Interaction-Aware Trajectory Prediction with Point Transformer

Yahui Liu, Xingyuan Dai, Jianwu Fang, Bin Tian, and Yisheng Lv[†]

***Abstract*— To ensure safe and efficient autonomous driving, trajectory prediction system must account for social interactions among road participants. Graph-based models are leading approaches in modeling social interactions for trajectory prediction, but they face the challenges of designing an appropriate graph structure and processing complex interactions. We consider that the participants in a scene are a set of unstructured points, which are similar to point cloud data. Inspired by point cloud learning networks, we view the road participants in a scene as point cloud in a two-dimensional coordinate system, and utilize Point Transformer aggregator to process the interactions on both local and global level. Besides, we present a multiplex fusion of social and temporal information for trajectory prediction. We perform extensive experiments on the Argoverse motion forecasting dataset, and the results demonstrate the superior performance of our model for multi-agent trajectory prediction.**

I. INTRODUCTION

Trajectory Prediction is critical for safe and efficient autonomous driving systems. To predict the future trajectories based on historical observations, the systems require taking into account the social interactions among various agents [1]–[4]. But it faces significant challenges due to the varying number of agents, permutation invariant, and complicated social interactions.

Leading approaches in modeling social interactions for trajectory prediction are graph-based models [5]–[15]. The graph representation constructs a graph where each road participant in a scene is a node, and the edges represent the relationships between the nodes. The graph can be constructed based on various criteria, such as Euclidean distance, group connectivity, or semantic similarity. There are several approaches to designing a graph structure for modeling social interactions among road participants:

- Range-based graph: This approach constructs a graph between road participants that are within a certain range of each other.
- Interaction-based graph: This approach constructs a graph between road participants that have a direct influence on each other's behavior.
- Attention-based graph: This approach uses an attention mechanism to weight the importance of various road participants and construct a graph that captures the most relevant interactions.
- Heterogeneous graph: This approach combines multiple graph structures to capture different types of interactions and create a more comprehensive representation of the environment.

While graph-based models have shown promise in social interactions among road participants for trajectory prediction in autonomous driving, however, there are still some challenges and limitations to graph-based models in trajectory prediction. One of the main challenges is the difficulty in designing an appropriate graph structure that captures the social interactions among road participants. Another challenge is the scalability of the models, particularly in large-scale traffic environments. Further research is needed to address these challenges and to explore the full potential of social interactions.

We consider that the participants in a scene are a set of unstructured points, which are similar to point cloud data. Point clouds are a set of data points in a three-dimensional coordinate system, and each point represents a specific position in space. Point cloud learning networks use these points as input to extract meaningful features. Starting from the perspective of node/point interaction relationships, we explore how to use the aggregators in point cloud learning networks to model the social interactions among road participants. We view the road participants in a scene as point cloud data in a two-dimensional coordinate system, and use the aggregator in point cloud learning network to model the social interactions among road participants. **The prominent advantage is that there is no need to manually specify the interaction structure and avoid the complex process of learning correlations.**

Two prior approaches TPCN [16] and UST [17] have explored the use of point cloud learning networks for trajectory prediction. Both models have demonstrated promising results in accurately predicting future trajectories and outperformed traditional methods. The above two methods only use the classic point cloud analysis network PointNet [18] to model the spatial interactions. However, the semantic relations between points are not considered. Recent researchers have proposed several advanced approaches to improve the performance of point cloud relation learning. Therefore, we use the Point Transformer aggregator [19] to model the social interactions. In addition, two standard Transformer encoders [20], [21] are used to capture the temporal correlations of the local interactions and global motion pattern. By aggregating

This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFB4703700, the National Natural Science Foundation of China under Grant 62271485, and the Fundamental Research Funds for the Central Universities, CHD, under Grant 300102343513.

Yahui Liu, Xingyuan Dai, Bin Tian and Yisheng Lv are with the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China.

Jianwu Fang is with the College of Transportation Engineering, Chang'an University, Xi'an 710064, China.

[†]Corresponding Author.

all the social and temporal correlations among local and global road participants, the future trajectories are predicted in a single forward pass by MLPs decoder. To model the multi-modal motion, we produce multiple socially plausible trajectories.

