I2T: From Intention Decoupling to Vehicular Trajectory Prediction Based on Prioriformer Networks

Yi Zhou, *Member, IEEE*, Zhangyun Wang, *Student Member, IEEE*, Nianwen Ning, Zhanqi Jin, Ning Lu, *Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—A reliable driving trajectory prediction of surrounding vehicles is an essential reference for decision-making and safe driving of an autonomous vehicle. Although predicting short-term trajectories can be well achieved, it is still very challenging for long-term prediction of trajectories since the prediction space grows exponentially. In this paper, we propose a novel architecture for trajectory prediction from factored intention estimation (I2T), which decouples the trajectory prediction space into a high-level space for intention estimation and a low-level space for motion prediction. The long-term dependencies between intention cues and future motions during driving are naturally extended to the internal sharing mechanism of I2T, leading to improved performance. Furthermore, we design a Prioriformer model to serve as the backbone network for I2T so that it can accurately capture the long-term dependency couplings related to the task of intention estimation or motion prediction. Prioriformer model adopts a personalized normalization method, which facilitates learning latent representations of long-term features and avoids getting stuck on local optimum. A designed multi-scale fusion encoder extracts features from various receptive fields and then learns richer information from the representation subspaces. An efficient non-autoregressive decoder reduces the pressure in long-term prediction of trajectories while avoiding cumulative errors. Experiments on three real-world motion datasets show that I2T can significantly outperform the state-of-the-art.**

***Index Terms*—Autonomous vehicles, intention estimation, trajectory prediction, Transformer.**

I. Introduction

Leveraging predictions of driving intentions and trajectories, autonomous vehicles can be aware of future movements of surrounding vehicles and identify hazardous driving conditions for planning routes proactively. Possible risks are therefore minimized by avoiding collisions with others caused by sudden reactive decisions. Autonomous vehicles equipped with intelligent sensing devices provide relevant information about the states, lanes, and maneuvers of surrounding vehicles, which can be utilized to predict the intentions and trajectories of target vehicles.

The future motions of vehicles are highly multimodal due to the uncertainty of the driving mode. Recent researchers have made significant progress in multimodal trajectory prediction by learning from real-world driving examples. Much of the current work performs well in short-term (1s-2s) trajectory prediction [1], [2], [3], [4]. However, as the trajectory prediction horizon increases, cumulative errors would lead to continuous degradation of the model performance. The failure of long-term prediction of trajectories results in incorrect decisions by autonomous vehicles, posing a serious threat to safe driving. Therefore, improving the long-term prediction accuracy of trajectories is an urgent and challenging task. It requires **(a)** long-term intention cues of multimodal future motions and **(b)** an extraordinary ability to capture long-term dependency couplings between observed scene contexts and future motions.

There are some previous studies on solving requirement **(a)** to improve the accuracy of trajectory prediction. The family of intention-bridging-based multimodal trajectory prediction methods [3], [4], [5], [6] pursues multimodality by traversing unimodal trajectories on bridged intention combinations. Restricted to the modeling mechanism, the number of predicted modes must correspond to the number of intention combinations and forward propagations of the model. This prediction paradigm suffers from scalability, accuracy, and efficiency bottlenecks in intention-aware-based trajectory prediction. Hence, for requirement **(a)**, we rethink the intention-aware multimodal motion prediction paradigm in this paper. We propose an end-to-end trajectory prediction architecture, I2T, which decomposes the trajectory prediction problem into intention estimation, trajectory prediction, and sample selection. As shown in Fig. 1, an intention estimator first predicts the probability distribution of high-level decision intention, i.e., the lane-change intention. Then, a motion predictor predicts low-level trajectories in the corresponding decision intention space for the target vehicle based on the high-level intention distribution and learned shared network weights. These trajectories are specified for gradient updating

Manuscript received 30 October 2022; revised 29 September 2023 and 16 January 2024; accepted 4 March 2024. Date of publication 21 March 2024; date of current version 1 August 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62176088, in part by the International Strategic Innovative Project of the National Key Research and Development Program of China under Grant 2023YFE0112500, and in part by the Program for Science and Technology Development of Henan Province under Grant 222102210067. The Associate Editor for this article was V. Pandi. *(Corresponding author: Nianwen Ning.)*

Yi Zhou, Zhangyun Wang, Nianwen Ning, and Zhanqi Jin are with the School of Artificial Intelligence, Henan University, Zhengzhou 450046, China, and also with the International Joint Research Laboratory for Cooperative Vehicular Networks of Henan, Zhengzhou 450046, China (e-mail: zhouyi@henu.edu.cn; zhyun@henu.edu.cn; nnw@henu.edu.cn; Jinzhanqi@henu.edu.cn).

Ning Lu is with the Department of Electrical and Computer Engineering, Queen's University, Kingston, ON K7L 3N6, Canada (e-mail: ning.lu@queensu.ca).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/TITS.2024.3375900

1558-0016 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

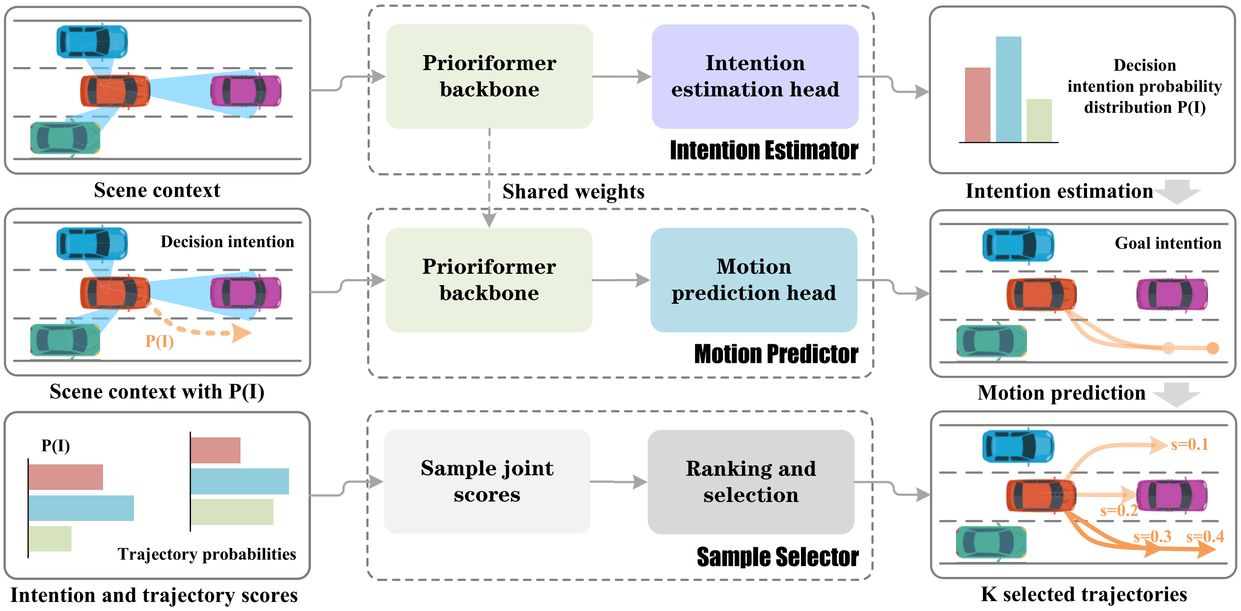


Fig. 1. Overview of I2T. I2T consists of an intention estimator, a motion predictor, and a sample selector. The intention estimator and the motion predictor share the Altioriformer network structure. The intention estimator outputs the decision intention probability distribution of target vehicle. The motion predictor inputs the scene context information with the intention probability distribution and merges shared weights containing a large amount of intention information to generate candidate trajectories based on the goal intention for target vehicle. The sample selector filters the most probable K trajectories by calculating the joint scores of decision intention and trajectory.

by the fine-grained goal intention condition, i.e., the motion endpoints of the target vehicle, enabling multimodal prediction. We refer to this process of decomposing the agent intention into the decision and goal intentions as intention decoupling. The internal sharing mechanism of I2T improves the long-term dependent coupling between driving intentions and future motions, allowing the motion predictor and the intention estimator to benefit from the training of each other. Finally, a sample selector calculates the joint scores of intention and trajectory, followed by filtering the most likely K trajectories.

The rational design of the I2T backbone network is crucial for the efficient implementation of requirement **(b)**. Recent studies [7], [8], [9] show that Transformer model holds potential in capturing long-term dependency couplings due to its $\mathcal{O}(1)$ theoretical signal transmission distance. Nevertheless, applying Transformer to the trajectory prediction needs to overcome a series of challenges. Vanilla Transformer [10] is originally designed for the language model. There is a gap between language features and trajectory features. Specifically, languages are human-made signals with high semantic and information density. Vehicle trajectories, as data generated by real-world vehicles in traffic scenarios, are sparse in semantic information, but rich in spatial-temporal information and statistical features. Thus, for requirement **(b)**, we strive to answer the question: whether Transformer can be improved to overcome this gap and encourage learning useful trajectory features while maintaining higher long-term prediction capability?

Vanilla Transformer has the following three limitations in solving tasks related to trajectory prediction: (1) The anisotropy of trajectories cannot be adequately expressed. The rich high-level semantic information contained in each word allows the model to easily learn the differences between words. By contrast, the semantic level of trajectories is lower, and the dependencies of trajectories at various moments are implicit. An effective method is expected to be constructed that maps the variability of trajectory points over time to a suitable level for capturing the dependencies better. (2) The inherent encoder structure of Vanilla Transformer cannot explicitly extract the features of the various trajectory phases. As an interpretable example, Fig. 9 in Appendix C shows that early, mid-term, and late observed trajectories contribute differently to the prediction performance. Hence, redesigning the encoder structure based on trajectory properties is imperative to enhance the capability of diverse feature extraction. (3) As a stepwise inference paradigm, autoregressive inference tends to be affected by cumulative errors of dynamic decoding, leading to a decrease in prediction accuracy with an increasing trajectory prediction horizon.

In this paper, these limitations are explicitly explored and solved. We design a backbone network called Prioriformer for the intention estimation and motion prediction tasks of I2T by mining the priori knowledge of trajectories. To overcome limitation (1), we develop an offline algorithm that estimates the priori mean, variance, and trajectory of each observed trajectory. Prioriformer captures the variability of trajectory points over time through a personalized normalization (PN) approach based on the priori mean and variance. Thus, latent representations and long-term dependencies of trajectories are better learned by the model with robust gradient updates. To overcome limitation (2), we design a multi-scale fusion encoder for Prioriformer to capture the spatial and temporal features of various receptive fields (trajectory phases that models can accept). Specifically, different contributing trajectory segments are represented by encoder blocks of various scales, then multi-scale fusion is employed to extract

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

essential and long-term features. To overcome limitation (3), Prioriformer employs a non-autoregressive decoder to generate the entire future trajectory at one step, which avoids substantial cumulative errors and computation costs associated with the autoregressive decoder. In addition, we make a further improvement by naturally extending the concept of the priori trajectory to input augmentation of the decoder, which improves the accuracy of non-autoregressive inference by alleviating the long-term prediction pressure of trajectories. We conduct extensive experiments to demonstrate the effectiveness of our method. The contributions of this paper are summarized as follows:

- I2T provides a novel paradigm for multimodal trajectory prediction, which achieves improvements beyond scalability, accuracy, and efficiency compared with traditional intention-bridging-based approaches. It decouples the long-term intentions of agents into decision and goal intentions, then marginalizes the joint distribution of trajectories in decoupled intention spaces. The internal sharing mechanism of I2T allows the long-term prediction of trajectories to benefit from the training and guidance of the intention estimator.
- Prioriformer running efficiently in I2T enables adequate learning of trajectory representations from the more moderate distributions generated by PN. Features from various receptive fields are mapped to different representation subspaces by the multi-scale fusion encoder, efficiently extracting significant features and preserving global dependencies during agent interactions. The non-autoregressive decoder is able to generate multimodal trajectories quickly and accurately.
- I2T provides an efficient solution for the long-term prediction of trajectories at both the architecture and model levels. Experiments on the NGSIM, HighD, and Argoverse real-world datasets show that our method achieves state-of-the-art performance in both intention estimation and trajectory prediction. We additionally perform granularity, parameter sensitivity, and ablation experiments to explore the impact of different experimental configurations on prediction performance.

