2021 IEEE Intelligent Vehicles Symposium (IV)
July 11-17, 2021. Nagoya, Japan

Trajectory Prediction for Autonomous Driving based on Multi-Head Attention with Joint Agent-Map Representation

Kaouther Messaoud[1,2], Nachiket Deo[2], Mohan M. Trivedi[2] and Fawzi Nashashibi[1]

2021 IEEE Intelligent Vehicles Symposium (IV) | 978-1-7281-5394-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/IV48863.2021.9576054

Abstract— **Predicting the trajectories of surrounding agents is an essential ability for autonomous vehicles navigating through complex traffic scenes. The future trajectories of agents can be inferred using two important cues: the locations and past motion of agents, and the static scene structure. Due to the high variability in scene structure and agent configurations, prior work has employed the attention mechanism, applied separately to the scene and agent configuration to learn the most salient parts of both cues. However, the two cues are tightly linked. The agent configuration can inform what part of the scene is most relevant to prediction. The static scene in turn can help determine the relative influence of agents on each other's motion. Moreover, the distribution of future trajectories is multimodal, with modes corresponding to the agent's intent. The agent's intent also informs what part of the scene and agent configuration is relevant to prediction. We thus propose a novel approach applying multi-head attention by considering a joint representation of the static scene and surrounding agents. We use each attention head to generate a distinct future trajectory to address multimodality of future trajectories. Our model achieves state of the art results on the nuScenes prediction benchmark and generates diverse future trajectories compliant with scene structure and agent configuration.**

I. INTRODUCTION

Autonomous vehicles navigate in highly-uncertain and interactive environments shared with other dynamic agents. In order to plan safe and comfortable maneuvers, they need to predict future trajectories of surrounding vehicles. The inherent uncertainty of the future makes trajectory prediction a challenging task. However, there is a structure to vehicle motion. Two cues in particular provide useful context to predict the future trajectories of vehicles: (1) The past motion of the vehicle of interest, and the motion of its neighbouring agents, and (2) the static scene structure including road and lane structure, sidewalks and crosswalks.

A major challenge in trajectory prediction is the high variability in both context cues. Static scene elements and agents in a traffic scene can have various configurations. Several existing machine learning models for trajectory prediction have employed the attention mechanism with single [22], [19], [15] or multiple heads [11], [12], [13] to learn the most salient subsets of the static scene and agent configuration, to make both cues tractable. However, a limitation of existing approaches is that they either consider just one of the two cues [22], [11], [12], or apply attention modules separately to representations of the agent configuration and static scene [19]. The two cues are tightly linked; each can inform the most salient parts of the other. For example, the presence of a nearby pedestrian could make a crosswalk in the path of the vehicle of interest a salient part of the static scene, as opposed to if there were no nearby pedestrians. On the other hand, the presence of a crosswalk would make a nearby pedestrian on the sidewalk a more salient part of the input context as opposed to if there was no crosswalk.

Another challenge in trajectory prediction is the multimodality of the distribution of future trajectories. The vehicle of interest could have one of several plausible intents, each of which would correspond to a distinct future trajectory. To address multimodality most existing approaches build one latent representation of the context [5], [6], [9], [23] and then generate multiple possible trajectories based on this representation. However, we believe that each possible future trajectory is conditioned on a specific subset of surrounding agents' behaviors and scene context features. For each possible intent, a different partial context is important to understand the future behavior.

To address the above challenges, we propose a model for multimodal trajectory prediction of vehicles, utilizing multi-head attention [21] with the following characteristics:

1) **Joint agent-map representation:** Unlike prior approaches, we use a joint representation of the agents and static scene (represented as an HD map) to generate keys and values for attention heads. This allows us to better model the inter-dependency of both cues.
2) **Attention heads specializing in prediction modes:** We model the predictive distribution as a mixture model, with each attention head specializing in one mixture component. This allows each attention head to weight different parts of the scene and agent configuration conditioned on agent intent.

We evaluate our model on the public NuScenes dataset [2]. Our model achieves state of the art results on the NuScenes prediction benchmark. It generates diverse future trajectories conform to the static scene.

