prediction accuracy, as *MHA-LSTM(+f)* outperforms all the methods. This consolidates our assumption that the relation between vehicles is not only related to their positions but also to their dynamics. The class of transportation (truck or car) also characterizes the speed and pattern of the motion. Therefore, these results indicate that multi-head attention better models the interdependencies of vehicle motion than convolutional social pooling. Moreover, this suggests that considering the relative importance of surrounding vehicles using both positions and dynamics when encoding the context is better than focusing on local dependencies.

2) Effects of Using Multiple Attention Heads: In order to evaluate the influence of the number of attention heads on the prediction accuracy, let us examine the RMSE values obtained by *MHA-LSTM* on the highD dataset with 2, 3 4, 5 and 6 attention heads on Table II. We notice that using several attention heads improves the prediction accuracy since each attention head represents a set of weights capturing one aspect of the effect of surrounding vehicles on the target vehicle. In addition, combining the attention vectors helps extract higher order relations. The best performance is reached with four attention heads.

We have conducted further experiments to evaluate the benefits of adding extra features, including explicit vehicle dynamics and class (*MHA-LSTM(+f)*). We observe that we outperform previous results when using different numbers of attention heads.

TABLE II
RMSE in Meters Over a 5-Second Prediction Horizon for Different Numbers of Attention Heads on the HighD Dataset

Heads Time(s)	2	3	4	5	6
1	0.21	0.20	**0.19**	0.20	0.21
2	0.61	0.61	**0.55**	0.57	0.59
3	1.20	1.19	**1.10**	1.13	1.16
4	1.96	1.99	**1.84**	1.87	1.92
5	2.95	3.01	**2.78**	2.83	2.93

TABLE III
RMSE in Meters Over a 5-Second Prediction Horizon for Different Attention Operations on the HighD Dataset

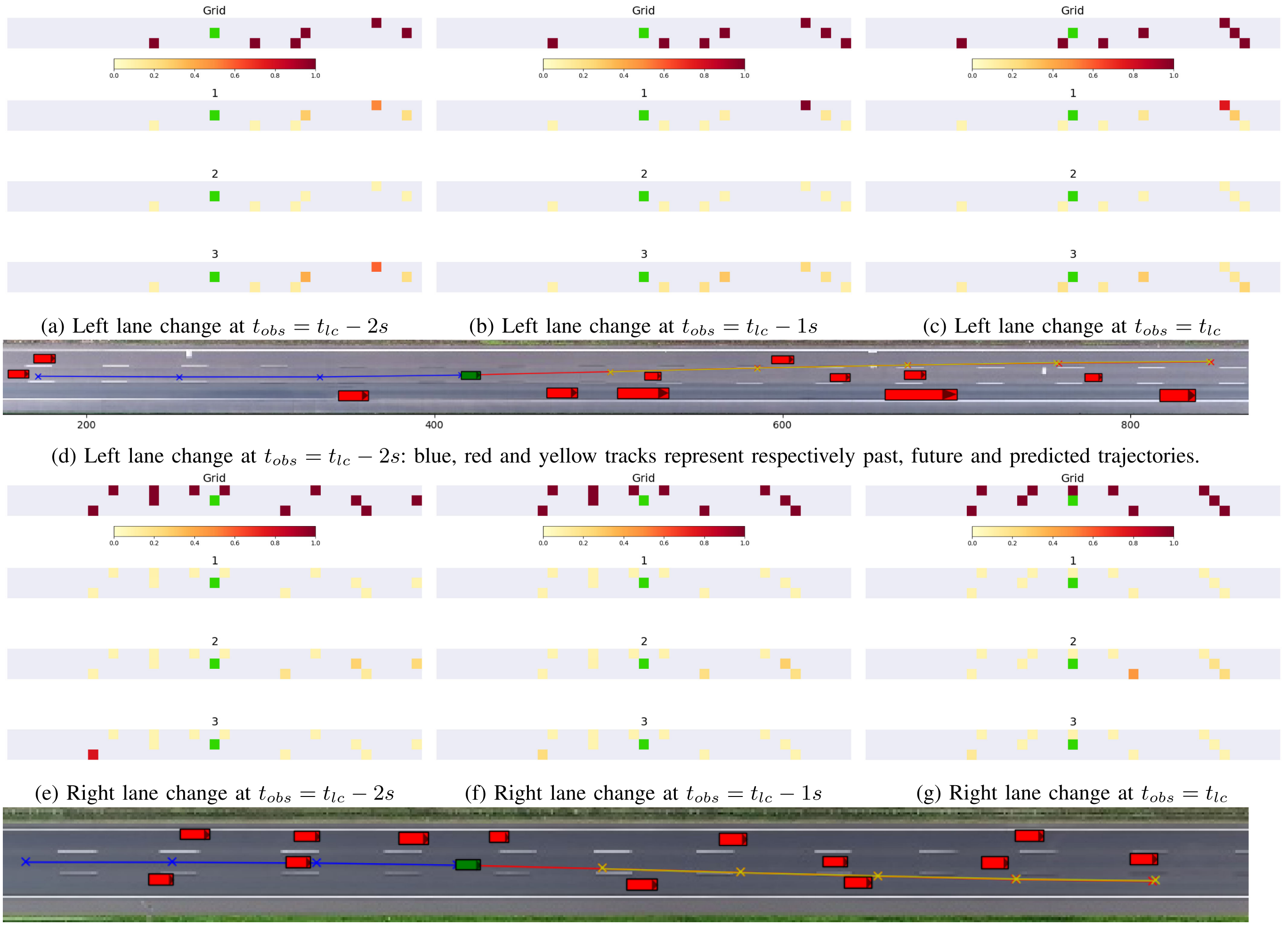
Methods Time(s)	$\alpha-$attention	Dot product	Concatenation
1	0.06	**0.06**	0.07
2	0.10	**0.09**	0.11
3	0.26	**0.24**	0.25
4	0.62	**0.59**	0.61
5	1.25	**1.18**	1.20