The following is a brief overview of the paper structure. Section II outlines a literature review of related work on vehicle motion prediction and Transformer model improvement. Our I2T architecture with its key components is described in Section III. Section IV presents the design of the backbone network (Prioriformer) for I2T. Section V analyzes the conducted experiments and results. Section VI concludes and outlooks this paper. Finally, we provide appendices containing virtual presentations of results and explanations of model designs.

II. Related Work

Motion prediction of traffic agents has been extensively studied in recent years. Traffic trajectories contain unique and distinct spatial-temporal features, thus efficient optimization of the model is critical to improving the long-term prediction accuracy of trajectories. Consequently, we focus on motion modeling approaches, priori trajectory models, and Transformer improvements, which motivate us to propose I2T.

A. Motion Prediction

Numerous studies have been conducted in predicting the motions of traffic agents [11], [12]. EA-Net [13] models spatial dependencies between vehicles and implicit graph-structured interactions through convolutional social pooling containing squeeze-and-extraction mechanism and graph attention networks. MID [14] represents the trajectory prediction task as a reverse process of motion uncertainty diffusion learned through a parameterized Markov chain. Several recent works reduce the prediction complexity by dividing the trajectory prediction task into multiple stages and then learning both stages together in an end-to-end manner. The prediction of road information, such as the end coordinates of target agent [15], [16], [17], [18], [19] and the lane to follow [20] and [21], is applied in the first stage to narrow down the trajectory inference. Su et al. [22] utilized the first stage to extract the features of traffic agents, followed by the second stage to divide interaction regions and predict trajectories. Sun et al. [23] divided the interacting agents into influencers and responders and then generated trajectories for both through marginal and conditional predictions, respectively. Inspired by these literature, we adopt the multi-stage architecture to model trajectory prediction.

The lane-change intentions are gradually applied to trajectory prediction as direct and effective information [24], [25], [26], [27]. Multimodal trajectory prediction methods based on intention bridging achieve inherent trajectory sets by embedding a combination of features for specific intentions. BRAM-ED [28] first identifies driving behavior through the Bi-GLSTM network, and then the designed behavioral attention mechanism guides the decoder to generate future trajectories based on changes in driving behavior. STDAN [5] and TrajTFNet [29] adaptively integrate temporal and social features in multimodal trajectory prediction through driving intention-specific feature fusion mechanism and intention-aware transformer decoder, respectively. The multimodal prediction accuracy, scalability, and efficiency of these methods are unsatisfactory due to limitations in the modeling mechanism.

B. Priori Trajectory Models

Priori trajectory models [30], [31], [32], [33] utilize the priori knowledge based on motion constraints to alleviate the difficulty and unreasonableness of trajectory prediction. NPSN [34] introduces a Quasi-Monte Carlo method and incorporates a learnable sampling network into the trajectory prediction network to ensure uniform coverage of the sampling space. Xu et al. [35] searched for similar scenarios in training data to predict the motion intentions of agents by simulating retrospective memory mechanisms in neuropsychology. Nevertheless, the modeling of these priori trajectory models is complex, and the performance is limited by the spatial resolution. Perceiving the limitations of these works, instead of referencing the concept of the priori trajectory model, we consider simpler, effective, and flexible concepts of the priori mean, variance, and trajectory as well as implement them into the designed Prioriformer model.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

C. Transformer Model Improvements

Some methods [36], [37] improve the prediction performance of models by enhancing the high-level semantic information of the decoder. The intention-aware decoder query generation module of iNATran [6] discovers critical features highly relevant to prediction from intermediate representations of intention inference to generate decoder queries in one step, overcoming the limitations of Transformer regarding inference speed and multimodal prediction. VNAGT [38] generates multiple plausible predictions with low latency leveraging a non-autoregressive query generation block that incorporates trajectory embeddings, spatial-temporal features, and positional embeddings. The self-attention distilling operation proposed by Informer [39] is able to reduce the temporal and spatial complexities significantly. However, as the model deepens vertically, the problem of over-distillation occurs, leading to the bottleneck of prediction accuracy.

III. A Holistic Architecture of I2T

A. I2T Overview

The methods and critical components of I2T are summarized in Fig. 1. It includes an intention estimator for estimating lane-change intentions, a motion predictor for inferring future trajectories, and a sample selector for ranking and filtering the optimal set of trajectories. The intention estimator and the motion predictor share the same Prioriformer network structure but different prediction heads. Prioriformer consists of personalized normalization of data, a multi-scale fusion encoder, and a non-autoregressive decoder, described in detail in Section IV. The motion predictor takes the intention probability distribution as scenario-enhancing information and merges the shared weights from the intention estimator. The sample selector estimates probabilities of candidate trajectories under each lane change intention through the maximum entropy model, then computes the joint prediction scores by the product of the intention and trajectory probabilities to filter out the most probable K trajectories. More detailed descriptions of each component are presented as follows.

B. Problem Formulation

We describe the trajectory prediction problem as predicting future trajectories of target vehicle based on the observed trajectories of land vehicles. Consider a training set with m vehicles, and each vehicle trajectory contains n state variables. At the moment k, trajectories of vehicle i is defined as follows:

$$x_{i,j}^{k} \in \mathcal{X}, \ \text{where} \ \ i \in \{1, \dots, m\}, \ \ j \in \{1, \dots, n\}, \quad (1)$$

where $\mathcal{X} \in \mathbb{R}^{m \times n}$ indicates the set of all trajectories in a dataset. Since the metadata provided by the different datasets varies, we record the specific state variables involved in Table I. Taking the observed trajectory of T_{obs} frames before k (containing the k-th frame) and the future trajectory of T_{pred} frames after k, the trajectory of vehicle i at any time k ($k \geq T_{obs}$) is defined as:

$$\boldsymbol{X}_{i,j} = \left\{ x_{i,j}^{k-T_{obs}+1}, \dots, x_{i,j}^{k}, x_{i,j}^{k+1}, \dots, x_{i,j}^{k+T_{pred}} \right\}, \quad (2)$$

TABLE I

State Variables Adopted on Different Datasets

Dataset	State variables
HighD	Coordinates, accelerations, and velocities in horizontal and vertical directions, lane ID, steering angle
NGSIM	Coordinates in horizontal and vertical directions, acceleration, velocity, lane ID, steering angle
Argoverse	Coordinates in horizontal and vertical directions, steering angle

[1] The steering angle is calculated manually according to metadata.

where the observed trajectory of vehicle i is denoted as:

$$\boldsymbol{H}_{i,j} = \left\{ x_{i,j}^{k-T_{obs}+1}, x_{i,j}^{k-T_{obs}+2}, \dots, x_{i,j}^{k} \right\}, \quad (3)$$

Our goal is to predict the future location of target vehicle i:

$$\boldsymbol{F}_{i,j'} = \left\{ x_{i,j'}^{k+1}, x_{i,j'}^{k+2}, \dots, x_{i,j'}^{k+T_{pred}} \right\}, \quad (4)$$

where $j' \in \{1, 2\}$ denotes the horizontal and vertical coordinates in the n state variables. As a model input, the motion features of target vehicle i are denoted as:

$$\mathcal{C}_{\text{target}} = \left\{ \boldsymbol{H}_{i,1}, \boldsymbol{H}_{i,2}, \dots, \boldsymbol{H}_{i,n} \right\}. \quad (5)$$

With the target vehicle as the center, prioritizing front, lane change direction, back, and non-lane change direction, relative displacements of three vehicles are extracted as the motion features of surrounding vehicles, which is expressed as:

$$\mathcal{C}_{\text{neighbour}} = \left\{ \Delta \boldsymbol{H}_{r,1}, \Delta \boldsymbol{H}_{r,2} | r \in \{1, 2, 3\} \right\}. \quad (6)$$

Predicting both the horizontal and vertical coordinates of the future trajectory for target vehicle i is our ultimate task:

$$\mathfrak{F} = \{ \boldsymbol{F}_{i,1}, \boldsymbol{F}_{i,2} \}. \quad (7)$$

We implement motion prediction with multimodal distribution through intention decoupling, which can be decomposed into two parts. The lane-changing decision intention $I \in \{I_s, I_l, I_r\}$ of the driver is first estimated. The fine-grained goal intention required to perform the vehicle movement, i.e., the vehicle motion endpoints $\vartheta \in \{\vartheta_1, \vartheta_2, \dots, \vartheta_M\}$, is further modeled in each intention space. The motion predictor specifies the corresponding trajectory for gradient updating based on the results produced by intention decoupling. Thus $\mathfrak{F}$ is a joint distribution over multiple modes, which we can marginalize to decompose the probability distribution according to the intention decoupling process:

$$P(\mathfrak{F} \mid \mathcal{C}_I) = \sum_{I,\vartheta} P(\mathfrak{F} \mid \mathcal{C}_I, I, \vartheta) P(\vartheta \mid \mathcal{C}_I, I) P(I \mid \mathcal{C}_I), \quad (8)$$

where $\mathcal{C}_I = \left\{ \mathcal{C}_{\text{target},} \mathcal{C}_{\text{neighbour}} \right\}$ is input features from the scene context. Intention decoupling reduces the complexity of multimodal joint distributions into tractable unimodal distributions by progressively capturing and controlling uncertainty.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

C. Intention Estimator

The pipeline of the intention estimator is depicted in the upper part of Fig. 1. The intention estimator learns lane-change intentions through the motion states of vehicles and interaction relations. The input to the intention estimator is $\mathcal{C}_I$. These features are normalized, encoded, and decoded in Prioriformer, and then the intention estimation head outputs the intention probability distribution:

$$P(I) = \text{Softmax}\left(f\left([\boldsymbol{X}_c]\right)\right), \tag{9}$$

where $\boldsymbol{X}_\text{c}$ denotes the inputs of the intention estimation head. $f(\cdot)$ represents the fully connected layer that aligns the predicted outputs with the target intentions. Softmax$(\cdot)$ is the normalization function that maps the outputs to probabilities. $P(I)$ denotes possibilities of lane keeping I_s, left lane change I_l, and right lane change I_r.

Finally, the loss function for training the intention estimator is defined as follows:

$$\mathcal{L}_I = \mathcal{L}_{ce}(P(I), E(I)), \tag{10}$$

where $\mathcal{L}_{ce}$ is the cross-entropy loss function. It avoids the slow gradient update and gradient diffusion in the back-propagation process. $E(I)$ represents the one-hot encoding for the ground truth intention of target vehicle.

D. Motion Predictor

The motion predictor is guided by the intention estimator for trajectory prediction. Input features for the motion predictor are derived from the set $\mathcal{C}_T = \left\{\mathcal{C}_{\text{target},}\mathcal{C}_{\text{neighbour},}\, P(I)\right\}$. The lane-change intention probabilities $P(I)$ and the shared weights containing a large number of intention representations from the intention estimator are merged into the motion predictor from both inputs and network facets, respectively. The motion prediction head holds three identically structured intention branches, each containing M module lists for trajectory regression. Prioriformer generates $3M$ candidate trajectories in each forward propagation and updates one of the module lists independently during each backpropagation as specified by the lane-changing intention and goal intention conditions to ensure multimodal properties. Inspired by the convolutional networks [40], [41], the structure of each module list is designed as follows:

$$\boldsymbol{X}_{out} = f\left(\text{MaxPool}\left(\sigma\left(\text{BN}\left(\text{Conv1d}\left([\boldsymbol{X}_d]\right)\right)\right)\right)\right), \tag{11}$$

where $\boldsymbol{X}_d$ denotes outputs of the Prioriformer decoder. Conv1d$(\cdot)$ performs an 1-D convolutional filters on $\boldsymbol{X}_d$. Its outputs are normalized by a BN$(\cdot)$ layer [42], then an ELU activation function σ [43] is applied to enhance the nonlinear expression ability. Finally, we downsample $\boldsymbol{X}_d$ through a max-pooling layer to alleviate the position sensitivity of the convolutional layer and align the predicted outputs with the target trajectories through a fully connected layer.