II. RELATED RESEARCH

Cross-agent interaction: Drivers and pedestrians operate in a shared space, co-operating with each other to perform safe maneuvers and reach their goals. Thus, most state of the art trajectory prediction approaches model cross-agent interaction. Alahi *et al.,* [1] propose *social pooling*, where the LSTM states of neighboring agents were pooled based on their locations in a 2-D grid to form *social tensors* as inputs to the prediction model. Deo *et al.* [5] extend this concept to model more distant interactions using successive convolutional layers. Messaoud *et al.,* [11] instead apply a multi-

[1] INRIA Paris, {kaouther.messaoud,fawzi.nashashibi}@inria.fr
[2] LISA, UCSD, {mkaouther,ndeo,mtrivedi}@ucsd.edu

978-1-7281-5394-0/21/$31.00 ©2021 IEEE

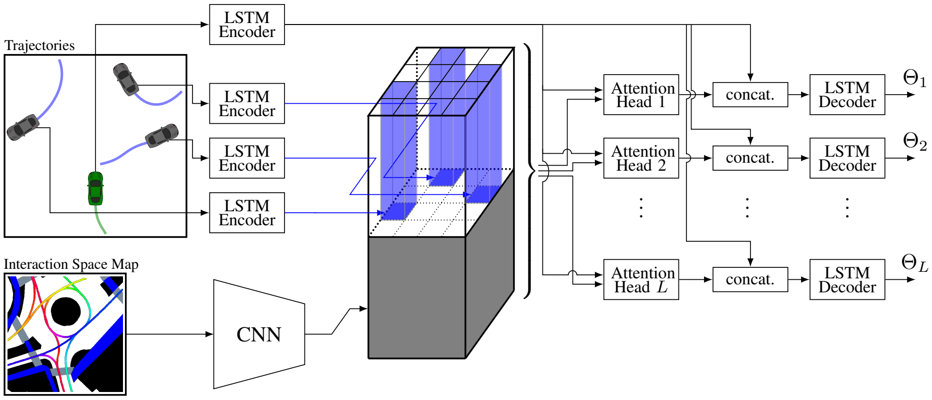


Fig. 1: **MHA-JAM** (MHA with Joint Agent Map representation): Each LSTM encoder generates an encoding vector of one of the surrounding agent recent motion. The CNN backbone transforms the input map image to a 3D tensor of scene features. A combined representation of the context is build by concatenating the surrounding agents motion encodings and the scene features. Each attention head models a possible way of interaction between the target (green car) and the combined context features. Each LSTM decoder receives an context vector and the target vehicle encoding and generates a distribution over a possible predicted trajectory conditioned on each context.

head attention mechanism to the social tensor to directly relate distant vehicles and extract a context representation. Our approach is closest to [11], however, we additionally consider the static scene information.

Agent-scene modeling: Static scene context in the bird's eye view is also an important cue that has been exploited in prior work for trajectory prediction. Zhao *et al.* [23] concatenate social tensors and an encoding of the scene in the bird's eye view and apply convolutional layers to extract a joint representation of the scene and surrounding agent motion. Sadeghian *et al.* [19] deployed two parallel attention blocks; a social attention for vehicle-vehicle interactions and a physical attention for vehicle-map interactions modeling. Yuan *et al.* [15] use two attention modules as well but deploy them sequentially by feeding the output of the cross-agent attention module as a query to the scene attention module. Unlike [15], [19], our model generates a joint representation of the scene and social tensor similar to [23], which we use as an input to multi-head attention modules to model the most salient parts of the scene and agent configuration. Recent work has also used auxiliary loss functions for generating scene-compliant trajectories [14], [15], [8]. We use the off-road loss proposed in [14] as an auxiliary loss function for training our model

Multimodal trajectory prediction: The future motion of agents is inherently multimodal, with multiple plausible trajectories conditioned on the agents' intent. Thus recent approaches learn one to many mappings from input context to multiple future trajectories. The most approach is sampling generative models such as conditional variational autoencoder (CVAE) [10] and Generative Adversarial Networks (GANs) [23], [9], [19]. Ridel *et al.* [18] predict the probability distributions over grids and generates multiple trajectory samples. In this paper, we utilise a mixture model similar to [4], [3], generating a fixed number L of plausible trajectories for an agent, where each trajectory is represented by a sequence of two-dimensional Gaussians and the probability associated with each of the L trajectories.