TABLE IV
Longitudinal and Lateral Errors in Meters Over a 5-Second Prediction Horizon for Different Maneuvers

Maneuver	RLC		LLC		LF	
Error	Long	Lat	Long	Lat	Long	Lat
1	0.07	0.03	0.20	0.03	0.05	0.01
2	0.12	0.06	0.32	0.07	0.07	0.02
3	0.34	0.18	0.42	0.19	0.22	0.06
4	0.79	0.43	0.88	0.45	0.54	0.14
5	1.43	0.76	1.74	0.78	1.10	0.22

Considering the trade-off between the complexity of calculation and the *MHA-LSTM(+f)* RMSE corresponding to different numbers of attention heads, we choose to deploy three attention heads in the experiments that follow.

3) Comparison of Attention Methods: In Table III, we show the performances of the three possible ways to compute the attention weights in *MHA-LSTM(+f)*, named α-attention, dot product attention, and concatenation attention presented in Section III-D. One can note that dot product and concatenation attentions outperform the α-attention. Therefore, we conclude that both the dynamics of the surrounding vehicles and their relationships with the target vehicle are of great importance for trajectory prediction.



(a) Left lane change at $t_{obs} = t_{lc} - 2s$ (b) Left lane change at $t_{obs} = t_{lc} - 1s$ (c) Left lane change at $t_{obs} = t_{lc}$

(d) Left lane change at $t_{obs} = t_{lc} - 2s$: blue, red and yellow tracks represent respectively past, future and predicted trajectories.

(e) Right lane change at $t_{obs} = t_{lc} - 2s$ (f) Right lane change at $t_{obs} = t_{lc} - 1s$ (g) Right lane change at $t_{obs} = t_{lc}$

(h) Right lane change at $t_{obs} = t_{lc} - 2s$: blue, red and yellow tracks represent respectively past, future and predicted trajectories.

Fig. 3. Three heads attention maps for two different lane change maneuvers. For visualisation, The target vehicle is added in green in the center of all the maps. The driving direction is from left to right.