The model only learns the deviation of the ground truth trajectories from the inverse normalized trajectories, which is much smaller than the deviation from $\boldsymbol{X}_{out}$. The inverse personalized normalization (the inverse operation of Eq. (22)) performed on $\boldsymbol{X}_{out}$ is expressed as:

$$\hat{\boldsymbol{F}}_{i,j'} = \boldsymbol{X}_{out} \times \sqrt{\text{var}(\hat{\boldsymbol{\xi}}_{i,j'}) + \varepsilon} + \text{mean}(\hat{\boldsymbol{\xi}}_{i,j'}), \tag{12}$$

where the priori mean and the priori variance of the horizontal or vertical coordinate channel are denoted by mean$(\hat{\boldsymbol{\xi}}_{i,j'})$ and var$(\hat{\boldsymbol{\xi}}_{i,j'})$, which are calculated by Eq. (20). The value of ε is the same as in Eq. (22).

Finally, the loss function for training the motion predictor is given by:

$$\mathcal{L}_T = \mathcal{L}_{mse}(\hat{\mathfrak{F}}^{*}, \mathfrak{F}), \tag{13}$$

where $\mathcal{L}_{mse}$ denotes the mean square error loss function. $\hat{\mathfrak{F}}^{*} = \{\hat{\boldsymbol{F}}_{i,1}, \hat{\boldsymbol{F}}_{i,2}\}$ denotes the trajectory that satisfies the condition of the goal intention, i.e., the minimum final displacement error, among the M predicted trajectories generated under the current intention space.

Improving the performance of trajectory prediction is the ultimate motivation of I2T. We ensure the dominance of the motion predictor in the gradient backpropagation by including the hyperparameter λ, thus the final loss function of I2T is formulated as:

$$\mathcal{L} = \lambda\mathcal{L}_I + \mathcal{L}_T. \tag{14}$$

Algorithm 1 Model Training Process With I2T

```
Input: Scene context $\mathcal{C}_I = \left\{\mathcal{C}_{\text{target},}\mathcal{C}_{\text{neighbour}}\right\}$;
       Future intentions $I$ and future trajectories $\mathfrak{F}$;
       Intention estimator $I_\Theta$ and trajectory predictor $T_\Theta$
Output: Network weights of $I_\Theta$ and $T_\Theta$
Set model hyperparameters and initialize network weights
Execute PN operation on $\mathcal{C}_I$ by Eq. (22) yields $\bar{\mathcal{C}}_I$
while No early stopping do
    repeat
        $c, \iota, \tau \leftarrow MiniBatch(\bar{\mathcal{C}}_I, I, \mathfrak{F})$
        $e \leftarrow Embedding(c)$
        $\hat{\iota} \leftarrow I_\Theta(e)$
        $Compute$ intention prediction accuracy $Acc$
        if $Acc$ exceeds the set threshold $\rho$ then
            $\tilde{e} \leftarrow Embedding(c, \hat{\iota})$
        else
            $\tilde{e} \leftarrow Embedding(c, \iota)$
        end if
        $\hat{\tau} \leftarrow T_\Theta(\tilde{e}, I_\Theta)$
        $Compute$ $\mathcal{L} = \lambda\mathcal{L}_I(\hat{\iota}, \iota) + \mathcal{L}_T(\hat{\tau}, \tau)$ by Eq. (14)
        $I_\Theta, T_\Theta \leftarrow Backward\,(\mathcal{L})$
    until Epoch end or model convergence
end while
```

Algorithm 1 summarizes the joint training process of the intention estimator and trajectory predictor of I2T, where I_Θ and T_Θ represent the body network of the intention estimator and trajectory predictor, respectively. c denotes the observed data of one batch fed into the model each time. ι and τ denote the ground truth for learning intention estimation and trajectory prediction, respectively. e is the representation of c after embedding through the 1D convolutional filter.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

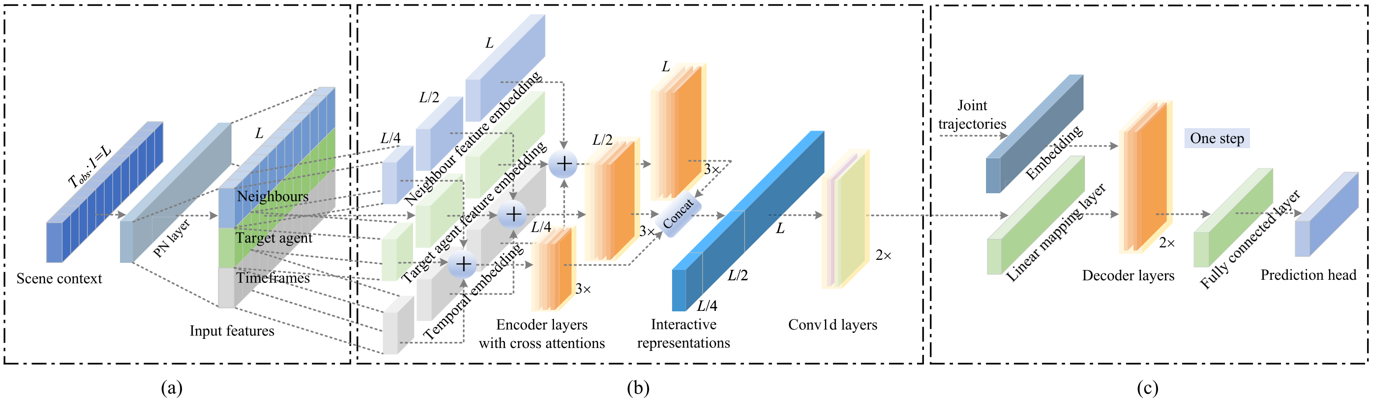


Fig. 2. Overview of Prioriformer. (a) Personalized normalization. PN layer extracts timeframe information and performs personalized normalization on observed trajectories. (b) Multi-scale fusion encoder. Trajectories after PN are sliced into L, $L/2$, and $L/4$ long segments from the tail-end to the front-end direction, followed by encoding the embedding of these segments by three encoder blocks of various scales. The concatenated outputs of the encoder blocks are finally fused by two convolutional layers. (c) Non-autoregressive decoder with frame compensation mechanism. It contains four various functional layers and completes the prediction task accurately and simultaneously by configuring the prediction head with the corresponding function.

$\tilde{e}$ is additionally embedded with lane-changing intentions compared to e. The $\hat{\iota}$ and $\hat{\tau}$ denote the predicted intentions and trajectories. During the initial stage of training intention estimator, the motion predictor may generate trajectories with large errors under the inaccurate guidance of the intention estimator. Therefore, in the early training of I2T, the motion predictor is trained using the teacher-forcing mechanism [44]. Specifically, when the intention estimation accuracy does not exceed the threshold ρ, the motion predictor is trained utilizing the ground truth intention ι with an error intention ratio of $1-\rho$ to ensure its better fault tolerance. Conversely, estimated intentions are taken to train the motion predictor. ρ is set to 0.9 in this paper.

E. Sample Selector

I2T is not limited to any trajectory classification or scoring methods [15], [23], [45]. In this paper, we employ the maximum entropy model to estimate the probability of M candidate trajectories under each lane change intention:

$$\mathfrak{L}_{\text{conf}} = \mathfrak{L}_{\text{ce}}(\varphi_I(\hat{\mathfrak{F}}|\mathcal{C}), \phi(\mathfrak{F})), \tag{15}$$

where $\phi(\mathfrak{F}) = \text{softmax}(-\max(\|\hat{\mathfrak{F}} - \mathfrak{F}\|_2^2))$ denotes the hand-crafted ground truth label, which is defined with the assistance of the maximum value of the distance between the predicted trajectories and the ground truth trajectory. The trajectory probability computational function $\varphi_I \in \{\varphi_{I_s}, \varphi_{I_l}, \varphi_{I_r}\}$ is modeled by two fully connected layers with skip-connect structures:

$$\varphi_I(\hat{\mathfrak{F}}|\mathcal{C}) = \text{softmax}(f\lfloor\Upsilon(f\lfloor\mathcal{C}, \hat{\mathfrak{F}}\rfloor), \hat{\mathfrak{F}}\rfloor), \tag{16}$$

where Υ represents the LeakyReLU activation function (negative slope = 0.1), and $\lfloor\cdot\rfloor$ denotes the concatenate operation. Trajectory probabilities are computed independently for each lane-change intention space. We freeze the weights of the Prioriformer to avoid gradient updates caused by $\hat{\mathfrak{F}}$ that interfere with the weights of the trained intention estimator and trajectory predictor.

Finally, the joint scores of trajectories are the product of the intention probabilities and the trajectory probabilities in the corresponding intention space:

$$\hat{\mathfrak{F}}_{\text{score}} = \text{softmax}(P(I) \cdot \varphi_I(\hat{\mathfrak{F}}|\mathcal{C})). \tag{17}$$

We select the optimal K trajectories from the predicted samples as the final results.

IV. Proposed Model

We formally introduce Prioriformer model and its core elements. A general overview of the model is shown in Fig. 2, and more details are presented in the following subsections.

A. Personalized Normalization

Neural network training requires constantly adjusting the network parameters to shift the internal covariates to accommodate changes in the input data distribution. Internal covariate shift causes the training process to easily fall into the gradient saturation region, reducing convergence speed and accuracy. Data distribution between trajectories varies significantly due to differences in driving styles and traffic scenarios. As shown in Appendix B in Fig. 8 (a & c), the traditional approach normalizes all trajectories using the uniform mean and variance, resulting in a highly variable distribution of inputs, which is prone to the internal covariate shift problem. Assuming that we can obtain the mean and variance of each trajectory, the distributions of all trajectories after PN converges to the standard Gaussian distribution, alleviating the internal covariate shift problem. Considering vehicles in the same scenario with similar driving intentions or historical trajectories, the future trajectories of the vehicles partially conform to the statistical law of historical data. Accordingly, we design an offline fitting algorithm to calculate the priori mean, variance, and trajectory of the target vehicle, followed by further proposing the personalized normalization method based on them. The priori mean, variance, and trajectory of the various intention spaces for the vehicle entail solving in the same way. To avoid redundancy, we do not embody the specific intention in the following equations of this subsection.

The displacement increases linearly in the driving direction, and the data size is also distinct for various lanes in the

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

vertical direction. To accurately describe vehicle behaviors (lane-change or lane-keeping) on diverse road sections, dimensionless processing is performed for individual channels of each trajectory:

$$\bar{\boldsymbol{X}}_{i,j} = \boldsymbol{X}_{i,j} - x_{i,j}^{k}. \tag{18}$$

$\bar{\boldsymbol{X}} = \{\bar{\boldsymbol{X}}_{1,1}, \bar{\boldsymbol{X}}_{1,2}, \ldots, \bar{\boldsymbol{X}}_{m,n}\}$ is denoted as a priori trajectory pool, i.e., all trajectories of the training set processed by Eq. (18). $\bar{\boldsymbol{H}}$ denotes the observed part of $\bar{\boldsymbol{X}}$, $\bar{\boldsymbol{H}}\neg_{i,j}$ denotes the subset of $\bar{\boldsymbol{H}}$ that does not include the target vehicle i, e.g., $\bar{\boldsymbol{H}}\neg_{1,1} = \{\bar{\boldsymbol{H}}_{2,1}, \bar{\boldsymbol{H}}_{3,1}, \ldots, \bar{\boldsymbol{H}}_{m,1}\}$, and the same is true for $\bar{\boldsymbol{X}}\neg_{i,j}$. The distance between observed trajectories is measured by:

$$\boldsymbol{\zeta}_{i,j}^{1:S} = \underset{\bar{X}\neg_{i,j}}{\arg\min} \sum_{j=1}^{n} \| \bar{\boldsymbol{H}}_{i,j} - \bar{\boldsymbol{H}}\neg_{i,j} \|, \tag{19}$$

where $\boldsymbol{\zeta}_{i,j}^{1:S}$ is the set of S trajectories with certain intention in the priori trajectory pool closest to the observed trajectory of target vehicle i.