III. Multi-head Attention with Joint Agent-Map Representation

A. Input Representation

Interaction space: We first define the *interaction space* of a target vehicle T as the area centered on its position at the prediction instant t_{pred} and oriented in its direction of motion, and denote it as $\mathcal{A}_T$. We consider all agents present in $\mathcal{A}_T$ and the static scene elements in it as inputs to our model. We consider $\mathcal{A}_T$ to extend $\pm 25m$ from the target vehicle T in the lateral direction, $40m$ in the longitudinal direction ahead of T and $10m$ behind it.

Trajectory representation: Each agent i in the interaction space is represented by a sequence of its states, for t_h past time steps between $t_{pred} - t_h$ and t_{pred},

$$S_i = [S_i^{t_{pred}-t_h}, \ldots, S_i^{t_{pred}}]. \quad (1)$$

Here, superscripts denote time, while subscripts denote agent index. Absence of subscripts implies all agents in $\mathcal{A}_T$, and absence of superscripts implies all time steps from $t_{pred} - t_h$ to t_{pred}. Each state is composed of a sequence of the agent relative coordinates x_i^t and y_i^t, velocity v_i^t, acceleration a_i^t and yaw rate $\dot{\theta}_i^t$,

$$S_i^t = (x_i^t, y_i^t, v_i^t, a_i^t, \dot{\theta}_i^t). \quad (2)$$

The positions are expressed in a stationary frame of reference; the origin is the position of the target vehicle at the prediction time t_{pred}. The y-axis and x-axis are oriented respectively toward the target vehicle's direction of motion and the direction perpendicular to it. We assume access to tracked vehicle locations and motion states over t_h, typically obtained via a vehicle tracker (e.g. [17]).

Map representation: We use a rasterized bird's eye view map similar to [4], to represent the static scene elements in

$\mathcal{A}_T$. The map includes the road geometry, drivable area, lane structure and direction of motion along each lane, locations of sidewalks and crosswalks. We denote the map as $\mathcal{M}$.

B. Multimodal Output Representation

We wish to estimate the probability distribution $P(Y|S,\mathcal{M})$ over the future locations Y of the target vehicle, conditioned on the past trajectories S of agents in $\mathcal{A}_T$, and the map $\mathcal{M}$. To account for multimodality of $P(Y|S,\mathcal{M})$, we model it as a mixture distribution with L mixture components. Each mixture component consists of predicted location co-ordinates at discrete time-steps over a prediction horizon t_f.

$$Y_l = [Y_l^{t_{pred}+1}, \dots, Y_l^{t_{pred}+t_f}],\ l = 1, \dots, L. \tag{3}$$

Here, superscripts denote time, while subscripts denote the mixture component. Each predicted location Y_l^t is modeled as a bivariate Gaussian distribution. Our model outputs the means and variances $\Theta_l^t = (\mu_l^t, \Sigma_l^t)$ for each mixture component at each time step. Additionally, it outputs the probabilities P_l associated with each mixture component.

C. Encoding layers

The encoding layers generate feature representations of the past trajectories S, and the map $\mathcal{M}$. They comprise:

Trajectory encoder: The state vector S_i^t of each agent is embedded using a fully connected layer to a vector e_i^t and encoded using an LSTM encoder from $t_{pred} - t_h$ to t_{pred},

$$h_i^t = LSTM(h_i^{t-1}, e_i^t; W_{enc}). \tag{4}$$

Here, h_i^t and h_T^t are the hidden states vector of the i^{th} surrounding agent and the target vehicle respectively at time t. All the LSTM encoders share the same weights W_{enc}.

Map encoder: We use a CNN to extract high level features from the map $\mathcal{M}$. The CNN outputs a map feature representation $\mathcal{F}_m$ of size (M, N, C_m), where (M, N) represent spatial dimensions of the feature maps and C_m the number of feature channels.