Our contributions are summarized as follows:

- We present a novel framework to model the social interaction for trajectory prediction. We use the Point Transformer to model the social interactions. To the best of our knowledge, this paper is the first attempt to adopt the Point Transformer in the context of modeling social interactions for trajectory prediction.
- In order to better integrate social and temporal information, we prioritize capturing social interactions before learning temporal correlations in the local interaction module. The global interaction module is exactly the opposite, which starts with learning motion pattern, followed by modeling global interactions.
- Extensive experiments are conducted on the Argoverse motion forecasting benchmark [22] to show the effectiveness of our approach.

II. RELATED WORK

In this section, we briefly review the node-based interaction representation for trajectory prediction and the literature related to point cloud analysis with point-based relation learning.

A. *Trajectory Prediction with Node-based Interaction*

Many approaches used different types of graph neural networks (GNNs) to model social interactions, where the vehicles are represented as graph nodes [5]–[14]. For instance, VectorNet [5] and TNT [6] utilize a fully-connected hierarchical graph, with each sub-graph containing the feature of an object (agent or map component) represented as a sequence of vectors. SCALE-Net [8] considers edge attributes and proposes interactions with edge feature isomorphic graphs, where edge features contain the relative states between two connected agents. Other recent methods [10], [12], [23] use road topology or lane constrains to design graph encoders for interaction modeling. Besides the aforementioned GNNs, attention mechanism is another top-performing idea of encoding the social interaction among agents [11], [15], [24]–[26]. Attention mechanism works by assigning weights to each participant, based on its importance for predicting the accurate trajectory. This allows the network to selectively attend to the most informative participants in a scene, while ignoring irrelevant or noisy participants. For example, LaneGCN [11] uses multiple attention mechanisms to interact features between vehicle nodes and lane maps information. HEAT [15] proposes a heterogeneous edge-enhanced graph attention network, which considers the heterogeneity of the road participants and the attribute of edges. Due to the unstructured node data, there are also approaches that use point cloud learning network to extract spatio-temporal features [16], [17].

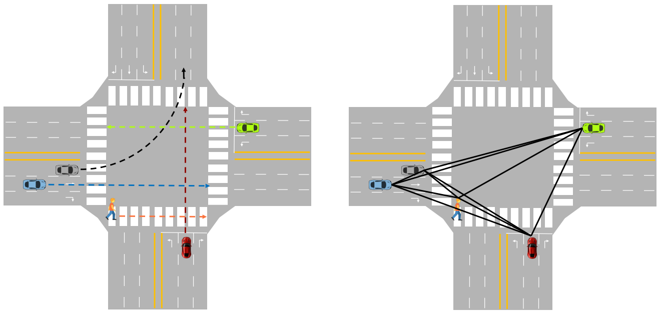


Fig. 1. Illustration of social interaction among the road participants in uncontrolled traffic scene.

B. *Point Cloud Analysis with Point-based Relation Learning*

The approaches of relation learning in point cloud analysis have become popular in recent years [19], [27]–[30]. The thrive of graph-based method starts from DGCNN [27], which learns on graphs dynamically updated at each layer. It proposed a local feature aggregation operator, named EdgeConv, which generates edge features that describe the semantic relationships between key points and their neighbors in feature space. RS-CNN [28] is another representative approach of local feature aggregation, it learns the relations within a local region by predefined geometric priors, but the low-level relation cannot fully represent the relation between two points. PointASNL [29] leverages non-local network to enhance the long-range dependency correlation learning. Point Transformer [19] and PointConT [30] applies Transformer-like operator to learn the relations between points or groups of points.

In this work, we explore the possibilities of point cloud learning aggregator to model the social interactions on both local level and global level, where road participants in a scene are viewed as a set of points.

III. METHODOLOGY

A. *Problem Statement*

Given a sequence of observed positions during time steps 1 to T_{obs} for a set of agents involved in a scene, the trajectory prediction problem aims to forecast the future positions of these agents over a specified time horizon T_{pred}. Formally, we have a set of N agents $\{p_i\}_{i=1}^{N}$ in a scene, and $p_i^t = (x_i^t, y_i^t)$ denotes the position of agent i at time step t. The observed information is:

$$\{p_i^t | i \in 1, \cdots N; t \in 1, \cdots, T_{\text{obs}}\}. \tag{1}$$

The task of trajectory prediction can be defined as:

$$\{\hat{p_i^t} | i \in 1, \cdots N; t \in T_{\text{obs}} + 1, \cdots, T_{\text{obs}} + T_{\text{pred}}\}. \tag{2}$$