We add weighting factors to $\boldsymbol{\zeta}_{i,j}^{1:S}$, which prevents the extremity of the priori trajectory. The mean and variance of the trajectory satisfy homogeneous additivity, and the variance is translation invariant. Therefore, the priori mean $\mathrm{mean}(\widehat{\boldsymbol{\zeta}}_{i,j})$, priori variance $\mathrm{var}(\widehat{\boldsymbol{\zeta}}_{i,j})$, and priori trajectory $\widehat{\boldsymbol{\zeta}}_{i,j}$ for the target vehicle i are formulated as:

$$\begin{aligned}
\mathrm{mean}(\widehat{\boldsymbol{\zeta}}_{i,j}) &= \sum_{s=1}^{S} \alpha_s \cdot \mathrm{mean}(\boldsymbol{\zeta}_{i,j}^{s}) + x_{i,j}^{k}, \\
\mathrm{var}(\widehat{\boldsymbol{\zeta}}_{i,j}) &= \sum_{s=1}^{S} \beta_s \cdot \mathrm{var}(\boldsymbol{\zeta}_{i,j}^{s}), \\
\widehat{\boldsymbol{\zeta}}_{i,j} &= \sum_{s=1}^{S} \gamma_s \cdot \boldsymbol{\zeta}_{i,j}^{s} + x_{i,j}^{k}, \\
s.t.\ \sum_{s=1}^{S} \alpha_s &= \sum_{s=1}^{S} \beta_s = \sum_{s=1}^{S} \gamma_s = 1,
\end{aligned} \tag{20}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_S]$ denotes the weighting factors of the S trajectories for fitting the priori mean, and the same is true for $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. The identical weighting factors are applied to the various channels of each vehicle. The optimal solutions of $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are determined by:

$$\begin{aligned}
\boldsymbol{\alpha}^{*} &= \underset{\boldsymbol{\alpha}}{\mathrm{argmin}} \sum_{i=1}^{m} \sum_{j=1}^{n} \left\| \mathrm{mean}(\boldsymbol{X}_{i,j}) - \mathrm{mean}(\widehat{\boldsymbol{\zeta}}_{i,j}) \right\|_1, \\
\boldsymbol{\beta}^{*} &= \underset{\boldsymbol{\beta}}{\mathrm{argmin}} \sum_{i=1}^{m} \sum_{j=1}^{n} \left\| \mathrm{var}(\boldsymbol{X}_{i,j}) - \mathrm{var}(\widehat{\boldsymbol{\zeta}}_{i,j}) \right\|_1, \\
\boldsymbol{\gamma}^{*} &= \underset{\boldsymbol{\gamma}}{\mathrm{argmin}} \sum_{i=1}^{m} \sum_{j=1}^{n} \left\| \boldsymbol{X}_{i,j} - \widehat{\boldsymbol{\zeta}}_{i,j} \right\|_1.
\end{aligned} \tag{21}$$

Fig. 5(a) shows parameter sensitivity experiments determining S and iterative accuracy. Additional details are provided in Appendix B: Algorithm 2 describes the general procedure for iterating the optimal solutions of Eq. (21) utilizing a heuristic search approach, and Fig. 8 illustrates the advantages of our personalized normalization approach over the conventional global normalization (GN) at macro and micro levels.

Finally, the personalized normalization approach is formulated as:

$$\hat{\boldsymbol{H}}_{i,j} = \frac{\boldsymbol{H}_{i,j} - \mathrm{mean}(\hat{\boldsymbol{\zeta}}_{i,j})}{\sqrt{\mathrm{var}(\hat{\boldsymbol{\zeta}}_{i,j}) + \varepsilon}}, \tag{22}$$

where ε is a compensation value to prevent division by zero errors and $\hat{\boldsymbol{H}}_{i,j}$ is the normalization result. Unlike global normalization, the individual channels of each observed trajectory are normalized, respectively.

B. Multi-Scale Fusion Encoder

As shown in Fig. 2(a), scene context information including target agent states $\mathcal{C}_{\text{target}}$ and the relative spatial relationships $\mathcal{C}_{\text{neighbour}}$ between vehicles are used as inputs, which are kept aligned on the timestamp. The structure of the multi-scale fusion encoder is shown in Fig. 2(b). We slice normalized trajectories after PN into segments of L, $L/2$, and $L/4$ lengths from the tail-end to the front-end direction. Agent states and neighbour features at each observation timestamp are represented as multi-dimensional vectors through the 1-D convolutional filter (kernel width = 3, stride = 1). The temporal embedding consists of the positional embedding [10] based on sine-cosine functions without learnable parameters and the local timestamp embedding [39] with learnable parameters, allowing subsequent self-attentive similarity computations to implicitly capture temporal interactions and dependencies between local contexts. Following this, representation subspaces are provided by three scales of encoder blocks, by which features are extracted from various sensory fields: global features, mid-term features, and the most critical late features are extracted by encoder blocks of L, $L/2$, and $L/4$ scales, respectively.

The structure of each encoder is that of a standard Transformer encoder replacing the self-attention mechanism with the cross-attention mechanism. Complex social interactions and potential spatial-temporal dependencies between vehicles are efficiently modeled by attention mechanisms in different scale encoders to avoid the cumbersome deployment of dividing traffic scenes into occupancy grids [3], [4]. Specifically, we perform the cross attention computation $\mathcal{A}(\bar{\mathbf{Q}}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}(\bar{\mathbf{Q}}\mathbf{K}^{\top}/\sqrt{\mathbf{d}})\mathbf{V}$ on the target agent and neighbour features at all timestamps, where $\bar{\mathbf{Q}}$ and $\mathbf{K}$, $\mathbf{V}$ are the parameter matrices of the target agent and the neighbour feature embeddings after head-splitting mapping, respectively. $\mathbf{d}$ is the dimension of $\mathbf{K}$. The spatial-temporal dynamic interaction dependencies between surrounding vehicles and the target agent are naturally represented by the attention weights. Finally, the concatenated outputs of the encoder blocks are fused by two convolutional layers. The multi-scale fusion encoder avoids the problem that secondary features of Informer are not involved in the model expression, which allows models to learn significant and essential information and long-term dependencies from various representation subspaces.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

C. Non-Autoregressive Decoder

The Vanilla Transformer adopts an autoregressive decoder, which can be formulated as:

$$\hat{\boldsymbol{F}}^{t} = \mathbb{D}\left(\hat{\boldsymbol{F}}^{1:(t-1)}, \boldsymbol{Z}^{1:(T_{pred}-t+1)}, \mathbb{E}(\boldsymbol{\mathcal{C}})\right), \tag{23}$$

where $\mathbb{E}(\cdot)$ denotes the encoding operation with data processing, and $\mathbb{D}(\cdot)$ denotes the decoding operation. $\boldsymbol{\mathcal{C}}$ indicates the model inputs, and $\boldsymbol{Z}^{1:(T_{pred}-t+1)}$ (generally filled with zeros) represents the tokens for remaining inputs to align the decoder outputs. The predicted value at the moment t depends on the previously generated prediction $\hat{\boldsymbol{F}}^{1:(t-1)}$. Due to cumulative errors during the dynamic decoding process, inference speed and prediction accuracy of the autoregressive decoder decrease as the prediction horizon lengthens. As a solution, we can take the copy $\tilde{\boldsymbol{\mathcal{C}}}$ derived from the observed trajectory $\boldsymbol{\mathcal{C}}$ and complementary tokens $\boldsymbol{Z}^{1:(T_{pred}-l)}$ as decoder inputs. Therefore, all target sequences are predicted concurrently:

$$\hat{\boldsymbol{F}}^{1:T_{pred}} = \mathbb{D}(\tilde{\boldsymbol{\mathcal{C}}}^{(k-l+1):k}, \boldsymbol{Z}^{1:(T_{pred}-l)}, \mathbb{E}(\boldsymbol{\mathcal{C}})), \tag{24}$$

where l is the input length of the observed trajectory. Nevertheless, large numbers of tokens are introduced in this design, which causes the decoder inputs to be of poor quality and negatively impacts the prediction performance. Therefore, we replace the tokens in Eq. (24) with the priori trajectory of future moments, generating the target trajectories simultaneously and accurately as follows:

$$\hat{\boldsymbol{F}}^{1:T_{pred}} = \mathbb{D}(\tilde{\boldsymbol{\mathcal{C}}}^{(k+p-T_{pred}+1):k}, \hat{\boldsymbol{\xi}}^{(k+1):(k+p)}, \mathbb{E}(\boldsymbol{\mathcal{C}}_{T})), \tag{25}$$

where p is the number of compensated frames, and $\hat{\boldsymbol{\xi}}^{(k+1):(k+p)}$ is the priori trajectory of target vehicle. For the intention estimation task, the decoder generates the distribution of lane-change intentions:

$$P(I) = \mathbb{D}(\tilde{\boldsymbol{\mathcal{C}}}^{(k+p-T_{pred}+1):k}, \hat{\boldsymbol{\xi}}^{(k+1):(k+p)}, \mathbb{E}(\boldsymbol{\mathcal{C}}_{I})). \tag{26}$$

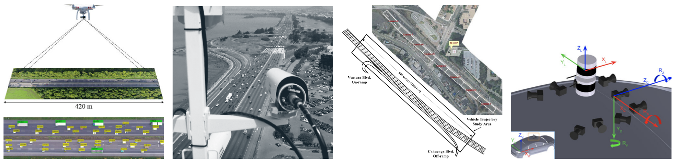
Fig. 2(c) illustrates the workflow of the non-autoregressive decoder in Prioriformer. The outputs of the multi-scale fusion encoder are linearly mapped to the equal dimension as the decoder inputs and then fed to the decoder together with the joint trajectories of the observed and priori trajectories. Finally, the prediction task is completed following a fully connected layer and a prediction head for the corresponding function.

V. Experimental Evaluations

A. Datasets

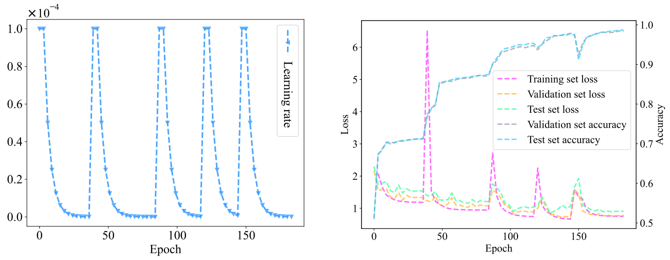
We conduct extensive experiments from various perspectives on three real-world datasets. Please refer to Fig. 3 for an overview and the details described as follows.

1) HighD [46]: A high-quality real-world vehicle trajectory dataset recorded at six different locations on German freeways, sampled at 25Hz. Fig. 3(a) illustrates its collection method. To further explore the long-term prediction performance at different granularity levels, we construct three datasets: HighD-A at 5Hz, HighD-B at 12Hz, and HighD-C at 25Hz.



(a) (b) (c) (d)

Fig. 3. Description of experimental datasets. (a) The HighD dataset captures vehicles from a bird's eye view over a road section approximately 420 m long with less than 10 cm of positioning error. (b) NGSIM I-80. (c) NGSIM US101. The NGSIM dataset is collected by multiple simultaneous digital cameras in high-rise buildings. (d) The Argoverse dataset is collected by a fleet of autonomous vehicles with vision and radar sensors.



(a) (b)

Fig. 4. Learning rate changing curve and training metric curves of I2T. (a) Learning rate. Each round sets the initial learning rate to $1e^{-4}$. The learning rate decays dynamically within the round, where the weight decay factor is 0.5. Until the loss no longer decreases, the learning rate is readjusted to $1e^{-4}$ and then the next round is started. (b) Training metrics. Corresponding to the learning rate settings in (a), the curves of the relevant metrics for the training set, validation set, and test set of the HighD dataset.

2) NGISM [47], [48]: The NGSIM dataset records vehicle trajectories on I-80 (Fig. 3(b)) and US101 (Fig. 3(c)) freeways in America, with a sampling frequency of 10Hz. Similarly to the HighD dataset, we construct the NGSIM-A dataset at 5Hz and the NGSIM-B dataset at 10Hz.

3) Argoverse [49]: The Argoverse motion forecasting dataset records realistic driving scenarios on urban streets in Miami and Pittsburgh, sampled at 10Hz. Fig. 3(d) depicts its collection approach. The Argoverse is only publicly available for training and validation sets. Following the existing work [50], we conduct tests utilizing the Argoverse validation set.

B. Experimental Details

1) Hyper-Parameter Tuning: We conduct a grid search for the parameter λ in Eq. (14), which is finally set to 0.1. The batch size is set to 32. In the encoder and decoder of Prioriformer, the number of attention heads is 8, the vector representation dimension is 512, and the feedforward network dimension is 2048. We train the model with the Adam optimizer, adjusting the learning rate with a combination of cyclic and dynamic decay, and set up an early stopping of training. The learning rate curve and the partial training process are visualized in Fig. 4. Parameters for the comparison methods are set as recommended.

