Density-Adaptive Model Based on Motif Matrix for Multi-Agent Trajectory Prediction

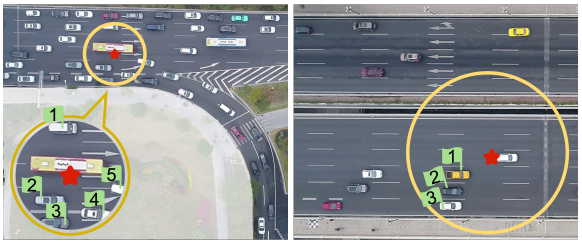
Di Wen[1,2] Haoran Xu[1,2] Zhaocheng He[1,2,3,*] Zhe Wu[2,*] Guang Tan[1,2] Peixi Peng[2,4]
[1] Sun Yat-sen University [2] PengCheng Laboratory [4] Peking University
[3] Guangdong Provincial Key Laboratory of Intelligent Transportation System
{wend25,xuhr9}@mail2.sysu.edu.cn, wuzh02@pcl.ac.cn,
{hezhch,tanguang}@mail.sysu.edu.cn, pxpeng@pku.edu.cn.

Abstract

*Multi-agent trajectory prediction is essential in autonomous driving, risk avoidance, and traffic flow control. However, the **heterogeneous traffic density** on **interactions**, which caused by physical laws, social norms and so on, is often overlooked in existing methods. When the density varies, the number of agents involved in interactions and the corresponding interaction probability change dynamically. To tackle this issue, we propose a new method, called **D**ensity-**A**daptive Model based on **M**otif **M**atrix for Multi-Agent Trajectory Prediction (DAMM), to gain insights into multi-agent systems. Here we leverage the **motif matrix** to represent dynamic connectivity in a higher-order pattern, and distill the interaction information from the perspectives of the spatial and the temporal dimensions. Specifically, in spatial dimension, we utilize multi-scale feature fusion to adaptively select the optimal range of neighbors participating in interactions for each time slot. In temporal dimension, we extract the temporal interaction features and adapt a pyramidal pooling layer to generate the interaction probability for each agent. Experimental results demonstrate that our approach surpasses state-of-the-art methods on autonomous driving dataset.*

1. Introduction

Multi-agent trajectory prediction is becoming increasingly attractive not only in academia but also in industry [18], particularly for automatic system [57], safety planning [63] and traffic flow control [28]. Existing methods [12, 19, 24, 34, 43, 57] often neglect the influence of varying traffic density on interactions between a target agent and its neighbors. They often treat all vehicles (agents) in the same scene equally, leading to the lack of adaptability. More exactly, it lacks of multi-scale neighbor selection and dynamic interaction analysis. Specifically, they do not adaptively choose participating neighbors and assume a constant interaction probability between the target agent and its neighbors.



(a) High density (b) Low density

Figure 1. The impact of heterogeneous traffic density in High-Density (1a) and Low-Density (1b) scenes. Here the star represents the target agent, the circles depict the selection ranges of neighbors and the numbers in the flags indicate agent ID. This highlights the significant influence of agent density on their interaction behaviors. Importantly, the range and interaction probability vary as the density changes across different time steps and among distinct agents.

To demonstrate the impact of heterogeneous traffic density and the necessity of density-adaptive method, we provide a visualization in Figure 1. When selecting multi-scale neighbors spatially, in high-density scenarios (Figure 1a), agents cluster closely due to space limitations, leading to fewer occurrences of interaction behaviors such as lane changes and overtaking maneuvers. Consequently, a small neighbor range (yellow circles) is used to model interactions with the target agent (red star). In contrast, a broader range is utilized in low-density situations (Figure 1b). For dynamic interaction probability in temporal dimension subsequent to neighbors selection, the interaction probabilities of neighbors (green flags with numbers) are unequal, and these probabilities may change in the subsequent moment due to variations in neighbors' velocity, position and other factors. For example, ID.1 has a higher probability to interact with the target agent than ID.3, yet the interaction

*Joint Corresponding Authors.

sequence might change in the next time step.

In this paper, we propose an innovative model called **D**ensity-**A**daptive Model based on **M**otif **M**atrix for Multi-Agent Trajectory Prediction (DAMM). With the aid of motif matrix, our model effectively captures the dynamic of interaction (further elaboration can be found in Section 3.2). Based on the motif matrix, our model consists of two essential components: spatial and temporal adaptation. In the Adaptive Spatial Interaction (ASI) block, we select neighbor ranges adaptively by considering multi-scale features. We then aggregate the motif matrix in the temporal dimension to capture the underlying interactions between the target agent and its neighbors. This process is akin to selecting nodes in a graph network. In the Adaptive Temporal Interaction (ATI) block, we determine the probability of interaction between agents based on the selections made in the ASI block. Higher probabilities during interaction with neighbors result in greater assigned weights. These varying weights constitute the edges in a graph network. Finally, the features of nodes and edges generated by ASI and ATI are input into the Graph Attention Network (GAT) [47] to extract attentive interaction features.

To summarize, the main contributions of this paper are:

1. Our model tackles the challenge of heterogeneous traffic densities, enabling dynamic capture of interaction features in neighbor selection and interaction probability.
2. We use motif matrix to achieve higher-order connectivity in clustering complex networks for fine-grained interaction modeling.
3. The results indicate that our model significantly enhances performance, with improvements of up to 19.77% in ADE (average displacement error) and 2.38% in FDE (final displacement error).

2. Related work

Trajectory Prediction. The methods for trajectory prediction can be broadly classified into four categories [18]. These include: (i) Physics-based methods, e.g., single trajectory [29], Monte Carlo [5] and Kalman filtering [41]; (ii) Classic machine learning methods, e.g., Gaussian process [22], dynamic Bayesian network [21], hidden Markov model [3] and support vector machine [40]; (iii) Deep learning methods, for instance, sequential network [48], generative model [20] and graph neural networks (GNN) [42]; and (iv) Reinforcement learning methods, e.g., deep inverse reinforcement learning [55], inverse reinforcement learning [1] and generative adversarial imitation learning [58].

Model of Multi-Agent. Multi-agent models play a crucial role in vehicle trajectory prediction by describing the relationship and influence between vehicles, which is a significant component of interaction. In SMART [44] simulates diverse trajectories in the top view and proposes a novel method that generates diverse predictions while considering scene semantics and multi-agent interactions. Grin [26] proposes a conditional deep generative model that integrates advancements in GNN. GroupNet [52] introduces a trainable multi-scale hypergraph to capture pairwise and group-wise interactions at multiple group sizes. Despite these advancements, current methods don't consider the impact of density on interaction, here we model for varying density for adaptive describe the interaction between agents.