D. Joint Agent-Map Attention

The first step in modeling vehicle-agents and vehicle-map interactions is to build a combined representation of the global context. To do so, we first build a *social tensor* as proposed in [1]. We divide $\mathcal{A}_T$ into a spatial grid of size (M, N). The trajectory encoder states of the surrounding agents $h_i^{t_{pred}}$ at the prediction instant t_{pred} are placed at their corresponding positions in the 2D spatial grid, giving us a tensor $\mathcal{F}_s$ of size (M, N, C_h), where C_h is the size of the trajectory encoder state. Figure 1 shows an example of a social tensor. Then, we concatenate the social tensor $\mathcal{F}_s$ and map features $\mathcal{F}_m$ along the channel dimension to generate a combined representation $\mathcal{F}$ of agent motions and the map,

$$\mathcal{F} = Concat(\mathcal{F}_s, \mathcal{F}_m). \tag{5}$$

We use the multi-head attention mechanism [21] to extract the salient parts of the joint agent-map representation $\mathcal{F}$ as shown in figure 2, with an attention head assigned to each of the L mixture components. For each attention head, the hidden state of the target vehicle $h_T^{t\ pred}$ is processed by a fully connected layer θ_l to give the query

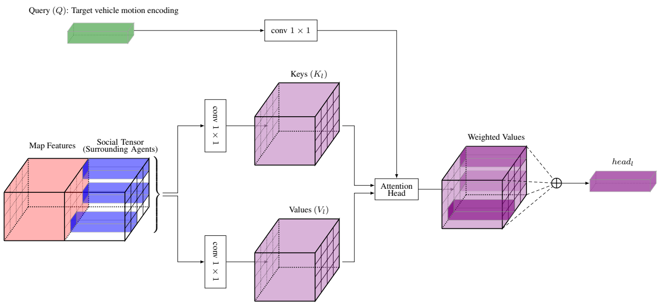


Fig. 2: Attention modules in MHA-JAM

$$Q_l = \theta_l(h_T^{t\ pred}; W_{\theta_l}). \tag{6}$$

The agent-map representation $\mathcal{F}$ is processed by 1×1 conv. layers ϕ_l and ρ_l to give keys K_l and values V_l,

$$K_l = \phi_l(\mathcal{F}; W_{\phi_l}), \tag{7}$$

$$V_l = \rho_l(\mathcal{F}; W_{\rho_l}). \tag{8}$$

The output A_l of each attention head is then calculated as a weighted sum of value vectors $V_l(m, n, :)$,

$$A_l = \sum_{m=1}^{M} \sum_{n=1}^{N} \alpha_l(m, n) V_l(m, n, :). \tag{9}$$

Here, $\alpha_l(m, n)$ weights the effect of each value vector $V_l(m, n)$,

$$\alpha_l(m, n) = Softmax(\frac{Q_l K_l^T(m, n, :)}{\sqrt{d}}), \tag{10}$$

where $Q_l K_l^T(m, n, :)$ is the dot product of Q_l and $K_l(m, n, :)$ and d is the dimension of Q_l and $K_l(m, n, :)$.

For each attention head we concatenate the output A_l with the target vehicle trajectory encoder state $h_T^{t_{pred}}$ to give a context representation z_l for each mixture component $l = 1, \dots, L$,

$$z_l = Concat(h_T^{t_{pred}}, A_l). \tag{11}$$

E. Decoding layer

Each context vector z_l, representing the selected information about the target vehicle's interactions with the context, and its motion encoding are fed to l LSTM Decoders. The decoders generate the predicted parameters of the distributions over the target vehicle's estimated positions of each trajectory for next t_f time steps,

$$\Theta_l^t = \Lambda(LSTM(h_l^{t-1}, z_l; W_{dec})). \tag{12}$$

All the LSTM decoders share the same weights W_{dec} and Λ is a fully connected layer. Similar to [4], we also output the probability P_l associated with each mixture component. To do so, we concatenate all the scene representation vectors

TABLE I: Results of comparative analysis on nuScenes dataset, over a prediction horizon of 6-seconds