2) Metrics: Different datasets or tasks have different recommended evaluation metrics as follows.

a) Metrics on intention estimation: The accuracy and recall metrics are employed to evaluate the performance of the I2T intention estimator.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

b) Metrics on NGSIM and HighD: The Root mean square error (RMSE) is employed to evaluate the trajectory prediction performance conducted on the NGSIM and HighD datasets, which is calculated by:

$$RMSE = \left(\frac{1}{2T_{\text{pred}}} \sum_{t=1}^{T_{\text{pred}}} \left\| \hat{\mathfrak{F}}^{t} - \mathfrak{F}^{t} \right\|^{2} \right)^{\frac{1}{2}}. \tag{27}$$

For granularity and ablation experiments, we add the average displacement error (ADE) and final displacement error (FDE) metrics to observe the effects of various components more accurately. The ADE and FDE metrics are calculated by:

$$ADE = \frac{1}{T_{\text{pred}}} \sum_{t=1}^{T_{\text{pred}}} \| \hat{\mathfrak{F}}^{t} - \mathfrak{F}^{t} \|,$$
$$FDE = \| \hat{\mathfrak{F}}^{T_{\text{pred}}} - \mathfrak{F}^{T_{\text{pred}}} \| . \tag{28}$$

c) Metrics on argoverse: Three metrics recommended by the Argoverse dataset are utilized, where minADE and minFDE are the minimum ADE and FDE of the multiple predictions, respectively. Miss rate (MR) is calculated as the percentage of the optimal predictions where the FDE is within 2.0 meters.

C. Baselines

We select highly convincing models in vehicle intention estimation or trajectory prediction tasks to validate the performance of I2T. We divide these methods according to the task or the dataset used for evaluations.

1) Methods Evaluated on Intention Estimation: 1) BTCL [51]: a self-supervised bidirectional trajectory comparison learning model. *2) CS-LSTM* [3]: a model based on LSTM encoder-decoder with convolutional social pooling. *3) AE-SVM* [52]: an interaction-aware prediction model consisting of a LSTM autoencoder and a SVM classifier. *4) STDAN* [5]: a spatial-temporal dynamic attention model for multimodal trajectory prediction of vehicles.

2) Methods Evaluated on NGSIM and HighD: 1) CS-LSTM [3] and *2) STDAN* [5] methods are described above. *3) S-LSTM* [1]: a LSTM-based model with the social pooling layer. *4) NLS-LSTM (N-LSTM)* [53]: a model combines local blocks and non-local multi-head attention mechanisms to capture impacts and interactions between vehicles. *5) MHA-LSTM (A-LSTM)* [54]: an approach to model higher-order interactions using a multi-head attention mechanism. *6) S-GAN* [2]: an approach employing a LSTM encoder-decoder with a pooling module as the trajectory generator. *7) PiP* [4]: a model based on a planning-informed trajectory prediction algorithm. *8) Informer* [39]: a long-term prediction model based on the ProbSparse self-attention mechanism. *9) TrajTFNet* [29]: an intention-aware Transformer-based multimodal trajectory prediction model for vehicles with adaptive social and temporal learning. *10) iNATran* [6]: a multimodal trajectory prediction model based on intention-aware non-autoregressive Transformer with social, temporal, and cross-attention learning.

3) Methods Evaluated on Argoverse: 1) CS-LSTM [3], *2) S-GAN* [2], and *3) BTCL* [51] methods are described above. *4) S-CVAE* [50]: a method for mitigating the social posterior collapse in interaction modeling variational autoencoder. *5) DESIRE* [55]: a method for combining static and dynamic scene contexts with a deep stochastic inverse optimal control-RNN encoder-decoder framework. *6) LaneAttention* [56]: a method for trajectory prediction employing a goal-oriented lane-attention module. *7) MID* [14]: a multimodal trajectory prediction method based on probabilistic diffusion models.

TABLE II
RESULTS OF DIFFERENT METHODS FOR VEHICLE DECISION INTENTION ESTIMATION ON HIGHD-A AND NGSIM-A DATASETS

Dataset	Method	Accuracy	Keep Recall	Left Recall	Right Recall
HighD-A	BTCL	0.935	0.976	0.761	**0.993**
	CS-LSTM	0.965	0.990	0.851	0.879
	AE-SVM	0.961	0.985	0.855	0.871
	Ours	**0.987**	**0.992**	**0.960**	0.985
NGSIM-A	BTCL	0.975	0.986	0.975	0.933
	CS-LSTM	0.973	0.991	0.520	0.454
	AE-SVM	0.976	**0.996**	0.475	0.353
	STDAN	0.979	0.995	0.559	0.414
	Ours	**0.986**	0.986	**0.985**	**0.935**
Argoverse	BTCL	0.931	0.934	0.933	0.894
	CS-LSTM	0.923	0.954	0.777	0.742
	AE-SVM	0.924	0.983	0.642	0.600
	Ours	**0.994**	**0.993**	**0.994**	**0.997**

D. Results and Analysis

We comprehensively analyze the performance of I2T in terms of intention estimation, unimodal and multimodal predictions, and inference costs.

1) Intention Estimation: The I2T intention estimator is evaluated on the NGSIM-A, HighD-A, and Argoverse datasets. Achieving optimization on all indicators is difficult due to the mutual influence of different evaluation metrics. Table II indicates that: (1) The I2T intention estimator performs best on the accuracy metric across all datasets, especially on the Argoverse dataset (6.3% improvement in accuracy). Prioriformer can efficiently perform the task of intention prediction, and the results can be accepted by the motion predictor. (2) CS-LSTM, STDAN, and AE-SVM are inferior to ours on Left Recall and Right Recall metrics on NGSIM, reflecting the superiority of Prioriformer in capturing the long-term dependence of intentions. Another reason is that they ignore the sample imbalance problem between lane change and lane keeping. The gradient is updated in the direction that makes the overall gain greater, thus the lane-keeping accuracy is the highest. We attenuate the sample imbalance by data augmentation of the training set. (3) The I2T intention estimator outperforms BTCL based on self-supervised learning. The internal sharing mechanism of I2T allows the intention estimator to benefit from the training of the motion predictor, which is more effective than the bidirectional trajectory comparison learning.

2) Trajectory Prediction ($K = 1$): Table III indicates the unimodal prediction results or multimodal prediction results based on probability scores for I2T and comparison methods on the NGSIM-A and HighD-A datasets (all baseline methods follow this granularity level). Table III ends with statistics on the average results of all methods on both datasets. We can observe that: (1) I2T significantly improves the performance of trajectory prediction in all datasets (the most shown in bold).

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

TABLE III

Results of Different Methods for Vehicle Trajectory Prediction on HighD-A and NGSIM-A Datasets

Method		S-LSTM	N-LSTM	A-LSTM	S-GAN	PiP	Informer	CS-LSTM		STDAN		TrajTFNet		iNATran		I2T(Ours)	
Mode		Unim			M = 1	M = 2	M = 2	M = 2	Unim	M = 3	Unim	M = 3	Unim	M = 3	Unim	M = 2	Unim
HighD-A	1s	0.19	0.20	0.19	0.30	0.17	0.42	0.23	0.22	-	0.19	0.05	**0.04**	**0.04**	**0.04**	0.16	0.07
	2s	0.57	0.57	0.55	0.78	0.52	0.56	0.65	0.61	-	0.27	0.08	**0.05**	**0.05**	**0.05**	0.41	0.11
	3s	1.18	1.14	1.10	1.46	1.05	0.84	1.29	1.24	-	0.48	0.24	0.21	0.21	0.21	0.53	**0.20**
	4s	2.00	1.90	1.84	2.34	1.76	1.12	2.18	2.10	-	0.91	0.61	0.58	0.54	0.54	0.73	**0.44**
	5s	3.02	2.91	2.78	3.41	2.63	1.42	3.37	3.27	-	1.66	1.21	1.15	1.11	1.10	0.91	**0.87**
NGSIM-A	1s	0.60	0.56	0.56	0.57	0.55	0.81	0.62	0.61	0.42	0.42	**0.34**	-	0.41	0.39	0.69	0.40
	2s	1.28	1.22	1.22	1.32	1.18	1.18	1.29	1.27	1.03	1.01	1.03	-	1.00	0.96	0.99	**0.91**
	3s	2.09	2.02	2.01	2.22	1.94	1.91	2.13	2.09	1.76	1.69	1.83	-	1.70	1.61	1.76	**1.51**
	4s	3.10	3.03	3.00	3.26	2.88	2.81	3.20	3.10	2.70	2.56	2.82	-	2.57	2.42	2.53	**2.21**
	5s	4.37	4.30	4.25	4.40	4.04	3.71	4.52	4.37	3.89	3.67	4.03	-	3.66	3.43	3.31	**3.09**
Mean	1s	0.40	0.38	0.38	0.44	0.36	0.62	0.43	0.42	-	0.31	**0.20**	-	0.23	0.22	0.43	0.24
	2s	0.93	0.90	0.89	1.05	0.85	0.87	0.97	0.94	-	0.64	0.56	-	0.53	**0.51**	0.70	**0.51**
	3s	1.64	1.58	1.56	1.84	1.50	1.38	1.71	1.67	-	1.09	1.04	-	0.96	0.91	1.15	**0.86**
	4s	2.55	2.47	2.42	2.80	2.32	1.97	2.69	2.60	-	1.74	1.72	-	1.56	1.48	1.63	**1.33**
	5s	3.70	3.61	3.52	3.91	3.34	2.57	3.95	3.82	-	2.67	2.62	-	2.39	2.27	2.11	**1.98**

[1] Unim mode denotes unimodal prediction. M denotes the number of candidate trajectories for each lane-change intention of supporting multimodal prediction methods (denoted by the (M) suffix below).

[2] The multimodal prediction method sets $K = 1$, i.e., only the predicted trajectory with the highest probability is selected. Thus, the comparison between unimodal prediction and probability-based multimodal prediction is fair.

TABLE IV

Comparison and Scalability Experiments of Multimodal Trajectory Prediction on NGSIM-A Dataset

Method		CS-LSTM	I2T	STDAN	I2T	I2T		
Mode		M = 2		M = 3		M = 4	M = 5	M = 6
Horizons	1s	0.51	0.37	0.42	0.36	0.35	0.34	0.34
	2s	0.97	0.77	0.82	0.72	0.69	0.66	0.65
	3s	1.44	1.09	1.16	0.93	0.84	0.77	0.73
	4s	2.04	1.47	1.60	1.15	0.97	0.85	0.76
	5s	2.91	2.12	2.35	1.70	1.46	1.31	1.21

[1] I2T does not employ the frame-compensation mechanism when $M > 2$.

[2] We select all predicted trajectories, i.e., $K = 3M$.

In particular, our model has an overwhelming advantage in the 3s-5s prediction phase, which demonstrates the success of our proposed I2T for long-term prediction of trajectories. (2) The prediction error of I2T shows a smooth and slow increase with the extension of the prediction horizon. Compared to the unimodal prediction family of LSTM-like models represented by A-LSTM, the RMSE metrics of I2T decreased by 44.9% (3s), 45.0% (4s), and 43.8% (5s), respectively. It can be attributed to the non-autoregressive decoder of Prioriformer that avoids cumulative errors of autoregression and the personalized normalization that makes model gradients adequately updated. (3) I2T is more competitive in terms of comprehensive metrics than iNATran, which possesses the optimal results among the comparison methods, with RMSE metrics decreasing by 5.5% (3s), 10.1% (4s), and 12.8% (5s), respectively. However, the short-term prediction performance of I2T is lower than that of TrajTFNet and iNATran methods. The internal sharing mechanism of I2T entails learning long-term cues regarding intentions, and the gradient will flow in a direction more conducive to overall gradient gains, i.e., long-term predictions are better optimized. Considering that the achievements of I2T in long-term prediction of trajectories are more prominent than the slight cost of short-term prediction and that the short-term prediction accuracy is sufficient, we do not perform additional optimizations. (4) The prediction performance of I2T(M) is worse than I2T. Some methods [3], [6], [57] observe a similar trend that the lower the number of choices K the worse the results when more modalities are trained, which reflects the gap between choices based on probabilistic predictions and actual results. I2T(M) still outperforms most comparison models in the long-term phase. In addition, the multimodal modeling mechanisms of CS-LSTM(M), TrajTFNet(M), and iNATran(M) are essentially traversals of unimodal results with fixed intentions, making the probability-based multimodal results of the three closer to unimodal results than I2T. However, this modeling mechanism is slow to improve performance when making multimodal predictions with large K.