Heterogeneous Traffic Density. Heterogeneous traffic flow density refers to the distribution of vehicles with different characteristics, such as sizes or speeds, within a specific area or road network. The impact of heterogeneous traffic density on vehicle trajectory prediction mainly includes these aspects: Firstly, it increases interaction complexity [8, 31, 51]. Furthermore, it causes changes in driving behavior, such as car-following and lane-changing [25, 35]. Besides, it causes changes in traffic flow [30, 39]. Therefore, accurate vehicle trajectory prediction requires accounting for heterogeneous traffic density. However, existing methods focus only on the before and after states without a dynamic process. So we model this process adaptively in spatial and temporal dimensions, which captures the interaction changes caused by varying density.

3. Preliminary

3.1. Problem Formulation

In a traffic scenario with N agents, referred to as vehicles V_i for $i \in \mathcal{N} = \{1, \cdots, N\}$, the history trajectory X of V_i over time steps $t \in [-t_-, 0]$ is represented as $X_i = (p_i^{-t_-}, \cdots, p_i^0)$, and the future trajectory Y over time steps $t \in [1, t_+]$ is represented as $Y_i = (p_i^1, \cdots, p_i^{t_+})$. Each state p_i^* is characterized by four dimensions, encompassing the agent's positions (x, y), speed, and heading.

We aim to predict the distribution of future trajectories $p(Y|X, Z)$ based on history trajectory X and feature information Z. For the future behavior of each agent, we introduce a set of latent variables of agent i as $\Phi = \{\Phi_i\}_{i \in \mathcal{N}}$, and rewrite the future trajectory distribution as

$$\begin{aligned} p(Y|X,Z) &= \sum\nolimits_i \left[p(Y|\Phi_i, X, Z) \cdot p(\Phi_i|X, Z) \right] \\ &\triangleq \sum\nolimits_i p_D \cdot p_\theta, \\ \text{with } [\Phi_i &= \Phi_{s_i} \cup \Phi_{t_i}] \wedge [\Phi_{s_i} \cap \Phi_{t_i} \neq \emptyset], \end{aligned} \tag{1}$$

where Φ_{s_i} corresponds to spatial features and Φ_{t_i} represents temporal features. We employ the Conditional Variational AutoEncoder (CVAE) framework for accuracy and generalization. For further details, please refer to Appendix 7.

3.2. Motif matrix

Dynamic capturing and describing of interactions among agents are crucial, involving the separation of agents inter-

acting with the target from the rest. This process resembles graph clustering. We utilize a network clustering framework known as network motifs [33] to portray this interaction at a higher-order pattern.

In this section, we utilize an unweighted and undirected network graph [59] $G = (V, E)$ with a node set V ($card\,(V) = N$) and an edge set E. The matrix A represents the adjacency matrix [50] corresponding to G. A graph cut is necessary to segregate the neighboring agents that interact with the target agent, forming a subgraph S ($S \subsetneqq V$) from the remaining set. Conversely, the set $\overline{S}$ ($\overline{S} = V \setminus S$) includes few neighbors interacting with the target agent. The cut of a node set S is denoted as cut(S), representing the number of edges with one endpoint in S and the other in the complementary set $\overline{S}$ [4]. The set S is considered a cluster [13] within graph G. Here, we employ a framework for identifying network clusters to gain insights into agents' interactions.

The complexity of agent interactions cannot be adequately captured [45] by simple connections. Therefore, we employ higher-order connectivity patterns to dynamically cluster complex networks [61] based on the adjacency matrix A. The matrix B encodes the edge pattern among the N nodes, with B being a binary matrix of size $k \times k$ ($k \leqslant N$), and $\mathcal{A} \subset \{1, \cdots, N\}$ is a set of anchor nodes. The network motifs are defined as:

Definition 1 *[56] Let $\chi_{\mathcal{A}}$ be a selection function that takes the subset of a $k-$tuple indexed by $\mathcal{A}$, set $(\cdot)$ be the operator that takes an (ordered) tuple to an (unordered) set, specifically*

$$set\left\{(v_1, \cdots, v_N)\right\} = \left\{v_1, \cdots, v_N\right\}.$$

*The set of **motifs** denotes by $M\,(B, \mathcal{A})$ is defined as*

$$\widetilde{M}\,(B, \mathcal{A}) = \left\{(set\,(v)\,, set\,[\chi_{\mathcal{A}}\,(v)]) \mid v \in V,\ A_v = B\right\},$$

where A_v is the $k \times k$ adjacent matrix on the subgraph included by the k nodes of the ordered vector v.

We denote M as the "motif matrix", as defined in Definition 1, capturing higher-order interactions among agents. The motif matrix is computed by considering agents' interactions as edges and systematically scanning the network to identify all instances of 3-node subgraphs. The method for connecting these subgraphs is illustrated in Figure 2. For example, in the Agent Graph (left part) the rectangle $abed$, where the side lengths are $be = 1$, $de = \sqrt{3}$ and $(ef \perp be) \wedge (ef = be)$, the contact range is $\sqrt{3}$. Notably, when considering 'a' as a target agent, the length $ae = 2$ exceeds the limit of $\sqrt{3}$ thus a and e are not connected. The matrix A records these connection relationships.

Since we focus solely on unidirectional interactions from neighbors to the target agent, we choose the 10th[1] motif

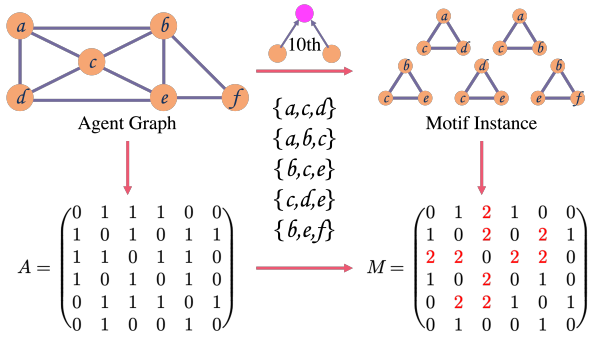


Figure 2. The process of transitioning from an adjacency matrix A to a motif matrix M. The set $\{a, b, c, d, e, f\}$ represents different agents. The 10th[1] indicates the chosen type, where the pink circle corresponds to the target agent. Elements in M indicate the count of duplicated edges within all triangles, with modifications highlighted in red. For instance, the edge ac is present in two triangles, resulting in $M_{1,3} = M_{3,1} = 2$.

type as depicted in Figure 2. The motif matrix M is obtained from agents' positions (x, y) in spatial and temporal dimensions, denoted as $M_{s,t}$. Section 4 details the specific decomposition.

4. Methodology

4.1. Overview of Framework

Our model, ***D***ensity-***A***daptive Model based on ***M***otif ***M***atrix for Multi-Agent Trajectory Prediction (DAMM, Figure 3), consists of **A**daptive **S**patial **I**nteraction (ASI) block and **A**daptive **T**emporal **I**nteraction (ATI) block. We sequentially treat each agent i in the scenes as the target agent in our model and focus the interaction between the target agent V_i and its neighbors $\{V_j\}_{j \neq i}^{N}$. As illustrated in Figure 3, we separate the neighbors interaction with target agent and utilize the motif matrix to describe this connectivity. Next, we decompose the model into spatial and temporal dimension. In the spatial dimension (ASI), we determine the agents involved in the interaction process by adaptively selecting the radius range. In the temporal dimension (ATI), we sort the interaction probabilities after making the spatial selections. After obtaining features from these blocks, we incorporate additional features to form node and edge features, which are then input into the Graph Attention Network (GAT). Ultimately, various trajectories are obtained using GAT.