	MinADE$_1$	MinADE$_5$	MinADE$_{10}$	MinADE$_{15}$	MinFDE$_1$	MinFDE$_5$	MinFDE$_{10}$	MinFDE$_{15}$	MissRate$_{5,2}$	MissRate$_{10,2}$	Off-Road Rate
Physics oracle	**3.69**	3.69	3.69	3.69	9.06	9.06	9.06	9.06	0.88	0.88	0.12
MTP [4]	4.42	2.22	1.74	1.55	10.36	4.83	3.54	3.05	0.74	0.67	0.25
Multipath [3]	4.43	**1.78**	1.55	1.52	10.16	**3.62**	2.93	2.89	0.78	0.76	0.36
CoverNet* [16]	-	2.62	1.92	-	11.36	-	-	-	0.76	0.64	0.13
Trajectron++* [20]	-	1.88	1.51	-	9.52	-	-	-	0.70	0.57	0.25
MHA-JAM	3.77	1.85	**1.24**	**1.03**	8.65	3.85	2.23	**1.67**	0.60	0.46	0.10
MHA-JAM (off-road)*	**3.69**	1.81	**1.24**	**1.03**	**8.57**	3.72	**2.21**	1.70	**0.59**	**0.45**	**0.07**

z_l, feed them to two successive fully connected layers and apply the softmax activation to obtain L probability values.

F. Loss Functions

Regression loss: In order to not penalize plausible trajectories generated by the model that do not correspond to the ground truth, we use a variant of the best of L regression loss for training our model similar to [9]. This encourages the model to generate a diverse set of predicted trajectories. Since we output the parameters of a bivariate Gaussian distribution at each time step for the L trajectories, we compute the negative log-likelihood (NLL) of the ground truth trajectory under each of the L modes output by the model, and consider the minimum of the L NLL values as the regression loss. The regression loss is given by

$$L_{reg} = \min_{l} \sum_{t=t_{pred+1}}^{t_{pred}+t_f} -log(P_{\Theta_l^t}(Y_l^t|S,\mathcal{M})). \tag{13}$$

Classification loss : In addition to the regression loss, we consider the cross entropy as used in [4], [3],

$$L_{cl} = -\sum_{l=1}^{L} \delta_{l^*}(l)log(P_l), \tag{14}$$

where δ_{l^*} is a function equal to 1 if $l = l^*$ and 0 otherwise. Here l^* is the mode corresponding to the minimum NLL in equation 13. Y_{l^*} is the predicted trajectory corresponding to l^* and P_{l^*} its predicted probability.

Off-road loss: Since the regression loss only affects the trajectory closest to the ground-truth, we consider the auxiliary loss function proposed in [14], [15] that penalizes points in any of the L trajectories that lie outside the road. The off-road loss L_{or} for each predicted location is the minimum distance of that location from the drivable area.

The overall loss for training the model is given by

$$L = L_{reg} + \lambda_{cl}.L_{cl} + \lambda_{or}.L_{or}, \tag{15}$$

where the weights λ_{cl} and λ_{or} are hyperparameters.

G. Implementation details

The input states are embedded in a space of dimension 32. We use an image representation of the scene map of size of $(500, 500)$ with a resolution of 0.1 meters per pixel. Similar to [16] representation, our input image extents are 40 m ahead of the target vehicle, 10 m behind and 25 m on each side. We use ResNet-50 pretrained on ImageNet to extract map features. This CNN outputs a map features of size $(28, 28, 512)$. On top of them, we place the trajectories encodings. We use $t_h = 2s$ and the deployed LSTM encoder and decoder are of 64 and 128 randomly initialized units respectively. We use $L = 16$ parallel attention operations applied on the vectors projected on spaces of size d=64.

IV. Experimental Analysis and Evaluations

A. Dataset

We train and evaluate our model using the publicly available nuScenes [2] dataset. The dataset was captured using vehicle mounted camera and lidar sensors driving through Boston and Singapore. It comprises 1000 *scenes*, each of which is a 20 second record, capturing complex inner city traffic. We train and evaluate our model using the official benchmark split for the nuScenes prediction challenge, with 32 186 prediction instances in the train set, 8 560 instances in the validation set, and 9 041 instances in the test set.

B. Baselines

We compare our model to five baselines, a physics based approach, and four recently proposed models that represent the state of the art for multimodal trajectory prediction. All deep learning based models generate up to $L = 25$ trajectories ($L = 16$ for all implemented methods) and their likelihoods. We report the results considering the k most probable trajectories generated by each model.