3) Scalability of Trajectory Prediction ($K > 1$): Selecting only one representative of multiple modes does not adequately reflect the multimodal performance of the model, and more downstream applications are beginning to utilize a small set of diverse predictions to improve vehicle driving safety. Thus, we conduct multimodal prediction experiments under the $K > 1$ setting on the NGSIM-A dataset. Table IV indicates that I2T(M) breaks through the accuracy and scalability bottleneck of the traditional intention-bridging-based multimodal modeling mechanism. Specifically, (1) I2T(M) significantly outperforms CS-LSTM(M) and STDAN(M). Intention-bridging-based multimodal prediction methods pursue multimodality by traversing the unimodal results on a specific combination of intentions. This fixed pattern wastes arithmetic power on an unreasonable intention space, making improving performance difficult when performing multimodal predictions with large K. (2) I2T(M) is superior to CS-LSTM(M) and STDAN(M) in terms of prediction

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

TABLE V
RESULTS OF DIFFERENT METHODS FOR MULTIMODAL TRAJECTORY PREDICTION ON THE ARGOVERSE DATASET

Method	minADE↓	minFDE↓	MR↓
CS-LSTM	1.51	2.92	0.52
S-GAN	1.49	2.91	0.44
S-CVAE	1.15	1.98	-
DESIRE	0.92	1.77	0.18
LaneAttention	1.05	2.06	-
BTCL	1.12	2.13	0.34
MID	1.14	2.08	0.38
I2T(M) (Ours)	**0.76**	**1.20**	**0.12**

[1] The symbol ”-” indicates that no metric data are recorded in the original paper, and no source code is provided.
[2] All methods set K = 6. I2T(M) sets M = 4 and does not employ the frame-compensation mechanism.

scalability. The CS-LSTM(M) method extends up to $M = 2$. Although STDAN(M) extends the mode to $M = 3$ by adding overtaking maneuvers based on CS-LSTM(M), this is close to the extension limit. The intention decoupling mechanism of I2T(M) could extend the prediction to more modes through fine-grained intentions, and its prediction error decreases significantly as M increases.

4) Multimodal Trajectory Prediction ($K = 6$): Since the Argoverse dataset involves complex map structures, we adopt the scene context modeling approach for the Argoverse dataset from previous work [58]. The map structures and the dynamic behaviors of road participants are first vectorized, and then the convolutional neural network is employed to aggregate the node features of individual polylines globally. The resulting scene context representations are fed to I2T(M) to learn the interactions between the structured maps and vehicles. Table II proves that I2T(M) has sufficient accuracy for intention estimation, hence we set $M = 4$ to improve the accuracy of trajectory prediction under a specific intention. As can be seen from Table V, I2T(M) exhibits the optimal performance on all metrics. Compared with the best results among the comparison methods, the metrics of our method dropped by 17.4% (minADE), 32.2% (minFDE), and 33.3% (MR), respectively. S-GAN, S-CVAE, DESIRE, and MID are probabilistic generation-based methods, which are inherently stochastic and inevitably encounter the mode blur problem. Therefore, these methods are inferior to our deterministic modeling approach. Our intention decoupling strategy models both coarse-grained decision and fine-grained goal intentions, outperforming CS-LSTM, LaneAttention, and BTCL methods that only consider decision intentions. We supplement the qualitative analysis of I2T(M) on the Argoverse dataset in Appendix A.

5) Inference Costs: We report the inference costs regarding Giga multiply-accumulate operations (GMACs) and time for some open-source methods and I2T(M) on the NGSIM-A dataset in Table VI. We calculate the GMACs with the official Python package thop. The batch size is set to 32, i.e., the future motions of 32 agents are predicted each time, meeting the requirements of autonomous vehicles. In general, the inference costs of our model is within an acceptable level. The distance traveled by vehicles with a speed of 120 km/h at 13.96ms forward is only 0.47 meters, much less than the 166.67 meters traveled in the 5s prediction period, which satisfies the safe driving of autonomous vehicles. In addition, taking the most common automotive millimeter-wave radar product as an example, its refresh cycle is 50ms-100ms. We are able to accomplish multiple inferences during its perception of the surrounding environment. The additional plan-coupling module of PiP(M) could introduce greater time consumption, which also depends on the degree of algorithmic optimization. The number of trajectories predicted by CS-LSTM(M) and STDAN(M) corresponds to the times of forward propagations of the model. From unimodal to multimodal prediction, the time costs of CS-LSTM(M) and STDAN(M) increase by 208.3% and 141.0%, respectively. The intention decoupling mechanism enables I2T(M) to generate multiple trajectories under each lane change intention at once for time-efficient optimization. Our method only increases the computational effort by 6.9% and the time cost by 21.4% after tuning the parameter M from 2 to 6. The time consumption of I2T(M) may be less than that of STDAN(M) if more modal predictions are made.

TABLE VI
COMPUTATIONAL AND TIME COSTS OF DIFFERENT MODELS FOR INFERENCE ON THE NGSIM-A DATASET

Methods	S-GAN	PiP	CS-LSTM		STDAN		I2T (Ours)		
Mode	M = 1	M = 2	Unim	M = 2	Unim	M = 3	Unim	M = 2	M = 6
GMACs	0.04	2.02	0.22	0.74	0.69	0.93	17.38	18.01	19.27
Time(ms)	11.77	42.29	1.20	3.69	3.89	9.38	11.00	13.96	16.95

TABLE VII
PERFORMANCE OF I2T WITH HIGHD DATASET AT VARIOUS LEVELS OF GRANULARITY

Dataset	Metric	1s	2s	3s	4s	5s
HighD-A (5Hz)	RMSE	0.16	0.41	0.53	0.73	0.91
	ADE	0.14	0.38	0.49	0.70	0.85
	FDE	0.28	0.69	0.96	1.51	1.90
HighD-B (12Hz)	RMSE	0.16	0.42	**0.46**	**0.63**	**0.86**
	ADE	0.13	0.37	**0.40**	**0.57**	0.81
	FDE	0.28	0.77	**0.86**	**1.23**	1.78
HighD-C (25Hz)	RMSE	**0.16**	**0.38**	0.59	0.70	0.87
	ADE	**0.13**	**0.35**	0.60	0.65	**0.79**
	FDE	**0.28**	**0.63**	1.11	1.53	**1.70**

E. Granularity Analysis

We conduct additional experiments on the HighD and NGSIM datasets. For more concise descriptions, granularity, parameter sensitivity, and ablation experiments all employ I2T to represent I2T(M), where M = 2. This subsection investigates the prediction performance of I2T at different granularities.

Based on equivalent experimental settings and model configurations, Table VII documents the performance of our method at different granularity levels on the HighD dataset. We can observe that: (1) I2T performs better on HighD-B and HighD-C than HighD-A. It is not affected by the increasing points, which demonstrates the long-term prediction capability of I2T. (2) When predicting short-term trajectories (1s-2s), I2T performs best on the HighD-C dataset, followed by HighD-B

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

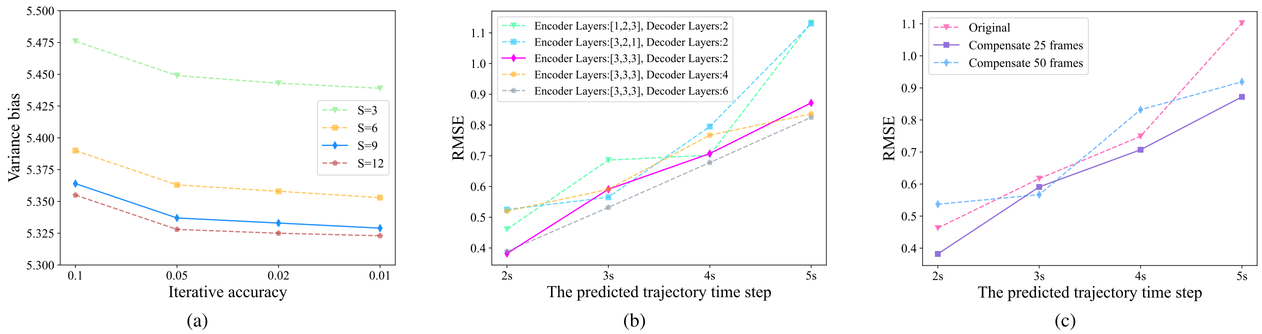


Fig. 5. Parameter sensitivity of three essential parts in Prioriformer. (a) The selection of fitting parameter S and iterative accuracy. (b) Combinations of encoder-decoder layers in Prioriformer. (c) The number of compensation frames for decoder inputs.

TABLE VIII
PERFORMANCE OF I2T WITH NGSIM DATASET AT VARIOUS LEVELS OF GRANULARITY

Dataset	Metric	1s	2s	3s	4s	5s
NGSIM-A (5Hz)	RMSE	0.69	0.99	1.76	2.53	3.31
	ADE	0.76	**1.08**	**1.83**	2.79	3.70
	FDE	1.14	1.95	3.53	4.23	**5.10**
NGSIM-B (10Hz)	RMSE	**0.68**	**0.97**	**1.66**	**2.46**	**3.23**
	ADE	**0.75**	1.09	1.86	**2.64**	**3.64**
	FDE	**1.11**	**1.32**	**2.44**	**3.38**	5.25

and HighD-A. Fine-granularity samples tend to contain richer detailed information, which is beneficial for short-term predictions. (3) When performing long-term (3s-5s) prediction, the model shows better performance on HighD-B than HighD-C. It is attributed to the sparse information at medium-granularity level that allows the model to capture global dependencies better. In contrast, redundant detailed information at fine-granularity level is detrimental to long-term prediction performance. (4) Trajectories with coarse-granularity level reduce model prediction accuracy, which is reflected in the consistently poor prediction performance on HighD-A. Coarse-granularity trajectories lose critical detail information, resulting in models with only limited inference performance from overly sparse and limited data information.

Table VIII records the performance of our method when the NGSIM dataset is at various levels of granularity. I2T also achieves better performance on the fine-granularity dataset NGSIM-B. The fourth conclusion of the granularity experiment of HighD dataset is reproduced here.

Our granularity experiments conclude that utilizing appropriate granularity levels at different prediction steps is beneficial to improving the performance of trajectory prediction. The conclusion also demonstrates that the long-used 5-Hz granularity level in the field of trajectory prediction may lead to sub-optimal prediction performance. To more accurately exploit the long-term prediction performance of trajectories, we conduct the following experiments of parameter sensitivity and ablation on the HighD-C dataset, which contains the highest number of points and shows better performance.

F. Parameter Sensitivity

We conduct a parameter sensitivity analysis of I2T by means of a trajectory prediction task under the HighD-C dataset.

1) Parameters of the Priori Variance: Fig. 5(a) depicts the relationship among the fitting parameter S, the iterative accuracy, and the variance bias (the bias between the ground truth variance and the priori variance). With increasing S and iterative accuracy, the variance bias decreases. To balance fitting accuracy and computational complexity, we set S to 9 and the iterative accuracy to 0.02 (shown by the solid blue line). Parameters of the priori mean and the priori trajectory also exhibit nearly identical relationships, which are not repeated.

2) Encoder-Decoder Layer Combinations: Fig. 5(b) depicts the experimental effects of different encoder-decoder layer combinations. The list of encoder layers sequentially represents the number of layers in encoder blocks of L,$L/2$, and $L/4$ size. When the late features of the observed trajectory are emphasized (The list of encoder layers is [1, 2, 3]), it gives a repeated short-term pattern to the model. The performance drops sharply when performing long-term prediction of trajectories. When the global features of the observed trajectory are emphasized (The list of encoder layers is [3, 2, 1]), the role of the essential features contained in the late trajectories is weakened, so its performance is unsatisfactory.

The decoder performance increases as layers are added, but the limited accuracy improvement is not enough to compensate for the cost of training and decoding time consumption. Finally, we set the multi-scale fusion encoder layers to the combination of [3, 3, 3] and the decoder layers to 2 (shown by the solid fuchsia line). In this setting, RMSE scores of the model are at a low level with moderate and smooth growth.