B. *Overall Architecture*

Fig. 2 shows an architectural overview of our proposed model with the name IPT. Our model consists of temporal encoder, social interaction module and decoder. The temporal encoder is composed of two Transformer encoders. The social interaction module includes local interaction module

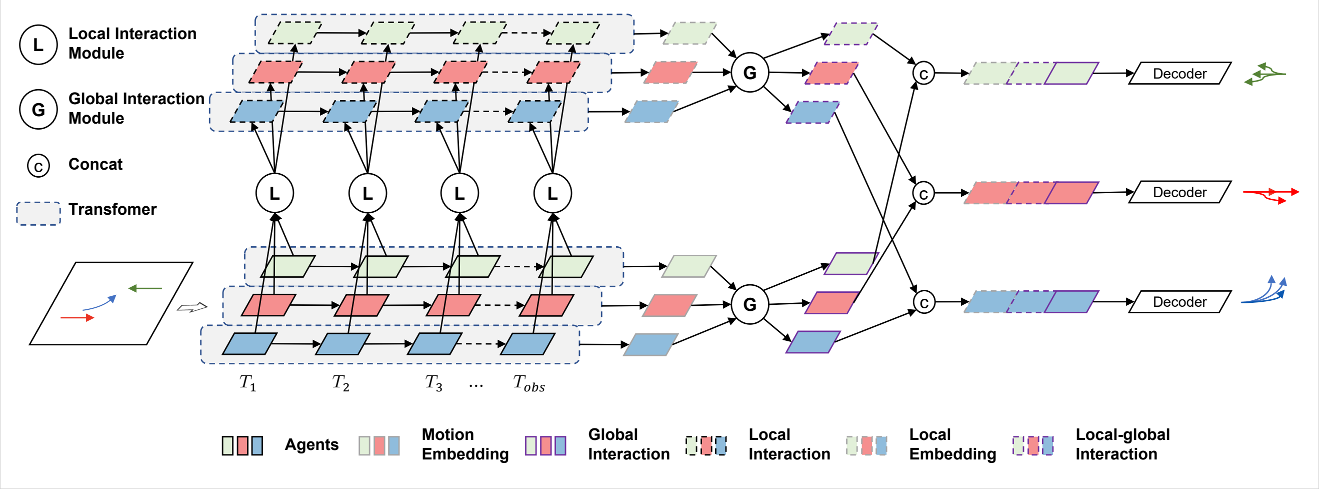


Fig. 2. Overall Architecture of our proposed IPT model.

and global interaction module, and the intermediate state fuses the social and temporal information on both local and global level. The decoder predicts the multi-modal future trajectories based on the intermediate state. The following subsections describe each component in detail.

C. Temporal Encoder

Each road agent has its own unique motion pattern, including different acceleration and preferred speed. Previous research has shown that the Transformer model have certain advantages in capturing sequence relationships. Therefore, we adopt a similar approach by using Transformer for each agent to obtain its motion temporal dependency. We employ temporal Transformer encoder on top of the local agent-agent interaction module and global motion state. For i-th agent at time step t, we get the motion embedding z_i^t and the local interaction embedding s_i^t from agent-agent interaction module.

$$\begin{aligned} Z_{\text{global}} &= \text{Transformer}_{\text{global}}(\{z_i^t\}_{t=1}^{T_{\text{obs}}}) \\ Z_{\text{local}} &= \text{Transformer}_{\text{local}}(\{s_i^t\}_{t=1}^{T_{\text{obs}}}) \end{aligned} \tag{3}$$

D. Social Interaction Module

In order to interact information across different agents in a scene, we propose to consider the agents in a scene as point cloud in 2D-coordinate space and leverage the recent progress in point cloud analysis. Since Point Transformer allows for aggregating information from neighbors by applying self-attention locally, we use the Point Transformer aggregator as our interaction operator.

