Safety-Balanced Driving-Style Aware Trajectory Planning in Intersection Scenarios With Uncertain Environment

Xiao Wang, *Senior Member, IEEE*, Ke Tang, Xingyuan Dai, Jintao Xu, Jinhao Xi, Rui Ai, Yuxiao Wang, Weihao Gu, and Changyin Sun, *Senior Member, IEEE*

***Abstract*—This paper proposes a two-stage trajectory planning method for self-driving vehicles (SDVs) in intersection scenarios with uncertain social circumstances while considering other traffic participants, which are human-driving vehicles (HDVs) with different driving styles. The mixture-of-experts approach is first utilized to learn from human-driving trajectory data to construct a multimodal motion planner, which uses a Transformer to model the interactions between vehicles by explicitly considering their driving styles to facilitate the integrated network to achieve scene-consistent multimodal trajectory prediction and candidate trajectory generation. Second, based on the generated trajectories for the SDV and the predicted trajectories for the other HDVs, each candidate planning trajectory is evaluated via a safety-balanced value function. After that, the trajectory with the highest value is selected for implementation. Such a method plans a safe and efficient driving trajectory in complex and uncertain scenarios. The experimental results demonstrate the efficiency and effectiveness of the designed method as well as the robustness and reasonableness of the SDVs' maneuver decisions at an intersection considering the behavioral dynamics of HDVs.**

***Index Terms*—Autonomous vehicles, trajectory planning, social interactions, uncertain circumstances.**

I. Introduction

TRAJECTORY planning at intersections on city roads where new participants join from time to time is one of the most common but challenging tasks in daily driving [1]. Sensing and responding to time-varying social situational changes are critical for self-driving vehicles (SDVs) to make safe, effective, and efficient decisions while interacting with surrounding traffic participants. Thus, their social complexity becomes the most challenging but crucial factor.

Learning and analyzing driving styles [2], sensing and predicting new participants' intentions, and self-maneuver-decision-making according to situational changes are the three main functions of an SDV when planning a trajectory through an intersection considering that the other participants are human-driving vehicles (HDVs) with consistent driving styles.

Fig. 1 shows a typical left-turn trajectory planning scenario with interference from oncoming traffic in an intersection. While driving, the SDV's desired turn-left trajectory planning should also account for traffic safety and avoid congestion. Therefore, if the driving style of the oncoming vehicle is radical with a solid intention to go straight, as shown in Fig. 1(a), the SDV needs to yield and slow down for collision avoidance. However, if the driving style of the oncoming vehicle is conservative with an intention to turn left, as shown in Fig. 1(d), the SDV directly performs a turn left decision. Thus, learning and analyzing the relevant participants' driving styles are vital for SDV trajectory planning.

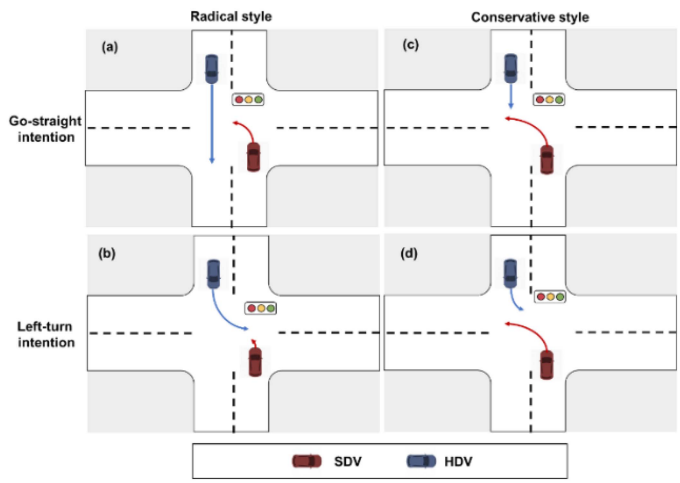


Fig. 1. The trajectory planning of SDVs will be influenced by different driving styles and the driving intentions of surrounding HDVs.

Manuscript received 30 November 2022; revised 7 January 2023; accepted 9 January 2023. Date of publication 26 January 2023; date of current version 19 May 2023. This work was supported by the National Natural Science Foundation of China under Grant 62173329. *(Corresponding author: Xiao Wang.)*

Xiao Wang is with the School of Artificial Intelligence, Anhui University, Hefei 230031, China, and also with the Haomo Technology Co., Ltd, Beijing 100192, China (e-mail: xiao.wang@ahu.edu.cn).

Ke Tang, Jintao Xu, Rui Ai, and Weihao Gu are with the Haomo Technology Co., Ltd, Beijing 100192, China (e-mail: yunfan@haomo.ai; xujintao@haomo.ai; luxiuyuan@haomo.ai; guweihao@haomo.ai).

Xingyuan Dai is with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the Qingdao Academy of Intelligent Industries, Qingdao 266114, China (e-mail: daixingyuan2015@ia.ac.cn).

Jinhao Xi and Yuxiao Wang are with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China, and also with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: xijinhao2020@ia.ac.cn; wangyuxiao2021@ia.ac.cn).

Changyin Sun is with the School of Artificial Intelligence, Anhui University, Hefei 230031, China (e-mail: cysun@ahu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIV.2023.3239903.