3) Compensated Frames for Decoder Input: Fig. 5(c) depicts the effect of the number of the priori trajectory compensation frames on the model prediction performance. The model performs best when decoder inputs are compensated for 25 frames, followed by 50 frames. As the prediction horizon increases, errors between the priori trajectory and the ground truth trajectory increase, and some extreme trajectories become increasingly inestimable. Thus more compensation frames are not better. To avoid causing instability of the decoder, we choose the priori trajectory within the first second (25 frames) of future moments to feed into the decoder (shown by the solid purple line).

G. Ablation Study

We subtract different components in turn to fully demonstrate their impact. Table IX summarizes the best results in each experiment after three training opportunities.

1) Ablation of PN and Decoder Frame Compensation Mechanism: We can observe that the prediction accuracy decreases

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

TABLE IX
ABLATION EXPERIMENTS OF THREE MAJOR SECTIONS IN I2T

Ablation items	Metric	1s	2s	3s	4s	5s
All features	RMSE	0.16	0.38	0.59	0.70	0.87
	ADE	0.13	0.35	0.60	0.65	0.79
	FDE	0.28	0.63	1.11	1.53	1.70
I2T†	RMSE	0.36	0.55	0.76	1.03	1.11
	ADE	0.40	0.60	0.82	1.10	1.04
	FDE	0.82	0.85	1.38	**2.10**	2.06
I2T‡	RMSE	0.45	0.87	0.96	1.08	1.18
	ADE	**0.50**	**0.97**	1.03	1.19	1.24
	FDE	**0.83**	**1.72**	1.67	1.79	2.58
I2T§	RMSE	0.46	0.92	1.23	1.24	1.53
	ADE	0.48	0.92	1.37	1.40	1.67
	FDE	0.83	1.64	2.32	2.36	3.25

[1] I2T† means that the backbone network of I2T uses Prior-iformer that removes PN and decoder frame compensation operations.
[2] I2T‡ further removes the multi-scale fusion encoder from the I2T† backbone network.
[3] I2T§ further removes the intention factors and the shared weights of the intention estimator from the I2T‡.
[4] The negative optimization term is highlighted in bold, i.e., the pre-ablation model performs worse than the post-ablation model.

significantly in all prediction ranges. The moderate data distribution after PN improves the training stability and allows models to learn trajectory representations better. Conversely, the data distribution of I2T† is relatively extreme, making it hard to reasonably learn latent trajectory representations. Gradient updates tend to be dominated by large gradients, leading models to fall into the local optimum.

2) Ablation of the Multi-Scale Fusion Encoder: I2T‡ removes the multi-scale fusion encoder from the backbone of I2T†. On average, RMSE metric increases by 16.0%, ADE metric by 19.6%, and FDE metric by 16.1%. The multi-scale fusion encoder extracts features from multiple receptive fields and maps them to various scales of representation subspaces. The model is able to learn richer information from representation subspaces of various trajectory phases, extracting essential features and retaining global dependencies, which brings an overall improvement in trajectory prediction performance.

3) Ablation of the Internal Sharing Mechanism: I2T§ removes intention factors and shared weights from I2T‡. Compared to I2T‡, I2T§ has an average increase of 15.4% in RMSE metric, 15.7% in ADE metric, and 17.4% in FDE metric. This occurrence exemplifies the significant role of intention factors and shared representations of the intention estimator in guiding the model performance improvement of the long-term prediction. The multi-task training framework boosts the quality of trajectory representations learning by leveraging additional beneficial information between two tasks.

However, the ADE and FDE metrics of I2T‡ are slightly higher than those of I2T§ in the short-term prediction phase (1s-2s), indicating that the single-task model I2T§ is sufficient for simple short-term prediction tasks to converge better. Although the internal sharing mechanism brings richer intention information, it is not as easy to train as the single-task model.

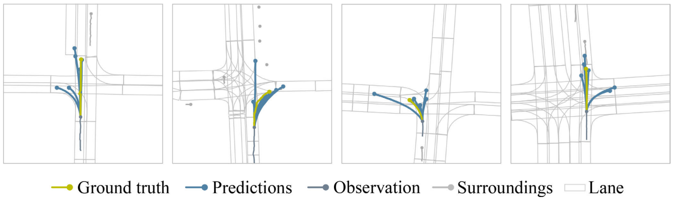


Fig. 6. Qualitative illustrations of multimodal trajectories generated by I2T(M) involving different modes on the Argoverse dataset.

VI. CONCLUSION

In this paper, an effective solution I2T has been developed for the long-term prediction of vehicle trajectories at both the architecture and model levels. I2T decouples trajectory prediction into intention estimation and motion prediction. The high-level intention is first estimated, followed by the low-level trajectory prediction utilizing the intention probability distribution and shared intention representations. Experiments with real-world trajectory datasets have demonstrated the potential of I2T for intention estimation and trajectory prediction.

Autonomous vehicles can leverage I2T-generated predictions to understand surrounding scenarios and make safer long-term decisions. I2T can be applied to advanced driver assistance systems as well to boost the active safety of conventional vehicles. In future work, we will model urban scenarios with complicated human-vehicle interactions and further extend the model to pedestrian trajectory prediction. Moreover, we plan to utilize trajectory predictions of surrounding vehicles to develop safe and efficient decisions for autonomous vehicles, addressing the problem of conservative driving.

APPENDIX A
QUALITATIVE RESULTS

Fig. 6 illustrates the effectiveness of our method in predicting different driving modes in complex urban traffic scenarios. I2T(M) reasonably generates multimodal trajectories in different decision intention spaces. The generated trajectories are meaningful ones that the agent may take under the corresponding driving scenario, and one or several of the predictions can better cover the ground truth trajectory.

Fig. 7 shows the visualized images of randomly selected trajectory predictions following our method and Informer, respectively. For each traffic condition, we cover the effect of trajectory prediction under lane keeping, left lane change, and right lane change intentions. We can observe that our method outperforms Informer in most scenarios. Specifically, the predictions of Informer exist for unreasonable lane change trajectories and large overshoot distances, while our method predicts better throughout the lane change process.

APPENDIX B
THE SOLUTION DETAILS AND INTERPRETABILITY OF THE PERSONALIZED NORMALIZATION APPROACH

Heuristic strategy guides the searching towards the most promising direction, which reduces the complexity of solving. Algorithm 2 describes the general procedure of iteratively

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

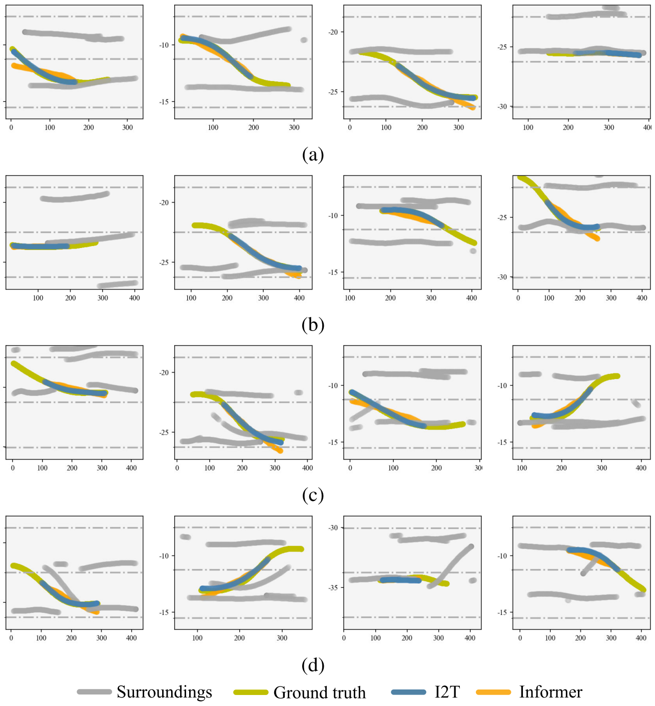


Fig. 7. Qualitative comparisons between our method and Informer under four traffic conditions on the HighD-C dataset. (a) Congestion-free traffic condition. (b) Slight congestion traffic condition. (c) Congestion traffic condition. (d) Multi-vehicle lane change traffic condition.

solving for the optimal solution of Eq. (21) with a heuristic search approach. With the basis of the rough results already obtained, the search range is gradually reduced next time, resulting in the precision of $\boldsymbol{\alpha}*$ being achieved to a desirable degree within a shorter period.

Algorithm 2 Iterate Best Fitting Parameters($\boldsymbol{\alpha}$ for example)

```
Input: The training set $\boldsymbol{X}$ and trajectory set $\boldsymbol{\zeta}_{i,j}^{1:S}$ to be fitted
Output: The optimal solution $\boldsymbol{\alpha}*$ of $\boldsymbol{\alpha}$ in Eq. (21)
Initialize $\delta, \theta, \boldsymbol{\alpha}, \boldsymbol{\alpha}*, u, u*$
while $\boldsymbol{\alpha}*$ does not meet the required precision do
    for All values of $\boldsymbol{\alpha}$ in $\theta$ range with $\delta$ precision do
        Fitting mean($\boldsymbol{\zeta}_{i,j}^{1:S}$) yields mean($\widehat{\boldsymbol{\zeta}}_{i,j}$) by Eq. (20)
        $u = \sum_{i=1}^{m} \sum_{j=1}^{n} \left\| \mathrm{mean}(\boldsymbol{X}_{i,j}) - \mathrm{mean}(\widehat{\boldsymbol{\zeta}}_{i,j}) \right\|_1$
        if $u < u*$ then
            $u* = u$
            $\boldsymbol{\alpha}* = \boldsymbol{\alpha}$
        end if
    end for
    Update $\delta$ to a more minor precision
    $\theta = [\boldsymbol{\alpha} * -3\delta, \boldsymbol{\alpha} * +3\delta]$
end while
return $\boldsymbol{\alpha}*$
```

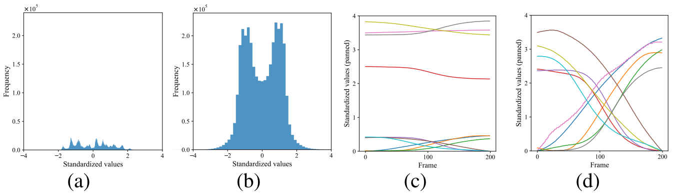


Fig. 8. Macro and micro perspectives of the distinction between PN and GN. (a) GN macro statistics. (b) PN macro statistics. (c) GN micro examples. (d) PN micro examples.

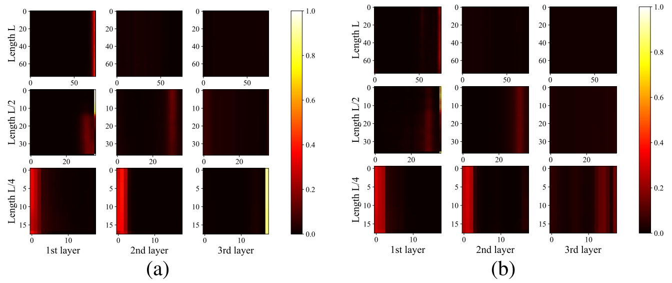


Fig. 9. Attention matrices of the multi-scale fusion encoder. (a) Attention matrices of intention estimator. (b) Attention matrices of motion predictor.

Fig. 8 illustrates the advantages of our personalized normalization approach over the conventional global normalization at macro and micro levels. Fig. 8(a) statistics the distribution of the globally normalized data. The normalized data are scattered over a broad range (Fig. 8(a) with an identical number of points as Fig. 8(b), but Fig. 8(a) with a lower frequency, proves this point). The extreme data distribution and sparse co-features render it difficult for the model to learn common knowledge from these data, leading to hard model training. Fig. 8(b) demonstrates the data distribution after personalized normalization. Data points are uniformly clustered around zero, and the number of similar points is approximately ten times greater than in Fig. 8(a). The moderate distribution facilitates the model convergence toward the global optimum.

Fig. 8(c) shows ten trajectories randomly sampled after global normalization. Each trajectory is mapped to an inactive interval, and the data points within the trajectory are flat over time and poorly featured. Fig. 8(d) depicts the results of personalized normalization of the identical ten trajectories (corresponding by color) as in Fig. 8(c). All trajectories are mapped to approximate intervals, and the differences in data points within each trajectory are extended to a reasonable level. It delivers a kind of gradient fairness to each trajectory, ensuring that each can be well trained.