4) Error evaluation per lane change: In order to complete the evaluation of our approach, we use the trained model *MHA-LSTM(+f)* to estimate the lateral and longitudinal errors obtained while carrying out right (RLC), left (LLC) lane change maneuvers or lane Following (LF) in the test set of the highD dataset (cf. Table III).

One can notice that the observed lateral error is low even during lane changes maneuvers (5% of the test data). This demonstrates the effectiveness of our method in predicting lane changes. It may also be observed that the longitudinal and lateral errors are greater during the LLC maneuvers. This may be due to the fact that the vehicle often speeds up when it performs LLC, which is not the case for the other maneuvers (LF or RLC).

F. Qualitative Analysis of Predictions

To understand which vehicles are taken into account by each attention head in our method, in Fig. 3 we present the attention maps corresponding to two lane change maneuvers carried out by the target vehicle. More precisely, a left lane change and a right lane change are shown and the attention maps are computed at times $t_{obs} = t_{lc} - 2\,s$, $t_{obs} = t_{lc} - 1\,s$ and $t_{obs} = t_{lc}$ where t_{lc} is the time of crossing the lane mark during the lane change maneuver. Each attention map corresponds to an attention head. The target vehicle is shown in green in the center of the attention map, the grey rectangular region corresponds to the 2D drivable area described by the grid H and the colors of the other vehicles indicate the attention weight associated to them in the attention head (they are darker when their attention weight increases).

We can remark that each attention head focuses on a subset of vehicles in the grid that are crucial to determining the future trajectory of the target vehicle. Moreover, like a human driver, most of the attention is directed to vehicles in front of the target vehicle, the vehicles behind it being less considered.

We also notice that, in each example, one attention head considers all the vehicles in the grid equally (attention head 2 for the left lane change and attention head 1 for the right lane change). Moreover, at time $t_{lc} - 2\,s$, attention map 3 is such that the most important vehicles belong to the target lane even though other vehicles are closer to the target vehicle in another lane. This consolidates our assumption that the closest neighbors do not always have the strongest influence on the target vehicle.

TABLE V
NEIGHBOR VEHICLES STATES

Example 1	Vehicle	Preceding	Lead		
	State	Sl	S +		
Example 2	Vehicle	Preceding	Following	Lead	Rear
	State	S +	F	Sl +	F -

Some other factors like the speed and the vehicle's lane are also essential for correctly estimating the importance of a neighbor. To emphasise that aspect, we consider the relative speeds of the vehicles surrounding the target vehicle and belonging either to the same lane as the target vehicle or to the target lane in examples 1 and 2. Table V summarizes the states of the considered interacting vehicles.

- Preceding, following: a vehicle belonging to the same lane as the target vehicle and preceding or following it.
- Lead, rear: a vehicle belonging to the target lane and positioned ahead or behind the target vehicle.
- S, Sl, F: same speed, slower, faster than the target vehicle respectively.
- $-$, $+$: decelerating, accelerating respectively.

In example 1, the preceding vehicle is slower than the target vehicle. The latter has two possible maneuvers: either to continue in the same lane and decelerate, or to accelerate and make a left lane change. In the left lane, the lead vehicle is distant to the target vehicle and has comparable velocity. This makes the lane change maneuver more likely.

In example 2, the preceding vehicle is accelerating and the following one is faster than the target vehicle. Therefore, the target vehicle has two options, either to accelerate or to make a right lane change.

In these two examples, we notice that even 2 seconds before performing a lane change, the target vehicle focuses mainly on the vehicles that belong to the target lane and which may have an influence on its future speed. Indeed, in both cases, the target vehicle performs the lane change while accelerating or decelerating according to the situation.

G. Multi-Modal Trajectory Prediction

Using multi-Modal Trajectory Prediction, we model the uncertainties of the future and acknowledge the existence of multiple possible paths. Generating one solution trajectory tends to average all the possible trajectories which may lead to unrealistic predicted behaviors. To address this problem, we use each attention head to specialize for a distinct class of driver behavior. In the following experiment, we use a combination of each of the three attention head and the global attention to generate three different possible trajectories.