Digital Object Identifier 10.1109/TIV.2023.3239903

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

Traditional rule-based planning methods set predefined modeling rules for the above-described complex situations. They usually make reasonable planning decisions by comprehensively considering the instant social interactions among traffic participants to guarantee a specific pass rate. However, these predefined rules cannot cover all the possible situations in an intersection. The constantly-occurring rule-uncovered corner cases cause rule-based planning methods to fail all the time. In addition, rule-based SDVs cannot transfer and share common knowledge about different scenarios with one another. This lack of ability to share information combined with the aforementioned constraints results in rule-based methods being unable to utilize the benefits of Big Data effectively and limits their compatibility with popular deep learning methods.

Data-driven trajectory planning methods are quite the opposite and can complementarily use Big Data, deep learning, and reinforcement learning methods to address intersection decision-making challenges. For them, enhancing scenario generalization and modeling uncertain environment are crucial for SDVs to implement safe and rational trajectory planning for different times and scenes. However, the state-of-the-art methods often fail to achieve the above two goals simultaneously. Bansal et al. proposed ChauffeurNet, which learns trajectory planning directly from expert data [3]. Although this method is more scalable than traditional rule-based planning methods, it does not consider the generalization of scenarios and predictions regarding environmental uncertainty. Thus, it cannot guarantee the safety of autonomous driving. To address environmental uncertainty, Diehl et al. proposed the UMBRELLA model [4], which solves the problem of prediction, planning and control of autonomous driving by modeling uncertainty in a partially observable environment. However, the model does not consider the diversity of planning trajectories of autonomous vehicles and lacks the ability of relational reasoning about the environmental entities, resulting in poor scene generalization for planning. Meanwhile, the single trajectory generated by the above method has difficulty ensuring the safety of driving, and the planned route does not necessarily meet the dynamic requirements of the downstream controller. To address this issue, multimodal trajectory generation considering environmental randomness is gaining attention among researchers [5]. However, existing multimodal trajectory planning methods tend to overlook the impact of social factors on the environment, making it difficult to ensure driving safety.

Parallel driving was proposed in the early 2000s to improve the safety, smartness, sensitivity, security, sustainability and efficiency of SDVs by training in a digitalized proving group [6], [7]. Recently, with the development of data-driven machine learning methods, scholars extended the idea to virtual-real interactive learning and testing and further proposed parallel learning [8], [9], [10], [11] and parallel testing methods [12]. At the same time, MIT, Waymo and Intel also noticed the importance of pretraining in virtual space and proposed parallel autonomy [13], data closed-loop platforms [3] and lane-changing models [14], respectively, which verified the effectiveness and recognition of parallel driving.

Our work builds on the research ideas regarding parallel driving and aims to design a parallel reasoning architecture and develop new methods for safe and efficient trajectory planning. We propose a two-stage trajectory planning method for SDVs in intersection scenarios with uncertain social interactions among other traffic participants. Driving style and intention uncertainty are considered to achieve better scenario generalization performance while guaranteeing driving safety. Specifically, we learn from expert-driving trajectory data and construct a mixture-of-experts (MoE) [5], [45] multimodal trajectory generator that is similar to the architecture described in [5]. A Transformer [15] is used to capture vehicles' interaction behaviors, and a driving-style aware module is incorporated with the network to implement scenario-consistent multimodal trajectory prediction and planning. Second, a safety verification module is designed to evaluate the candidate planning trajectory obtained during the first stage; then, the trajectory with the highest safety score is selected for the downstream controllers.

Trajectory planning at intersections on city roads where new participants continually join has been one of the most common but challenging tasks for SDVs. To this end, we focused on the social complexity of driving in intersections and proposed a safety-balanced driving-style-aware trajectory planning method. The main contributions of this work include the following:

1) A two-stage autonomous driving trajectory planning method is designed, and the method improves the safety, rationality, and robustness of self-driving strategies.
2) A candidate trajectory generation method that considers the driving styles and driving intentions of surrounding vehicles is proposed. The method improves an SDV's ability to predict the multimodal trajectory of surrounding vehicles and thus helps SDVs generate safe and interpretable trajectories in interaction scenarios.
3) A safety verification mechanism based on candidate trajectory prediction is proposed to balance the safety and efficiency of the planned trajectory in response to the surrounding uncertain vehicle behaviors.
4) The experimental results demonstrate that the designed method plans safe and efficient trajectories in intersections with uncertain environments by explicitly considering the driving style of the surrounding vehicles.

The remainder of this paper is organized as follows. Section II presents the related works. Section III introduces the methodology of the proposed two-stage planning method. Section IV illustrates how the proposed method is implemented, verified and evaluated and elaborates on the results and explanations of the experiments. Section V concludes the work in this paper and presents our perspective on the development of trajectory planning in SDVs.

II. Related Work

Autonomous vehicles must negotiate with other traffic participants regarding the right to use the road while driving to achieve their goals of safe and efficient driving in complex social traffic scenarios [59]. The safe path planning of SDVs in such complex situations can be formulated by predicting and analyzing the driving styles and intentions of others and making appropriate in-time maneuvering decisions. The main functions include (1) learning and predicting the driving styles and intentions of other

traffic participants with mobility capabilities; (2) generating feasible multimodal paths based on road traffic conditions and the styles and intentions of potentially conflicting vehicles; (3) performing safety verification on the generated paths; and (4) selecting the safety-balanced optimal driving path.