Appendix C
Interpretability of Multi-Scale Fusion Encoder

Fig. 9 shows attention matrices of encoder blocks for the intention estimator and the motion predictor, respectively. The sizes of the encoder blocks employed are L, $L/2$, and $L/4$ from top to bottom (corresponding to the input length). Each encoder block contains three encoder layers, yielding a total of nine attention matrices. Most attention matrices contain relatively large values towards the end when the input length is L or $L/2$. Therefore, the closer to the end of the observed trajectory, the greater the data contribution. The multi-scale fusion encoder extracts richer features of diverse importance through various receptive fields.

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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 961–971.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

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

[3] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1468–1476.

[4] H. Song, W. Ding, Y. Chen, S. Shen, M. Y. Wang, and Q. Chen, "PiP: Planning-informed trajectory prediction for autonomous driving," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 598–614.

[5] X. Chen, H. Zhang, F. Zhao, Y. Hu, C. Tan, and J. Yang, "Intention-aware vehicle trajectory prediction based on spatial–temporal dynamic attention network for Internet of Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 19471–19483, Oct. 2022.

[6] X. Chen, H. Zhang, F. Zhao, Y. Cai, H. Wang, and Q. Ye, "Vehicle trajectory prediction based on intention-aware non-autoregressive transformer with multi-attention learning for Internet of Vehicles," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022.

[7] P. H. Martins, Z. Marinho, and A. Martins, "∞-former: Infinite memory transformer," in *Proc. Annu. Meeting Assoc. Comput. Linguist. (ACL)*, 2022, pp. 5468–5485.

[8] A. Dosovitskiy, L. Beyer, and A. Kolesnikov, "An image is worth 16 × 16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 1–12.

[9] Z. Liu et al., "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10012–10022.

[10] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inform. Process. Syst. (NIPS)*, 2017, pp. 5998–6008.

[11] W. Zhu, Y. Liu, M. Zhang, and Y. Yi, "Reciprocal consistency prediction network for multi-step human trajectory prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 6, pp. 6042–6052, 2023.

[12] Y. Zhang, W. Guo, J. Su, P. Lv, and M. Xu, "BIP-Tree: Tree variant with behavioral intention perception for heterogeneous trajectory prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9584–9598, 2023.

[13] Y. Cai et al., "Environment-attention network for vehicle trajectory prediction," *IEEE Trans. Veh. Technol.*, vol. 70, no. 11, pp. 11216–11227, Nov. 2021.

[14] T. Gu et al., "Stochastic trajectory prediction via motion indeterminacy diffusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17113–17122.

[15] J. Gu, C. Sun, and H. Zhao, "DenseTNT: End-to-end trajectory prediction from dense goal sets," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 15303–15312.

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

[17] K. Mangalam et al., "It is not the journey but the destination: Endpoint conditioned trajectory prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 759–776.

[18] L. Fang, Q. Jiang, J. Shi, and B. Zhou, "TPNet: Trajectory proposal network for motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6796–6805.

[19] H. Zhao et al., "TNT: Target-driven trajectory prediction," in *Proc. Conf. Robot Learn. (CoRL)*, vol. 155, Aug. 2020, pp. 895–904.

[20] B. Kim, S. H. Park, S. Lee, E. Khoshimjonov, and D. Kum, "LaPred: Lane-aware prediction of multi-modal future trajectories of dynamic agents," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14636–14645.

[21] J. Wang, T. Ye, Z. Gu, and J. Chen, "LTP: Lane-based trajectory prediction for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17134–17142.

[22] Z. Su, C. Wang, D. Bradley, C. Vallespi-Gonzalez, C. Wellington, and N. Djuric, "Convolutions for spatial interaction modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 6573–6582.

[23] Q. Sun, X. Huang, J. Gu, B. C. Williams, and H. Zhao, "M2i: From factored marginal trajectory prediction to interactive prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 6543–6552.

[24] Z. Yan, K. Yang, Z. Wang, B. Yang, T. Kaizuka, and K. Nakano, "Intention-based lane changing and lane keeping haptic guidance steering system," *IEEE Trans. Intell. Vehicles*, vol. 6, no. 4, pp. 622–633, Dec. 2021.

[25] T. Rehder, A. Koenig, M. Goehl, L. Louis, and D. Schramm, "Lane change intention awareness for assisted and automated driving on highways," *IEEE Trans. Intell. Vehicles*, vol. 4, no. 2, pp. 265–276, Jun. 2019.

[26] A. Zyner, S. Worrall, and E. Nebot, "Naturalistic driver intention and path prediction using recurrent neural networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1584–1594, Apr. 2020.

[27] S. Cong, W. Wang, J. Liang, L. Chen, and Y. Cai, "An automatic vehicle avoidance control model for dangerous lane-changing behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 8477–8487, Jul. 2022.

[28] L. Li, W. Zhao, C. Wang, Q. Chen, and F. Chen, "BRAM-ED: Vehicle trajectory prediction considering the change of driving behavior," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 6, pp. 5690–5700, Dec. 2022.

[29] Y. Hu and X. Chen, "Intention-aware transformer with adaptive social and temporal learning for vehicle trajectory prediction," in *Proc. 26th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2022, pp. 3721–3727.

[30] T. Zhang, W. Song, M. Fu, Y. Yang, and M. Wang, "Vehicle motion prediction at intersections based on the turning intention and prior trajectories model," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 10, pp. 1657–1666, Oct. 2021.

[31] T. Phan-Minh, E. C. Grigore, F. A. Boulton, O. Beijbom, and E. M. Wolff, "CoverNet: Multimodal behavior prediction using trajectory sets," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 14062–14071.

[32] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov, "Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction," in *Proc. Conf. Robot Learn., (PMLR)*, Oct. 2019, pp. 86–99.

[33] K. Guo, W. Liu, and J. Pan, "End-to-end trajectory distribution prediction based on occupancy grid maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2232–2241.

[34] I. Bae, J.-H. Park, and H.-G. Jeon, "Non-probability sampling network for stochastic human trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 6477–6487.

[35] C. Xu, W. Mao, W. Zhang, and S. Chen, "Remember intentions: Retrospective-memory-based trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 6488–6497.

[36] Y. Wang, F. Tian, D. He, T. Qin, C. Zhai, and T.-Y. Liu, "Non-autoregressive machine translation with auxiliary regularization," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 5377–5384.

[37] J. Guo, X. Tan, D. He, T. Qin, L. Xu, and T.-Y. Liu, "Non-autoregressive neural machine translation with enhanced decoder input," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2019, pp. 3723–3730.

[38] X. Chen, H. Zhang, Y. Hu, J. Liang, and H. Wang, "VNAGT: Variational non-autoregressive graph transformer network for multi-agent trajectory prediction," *IEEE Trans. Veh. Technol.*, vol. 72, no. 10, pp. 12540–12552, May 2023.

[39] H. Zhou et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 11106–11115.

[40] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5693–5703.

[41] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[42] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[43] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2016, pp. 1–12.

[44] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Comput.*, vol. 1, no. 2, pp. 270–280, Jun. 1989.

[45] X. Mo, Z. Huang, Y. Xing, and C. Lv, "Multi-agent trajectory prediction with heterogeneous edge-enhanced graph attention network," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 9554–9567, Jul. 2022.

[46] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highD dataset: A drone dataset of naturalistic vehicle trajectories on German highways for validation of highly automated driving systems," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2118–2125.

[47] J. Colyar and J. Halkias, "US highway I-80 dataset," Federal Highway Admin., Washington, DC, USA, Tech. Rep. FHWA-HRT-06-137, 2006, pp. 1–14.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.

[48] J. Colyar and J. Halkias, "US highway 101 dataset," Federal Highway Admin., Washington, DC, USA, Tech. Rep. FHWA-HRT-07-030, 2007, pp. 27–69.

[49] M. Chang et al., "Argoverse: 3D tracking and forecasting with rich maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 8748–8757.

[50] C. Tang, W. Zhan, and M. Tomizuka, "Exploring social posterior collapse in variational autoencoder for interaction modeling," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2021, pp. 8481–8494.

[51] Y. Zhou, H. Wang, N. Ning, Z. Wang, Y. Zhang, and F. Liu, "A bidirectional trajectory contrastive learning model for driving intention prediction," *Complex Intell. Syst.*, vol. 9, no. 4, pp. 4301–4315, Aug. 2023.

[52] T. Westny, E. Frisk, and B. Olofsson, "Vehicle behavior prediction and generalization using imbalanced learning techniques," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, Sep. 2021, pp. 2003–2010.

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

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

[55] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 336–345.

[56] J. Pan et al., "Lane-attention: Predicting vehicles' moving trajectories by learning their attention over lanes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 7949–7956.

[57] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "HOME: Heatmap output for future motion estimation," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, Sep. 2021, pp. 500–507.

[58] J. Gao et al., "VectorNet: Encoding HD maps and agent dynamics from vectorized representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Aug. 2020.

Yi Zhou (Member, IEEE) received the B.S. degree in electronic engineering from the First Aeronautic Institute of Air Force, China, in 2002, and the Ph.D. degree in control system and theory from Tongji University, China, in 2011. He is currently a Full Professor and the Deputy Dean with the School of Artificial Intelligence, Henan University, China. He is also the Director of the International Joint Research Laboratory for Cooperative Vehicular Networks, Henan, China. His research interests include vehicular cyber-physical systems and multi-agent collaboration.

Zhangyun Wang (Student Member, IEEE) is currently pursuing the M.S. degree with Henan University, Zhengzhou, China. His current research interests include graph neural networks, probabilistic diffusion modeling, and self-supervised learning for multimodal motion forecasting of intelligent agents.

Nianwen Ning received the Ph.D. degree in computer science and technology from Beijing University of Posts and Communications in 2021. He is currently a Lecturer with the School of Artificial Intelligence, Henan University. His research interests include graph data mining and intelligent transportation.

Zhanqi Jin is currently pursuing the M.S. degree with Henan University, Zhengzhou, China. His current research interests include UAV-assisted communications and intelligent reflective surfaces.

Ning Lu (Member, IEEE) received the B.Eng. and M.Eng. degrees in electrical engineering from Tongji University, Shanghai, China, in 2007 and 2010, respectively, and the Ph.D. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 2015. From 2015 to 2016, he was a Post-Doctoral Fellow with the Coordinated Science Laboratory, University of Illinois at Urbana–Champaign, Champaign, IL, USA. He was an Intern with the National Institute of Informatics, Tokyo, Japan, in the Summer of 2009. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Queen's University, Kingston, ON, Canada. Prior to joining Queen's University, he was an Assistant Professor with the Department of Computing Science, Thompson Rivers University, Kamloops, BC, Canada. He has published more than 60 papers in top IEEE journals and conferences, including IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, *ACM MobiHoc*, and *IEEE INFOCOM*. His current research interests include scheduling, distributed algorithms, and reinforcement learning for wireless communication networks.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include network resource management, wireless network security, the Internet of Things, 5G and beyond, and vehicular networks. He is a registered Professional Engineer of Ontario, Canada; an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, a Chinese Academy of Engineering Foreign Member, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and the IEEE Communications Society. He received Canadian Award for Telecommunications Research from the Canadian Society of Information Theory (CSIT) in 2021, the R. A. Fessenden Award from IEEE, Canada in 2019, the Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, the James Evans Avant Garde Award from the IEEE Vehicular Technology Society in 2018, the Joseph LoCicero Award in 2015, the Education Award from the IEEE Communications Society in 2017, and the Technical Recognition Award from the Wireless Communications Technical Committee in 2019 and the AHSN Technical Committee in 2013. He also received the Excellent Graduate Supervision Award from the University of Waterloo in 2006 and the Premier's Research Excellence Award (PREA) from the Province of Ontario, Canada, in 2003. He served as the Technical Program Committee Chair/Co-Chair for IEEE GLOBECOM 2016, IEEE Infocom 2014, IEEE VTC 2010 Fall, and IEEE GLOBECOM 2007, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. He was the Vice President for Technical and Educational Activities, the Vice President for Publications, a Member-at-Large on the Board of Governors, the Chair of the Distinguished Lecturer Selection Committee, and a member of the IEEE Fellow Selection Committee of the ComSoc. He is the President of the IEEE Communications Society. He served as the Editor-in-Chief for the IEEE INTERNET OF THINGS JOURNAL, *IEEE Network*, and *IET Communications*.

Authorized licensed use limited to: University of Waterloo. Downloaded on August 08,2024 at 19:27:56 UTC from IEEE Xplore. Restrictions apply.