Physics oracle : The physics based model introduced in [16] computes the minimum average point-wise Euclidean distance over the predictions generated by four models.

Multiple-Trajectory Prediction (MTP) [4]: The MTP model uses a CNN over a rasterized representation of the scene and the target vehicle state to generate a fixed number of trajectories (modes) and their associated probabilities. We use the implementation of this model developed by [16].

Multipath [3]: The Multipath model uses a CNN with with same inputs as MTP. It deploys fixed *anchors* obtained from the train set to represent the modes and outputs residuals with respect to anchors in its regression heads. We implement MultiPath as described in [3].

CoverNet [16]: The CoverNet model formulates multimodal trajectory prediction purely as a classification problem. The model predicts the likelihood of a fixed trajectory set, conditioned on the target vehicle state.

Trajectron++ [20]: is a graph-structured recurrent model that predicts the agents trajectories while considering agent motions and heterogeneous scene data.

[*] Results reported in the nuScenes challenge Leaderboard: https://evalai.cloudcv.org/web/challenges/challenge-page/591/leaderboard/1659

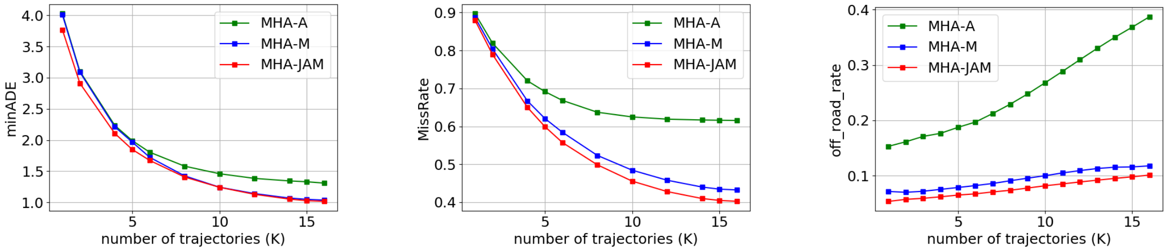


Fig. 3: Plots of the metrics MinADE$_k$, MissRate$_{k,2}$ and off-road rate for the k likeliest predicted trajectories.

C. *Metrics*

MinADE$_k$ and MinFDE$_k$: We report the minimum average and final displacement errors over k most probable trajectories similar to prior approaches for multimodal trajectory prediction [9], [10], [4], [3], [7]. The minimum over k avoids penalizing the model for generating plausible future trajectories that don't correspond to the ground truth.

Miss rate: For a given distance d, and the k most probable predictions generated by the model, the set of k predictions is considered a miss based on,

$$\text{Miss}_{k,d} = \begin{cases} 1 \text{ if } \min_{\hat{\mathbf{y}} \in P_k} \left(\max_{t=t_{pred}}^{t=t_f} \left\| \mathbf{y}^t - \hat{\mathbf{y}}^t \right\| \right) \geqslant d. \\ 0 \qquad\qquad \text{otherwise} \end{cases} \tag{16}$$

The miss rate MissRate$_{k,d}$ computes the fraction of missed predictions over the test set.

Off-road rate: Similar to [7], we consider the off-road rate, which measures the fraction of predicted trajectories that fall outside the drivable area of the map.

D. *Quantitative Results*

We compare our model **MHA-JAM** (MHA with joint agent map representation trained with off-road loss) to various baselines in table I. Our model outperforms all baselines on 9 of the 11 reported metrics, while being second on the remaining two, representing the state of the art on the nuScenes benchmark as of writing this paper.

For the MinADE$_k$ and MinFDE$_k$ metrics, our model achieves the best results for $k \in \{1, 10, 15\}$ and second best to Multipath [3] when $k = 5$. Having the best performance for $k \in \{10, 15\}$ shows that our method generates a diverse set of plausible trajectories that match the ground truth. However, for $k = 5$, our classifier doesn't seem to succeed in selecting the closest trajectories to the ground truth among the 5 most probable ones, while the Multipath classifier does.