Point Transformer operates on the point cloud and computes the features of each point by attention over its neighbors, following a vector attention strategy [31]. We formulate the Point Transformer layer in Eq. (4):

$$\begin{aligned} \mathbf{x}_i' &= \sum_{j \in \mathcal{N}_i} \alpha_{i,j} (\mathbf{W}_v \mathbf{x}_j + \delta_{ij}) \\ \alpha_{i,j} &= \text{softmax}\left(\gamma_\Theta(\mathbf{W}_q \mathbf{x}_i - \mathbf{W}_k \mathbf{x}_j + \delta_{i,j})\right) \\ \delta_{i,j} &= h_\Theta(\mathbf{p}_i - \mathbf{p}_j) \end{aligned} \tag{4}$$

where $\mathcal{N}_i$ is the set of i-th agent's local neighbors, $\mathbf{W}_{q,k,v}$ are learnable matrices for linear projection. $\alpha_{i,j}$ and $\delta_{i,j}$ are the attention coefficients and the positional embedding with γ_Θ and h_Θ denoting embedding function ($i.e.$ MLP blocks).

We first divide the scene into a set of local regions centered on each agent. In each local region, we aggregate the agent neighbor-related context features based on the Point Transformer operator. Then, in order to compensate the limited local field and capture the long-range dependencies in the scene, we take two different ways for global interaction module. One is the implementation of global interaction among the local region centered on the agents, and the other is directly modeling the motion pattern of all observed trajectories in the scene.

For the interaction between road participants and maps, it is planned to merge the local map information into the social-temporal features of the local central agent. Map information consists of lane centerlines, road intersections, turn directions, and traffic controls. We incorporate the local map information a_ξ into the embeddings z_ξ. With the central agent's social-temporal features z_i, the interaction can be defined as:

$$\begin{aligned} \mathbf{z}_i' &= \sum_{\xi \in \mathcal{N}_i} \alpha_{i,\xi} (\mathbf{W}_v \mathbf{z}_\xi + \delta_{i\xi}) \\ \alpha_{i,\xi} &= \text{softmax}\left(\gamma_\Theta(\mathbf{W}_q \mathbf{z}_i - \mathbf{W}_k \mathbf{z}_\xi + \delta_{i,\xi})\right) \\ \delta_{i,\xi} &= \phi_{\text{map}}\left(\left[(\mathbf{p}_\xi^1 - \mathbf{p}_\xi^0), (\mathbf{p}_\xi^0 - \mathbf{p}_i^{T_{\text{obs}}}), a_\xi\right]\right) \end{aligned} \tag{5}$$

E. Fusion of Social and Temporal Information

In the temporal encoder, two Transformer encoders are used to model the motion pattern and the temporal correlations of local interactions, respectively. In order to further integrate these two parts to accomplish the fusion of the temporal dependency and social interactions, we first extract social interaction relationships in the local interaction module, and then learn temporal correlations. Moving on to the global interaction module, we prioritize learning motion pattern before modeling global interaction relationships. In addition, we apply skip-connections to concatenate the local information and global information.

F. Multi-modal Decoder

Since the road environment is stochastic, a single prior trajectory can lead to multiple future trajectories. Following HiVT [32], we parameterize the distribution of future trajectories as a mixture Laplace distribution model. For each agent i and each component k, the Laplace density at time step t is parameterized by the location $\mu_{i,k}^t$ and the scale $b_{i,k}^t$.

$$f(\{p_i^t\}_{t=1}^{T_{\text{pred}}}) = \sum_{k=1}^{K} \pi_{i,k} \prod_{t=1}^{T_{\text{pred}}} \text{Laplace}(p_i^t | \mu_{i,k}^t, b_{i,k}^t) \quad (6)$$

where $\pi_{i,k}$ is the mixing coefficient of the k-th mixture component for i-th agent. We use an MLP to generate the $\pi_{i,k}$, and two side-by-side MLPs to predict $\mu_{i,k}^t$ and $b_{i,k}^t$.

G. Loss Function

We use the sum of regression loss $\mathcal{L}_{\text{reg}}$ and classification loss $\mathcal{L}_{\text{cls}}$ for end-to-end training:

$$\mathcal{L} = \mathcal{L}_{\text{reg}} + \lambda \mathcal{L}_{\text{cls}}, \quad (7)$$

where $\lambda = 1.0$. For regression loss, we use the negative log-likelihood of Eq.(6):

$$\mathcal{L}_{\text{reg}} = -\frac{1}{N T_{\text{pred}}} \sum_{i=1}^{N} \sum_{t=1}^{T_{\text{pred}}} \log \text{P}(p_i^t | \hat{\mu}_i^t, \hat{b}_i^t) \quad (8)$$

where $\text{P}(\cdot|\cdot)$ is the Laplace probability density function. In addition, we adopt the Winner-Takes-All (WTA) strategy [29]. WTA only conducts backpropagation on the best-predicted trajectory $\hat{k}$, which has the minimum final displacement error among K predicted trajectories. So $\{\hat{\mu}_i^t\}_{t=1}^{t=T_{\text{pred}}}, \{\hat{b}_i^t\}_{t=1}^{t=T_{\text{pred}}}$ are the locations and the scales of the best-predicted trajectory for i-th agent.