Prior models concentrate on static interactions, while our emphasis is on capturing dynamic density changes, allowing for higher-order connectivity to extract features for modeling heterogeneous traffic density.

4.2. Adaptive Spatial Interaction

In the ASI block (Figure 4a), it is of particular significance to determine which neighbors interact with the target agent

[1] Higher-order organization of complex networks [4].

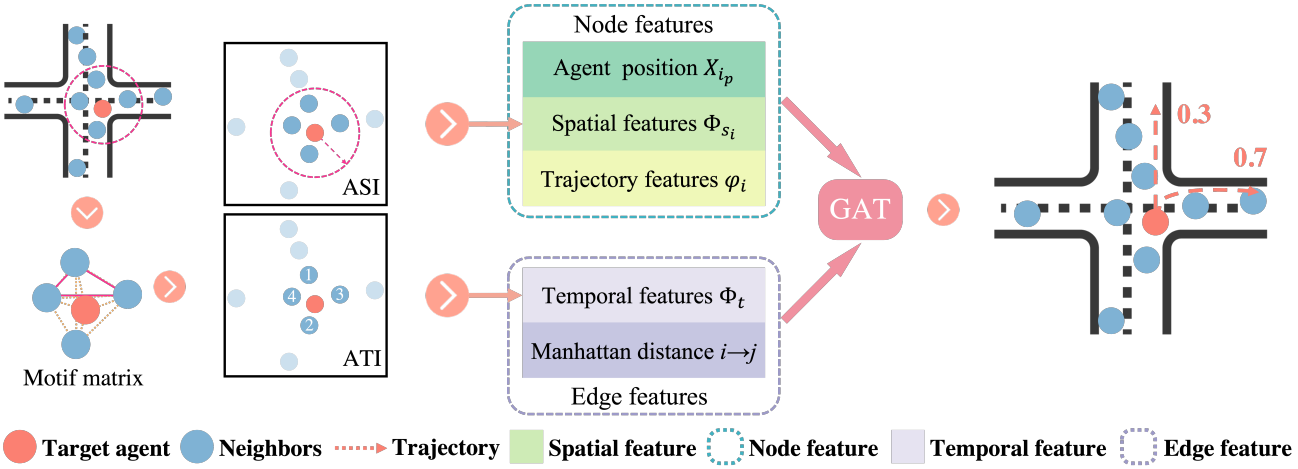
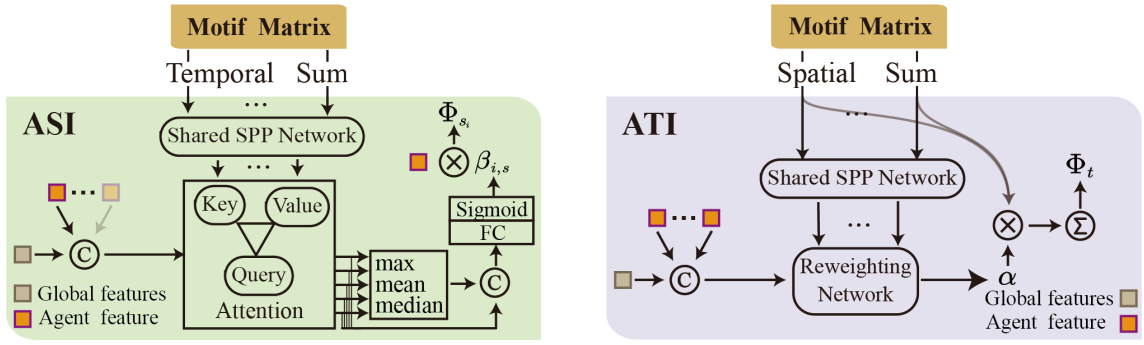


Figure 3. The DAMM model is composed of a spatial block (termed ASI) and a temporal block (termed ATI), modeling the process of interaction between target agent and its neighbors using density adaptation. ASI determines the range (pink dotted circle) for multi-scale neighbors selection, while ATI reflects the dynamic interaction analysis (where numbers represent interaction probabilities). Features from ASI and ATI are integrated and processed by the GAT to predict various trajectories (where number denotes generation probability).



(a) Adaptive Spatial Interaction

(b) Adaptive Temporal Interaction

Figure 4. The Adaptive Spatial Interaction (ASI) block and Adaptive Temporal Interaction (ATI) block. The symbol Φ_* denotes features from different blocks, with β and α are the coefficients corresponding to the features in each block.

within the ranges. To address this, we acquire fusion features β derived from both motif matrix and agent, subsequently integrating them with each agent feature φ_i. These features constitute a subset of the node of the GAT.

To achieve this, we must obtain the motif matrix $M_{s,t}$ in the spatial dimension like $M_{s,*}$. Subsequently, we erase the temporal dimension and obtain a motif matrix $\widehat{M}_s$ for each radius as:

$$\widehat{M}_s \triangleq \sum\nolimits_t M_{s,t} \ (s = s_1, \cdots, S), \tag{2}$$

here, s represents the range for selecting neighbors, ranging from s_1 to S. In this context, we obtain the motif matrices in the spatial dimension as $\widehat{M}_{s_1}, \ldots, \widehat{M}_S$.

To extract primary features $\varphi_s^{\widehat{M}}$ from the motif matrix $\widehat{M}_s$, we utilize the Spatial Pyramid Pooling (SPP) network [17] to aggregate the features from each $\widehat{M}_s$ into a unified dimension. This process is encoded by

$$\varphi_s^{\widehat{M}} = \mathrm{LN}\left\{\mathrm{FC}\left[\mathrm{SPP}\left(\widehat{M}_s\right)\right]\right\}, \tag{3}$$

with Fully-Connected (FC) layer and Layer Normalization (LN). For fusing features from both agents and motif matrices in the spatial dimension, we utilize a multi-head attention mechanism. Specifically, queries Q concatenate agents feature while keys K and values V capture information from different motif matrices. The formulation of this process for each agent i can be described as follows:

$$\begin{aligned} &\mathrm{MultiHead}\left(Q_i, K_s, V_s\right) \\ &= \left[W^Q\left(\varphi_i; \varphi_G\right), W^K \varphi_s^{\widehat{M}}, \ W^V \varphi_s^{\widehat{M}}\right], \end{aligned} \tag{4}$$

where $i \in \mathcal{N}$ and W^* represents corresponding weight matrix. We use a semicolon to denote concatenation, while a comma represents components of a vector in this paper.