A. Driving Style Recognition

Driving style refers to how a driver maneuvers in different driving scenarios under different external factors, such as time and weather [2]. Due to the social complexity of traffic situations and uncertainty about the drivers' states, they may exhibit different driving styles even in the same scenario. Predicting their driving style helps improve the safety and efficiency of SDV trajectory planning.

The current related algorithms are mainly classified into rule-based, model-based, and machine learning-based methods. Early rule-based approaches use a priori knowledge and manage a limited number of variables and data to classify driving styles. Model-based approaches use a set of predefined feature equations to describe driving styles by adjusting parameters to fit the driving style data [16], [17], [18]. These methods can have a high recognition ability in terms of driving styles, but it is more complicated for the model's design.

Machine learning-based methods can be specifically subdivided into three categories: (1) unsupervised training methods that statistically analyze the data with input features to achieve classification [19], [20]; (2) supervised training methods such as the K-nearest neighbors (KNN), neural networks, decision trees, random forest, and Markov models [21], [22], [23], [24], which use known sample features and corresponding styles for training; and (3) unsupervised and supervised hybrid methods that combine the advantages of both to improve the overall driving style recognition performance; one example involves combining K-means and the support vector machine (SVM) [25]. The unsupervised training method has a wide range of applications, but the output lacks interpretability; the supervised training method has fewer model parameters and is easy to understand, but the model's accuracy could be more satisfactory. The combination of unsupervised and supervised methods can make use of the advantages of both to improve the overall driving style recognition performance.

B. Driving Intention Inference

Driving intention inference refers to reasoning about the actions of surrounding vehicles over time in the upcoming time step [26]. Different vehicles in the same scenario have different driving intentions and generate different driving trajectories. Accurately inferring other traffic participants' intentions helps SDVs plan proper and safe paths.

Driving intention inference methods include probability-based methods and learning-based methods [27]. Probability-based methods model vehicle behavior through dynamics and kinetic functions, and model uncertainty through normal distributions. Learning-based methods infer driving intentions mainly through data-driven vehicle state information regarding the environment. Zyner et al. [28] used long short-term memory networks to identify vehicle intentions at intersections. Lee et al. [29] proposed using convolutional neural networks (CNNs) to predict and infer traffic participants' lane-changing intentions and improve safety and ride comfort through model predictive control. Ding et al. [30] proposed a driving intention prediction method for highway scenes that predicts vehicle intentions by considering vehicles' interaction behavior. The disadvantage of the above research is that it is difficult to use driver behavior data effectively and to fully consider the impact of driving style on driving intention. This paper develops a data-driven trajectory planning method to infer intention by considering the driver's style and predicting a reasonable driving trajectory.

C. Data-Driven Trajectory Prediction

In recent years, researchers have focused on autonomous driving trajectory prediction tasks based on deep learning models due to the increasing complexity of datasets in terms of volume and content [57]. Based on data-driven approaches, a large amount of data can be used to train the model. Most of these approaches consist of an encoder-decoder architecture. Early CNN-based [31] and recurrent neural network (RNN)-based [32] methods extract spatial features and time series information by using a bird's-eye view and historical trajectories, respectively. However, such methods encounter scene information loss and long-distance dependence problems. More importantly, the interaction information between vehicles is missed. Although graph neural networks [33] have been used in later studies to model interaction information by vectorization, these models are "black boxes" and have poor model interpretability and unimodal prediction trajectories. To solve the above problems, the generative adversarial network (GAN) is used to model the uncertainty of environmental information and generate multimodal prediction trajectories [34]. However, the GAN-based approach suffers from the problem of pattern collapse [35], which causes the training model to collapse and prevents it from converging. In addition, the Social-GAN model ignores the driving style and intention changes of surrounding vehicles, which is inconsistent with reality. To solve this problem, the leader-follower game controller model [36] uses a game-theoretic approach to model the driving style and intention classification of surrounding vehicles. The attention-based mechanism approach [37] is a popular strategy that has recently attracted the attention of researchers. This approach can quickly focus on high-value factors, especially the Transformer method [38], which has been adopted from the natural language processing field to autonomous driving. The model can generate highly accurate multimodal trajectories. In this paper, we use a data-driven planning approach and use a bird's-eye view semantic map as input, which consists of a high-definition (HD) map and agents. To account for the uncertainty of HDV driving behavior, the interactions between vehicles are modeled by a Transformer, and the driving style of the HDVs is explicitly modeled. The decoder outputs a multimodal predictive trajectory based on driving style with safety-sensitivity and interpretability.

D. Safety-Sensitive Trajectory Planning

As one of the core modules of the autonomous driving system, the trajectory planning module is used to receive environmental

information input from the sensing module and output the trajectory for the downstream control module. The trajectory planning module first predicts the future trajectory using environmental and interaction information and generates a safe and interpretable planning trajectory based on the generated multimodal trajectory. Reasonable path planning is the key to achieving safe autonomous driving. This paper divides the trajectory planning task into data-driven unimodal and safety-sensitive multimodal trajectory planning.