Table VI and Figure 4 show the RMSE in meters over a 5-second prediction horizon for the generated trajectories using three attention heads for the RMSE values obtained as follows:

- Min and Max RMSE were computed by selecting at each instant the trajectory having respectively the minimum and maximum RMSE.
- H_l represents the RMSE of the trajectory generated by the attention head l, $l \in [1, 3]$.

TABLE VI
RMSE IN METERS OVER A 5-SECOND PREDICTION HORIZON FOR THE GENERATED TRAJECTORIES HIGHD DATASET

Time(s)	Min	Max	H_1	H_2	H_3	Proba
1	0.07	0.13	0.08	0.11	0.10	0.08
2	0.12	0.21	0.14	0.20	0.13	0.14
3	0.19	0.53	0.36	0.45	0.30	0.31
4	0.36	1.18	0.81	0.91	0.73	0.67
5	0.69	2.13	1.51	1.58	1.38	1.24

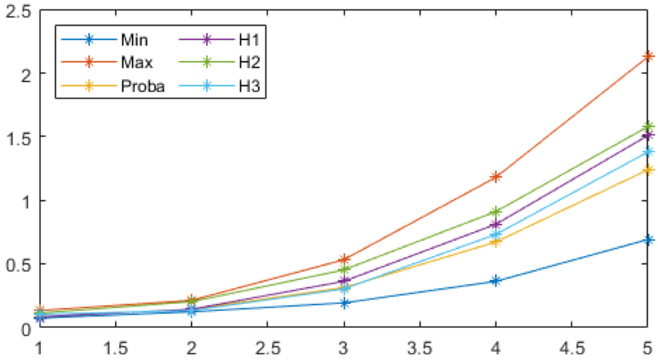


Fig. 4. RMSE in meters over a 5 second prediction horizon for the generated trajectories.

- Proba RMSE is obtained by computing the RMSE values of the trajectory computed by one of the three attention heads and having the maximum probability at t_{obs}.

We notice that one of the generated possible trajectories presents lower prediction error than the one solution trajectory by comparing the Min RMSE to the results in Table VI. Moreover, choosing the trajectory that has the best probability of predicting the target trajectory gives better results than systematically selecting the trajectory computed by one attention head (either H_1, H_2 or H_3). However, the network for trajectory selection does not always guide us to the trajectory with minimum loss, which justifies the difference between the min and probability based losses.

V. CONCLUSION

This work proposed an adapted attention-based method for modeling vehicle interactions during the tasks of vehicle trajectory prediction on highways. We extended our first method to acknowledge the future uncertainties and generate a multi-modal solution presenting different possible future trajectories. The proposed method caused each attention head to specialize in extracting interaction features characterizing a distinct class of driver behavior without requiring explicit labels. Experiments showed that our approach *MHA-LSTM(+f)* significantly outperforms the state-of-the-art on two naturalistic large-scale driving datasets based on the RMSE metric. Furthermore, the presented visualisation of the attention maps enabled us to recognize the importance and the dependencies between vehicles. It confirmed that the attention is directed based on the future maneuver. This justified our choice to use each attention head to generate a possible future trajectory.

Our evaluation results confirmed our intuitions: the importance of the relative dynamics and the efficiency of multi-head attention mechanism in modeling interactions between vehicles to predict vehicle trajectories in a highway scenario.

Our proposed approach can be extended to consider heterogeneous and mixed traffic scenarios with different road users, such as buses, trucks, cars, scooters, bicycles, or pedestrians. However, further information about the road structure should be integrated in our model for better representation of different driving scenes.

References

[1] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Non-local social pooling for vehicle trajectory prediction," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2019, pp. 975–980.

[2] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," 2019, *arXiv:1905.06113*.

[3] D. Lenz, F. Diehl, M. T. Le, and A. Knoll, "Deep neural networks for Markovian interactive scene prediction in highway scenarios," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2017, pp. 685–692.