The feature of each agent φ_i and the global feature φ_G in Eq. (4) are obtained via

$$\begin{aligned} \varphi_i &= \text{LSTM}\left[\text{MLP}\left(X_i\right)\right], \\ \varphi_G &= \text{MLP}\left[\text{PointNet}\left(X_1; \cdots; X_N\right)\right]. \end{aligned} \tag{5}$$

Here LSTM and MLP represents Long Short Term Memory and Multi-Layer Perceptron respectively. We employ PointNet [38] to maintain consistent dimensions for agents across various scenes. To mitigate the influence of large dot product values, we use scaled dot-product attention, that is

$$\begin{aligned} \beta_{i,s} &= \text{MultiHead}\left(Q_i, K_s, V_s\right) \\ &= \left(\text{head}_1; \cdots; \text{head}_h\right) W^O, \\ \text{with head}_k &= \text{Attention}\left(Q_i W_k^A, K_s W_k^A, V_s W_k^A\right). \end{aligned}$$

In order to mitigate the information loss of motif matrix feature $\varphi_s^{\widehat{M}}$ due to SPP in Eq. (3), we apply three statistical functions: maximum, minimum, and median (using the first three letters for brevity) to β_i, derived from $\{\beta_{i,s_1}, \cdots, \beta_{i,S}\}$. Then, we combine the original features with the statistical features through

$$\beta_i^{\widehat{M}} = \text{MLP}\left(\beta_{i,s_1}; \cdots; \beta_{i,S}; \beta_{i,\text{max}}; \beta_{i,\text{min}}; \beta_{i,\text{med}}\right). \tag{6}$$

Finally, the decomposed feature of agent i in the spatial dimension is represented as $\Phi_{s_i} = \beta_i^{\widehat{M}} \odot \varphi_i$. The symbol $\odot$ represents the Hadamard product. This feature Φ_{s_i} constitutes a portion of the node features in the GAT. More detail information about the GAT can be found in Appendix 8 due to space limitations.

4.3. Adaptive Temporal Interaction

In ATI block (Figure 4b), our goal is to identify the neighbors that exert a more significant influence on the target agent following filtration from ASI block. To achieve this, we must derive an optimal coefficient α through feature fusion to aggregate matrices $\widetilde{M}_*$ as $\sum \alpha \widetilde{M}_*$. In this context, we necessitate a subset of edge features for the GAT.

As in Section 4.2, here our objective is to extract the temporal feature from the motif matrix $M_{s,t}$ like $M_{*,t}$. The motif matrix $\widetilde{M}_t$ at each time step can be obtained using an approach analogous to that in Eq. (2):

$$\widetilde{M}_t, \triangleq \sum\nolimits_s M_{s,t} \ \ (t = t_1, \cdots, T),$$

here T is the horizon of the trajectory sequence, and t_1 is the initial time step. We integrate the features of $\widetilde{M}_*$ through

$$\varphi^{\widetilde{M}} = \text{LN}\left\{\text{MLP}\left[\text{SPP}\left(\widetilde{M_{t_1}}\right); \cdots; \text{SPP}\left(\widetilde{M_T}\right)\right]\right\}. \tag{7}$$

In contrast to Eq. (3), we extract the features simultaneously. We aggregate $\varphi^{\widetilde{M}}$ and its statistical functions as Eq. (6) to prevent information loss as follows:

$$\alpha^{\widetilde{M}} = \text{MLP}\left(\varphi^{\widetilde{M}}; \varphi_{\text{max}}^{\widetilde{M}}; \varphi_{\text{min}}^{\widetilde{M}}; \varphi_{\text{med}}^{\widetilde{M}}\right).$$

The weighting of the temporal motif matrix is determined by both the global trajectory feature φ_G and the resulting temporal motif features $\alpha^{\widetilde{M}}$, which are obtained from the reweighting network via

$$\alpha^{\widetilde{M}} = \left(\alpha_{t_1}, \cdots, \alpha_T\right) \leftarrow \text{Reweighting}\left(\varphi_G; \alpha^{\widetilde{M}}\right).$$

Finally, the decomposed feature of agent i in the temporal dimension is computed as follows: $\Phi_{t_i} = \sum_{t=t_1}^{T} \alpha_t \widetilde{M}_t$. This Φ_{t_i} contributes to the edge features in the GAT. For additional details, please refer to Appendix 8.

4.4. Training the Loss

Our loss function consists of three components: distance loss L_{DIS}, selection binary loss L_{SEL}, and binary cross-entropy loss L_{KLD}. For each agent i, the L_{DIS} term quantifies the distance between the predicted trajectory $\overline{Y_i}$ and the ground truth (the actual lane for the agent's driving) using the L^2 norm. As we generate multiple trajectories using the optional top$-k$ configuration, we employ Binary Cross Entropy (BCE) loss:

$$L_{\text{SEL}} = \text{BCE}\ \left\{g^m, \text{softmax}\left[\text{MLP}\left(\varphi_i; \varphi_{GAT}; \varphi_L\right)\right]\right\},$$

where g^m is the one-hot vector of the ground truth lane. The symbol φ_* represents the feature, with the subscript i denote the features of agents, GAT net, and lanes.

L_{KLD} ensures the similarity of the latent features sampled from the prior and the posterior through Kullback-Leibler divergence. The overall loss function is a weighted sum of the three components:

$$Loss = L_{\text{DIS}} + \lambda_1 L_{\text{SEL}} + \lambda_2 L_{\text{KLD}}, \tag{8}$$

where (λ_1, λ_2) are trade-off parameters. Typically, λ_2 increases with the number of training epochs.

5. Experiments

5.1. Fundamental Information

Our method is evaluated on two real-world datasets: nuScenes [6] and Argoverse [9], and they offer trajectories for each target agent. NuScenes comprises 245,414 trajectory instances, predicting a 6-second future trajectory based on a 2-second past trajectory. Argoverse provides 323,557 scenarios, predicting a 3-second future trajectory from a 2-second past trajectory. For further details on the datasets and experimental setup, please refer to Appendix 9. We employ two widely recognized metrics to quantitatively assess the model: Average Displacement Error (ADE_K) and Final Displacement Error (FDE_K) for evaluating the top K predictions. These metrics are prevalent in trajectory prediction and are compared to each other [7].

Table 1. Baseline comparisons on the nuScene and Argoverse datasets[†]

Method	ADE_1	FDE_1	ADE_5	FDE_5	ADE_{10}	FDE_{10}	Method	ADE_1	FDE_1	ADE_6	FDE_6
AgentFormer [57]	-	-	1.86	3.89	1.45	2.86	AutoBot [16]	-	-	0.89	1.41
LDS-AF [32]	-	-	2.06	4.60	1.66	3.58	MP++* [46]	1.62	3.61	0.79	1.21
THOMAS [15]	-	6.71	**1.33**	-	1.04	-	THOMAS [15]	1.67	3.59	0.94	1.44
PreTraM [53]	-	-	1.70	4.15	1.45	3.22	HiVT [62]	1.60	3.53	0.77	1.17
Aware [60]	5.58	11.47	-	-	1.67	2.66	Aware [60]	1.61	3.54	0.86	1.31
MUSE-VAE [23]	-	-	1.38	**2.90**	1.09	2.10	LTP [49]	1.62	3.55	0.83	1.30
GOHOME [14]	-	-	1.42	-	1.15	-	GOHOME [14]	1.70	3.68	0.89	1.29
GATraj [10]	-	-	1.87	4.08	1.46	2.97	ADAPT [2]	1.59	3.50	0.79	1.17
Real-Time [27]	3.56	8.63	1.60	3.34	1.23	2.32	R-Pred [11]	1.58	3.47	**0.76**	**1.12**
Context-Aware [54]	3.54	8.24	1.59	3.28	-	-	FRM [36]	-	-	0.82	1.27
DAMM	**2.84**	**6.59**	1.39	3.14	**1.02**	**2.05**	DAMM	**1.57**	**3.42**	**0.76**	1.29

[†] The underline and **bold** number represent the best results in the baseline and global respectively.