The data-driven unimodal trajectory planning approach is mainly based on imitation learning, where driving strategies are learned from expert data. Early trajectory planning based on imitation learning used behavioral cloning (BC) [4]. However, BC relies on expert data, experiences cascading errors and suffers from low safety in unknown scenarios. Although the data aggregation method solves the problem of cascading errors, the human cost increases. In a subsequent study, the generative adversarial imitation learning (GAIL) model was proposed to solve the cascade error problem in the BC model [39]. GAIL consists of a generator and a discriminator. The generator is trained to generate expert samples, and the discriminator is trained to generate samples that are distinguishable from actual samples. These data-driven approaches focus on applying sequential decision-making techniques to simulate human drivers. This reward-based learning model avoids the drawbacks of short-sighted behavioral choices in the behavioral cloning approaches. GAIL can generate long-term trajectories that are immune to cascading errors. Kuefler et al. applied GAIL to a highway driving scenario [40]. However, the above approach does not consider driving style and driving intentions, which are the main factors that constitute uncertainty. ChauffeurNet [3] takes a bird's-eye view as input and extracts environmental information. A mixture of actual and simulated data is used to train the model. The output predicted trajectory is used as expert data to learn recovery after drifting and avoid dangerous behaviors such as collision, swaying, and lane departure by combining the behavioral changes of experts. UMBRELLA [4] jointly solves the prediction, planning and control problems of SDVs in an interpretable learning manner. This method trains the model from offline data through a model-based method while considering the partial observability of the environment and uncertainty.

The data-driven multimodal trajectory planning approaches incorporate the influence of the SDV on other agents during the planning process. A representative study first predicted the trajectories of surrounding vehicles, generated multimodal trajectory candidates, and then selected the final trajectory via rule-based metrics [5]. Although the data-driven works have high-precision inference for planning, they have not considered the influence of the road agent's driving style. Therefore, it is difficult to ensure the rationality and consistency of trajectory planning. In this paper, we design a driving-style-aware multimodal planning method and propose a safety assessment link based on safety-sensitivity as a trajectory selection method to achieve safe and efficient planning.

III. Methodology

This section first describes the trajectory planning problem under the interaction between SDVs and HDVs. Then, we propose a two-stage trajectory planning method that contains driving-style-aware candidate trajectory generation and safety-verification-based trajectory selection. The proposed method learns the driving behavior of human drivers in the interaction scenario by explicitly modeling the uncertainty of driving styles and the intentions of surrounding HDVs to generate reasonable driving trajectories. Moreover, we perform safety verification on the candidate trajectories to improve the driving feasibility of SDVs in complex interaction scenarios.

A. Problem Formulation

Guided by a predefined route, the trajectory planner accounts for the multiobjective requirements of the vehicle and the corresponding dynamics constraints and outputs the optimal executable trajectory. We formulate the driving scenario consisting of SDVs and different numbers of interactive traffic participants as a discrete-time system in continuous space [41]. The SDV is denoted as A^0, and other traffic participants (i.e., HDVs) are denoted as $A^1, \ldots, A^N$. It should be noted that the other traffic participants can be vehicles, bicycles, or pedestrians, which the agent displays. The state of agent i at time t is denoted as s_t^i, and the corresponding trajectory is τ_t^i. Apart from the agents, the scenario context, including a vectorized HD map with traffic signals, is also considered. The context is represented as M_t. At time t, given the historical state $X_t = s_{t-H:t}^{0:N}$ of the SDV and the HDVs from the previous H steps and with the scene context M_t, the goal of trajectory planning is to learn a planner function that generates the future trajectory of the SDV

$$\tau_{t+T}^0 = f(X_t, M_t), \tag{1}$$

while ensuring that the trajectory meets the requirements of safety and efficiency.

B. Model Framework

In the interaction scenario, the uncertainty of the driving behavior of surrounding agents causes SDV safe planning difficulties. If the SDV intends to plan a safe trajectory, it needs to accurately predict the movements of other vehicles, which are influenced by their drivers' driving styles and driving intentions. Meanwhile, the surrounding vehicle motions will also be influenced by the future trajectory of the SDV. Therefore, the SDV needs to fully consider the environmental uncertainty caused by these factors to ensure safe and efficient planning behavior. To address the difficulty of planning in intersection scenarios with uncertainties and complex social interactions, we propose a two-stage trajectory planning method that contains candidate trajectory generation based on HDV driving style recognition, driving intention judgment, and trajectory selection based on safety verification.

The model architecture is shown in Fig. 2. First, we construct a multimodal joint prediction and planning module considering