For the classification loss, we adopt the soft displacement error as target probability π_k and use cross-entropy loss to optimize the mixing coefficients:

$$\mathcal{L}_{\text{cls}} = \sum_{k=1}^{K} -\pi_k \log(\hat{\pi}_k) \quad (9)$$

IV. EXPERIMENTS

A. Experimental Setups

1) Dataset: We evaluate the proposed model on the widely used Argoverse1 motion forecasting dataset [22]. There are 323,557 scenarios collected in Miami and Pittsburgh, split into 205,942 for training, 39,472 for validation, and 78,143 for testing. The dataset includes multiple object types, tagged with AV, agent and others. Each scenario is 5 seconds long sampled at 10 Hz. The training and validation sets contain the full 5-second observations, while the test set only provides the first 2-second motion. The Argoverse motion forecasting challenge requires to predict the next 3-second motion of the target agents. In addition to the motion dataset, we also use the HD-maps provided by the Argoverse.

2) Metrics: Following the previous works, we adopt the Average Displacement Error (ADE), the Final Displacement Error (FDE) and the Miss Rate (MR) as metrics. ADE is calculated by taking the mean of the Euclidean distances between the predicted and ground-truth positions at each time step, while FDE measures the Euclidean distance between the predicted endpoint and the ground-truth endpoint. For multi-modal predictions, minimum ADE and minimum FDE are used by selecting the smallest value of ADE and FDE across all predictions. MR is the percentage of the forecasted trajectories exceeding 2.0 meters of ground truth according to endpoint error.

3) Implementation details: During preprocessing, we first apply some common preprocessing steps as previous works [11], [13], including coordinate transformation of each scenario and displacement of each trajectory. We transform the coordinate in each scenario to be originated at the agent position at $t = T_{obs}$ (orientation between the position at $t = T_{obs}$ and $t = T_{obs} - 1$ as positive x-axis), and calculate the displacement Δp_i^t ($\Delta p_i^t = p_i^t - p_i^{t-1}$) of each trajectory. All local regions have a radius of 50 meters. We used the AdamW optimizer with an initial learning rate of 3×10^{-4} and weight decay of 1×10^{-4} to train our model for 64 epochs using a batch size of 32. Besides, The learning rate is decayed using the cosine annealing scheduler. All the experiments are performed on two Tesla V100 GPUs. We fix the random seed in all experiments to eliminate the influence of randomness.

B. Quantitative Results

In Table I and Table II, we compare our model with previous works on the Argoverse 1 validation and online test datasets, respectively. The metrics are minADE, minFDE and MR for $K = 6$, and the leaderboard is ranked by minFDE for $K = 6$. We can observe that our model achieves excellent results on the validation set. It also achieves very competitive performance on the test set without any ensemble strategies.