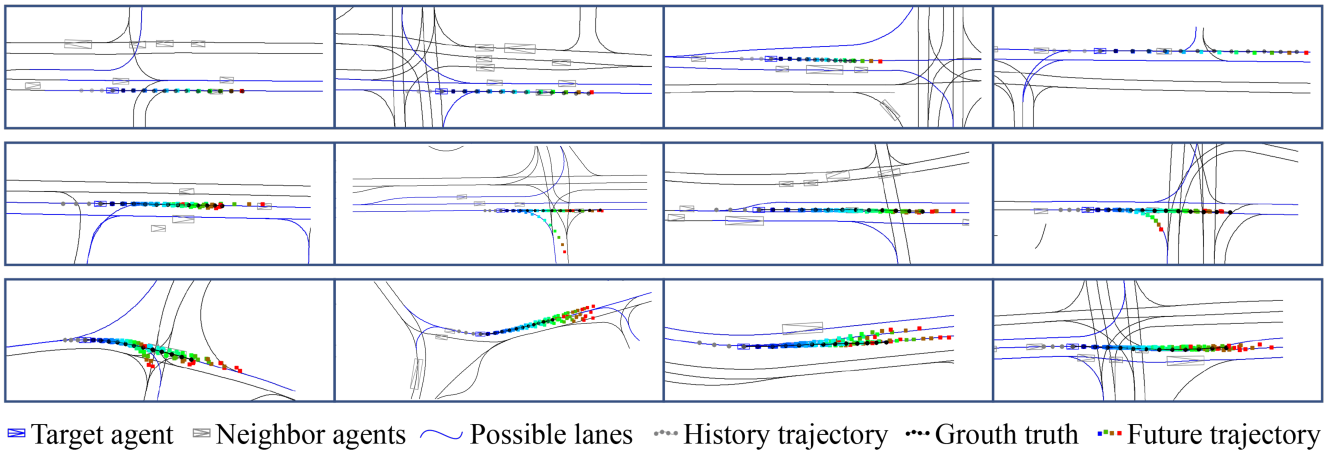


Figure 5. The upper, middle, and lower rows represent the trajectory cases generated by $K = 1, 5$, and 10, respectively. Each row depicts the predicted trajectory of the target agent (light blue box) and neighboring agents (gray box). The predicted future trajectories are shown in various cases, with start points in blue and end points in red. The ground truth trajectory is represented in black for clarity.

5.2. Results

Quantitative Results. We assess the performance of the proposed model alongside existing methods when $K = 1$, 5 and 10. the results in Table 1 highlight the best-performing methods in **red bold** and the second best methods in blue underline. Our model outperforms other methods in predicting agent trajectories.

In the nuScenes dataset, ADE1 improved from 3.54 to 2.84, a 19.77% increase in performance, while FDE10 improved from 2.10 to 2.05, a 2.38% boost. In the Argoverse dataset, we observed a maximum improvement of 9.52% in ADE ($K = 6$) and 4.37% in FDE ($K = 1$).

Our model markedly reduces errors in both ADE and FDE, particularly excelling in ADE, demonstrating its stability in predicting distant future positions. This proficiency is attributed to the design of DAMM, effectively capturing dependencies by dynamically extracting and describing higher-order features for any agent at previous time steps while predicting the agent's future position.

Prediction Example. Figure 5 demonstrates the effectiveness of our model in generating precise and realistic multi-agent trajectory predictions for K =1, 5 and 10.

The upper showcase the case $K = 1$. The results clearly demonstrate that the DAMM model is capable of generating trajectories that closely match the real trajectory, without any instances of trajectories deviating from the road. This indicates the effectiveness of our approach in modeling the dependencies between the agents and the environment.

The middle and lower are generated by $K = 10$ and 15, which reflect the multi-agent trajectory distribution. These trajectories strike a balance between accuracy and diversity and enable us to better understand the range of possible outcomes for the given scene. The results illustrate the proficiency of our model in capturing scene context and simulating interactions to produce precise and various trajectories. At present, there's no additional filtering or extraction of lane information. Enhancing this aspect constitutes a central focus for our upcoming endeavors.

Table 2. Ablation study conducted on nuScenes. The percentage means the values greater than M_1.

M_*	STMM	ATI	ASI	ADE_1/FDE_1	Percentage	ADE_5/FDE_5	Percentage	ADE_{10}/FDE_{10}	Percentage
M_1	×	×	×	3.05/7.12	-	1.44/3.25	-	1.10/2.14	-
M_2	√	×	√	2.98/6.91	2.30%/2.95%	1.42/3.16	1.39%/2.77%	1.08/2.04	1.82%/4.67%
M_3	√	√	×	2.97/6.97	2.62%/2.11%	1.41/3.23	2.08%/0.62%	1.09/2.09	0.91%/2.34%
M_4	√	√	√	2.84/6.59	9.89%/7.44%	1.39/3.14	3.47%/3.38%	1.09/2.05	0.91%/4.21%

5.3. Ablation Studies

5.3.1 Impact of Block

Table 2 shows the evaluation results of each component of the DAMM model, including a traditional interaction method M_1 (using adjacent matrix instead of Spatial-Temporal Motif Matrix), only spatial adaptive (method M_2), only temporal adaptive (method M_3), and the complete model DAMM (method M_4).

Interaction without Motif Matrix. This section presents the results obtained equitable extraction of features from all agents using the adjacency matrix. For M_1, the FDE_1 metric is 7.12, which is 0.41 higher than the minimum baseline value of 6.71. However, this approach for feature extraction is inappropriate and results in unrealistic outcomes in real-world scenarios. This underscores the importance of our proposed adaptive model, which considers the varying inaccurate probabilities between different agents.

Effects of ASI block. The effectiveness of the ASI block in reducing trajectory prediction errors compared to M_1 through adaptive neighbors selection is clearly demonstrated by the 4.67% increase in FDE_{10}. This is because density changes are a chronic and relatively steady process, which requires more time to reflect the advantages of radius ranges. With multi-scale fusion, the model can better capture the spatial density features of the scene and adaptively adjust the neighbor radius for each agent, resulting in more accurate trajectory predictions.