Moreover, our method presents significant improvements compared to others when considering miss rate and off-road rate metrics. Having the lowest miss rate suggests that our predicted trajectories are less likely to deviate from the ground truth over a threshold of $d = 2m$. In addition, our model achieves significantly lower off-road rates. Therefore, it generates scene compliant trajectories.

E. *Ablation Experiments:*

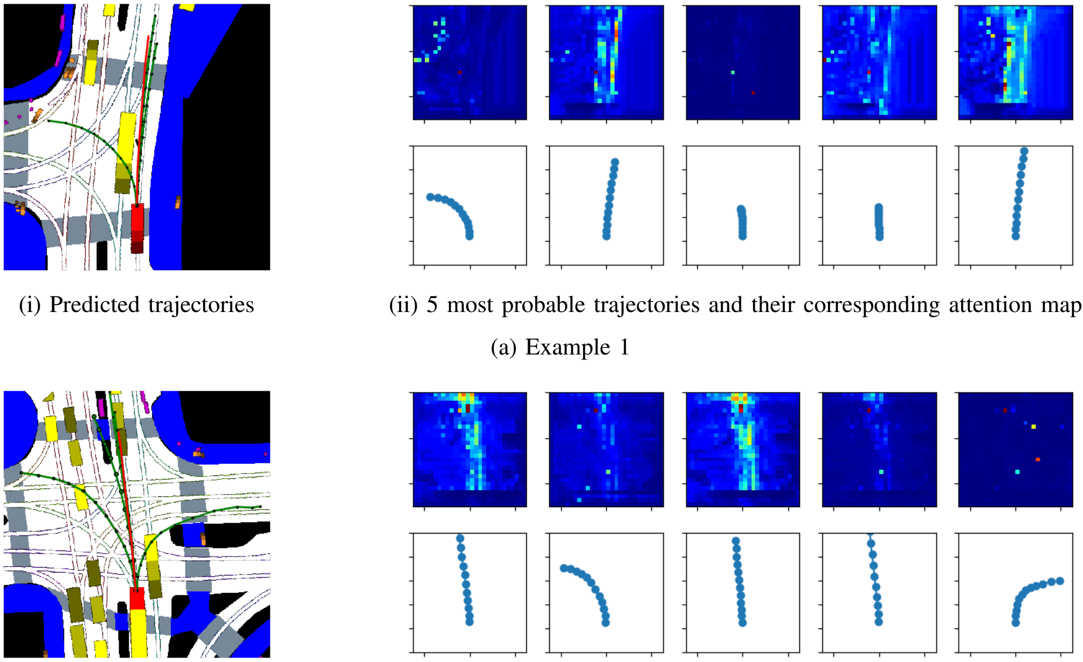
To get a deeper insight on the relative contributions of input cues affecting the overall performance, we compare our model MHA-JAM to two models: (1) MHA with purely agent inputs (MHA-A), and (2) MHA with purely map inputs (MHA-M). When considering only the surrounding agents without any information about the scene structure (MHA-A), the model shows poor results according to the three metrics MinADE$_k$, missRate$_{k,2}$, and off-road rate (cf. Figure 3). This highlights the importance of the map information to make more accurate and scene compliant predictions. Since MHA-JAM has the best performance, we infer that considering the surrounding agents enhances the prediction accuracy.

F. *Qualitative Results*

Figure 4 presents two examples of vehicle trajectory prediction, their corresponding 5 most probable generated trajectories and their associated attention maps. We notice that our proposed model MHA-JAM (off-road) successfully predicts diverse possible maneuvers; straight and left for the first Example 4a and straight, left and right for the second Example 4b. In addition, it produces different attention maps which implies that it learnt to create specific context features for each predicted trajectories. For instance, the attention maps of the going straight trajectories, assign high weights to the drivable area in the straight direction and to the leading vehicles (the dark red cells). Moreover, they show focus on relatively close features when performed with low speed and further ones with high speed (cf. Example 4a). For the left and right turns, in both examples, the corresponding attention maps seem to assign high weights to surrounding agents that could interact with the target vehicle while performing those maneuvers. For instance, in the left turn (cf. Example 4b), the attention map assigns high weights to vehicles in the opposite lane turning right.