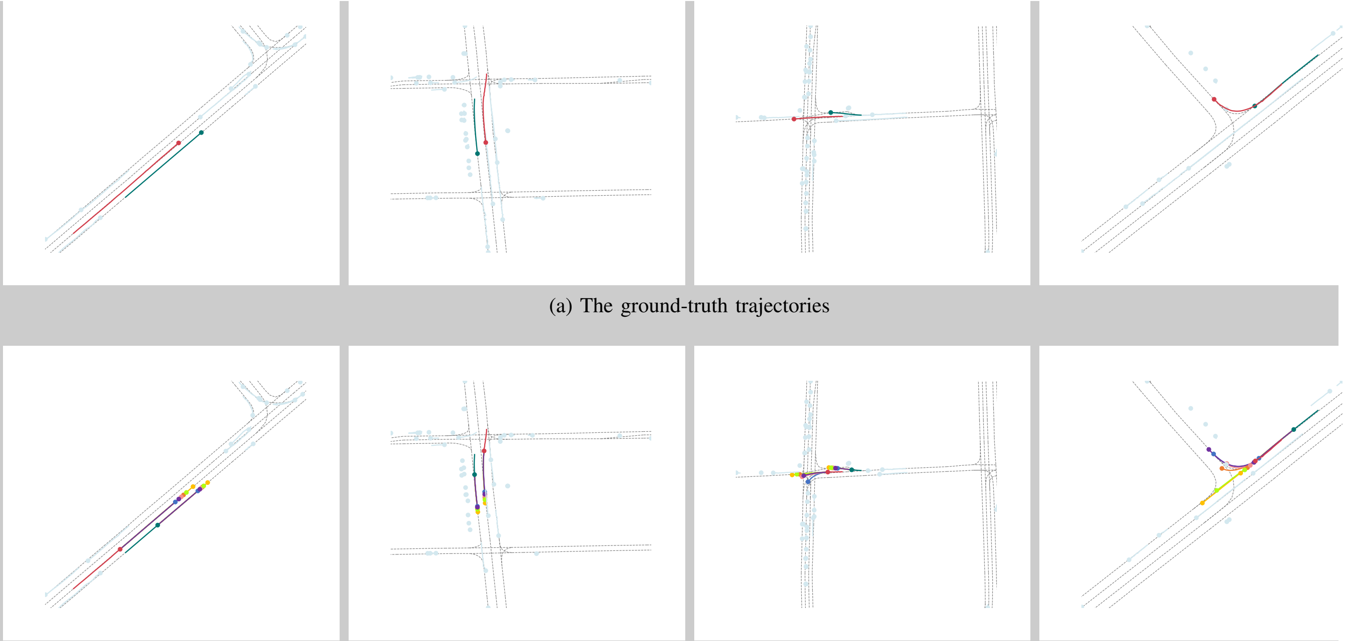
TABLE I

THE RESULTS ON THE ARGOVERSE 1 VALIDATION SET

Methods	**minFDE**	**minADE**	**MR**
LaneRCNN [10]	1.19	0.77	0.082
TNT [6]	1.29	0.73	0.093
mmTransformer [33]	1.21	0.72	0.092
DenseTNT [7]	1.05	0.73	0.098
SSL-Lanes [34]	1.01	0.70	**0.086**
TPCN [16]	1.15	0.73	0.11
HiVT [32]	0.96	0.66	0.09
IPT (ours)	**0.95**	**0.65**	0.09

C. Qualitative Results

Fig. 3 shows qualitative results of our proposed model prediction examples on four diverse sequences of the Argoverse validation set. From the visualizations, it is evident that the model has a strong capability to accurately predict multi-modal trajectories.



(a) The ground-truth trajectories

(b) The past observed trajectories and multimodal predictions

Fig. 3. Qualitative results of our model on the Argoverse validation set.

TABLE II

THE RESULTS ON THE ARGOVERSE 1 ONLINE TEST SET

Methods	**minFDE**	**minADE**	**MR**
LaneRCNN [10]	1.4526	0.9038	0.1232
GOHOME [35]	1.4503	0.9425	0.1048
TNT [6]	1.4457	0.9097	0.1656
THOMAS [36]	1.4388	0.9423	0.1038
mmTransformer [33]	1.3383	0.8436	0.1540
MultiModalTransformer [37]	1.2905	0.8372	0.1429
DenseTNT [7]	1.2815	0.8817	0.1258
SSL-Lanes [34]	1.2493	0.8401	0.1326
TPCN [16]	1.2442	0.8153	0.1333
HiVT [32]	1.1693	0.7735	0.1267
IPT (ours)	1.2272	0.7959	0.1349

TABLE III

ABLATION STUDY OF THE MAP INFORMATION AND EMBEDDING DIMENSION ON THE ARGOVERSE VALIDATION SET.

Map	Embedding	minFDE	minADE	MR
✓	64	1.02	0.68	0.101
✓	128	**0.95**	**0.65**	**0.091**
	128	1.14	0.72	0.122

TABLE IV

ABLATION STUDY OF THE GLOBAL INTERACTION MODULE AND GRAPH MESSAGE PASSING OPERATORS ON THE ARGOVERSE VALIDATION SET. GMP: GRAPH MESSAGE PASSING.