Effects of ATI block. This experiment is designed to assess the ASI block's performance in predicting interaction probability. The results demonstrate that the ASI block outperforms M_1 with improvements in ADE by 0.08 and FDE by 0.15 when predicting the top 1 trajectory. As the number of trajectory predictions increases, the prediction error becomes smaller and smaller, due to the increased diversity generated by the interaction and more options for calculation and correction of the model.

Adaptive Spatial-Temporal. It is evident that jointly modeling the temporal and spatial dimensions in an adaptive manner leads to superior performance compared to single model (compare to M_1, M_2 and M_3). This finding confirms the significance of agent interactions in trajectory prediction, and our proposed method effectively handles heterogeneous traffic density. The most significant improvement in each metric is observed when $K = 1$, resulting in a performance increase of 6.89% and 7.44%, respectively.

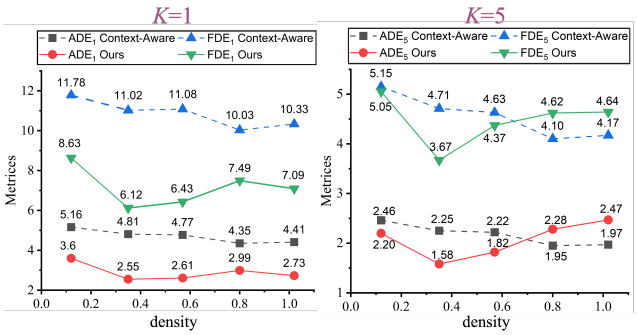


Figure 6. Density comparison with Context-Aware [54] visualization results. We represent our method with solid lines in green and red, while [54] is depicted with dashed lines in blue and gray.

5.3.2 Density

To present the results with clarity regarding density, we employ the metric "number of agents per unit area". As illustrated in Figure 1, it is defined as a circle given by $\text{Density} = n_s/\pi d_{\max}^2$, where n_s represents the number of agents in the s-th scene, and $d_{\max}$ is the maximum distance between the target agent and its neighbors.

In this context, we categorize the density of the test set in the nuScenes dataset into five intervals ranging from 0.01 to 1.14, as illustrated in Figure 7. We visually demonstrate our method by comparing it with Context-Aware [54], and the results are presented in Figure 6.

When $K = 1$, our method consistently demonstrates a significant reduction in displacement error, with an average improvement of 1.8 in ADE_1 and 3.7 in FDE_1. With $K = 5$, our metrics perform well over most of the intervals, particularly in the second interval, where there is a remarkable 29.58% improvement in ADE_5 and a 22.00% improvement in FDE_5. It's worth noting that in the fourth and fifth intervals, our metrics are inferior to those of [37], but they remain competitive. This is due to the fact that Figure 7 reveals that the last two intervals account for only 6.59% of the total density range.

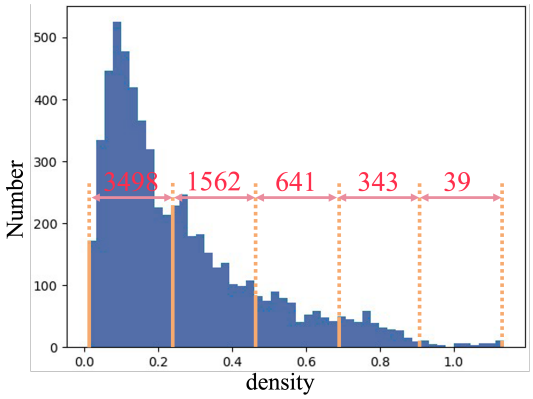


Figure 7. The number of scenes at various densities. The interval is divided by five, marked by a vertical line with numbers on it.

5.3.3 Multi-Scale

In our proposed method, we take into account multi-scale interactions among agents in the ASI block. Here, we present the results in Table 3 with single neighbor selection ranges R_S compare with ours R_O. In this setup R_S, we set the neighbor radius to 50m.

Table 3. Comparisons between single-scale R_S and multi-scale R_O methods. The $\uparrow_{\%}$ indicates the percentage increase in values compared to R_S.

Metric	ADE_1	FDE_1	ADE_5	FDE_5	ADE_{10}	FDE_{10}
R_S	3.12	7.48	1.49	3.36	1.17	2.49
R_O	2.84	6.59	1.39	3.14	1.02	2.05
$\uparrow_{\%}$	8.80%	8.96%	6.88%	6.26%	12.89%	7.57%

Significantly, the observed errors in this configuration are noticeably greater than those in our proposed model R_O. The results show that without employing multi-scale, the errors are more substantial compared to when it is utilized. Through multi-scale neighbor selection, we achieve an average improvement of 0.27 in terms of the metric value, which translates to a 8.56% improvement in percentage.

5.3.4 Impact of Loss

To assess the effectiveness of the loss functions Eq. (8), we present the results for each component in Table 4. Here, A_1 represents L_{DIS}, A_2 represents L_{DIS} and L_{SEL}, A_3 represents L_{DIS} and L_{KLD} and A_4 represents all loss functions. Compared to the A_1 experiment, the metrics of the other settings demonstrate improvements, resulting in an average increase of 31.54% in ADE and 32.47% in FDE. For further details on this section, please refer to Appendix 10.

5.4. Failure Cases

Here, we showcase instances of failures in Fig.8. The failure cases consider more realistic situations, such as turns to avoid collisions.

Table 4. Ablation study of Loss.

A_*	ADE_1	FDE_1	ADE_5	FDE_5	ADE_{10}	FDE_{10}
A_1	2.99	6.81	2.34	5.40	1.93	4.30
A_2	2.52	5.61	2.52	5.81	2.32	5.23
A_3	3.42	7.98	1.77	4.00	1.29	2.65
A_4	2.84	6.59	1.39	3.14	1.02	2.05

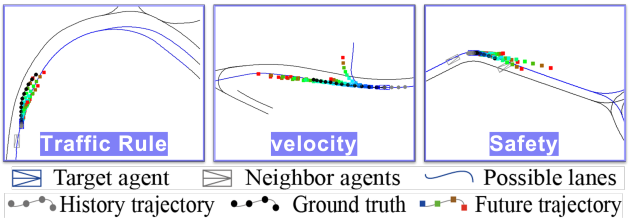


Figure 8. Some failure cases.

Our model precisely predict trajectories in compliance with traffic regulations, exemplified by a vehicle's straight trajectory perfectly aligning with our predictions. Variations in velocity lead to slight deviations in our forecasts. Prioritizing safety, our model predicts essential maneuvers, including deceleration and evasive actions, to ensure that the projections accurately mirror realistic reactions to evolving traffic situations, even when this involves straying from lane lines.

6. Conclusion

In this paper, we propose a density-adaptive model utilizing the motif matrix to capture interaction features in scenarios with heterogeneous traffic density. By decomposing the model in both spatial and temporal dimensions, it offers a detailed description of the interaction process, resulting in more accurate and diverse future trajectories for multi-agents. Experimental results on established benchmarks demonstrate the superior performance of our framework compared to state-of-the-art methods across various metrics. In the future, this approach harbors significant potential for widespread applications in autonomous vehicles. On an individual vehicle level, it supports safe decision-making and route optimization. From a macro perspective, it aids in traffic control and accident reduction.