V. Concluding Remarks

This work tackled the task of vehicle trajectory prediction in an urban environment while considering interactions between the target vehicle, its surrounding agents and scene. To this end, we deployed a MHA-based method on a joint agents and map global context representation. The model enabled each attention head to explicitly extract specific agents and scene features that help to infer the driver's diverse possible behaviors. Furthermore, the visualization of the attention maps reveals the importance of joint agents and map features and the interactions occurring during the execution of each possible maneuver. Experiments showed that our proposed



(i) Predicted trajectories

(ii) 5 most probable trajectories and their corresponding attention maps

(a) Example 1

(i) Predicted trajectories

(ii) 5 most probable trajectories and their corresponding attention maps

(b) Example 2

Fig. 4: Examples of produced attention maps and trajectories with MHA-JAM (off-road) model

approaches outperform the existing methods according to most of the metrics considered, especially the off-road metric. This highlights that the predicted trajectories comply with the scene structure.

References

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *IEEE Conference CVPR*, June 2016.

[2] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. *arXiv:1903.11027*, 2019.

[3] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. *ArXiv*, abs/1910.05449, 2019.

[4] H. Cui, V. Radosavljevic, F. Chou, T. Lin, T. Nguyen, T. Huang, J. Schneider, and N. Djuric. Multimodal trajectory predictions for autonomous driving using deep convolutional networks. *ICRA*, 2019.

[5] N. Deo and M. M. Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *CVPRW*, 2018.

[6] N. Deo and M. M. Trivedi. Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs. In *IV*, 2018.

[7] N. Deo and M. M. Trivedi. Trajectory forecasts in unknown environments conditioned on grid-based plans. *ArXiv*, abs/2001.00735, 2020.

[8] R. Greer, N. Deo, and M. Trivedi. Trajectory prediction in autonomous driving with a lane heading auxiliary loss. *arXiv preprint arXiv:2011.06679*, 2020.

[9] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *CVPR*, June 2018.

[10] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr, and M. Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *CVPR*, 2017.

[11] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi. Non-local social pooling for vehicle trajectory prediction. In *IEEE Intelligent Vehicles Symposium, IV*, pages 975–980, June 2019.

[12] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi. Relational recurrent neural networks for vehicle trajectory prediction. In *IEEE Intelligent Transportation Systems Conference, ITSC*, 2019.

[13] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi. Attention Based Vehicle Trajectory Prediction. *IEEE Transactions on Intelligent Vehicles*, Apr. 2020.

[14] M. Niedoba, H. Cui, K. Luo, D. Hegde, F. Chou, and N. Djuric. Improving movement prediction of traffic actors using off-road loss and bias mitigation. In *Workshop ML4AD*, 2019.

[15] S. H. Park, G. Lee, M. Bhat, J. Seo, M.-S. Kang, J. Francis, A. R. Jadhav, P. P. Liang, and L.-P. Morency. Diverse and admissible trajectory forecasting through multimodal context understanding. *ArXiv*, abs/2003.03212, 2020.

[16] T. Phan-Minh, E. C. Grigore, F. A. Boulton, O. Beijbom, and E. M. Wolff. Covernet: Multimodal behavior prediction using trajectory sets. *CoRR*, abs/1911.10298, 2019.

[17] A. Rangesh, P. Maheshwari, M. Gebre, S. Mhatre, V. Ramezani, and M. M. Trivedi. Trackmpnn: A message passing graph neural architecture for multi-object tracking. *arXiv preprint arXiv:2101.04206*, 2021.

[18] D. A. Ridel, N. Deo, D. F. Wolf, and M. Trivedi. Scene compliant trajectory forecast with agent-centric spatio-temporal grids. *IEEE Robotics and Automation Letters*, 5:2816–2823, 2019.

[19] 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 *CVPR*, June 2019.

[20] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control. *ArXiv*, abs/2001.03093, 2020.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[22] A. Vemula, K. Muelling, and J. Oh. Social attention: Modeling attention in human crowds. In *ICRA*, May 2018.

[23] T. Zhao, Y. Xu, M. Monfort, W. Choi, C. Baker, Y. Zhao, Y. Wang, and Y. N. Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *CVPR*, June 2019.