Global	GMP	minFDE	minADE	MR
	PT	1.09	0.71	0.113
✓	PT	**1.02**	**0.68**	**0.101**
✓	Transformer	1.03	**0.68**	0.102
✓	CGConv	1.06	0.70	0.106

D. Ablation Study

For ablation study, we investigate our model with different control settings. In Table III, we show the results of the ablation study conducted on the validation set. First, We explore the impact of different embedding dimensions on model performance. We conduct experiments based on a small model with 64 embedding dimensions and a large model with 128 embedding dimensions. We find that higher embedding dimensions can further improve the model performance, but we keep 128 embedding dimensions for higher efficiency. Besides, without the incorporation of map information, the model suffers from performance drop by a large margin. This indicates that map information has a crucial role in trajectory prediction since the trajectories are constrained by the geometry of the lane.

In addition to the models with different control settings, we investigate a variation of 64-dimension model to analyze the contributions of different parts of our model, shown in Table IV. In this case, we only use the temporal correlations of local interactions and ignore the global motion pattern. For a more in depth performance analysis, we choose different graph message passing operators to consider the social interactions in the scene. The results validate the effectiveness of the global motion pattern component and Point Transformer aggregator as graph message passing operator. While prior approaches applied Transformer and observed an improvement in the performance, our experiments confirm that Transformer is indeed able to learn social interactions in the scene, but the point cloud relation learning aggregator can express the features of spatial relative information more.

V. Conclusion

In this work, we propose a novel framework for trajectory prediction. We use the Point Transformer aggregator to model the social interactions between agents on both local and global level. Besides, we use two Transformer encoders to capture the motion pattern and the temporal correlations of local interactions. Quantitative experiments show that the multi-hierarchical social interaction network provides a powerful tool for modeling the complex interactions that occur in autonomous driving systems, and the point cloud learning network is an innovative application to the field of trajectory prediction. We hope that these findings could shed new light on the network design for trajectory prediction.

References

[1] L. Chen, Y. Zhang, B. Tian, Y. Ai, D. Cao, and F.-Y. Wang, "Parallel driving os: A ubiquitous operating system for autonomous driving in cpss," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 4, pp. 886–895, 2022.

[2] L. Chen, Y. Li, C. Huang, B. Li, Y. Xing, D. Tian, L. Li, Z. Hu, X. Na, Z. Li, S. Teng, C. Lv, J. Wang, D. Cao, N. Zheng, and F.-Y. Wang, "Milestones in autonomous driving and intelligent vehicles: Survey of surveys," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1046–1056, 2023.

[3] W. Wang, L. Wang, C. Zhang, C. Liu, and L. Sun, "Social interactions for autonomous driving: A review and perspectives," *Foundations and Trends in Robotics*, vol. 10, no. 3-4, pp. 198–376, 2022.

[4] J. Zhao, T. Qu, X. Gong, and H. Chen, "Interaction-aware personalized trajectory prediction for traffic participant based on interactive multiple model," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 3, pp. 2184–2196, 2023.

[5] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "Vectornet: Encoding hd maps and agent dynamics from vectorized representation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 525–11 533.

[6] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, C. Li, and D. Anguelov, "Tnt: Target-driven trajectory prediction," in *Proc. Conf. Robot Learning*, 2021, pp. 895–904.

[7] J. Gu, C. Sun, and H. Zhao, "DenseTNT: End-to-end trajectory prediction from dense goal sets," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2021, pp. 15 303–15 312.

[8] H. Jeon, J. Choi, and D. Kum, "Scale-net: Scalable vehicle trajectory prediction network under random number of interacting vehicles via edge-enhanced graph convolutional neural network," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems (IROS)*, 2020, pp. 2095–2102.

[9] Y. Huang, H. Bi, Z. Li, T. Mao, and Z. Wang, "STGAT: Modeling spatial-temporal interactions for human trajectory prediction," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2019, pp. 6272–6281.

[10] W. Zeng, M. Liang, R. Liao, and R. Urtasun, "Lanercnn: Distributed representations for graph-centric motion forecasting," in *Proc. IEEE. Int. Conf. Intelligent Robots and Systems (IROS)*, 2021, pp. 532–539.

[11] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning lane graph representations for motion forecasting," in *Proc. European Conf. Computer Vision (ECCV)*, 2020, pp. 541–556.

[12] B. Kim, S. H. Park, S. Lee, E. Khoshimjonov, D. Kum, J. Kim, J. S. Kim, and J. W. Choi, "Lapred: Lane-aware prediction of multi-modal future trajectories of dynamic agents," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 14 636–14 645.