[4] D. J. Phillips, T. A. Wheeler, and M. J. Kochenderfer, "Generalizable intention prediction of human drivers at intersections," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2017, pp. 1665–1670.

[5] F. Altché and A. de La Fortelle, "An LSTM network for highway trajectory prediction," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, Oct. 2017, pp. 353–359.

[6] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2018, pp. 1179–1184.

[7] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proc. IEEE Conf. Comput. Vision Pattern Recogn. Workshops*, Jun. 2018, pp. 1468–1476.

[8] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Relational recurrent neural networks for vehicle trajectory prediction," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Oct. 2019, pp. 1813–1818.

[9] H. Veeraraghavan, N. Papanikolopoulos, and P. Schrater, "Deterministic sampling-based switching Kalman filtering for vehicle tracking," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Sep. 2006, pp. 1340–1345.

[10] A. Barth and U. Franke, "Where will the oncoming vehicle be the next second?" in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2008, pp. 1068–1073.

[11] R. T. Moreo and M. A. Z. Izquierdo, "IMM-based lane-change prediction in highways with low-cost GPS/INS," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 180–185, Mar. 2009.

[12] G. Aoude, J. Joseph, N. Roy, and J. How, "Mobile agent trajectory prediction using Bayesian nonparametric reachability trees," in *Proc. AIAA Infotech at Aerosp. Conf. Exhibit*, Mar. 2011.

[13] Q. Tran and J. Firl, "Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2014, pp. 918–923.

[14] C. Hermes, C. Wohler, K. Schenk, and F. Kummert, "Long-term vehicle motion prediction," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2009, pp. 652–657.

[15] J. Schlechtriemen, A. Wedel, J. Hillenbrand, G. Breuel, and K. Kuhnert, "A lane change detection approach using feature ranking with maximized predictive power," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2014, pp. 108–114.

[16] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 4363–4369.

[17] E. Käfer, C. Hermes, C. Wöhler, H. Ritter, and F. Kummert, "Recognition of situation classes at road intersections," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 3960–3965.

[18] D. Sierra González, J. S. Dibangoye, and C. Laugier, "High-speed highway scene prediction based on driver models learned from demonstrations," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, Nov. 2016, pp. 149–155.

[19] D. Sierra González, V. Romero-Cano, J. S. Dibangoye, and C. Laugier, "Interaction-aware driver maneuver inference in highways using realistic driver models," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, Oct. 2017, pp. 1–8.

[20] Y. Li, J. Song, and S. Ermon, "Infogail: Interpretable imitation learning from visual demonstrations," in *Proc. Adv. Neural Inf. Process. Syst. 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017, pp. 3812–3822.

[21] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Proc. Adv. Neural Inf. Process. Syst. 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2016, pp. 4565–4573.

[22] N. Rhinehart, R. McAllister, and S. Levine, "Deep imitative models for flexible inference, planning, and control," 2018, *arXiv:1810.06544*. [Online]. Available: http://arxiv.org/abs/1810.06544

[23] W. Ding, J. Chen, and S. Shen, "Predicting vehicle behaviors over an extended horizon using behavior interaction network," in *Proc. Int. Conf. Robot. Autom.*, May 2019, pp. 8634–8640.

[24] 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 *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, Nov. 2018, pp. 1441–1446.

[25] H. Cui *et al.*, "Multimodal trajectory predictions for autonomous driving using deep convolutional networks," 2018, *arXiv:1809.10732*.

[26] D. A. Ridel, N. Deo, D. F. Wolf, and M. M. Trivedi, "Scene compliant trajectory forecast with agent-centric spatio-temporal grids," 2019, *arXiv:1909.07507*.

[27] T. Zhao *et al.*, "Multi-agent tensor fusion for contextual trajectory prediction," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2019, pp. 12118–12126.

[28] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2018, pp. 2255–2264.

[29] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive GAN for predicting paths compliant to social and physical constraints," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2019, pp. 1349–1358.