Acknowledgement. The study was funded by the National Natural Science Foundation of China (U21B2090, 62102207), National Key Research and Development Program of China (2023YFB4301900), Shenzhen Science and Technology Plan Project Key Technical Tackling Project (JSGG20220831094604008), Guangzhou Science and Technology Plan Project Key Field RD Project (202206010056) and Major Key Project of PengCheng Laboratory (PCL2023A08). Computing support was provided by Pengcheng Cloudbrain.

References

[1] Rushdi Alsaleh and Tarek Sayed. Modeling pedestrian-cyclist interactions in shared space using inverse reinforcement learning. *Transportation research part F: traffic psychology and behaviour*, 70:37–57, 2020. 2

[2] Görkay Aydemir, Adil Kaan Akan, and Fatma Güney. Adapt: Efficient multi-agent trajectory prediction with adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8295–8305, 2023. 6

[3] Samet Ayhan and Hanan Samet. Aircraft trajectory prediction made easy with predictive analytics. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 21–30, 2016. 2

[4] Austin R Benson, David F Gleich, and Jure Leskovec. Higher-order organization of complex networks. *Science*, 353(6295):163–166, 2016. 3

[5] Adrian Broadhurst, Simon Baker, and Takeo Kanade. Monte carlo road safety reasoning. In *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.*, pages 319–324. IEEE, 2005. 2

[6] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*, 2019. 5, 2

[7] Sergio Casas, Cole Gulino, Simon Suo, Katie Luo, Renjie Liao, and Raquel Urtasun. Implicit latent variable model for scene-consistent motion forecasting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*, pages 624–641. Springer, 2020. 5

[8] Rohan Chandra, Uttaran Bhattacharya, Aniket Bera, and Dinesh Manocha. Traphic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8483–8492, 2019. 2

[9] Ming-Fang Chang, John W Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 5, 2

[10] Hao Cheng, Mengmeng Liu, Lin Chen, Hellward Broszio, Monika Sester, and Michael Ying Yang. Gatraj: A graph-and attention-based multi-agent trajectory prediction model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 205: 163–175, 2023. 6

[11] Sehwan Choi, Jungho Kim, Junyong Yun, and Jun Won Choi. R-pred: Two-stage motion prediction via tube-query attention-based trajectory refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8525–8535, 2023. 6

[12] Nachiket Deo and Mohan M Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1468–1476, 2018. 1

[13] Gary William Flake, Robert E Tarjan, and Kostas Tsioutsiouliklis. Graph clustering and minimum cut trees. *Internet Mathematics*, 1(4):385–408, 2004. 3

[14] Thomas Gilles, Stefano Sabatini, Dzmitry Tsishkou, Bogdan Stanciulescu, and Fabien Moutarde. Gohome: Graph-oriented heatmap output for future motion estimation. In *2022 international conference on robotics and automation (ICRA)*, pages 9107–9114. IEEE, 2022. 6

[15] Thomas Gilles, Stefano Sabatini, Dzmitry Tsishkou, Bogdan Stanciulescu, and Fabien Moutarde. Thomas: Trajectory heatmap output with learned multi-agent sampling. In *International Conference on Learning Representations*, 2022. 6

[16] Roger Girgis, Florian Golemo, Felipe Codevilla, Martin Weiss, Jim Aldon D'Souza, Samira Ebrahimi Kahou, Felix Heide, and Christopher Pal. Latent variable sequential set transformers for joint multi-agent motion prediction. *arXiv preprint arXiv:2104.00563*, 2021. 6

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(9):1904–1916, 2015. 4

[18] Yanjun Huang, Jiatong Du, Ziru Yang, Zewei Zhou, Lin Zhang, and Hong Chen. A survey on trajectory-prediction methods for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 7(3):652–674, 2022. 1, 2

[19] Boris Ivanovic and Marco Pavone. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2375–2384, 2019. 1

[20] Boris Ivanovic, Karen Leung, Edward Schmerling, and Marco Pavone. Multimodal deep generative models for trajectory prediction: A conditional variational autoencoder approach. *IEEE Robotics and Automation Letters*, 6(2):295–302, 2020. 2

[21] Yuande Jiang, Bing Zhu, Shun Yang, Jian Zhao, and Weiwen Deng. Vehicle trajectory prediction considering driver uncertainty and vehicle dynamics based on dynamic bayesian network. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2022. 2

[22] Kihwan Kim, Dongryeol Lee, and Irfan Essa. Gaussian process regression flow for analysis of motion trajectories. In *2011 International Conference on Computer Vision*, pages 1164–1171. IEEE, 2011. 2

[23] Mihee Lee, Samuel S Sohn, Seonghyeon Moon, Sejong Yoon, Mubbasir Kapadia, and Vladimir Pavlovic. Musevae: multi-scale vae for environment-aware long term trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2221–2230, 2022. 6

[24] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning. *Advances in neural information processing systems*, 33:19783–19794, 2020. 1

[25] Jie Li, Han Shi, Yue Guo, Guangjie Han, Ruiyun Yu, and Xingwei Wang. Tragcan: Trajectory prediction of heterogeneous traffic agents in iov systems. *IEEE Internet of Things Journal*, 10(8):7100–7113, 2022. 2

[26] Longyuan Li, Jian Yao, Li Wenliang, Tong He, Tianjun Xiao, Junchi Yan, David Wipf, and Zheng Zhang. Grin: Generative relation and intention network for multi-agent trajectory prediction. *Advances in Neural Information Processing Systems*, 34:27107–27118, 2021. 2

[27] Linhui Li, Xuecheng Wang, Dongfang Yang, Yifan Ju, Zhongxu Zhang, and Jing Lian. Real-time heterogeneous road-agents trajectory prediction using hierarchical convolutional networks and multi-task learning. *IEEE Transactions on Intelligent Vehicles*, 2023. 6

[28] Maohan Liang, Ryan Wen Liu, Yang Zhan, Huanhuan Li, Fenghua Zhu, and Fei-Yue Wang. Fine-grained vessel traffic flow prediction with a spatio-temporal multigraph convolutional network. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):23694–23707, 2022. 1

[29] Chiu-Feng Lin, A Galip Ulsoy, and David J LeBlanc. Vehicle dynamics and external disturbance estimation for vehicle path prediction. *IEEE Transactions on Control Systems Technology*, 8(3):508–518, 2000. 2

[30] Shuncheng Liu, Xu Chen, Ziniu Wu, Liwei Deng, Han Su, and Kai Zheng. Hega: Heterogeneous graph aggregation network for trajectory prediction in high-density traffic. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1319–1328, 2022. 2

[31] Yuexin Ma, Xinge Zhu, Sibo Zhang, Ruigang Yang, Wenping Wang, and Dinesh Manocha. Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6120–6127, 2019. 2