[13] J. Schmidt, J. Jordan, F. Gritschneder, and K. Dietmayer, "Crat-pred: Vehicle trajectory prediction with crystal graph convolutional neural networks and multi-head self-attention," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. Institute of Electrical and Electronics Engineers Inc., 2 2022, pp. 7799–7805.

[14] D. Xu, X. Shang, Y. Liu, H. Peng, and H. Li, "Group vehicle trajectory prediction with global spatio-temporal graph," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1219–1229, 2023.

[15] X. Mo, Z. Huang, Y. Xing, and C. Lv, "Multi-agent trajectory prediction with heterogeneous edge-enhanced graph attention network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9554–9567, 2022.

[16] M. Ye, T. Cao, and Q. Chen, "TPCN: Temporal point cloud networks for motion forecasting," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 11 318–11 327.

[17] H. He, H. Dai, and N. Wang, "Ust: Unifying spatio-temporal context for trajectory prediction in autonomous driving," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems (IROS)*, 2020, pp. 5962–5969.

[18] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 7 2017, pp. 652–660.

[19] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun, "Point transformer," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 12 2021, pp. 16 239–16 248.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Conf. Neural Information Processing Systems (NeurIPS)*, 2017, pp. 5998–6008.

[21] Y. Tian, Y. Wang, J. Wang, X. Wang, and F.-Y. Wang, "Key problems and progress of vision transformers: The state of the art and prospects," *Acta Automatica Sinica*, vol. 48, no. 4, pp. 957–979, 2022.

[22] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3d tracking and forecasting with rich maps," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[23] M. Liu, H. Cheng, L. Chen, H. Broszio, J. Li, R. Zhao, M. Sester, and M. Y. Yang, "LAformer: Trajectory prediction for autonomous driving with lane-aware scene constraints," *arXiv preprint arXiv:2302.13933*, 2023.

[24] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Attention based vehicle trajectory prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 1, pp. 175–185, 2021.

[25] Y. Yuan, X. Weng, Y. Ou, and K. Kitani, "AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2021, pp. 9793–9803.

[26] L. Lin, W. Li, H. Bi, and L. Qin, "Vehicle trajectory prediction using lstms with spatial–temporal attention mechanisms," *IEEE Intelligent Transportation Systems Magazine*, vol. 14, no. 2, pp. 197–208, 2022.

[27] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graphics*, vol. 38, pp. 1–12, 10 2019.

[28] Y. Liu, B. Fan, S. Xiang, and C. Pan, "Relation-shape convolutional neural network for point cloud analysis," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8895–8904.

[29] X. Yan, C. Zheng, Z. Li, S. Wang, and S. Cui, "PointASNL: Robust point clouds processing using nonlocal neural networks with adaptive sampling," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 3 2020, pp. 5588–5597.

[30] Y. Liu, B. Tian, Y. Lv, L. Li, and F.-Y. Wang, "Point cloud classification using content-based transformer via clustering in feature space," *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 8, pp. 1714–1722, 2023.

[31] H. Zhao, J. Jia, and V. Koltun, "Exploring self-attention for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 073–10 082.

[32] Z. Zhou, L. Ye, J. Wang, K. Wu, and K. Lu, "HiVT: Hierarchical vector transformer for multi-agent motion prediction," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2022.

[33] Y. Liu, J. Zhang, L. Fang, Q. Jiang, and B. Zhou, "Multimodal motion prediction with stacked transformers," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 7577–7586.

[34] P. Bhattacharyya, C. Huang, and K. Czarnecki, "SSL-Lanes: Self-supervised learning for motion forecasting in autonomous driving," in *Proc. Conf. Robot Learning (CoRL)*, 2023, pp. 1793–1805.

[35] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "GOHOME: Graph-oriented heatmap output for future motion estimation," in *Int. Conf. Robotics and Automation (ICRA)*, 2022, pp. 9107–9114.

[36] T. Gilles, S. Sabatini, D. Tsishkou, and B. Stanciulescu, "THOMAS: trajectory heatmap output with learned multi-agent sampling," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2022.

[37] Z. Huang, X. Mo, and C. Lv, "Multi-modal motion prediction with transformer-based neural network for autonomous driving," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2022, pp. 2605–2611.