[30] A. Zyner, S. Worrall, J. Ward, and E. Nebot, "Long short term memory for driver intent prediction," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2017, pp. 1484–1489.

[31] 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 *Proc. IEEE Intell. Veh. Symp.*, Jun. 2018, pp. 1672–1678.

[32] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. F. Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2016, pp. 961–971.

[33] L. Hou, L. Xin, S. Li, B. Cheng, and W. Wang, "Interactive trajectory prediction of surrounding road users for autonomous driving using structural-LSTM network," *IEEE Trans. Intell. Transp. Syst.*, vol. 27, no. 11, pp. 4615–4625, Nov. 2020.

[34] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. 30*. Red Hook, NY, USA: Curran Associates, Dec. 2017, pp. 5998–6008.

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

[36] J. Colyar and J. Halkias, "US Highway I-80 dataset," in *Federal Highway Administration (FHWA)*, Tech. Rep. FHWA-HRT-06-137, 2006.

[37] A. Santoro *et al.*, "A simple neural network module for relational reasoning," in *Proc. Adv. Neural Inf. Process. Syst. 30*. Red Hook, NY, USA: Curran Assoc., Dec. 2017, pp. 4967–4976.

[38] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[39] 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 *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, Nov. 2018, pp. 2118–2125.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representations*, May 2015.

[41] A. Paszke *et al.*, "Automatic differentiation in pytorch," in *Proc. NIPS Autodiff Workshop: Future Gradient-Based Mach. Learn. Softw. Techn.*, Dec. 2017.

[42] B. Coifman and L. Li, "A critical evaluation of the next generation simulation (NGSIM) vehicle trajectory dataset," *Transp. Res. Part B: Methodological*, vol. 105, pp. 362–377, Nov. 2017.

Kaouther Messaoud is currently a Ph.D. student at INRIA in Paris and member of Robotics and Intelligent Transportation Systems (RITS) team. She is also collaborating with LISA lab team at UCSD. She received her engineering degree at Tunisia Polytechnic School. Her research interests are modeling vehicles interactions and predicting drivers behaviors using deep learning based approaches.

Itheri Yahiaoui is an Assistant Professor of Computer Science at Université de Reims Champagne-Ardenne and a member of the CReSTICLab. She is also an Associate Researcher in the RITS research group of INIRA Paris. She received her PhD degree in Computer Science, in 2003, from the Ecole Nationale Supérieure des Télécommunications (TELECOM Paris). Her research interests include multimedia indexing, pattern recognition, image analyses and time series forcasting.

Anne Verroust-Blondet is a senior research scientist in the RITS research group of Inria Paris. She obtained her “Thèse de 3e cycle” and her “Thèse d’Etat” in Computer Science (respectively in database theory and in computer graphics) from the University of Paris-Sud. Her current research interests include 2D and 3D object recognition and geometric modeling, environment perception, decision and planning in the context of intelligent transportation systems.

Fawzi Nashashibi received the master’s degree in automation, industrial engineering, and signal processing from the Laboratory for Analysis and Architecture of Systems/Centre Nationnal de la Recherche Scientifique (LAAS/CNRS), Toulouse, France, the Ph.D. degree in robotics from the LAAS/CNRS Laboratory, Toulouse University, and the HDR Diploma (accreditation to research supervision) from Pierre and Marie Curie University (Paris VI), Paris, France. Since 1994, he has been a Senior Researcher and the Program Manager with the Robotics Center, Mines ParisTech, Paris. Since 2010, he has been a Senior Researcher and a Program Manager with the RITS Team, Inria, Paris-Rocquencourt, France. He played key roles in over 50 European and national French projects, some of which he has coordinated. He is the author of numerous publications and patents in the field of ITS and ADAS systems. In this field, he is known as an international expert. He is a member of the IEEE ITS Society and the Robotics and Automation Society. He is an Associate Editor of several IEEE international conferences, such as ICRA, IROS, IV, ITSC, and ICARCV.