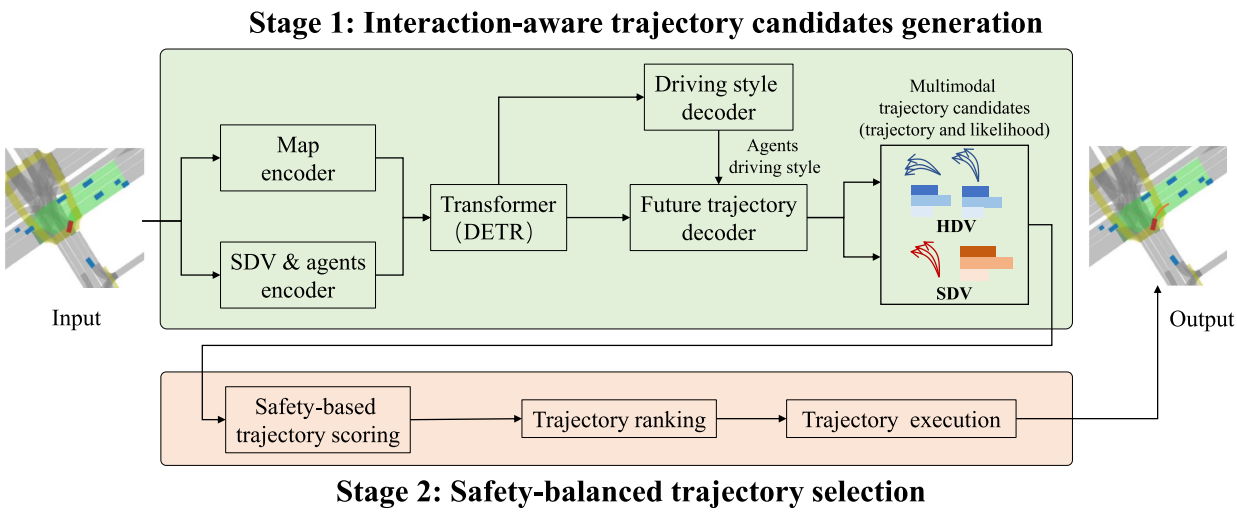


Fig. 2. Framework of the two-stage safety-balanced driving-style-aware trajectory planner.

vehicle driving style during the candidate trajectory generation stage. The module accepts the bird's-eye view semantic graph as input, which includes an HD map and other vehicle agents. The vehicle interaction is encoded through a Transformer. Meanwhile, the driving styles of the surrounding HDVs are explicitly modeled, and the predicted likelihood of the vehicle driving style is used as a conditional constraint to assist the trajectory decoding network in achieving interaction-aware joint prediction and planning. The process produces candidate trajectories for the SDV. During the trajectory selection stage, the safety-sensitive trajectory evaluation function scores each candidate trajectory based on the predicted trajectories of the other vehicles in the first stage as well as other road information. Finally, the trajectory with the safest score is selected for controller execution. We detail the two stages in Sections III-C and III-D.

C. Driving Style-Aware Joint Prediction and Planning

This subsection introduces the proposed joint prediction and planning module considering the HDV driving styles. We first present the inputs and outputs of the module. Then, we design the network structure, and finally, we propose the training method and optimization objectives.

1) Input and Output Representation: The joint prediction and planning network generates both SDV multimodal trajectory candidates and surrounding HDV multimodal predictive trajectories based on the currently observed vectorized bird's-eye view states. The specific representations of its input and output are as follows:

Inspired by recent vectorization representation methods [5], [42], we describe the driving scenario in vector form, and the following data are used as inputs to the joint prediction and planning network: The input contains dynamic entities, static entities, and extra information. The dynamic entities include the SDV's current and past poses, which include the coordinates and yaw angles of the last H steps and the HDV's current and past poses and their types. The static entities include a static HD map with lanes, pedestrian crossings, intersections, and traffic light states. The extra information includes routes that indicate the middle lanes that the SDVs should follow. Each input element is encoded as a reference frame centered on the SDV and contains the element type as an additional feature.

The model's output consists of three components: multimodal trajectory planning, driving style recognition and multimodal trajectory prediction for the surrounding vehicles.

The trajectory planning output consists of K future SDV trajectories and the corresponding probability distributions. Each SDV trajectory is defined as a set of T discrete states

$$\tau_t^i = \{x_t^i, y_t^i, \theta_t^i\}, \tag{2}$$

where t is the time step in the $[1, T]$ interval, and x, y, θ denotes the horizontal and vertical coordinates and yaw of the vehicle midpoint relative to the current moment, respectively. These values constitute the vehicle pose. The probability distribution $p^k = p(\tau^k | X_t, M_t)$ is defined as the likelihood of the K SDV trajectories given the current state; this likelihood is calculated from the human driving trajectory data and used to model the uncertain driving behavior of human experts in the current state. The computation of the multimodal planning trajectories with the corresponding trajectory probability likelihoods is presented in Section III-C-3.

The driving style identification output labels the instantaneous driving styles of the N HDVs in the scene. Each HDV's predicted driving style output is represented as a three-dimensional one-hot encoding, corresponding to radical, moderate, and conservative driving styles.

The model's trajectory prediction output contains the future trajectories of surrounding agents and a probability distribution that is used to evaluate the likelihood that each agent follows the K possible future trajectories. Similar to the planning output, the predicted trajectories of each agent are defined as a discrete set of states

$$\tau_t^n = \{x_t^n, y_t^n, \theta_t^n\}. \tag{3}$$

Each probability distribution q^n, $n = 1, \ldots, N$ is defined as K possible future trajectories of agent n. This probability distribution is used to select the most likely trajectories.

2) Model Architecture: The architecture of the joint prediction and planning model based on driving styles is shown in Fig. 2. The overall architecture of each module is constructed using neural networks, and the backbone network design is similar to [5]. The main difference in our network lies in the intermediate layer, which recognizes the driving style of the surrounding vehicles. The designed network, which employs end-to-end joint training, ensures that our network effectively models the diversity of driving styles and the intentions of vehicles in the intersection scenario. The model can better characterize the intersection state and help the planner to generate safe and efficient trajectories.