[32] Yecheng Jason Ma, Jeevana Priya Inala, Dinesh Jayaraman, and Osbert Bastani. Likelihood-based diverse sampling for trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13279–13288, 2021. 6

[33] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594): 824–827, 2002. 3

[34] Xiaoyu Mo, Zhiyu Huang, Yang Xing, and Chen Lv. Multi-agent trajectory prediction with heterogeneous edge-enhanced graph attention network. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):9554–9567, 2022. 1

[35] Sangram Krishna Nirmale, Abdul Rawoof Pinjari, and Anshuman Sharma. A discrete-continuous multi-vehicle anticipation model of driving behaviour in heterogeneous disordered traffic conditions. *Transportation Research Part C: Emerging Technologies*, 128:103144, 2021. 2

[36] Daehee Park, Hobin Ryu, Yunseo Yang, Jegyeong Cho, Jiwon Kim, and Kuk-Jin Yoon. Leveraging future relationship reasoning for vehicle trajectory prediction. In *International Conference on Learning Representations (ICLR 2023)*. Eleventh International Conference on Learning Representations, 2023. 6

[37] Bimsara Pathiraja, Shehan Munasinghe, Malshan Ranawella, Maleesha De Silva, Ranga Rodrigo, and Peshala Jayasekara. Class-aware attention for multimodal trajectory prediction. *arXiv preprint arXiv:2209.00062*, 2022. 7

[38] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 5

[39] Zhen Sean Qian, Jia Li, Xiaopeng Li, Michael Zhang, and Haizhong Wang. Modeling heterogeneous traffic flow: A pragmatic approach. *Transportation Research Part B: Methodological*, 99:183–204, 2017. 2

[40] Zhibin Qiu, Jiangjun Ruan, Daochun Huang, Ziheng Pu, and Shengwen Shu. A prediction method for breakdown voltage of typical air gaps based on electric field features and support vector machine. *IEEE Transactions on Dielectrics and Electrical Insulation*, 22(4):2125–2135, 2015. 2

[41] Jens Schulz, Constantin Hubmann, Julian Löchner, and Darius Burschka. Multiple model unscented kalman filtering in dynamic bayesian networks for intention estimation and trajectory prediction. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 1467–1474. IEEE, 2018. 2

[42] Zihao Sheng, Yunwen Xu, Shibei Xue, and Dewei Li. Graph-based spatial-temporal convolutional network for vehicle trajectory prediction in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):17654–17665, 2022. 2

[43] Haoran Song, Wenchao Ding, Yuxuan Chen, Shaojie Shen, Michael Yu Wang, and Qifeng Chen. Pip: Planning-informed trajectory prediction for autonomous driving. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 598–614. Springer, 2020. 1

[44] NN Sriram, Buyu Liu, Francesco Pittaluga, and Manmohan Chandraker. Smart: Simultaneous multi-agent recurrent trajectory prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 463–479. Springer, 2020. 2

[45] William G Underwood, Andrew Elliott, and Mihai Cucuringu. Motif-based spectral clustering of weighted directed networks. *Applied Network Science*, 5(1):1–41, 2020. 3

[46] Balakrishnan Varadarajan, Ahmed Hefny, Avikalp Srivastava, Khaled S Refaat, Nigamaa Nayakanti, Andre Cornman, Kan Chen, Bertrand Douillard, Chi Pang Lam, Dragomir Anguelov, et al. Multipath++: Efficient information fusion and trajectory aggregation for behavior prediction. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 7814–7821. IEEE, 2022. 6

[47] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017. 2, 1

[48] Chengxin Wang, Shaofeng Cai, and Gary Tan. Graphtcn: Spatio-temporal interaction modeling for human trajectory

prediction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3450–3459, 2021. 2

[49] Jingke Wang, Tengju Ye, Ziqing Gu, and Junbo Chen. Ltp: Lane-based trajectory prediction for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17134–17142, 2022. 6

[50] Eric W Weisstein. Adjacency matrix. *https://mathworld. wolfram. com/*, 2007. 3

[51] Song Xiao, Kai Chen, Xiaoxiang Ren, and Haitao Yuan. Pedestrian trajectory prediction in heterogeneous traffic using facial keypoints-based convolutional encoder-decoder network. *ACM Transactions on Internet Technology*, 22(4): 1–14, 2022. 2

[52] Chenxin Xu, Maosen Li, Zhenyang Ni, Ya Zhang, and Siheng Chen. Groupnet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6507, 2022. 2

[53] Chenfeng Xu, Tian Li, Chen Tang, Lingfeng Sun, Kurt Keutzer, Masayoshi Tomizuka, Alireza Fathi, and Wei Zhan. Pretram: Self-supervised pre-training via connecting trajectory and map. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIX*, pages 34–50. Springer, 2022. 6

[54] Pei Xu, Jean-Bernard Hayet, and Ioannis Karamouzas. Context-aware timewise vaes for real-time vehicle trajectory prediction. *IEEE Robotics and Automation Letters*, 8(9): 5440–5447, 2023. 6, 7

[55] Yifan Xu, Theodor Chakhachiro, Tribhi Kathuria, and Maani Ghaffari. Solo t-dirl: Socially-aware dynamic local planner based on trajectory-ranked deep inverse reinforcement learning. *arXiv preprint arXiv:2209.07996*, 2022. 2

[56] Hao Yin, Austin R Benson, Jure Leskovec, and David F Gleich. Local higher-order graph clustering. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 555–564, 2017. 3

[57] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9813–9823, 2021. 1, 6

[58] Qichao Zhang, Yinfeng Gao, Yikang Zhang, Youtian Guo, Dawei Ding, Yunpeng Wang, Peng Sun, and Dongbin Zhao. Trajgen: Generating realistic and diverse trajectories with reactive and feasible agent behaviors for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 23 (12):24474–24487, 2022. 2

[59] Yilin Zhang and Karl Rohe. Understanding regularized spectral clustering via graph conductance. *Advances in Neural Information Processing Systems*, 31, 2018. 3

[60] Yiqi Zhong, Zhenyang Ni, Siheng Chen, and Ulrich Neumann. Aware of the history: Trajectory forecasting with the local behavior data. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 393–409. Springer, 2022. 6

[61] Dawei Zhou, Jingrui He, Hasan Davulcu, and Ross Maciejewski. Motif-preserving dynamic local graph cut. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1156–1161. IEEE, 2018. 3

[62] Zikang Zhou, Luyao Ye, Jianping Wang, Kui Wu, and Kejie Lu. Hivt: Hierarchical vector transformer for multi-agent motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8823–8833, 2022. 6

[63] Dekai Zhu, Guangyao Zhai, Yan Di, Fabian Manhardt, Hendrik Berkemeyer, Tuan Tran, Nassir Navab, Federico Tombari, and Benjamin Busam. Ipcc-tp: Utilizing incremental pearson correlation coefficient for joint multi-agent trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5507–5516, 2023. 1