The overall network can be divided into the encoder and the decoder. The encoder maps the SDV-centered scene vector states to hidden embeddings. We apply elementwise point encoders to extract vectorized representations of the intersection state to effectively model the static and dynamic element relationships in the intersection scene. Specifically, we first consider the local relationships of each entity in the scene and encode the state of each vectorized entity (e.g., static map and agents) in the input scene separately. The point features in each entity state are embedded with the same length by multilayer perceptrons (MLPs), while the location information of the points is encoded by sinusoidal encoding. The point set of each entity embedding together with the corresponding location encoding is fed into the 3-layer PointNet [43], which outputs the feature encoding of each entity. The encoding of all entities has the same dimensions. Then, we consider the global relationship between the entities in the scene. Based on the encoding of each entity, we perform global feature aggregation using the DETR architecture [44], which is an encoder-decoder Transformer. Based on the extracted local features of the vehicles via PointNet, the Transformer, which is equipped with self-attention modules, captures the global scene features. The method involves encoding the relationships among all of the input elements (SDVs, road agents, dynamic maps, and lane lines) using the Transformer encoder and querying the characteristics of the SDV and agents using the transformer decoder. The query of the Transformer decoder is formed using a set of learnable embeddings, which are created by adding the SDV embeddings output by the PointNet network to learnable ensembles to generate distinct queries for each of the predicted future SDV trajectories.

The decoder in our architecture contains two modules. The driving style decoder outputs the driving styles of the surrounding HDVs based on scene state encoding. The trajectory decoder jointly outputs the predicted trajectories of the surrounding vehicles with the trajectory candidates planned by the SDV based on the intermediate layer features of the scene state encoding and the driving style decoder. Specifically, the driving style decoder consists of a scaled dot product attention layer and an MLP layer. Its output predicts the probability of three driving styles (radical, moderate, and conservative) for N agents. The planning trajectory decoder and the prediction trajectory decoder share the scaled dot product attention layer and contain separate MLP layers, which are built on top of the shared layer, for prediction and planning. The input to the shared layer contains the merged embedding of the scene state encoding and the intermediate layer output of the driving style decoder. The trajectory decoding network outputs the predicted multimodal trajectories of the agents with the multimodal planning trajectory and the corresponding probabilistic likelihood of each trajectory.

3) Training Process and Objective Function: Inspired by [5], we use imitation learning to train the joint predictive planning network, and the training objective contains three components: the distance between the predicted SDV trajectory and the human expert trajectory, the distance between the predicted agent trajectory and its actual future trajectory, and the distance between the predicted agent driving style and its driving style label. The driving style labels are obtained by clustering in the offline dataset before the training process.

The joint prediction and planning process is represented as an MoE model to forecast multiple trajectories for the SDV and the surrounding agents. Additionally, the probability distribution for each trajectory set corresponding to expert selection is obtained. To avoid the pattern collapse problem during expert training and expert selection, we use a greedy approach in which the maximum probability trajectory for the loss calculation is chosen. We formulate a matching cost between the predicted and target trajectories and probabilities so that the expert with the lowest cost becomes the winner. The matching cost for each SDV trajectory is calculated, and a trajectory is selected via

$$i^* = \arg_i \min L_{IL}^i + \lambda(1 - p^i), \tag{4}$$

where

$$L_{IL}^i = \sum_{t=1}^{T} ||\tau_t^i - \hat{\tau}_t|| + \beta L_{reg}^i \tag{5}$$

and p^i is the predicted probability of trajectory τ^i. L_{reg}^i is the trajectory smoothness regularization term. Then, the loss is minimized

$$L = L_{IL}^{i^*} + \mu L_{NLL}^{i^*}, \quad \text{where} \;\; L_{NLL}^{i^*} = -\log p^{i^*}. \tag{6}$$

The objection trades between imitation loss and multimodal loss for trajectory planning.

The network for driving style recognition is trained in an end-to-end manner with the prediction and planning network. The K-means clustering algorithm [2], [46], [47], [48] is used to generate instantaneous driving style labels based on the vehicle trajectories. Driving style recognition consists of three steps: generating trajectory datasets, generating velocity and acceleration datasets from the trajectory datasets, and identifying the driving style of the surrounding human drivers in each scene. Each scene in the training set creates a dataset by capturing the speed and acceleration of the vehicle trajectory and obtains the driving style label through unsupervised K-means clustering. The input to the K-means clustering consists of vector samples representing the speed or acceleration data of each vehicle in each scene. The obtained label of each car in the scene is encoded into a three-dimensional one-hot embedding, representing the

cluster's three driving style labels: radical, moderate, and conservative. Our driving style recognition model identifies the driving styles of the surrounding vehicles based on their current and past states, and the objective function is designed as a cross-entropy function of the predicted driving style $\hat{Y}^i_{DS}$ and the label Y^i_{DS}

$$L_{DS} = H_c(\hat{Y}^i_{DS}, Y^i_{DS}), \tag{7}$$

where H_c denotes the cross-entropy function.

D. Safety-Balanced Trajectory Selection

The candidate trajectories generated by the joint prediction and planning module during the first stage are used as input for the second stage, and the trajectory with the highest safety value is selected for execution.

We account for the generated SDV's multimodal trajectory and the predicted HDVs' trajectories when performing trajectory selection. A collision check is performed on each future SDV trajectory τ^i and the predicted most likely trajectory of the HDVs. Safety verification is performed by calculating the overlapping area between the vehicle bounding boxes. The shortest distance between the bounding boxes is denoted as $C(\tau^i)$. Then, the distance between the SDVs from the lane median, which is denoted by $F(\tau^i)$, is also used to evaluate the reasonableness of the SDV trajectory. The above two indicators are used to select a safe and reasonable trajectory. For each candidate trajectory, when the distance between the SDV bounding box and that of surrounding vehicles is large and the distance between the SDV and the centerline is small, the planned trajectory aligns with our expected requirements. The final trajectory value function is obtained by

$$V(\tau^i) = C(\tau^i) - F(\tau^i). \tag{8}$$

The trajectory chosen based on this value function allows the SDV to maintain a safe distance from other vehicles and follow the lane centerline as much as possible. Thus, the approach improves the safety of the SDV during online inference.

IV Experiments and Evaluation

A. Dataset

To train and test our model, we use the Lyft motion prediction dataset [49]. This dataset was collected from real vehicles and contains actual driving samples from complex urban routes in Palo Alto, California. The dataset contains a variety of real-world driving scenarios, such as driving in multiple lanes, turning, and vehicle interactions at intersections. The perception system preprocesses the data to obtain the precise location of the surrounding vehicles over time. In addition, the dataset contains an HD map that provides the location and status of lane lines, crosswalks and traffic lights.

B. Baseline

We compare our proposed algorithm with four strong BC-based planner baselines.

1) Behavioral cloning with image-based scene representation (BC-I). BC-I is a standard BC planner based on the ResNet-50 backbone. The planner's input is an ego-centered scene raster image without historical information.
2) Behavioral cloning with image-based scene representation and trajectory perturbations (BC-IP). Similar to ChauffeurNet [3], the method adds synthetic perturbations to the ego trajectory in the dataset during training.
3) Behavioral cloning with vector-based scene representation (BC-V). BC-V is a standard BC planner based on the PointNet backbone. The planner's input is the ego-centered scene vector, which is the same as the scene representation in this paper.
4) Behavioral cloning with vector-based scene representation and trajectory perturbations (BC-VP). This method adds synthetic perturbations to the ego trajectory for vector-based scene representation.

C. Evaluation Indicator

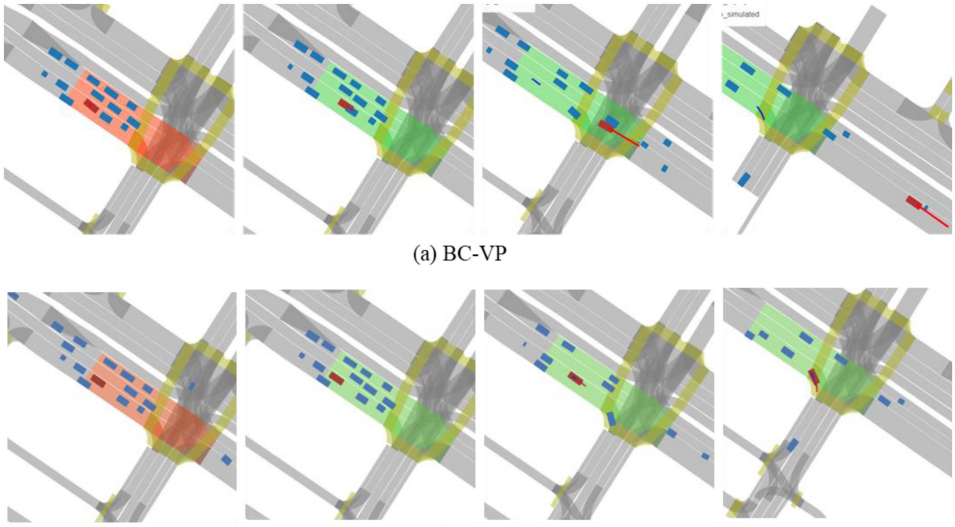
We use the following metrics to comprehensively evaluate the planner performance. These metrics reflect imitation performance and safety. It should be noted that the first four metrics are for closed-loop tests; the last two are for open-loop tests.

1) *Collision*: Collision of the SDV with any other intelligent body, including front, rear and side collisions with the SDV.
2) *L2*: L2 displacement error in meters from the base expert position in the driving log.
3) *Off-road events*: This event is triggered if the lateral distance between the planned trajectory and the reference trajectory is more than 2 m. The metric reflects that the vehicle is running off roads or into the opposite lanes.
4) *Interventions*: This composite metric evaluates safety failures by considering collisions and off-road events.
5) *ADE:* The average displacement error for the planning trajectory.
6) *FDE:* The final displacement error for the planning trajectory.

D. Experimental Results

We performed closed-loop and open-loop tests to evaluate the model's performance. During the closed-loop test, we validated the safety of the proposed two-stage trajectory planning approach. During the open-loop test, we focused on the distance between the planned and actual human driving trajectories.

1) Closed-Loop Test: We compare the planning performance of the proposed model with the four BC planning benchmarks through closed-loop tests. The experimental results in 1000 scenarios show that the proposed method has a lower collision rate than all baselines. In the 1000 scenarios, our method resulted in a total of 62 collisions, which is the lowest for all compared planners. This result indicates that the proposed two-stage method is more safety-sensitive during the planning process. In addition, the trajectories planned by our model are more reasonable. Our planner's number of off-road events is the lowest for all compared planners.



(a) BC-VP

(b) Ours

Fig. 3. Visualization of the closed-loop test in a complex intersection with uncertainty.

TABLE I
PLANNING METRICS FOR BASELINES AND OUR METHOD ON THE CLOSED-LOOP TEST

Model	Collisions (front/rear/side)	Imitation (L2)	Off-road events	Interventions
BC-I	22/130/80	**3**	719	951
BC-IP	1/222/1	15	23	247
BC-V	69/295/44	45	533	941
BC-VP	1/119/3	20	28	151
Ours	**1/60/1**	14	**15**	**77**

TABLE II
PLANNING METRICS FOR BASELINES AND OUR METHOD ON THE OPEN-LOOP TEST

Model	ADE	FDE
BC-VP	0.924	1.630
Ours	**0.447**	**1.105**

TABLE III
PLANNING METRICS FOR ABLATION STUDY ON THE CLOSED-LOOP TEST

Model	Collisions (front/rear/side)	Imitation (L2)	Off-road events	Interventions
Ours w/o DS and SV	1/113/2	17	18	134
Ours w/o SV	1/95/1	14	15	112
Ours	**1/60/1**	**14**	**15**	**77**

We find that perturbation is an important technique for addressing the covariance shift problem [3] in closed-loop tests. Specifically, BC planners with trajectory perturbations, such as BC-IP, BC-VP, and our proposed method, all show improved planning performance in terms of safety and reasonableness compared to BC planners without trajectory perturbations, such as BC-I and BC-V. Our results show that using trajectory perturbation as a data augmentation method during training can significantly reduce the mismatch between the training and inference-time distributions, improving the generalization of the planner in closed-loop tests. This conclusion holds for both image-based and vector-based planners.

To further validate the reasonableness of our proposed planner, we visualize the planning behavior of the proposed method with the benchmark method for a complex interaction scenario. As shown in Fig. 3, our method can cause the SDV in red to stop at the intersection when it displays the red light and resume movement when it displays the green light while maintaining a safe distance from other vehicles and driving along the route. The planning trajectory is close to the blue reference trajectory of human drivers. In contrast, the vehicle based on the baseline BC-VP planner runs the red light and hits other vehicles while not following the routes.

2) Open-Loop Test: We also evaluated the imitation performance of the proposed planner and BC-VP planners using open-loop tests. The performance metric reflects the closeness of the planning trajectory by the BC planners to the reference trajectory by human drivers for in-distribution scenarios. The experimental results are shown in Table II. Our planner achieves a lower value than the benchmark on both the ADE and FDE metrics, indicating that our method is more capable of imitating expert driving behavior.

3) Ablation Study: To validate the effectiveness of the different modules in the proposed method, we performed ablation study using closed-loop tests. The ablation module includes driving style recognition (DS) and safety verification (SV).

The closed-loop test results are shown in Table III. The imitation and off-road event metrics indicate that, without considering driving style, the planned vehicle trajectory greatly differs from the actual reference trajectory. This result indicates that the driving styles of other drivers partially influence human driving trajectories. Therefore, considering the driving styles of the

surrounding vehicles will help to achieve reasonable planning for the SDV.

The safety verification module has also been proven to be effective. The experimental results show that using safety verification can reduce the number of collisions compared to not using safety verification. This result demonstrates that the combination of multimodal trajectory generation and safety verification can maximize safety during the SDV driving process.

V. Conclusion

In this paper, we propose a safety-balanced two-stage trajectory planning method that considers the driving styles of other HDVs to address the difficulties of SDV trajectory planning caused by the behavioral uncertainty of surrounding traffic participants in intersection scenarios. The two-stage planning approach consists of multimodal candidate trajectory generation based on driving style and trajectory selection based on safety-balanced verification. During the first stage, the model predicts the future trajectory of the surrounding HDVs based on driving style recognition and intention inference and simultaneously generates multimodal candidate trajectories that conform to the human driving experience of the autonomous vehicle in the interaction scenario. During the second stage, based on the generated candidate trajectories and the predicted HDV trajectories, we verify and score the trajectories with a safety value according to the rules constructed from experience and select the safest driving path for final execution.

In the proposed two-stage approach, the first stage learns the driving style and intention of the HDVs to better predict the driving behavior of HDVs and generates feasible trajectory candidates. The second stage further improves driving safety in complex interaction scenarios through safety verification.

In the future, we plan to integrate the proposed two-stage approach with the cognitive world model to better generalize SDV trajectory planning. The world model [50] can be used to simulate the prediction of various possible traffic events during the driving process, provide the potential to evaluate planning behavior and ensure safe and reasonable trajectory planning in the actual system. In addition, the world model is updated through continuous learning based on empirical data to ensure the cognitive alignment of the world model with the real world. In this way, we will integrate parallel planning with parallel vision [51] and parallel control [52], [56] to guarantee safe and efficient autonomous driving in complex road environments. We think parallel driving and parallel testing can be incorporated as a closed-loop system and promote autonomous driving to the 6S world, which is safe in the physical world, secure in the cyber world, sustainable in the ecological world, sensitive to individual needs, services for all, and smart in parallel intelligent societies [53], [54], [55], [58], [60].

References

[1] Z. Zhu and H. Zhao, "A survey of deep RL and IL for autonomous driving policy learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14043–14065, Sep. 2022.

[2] C. M. Martinez, M. Heucke, F.-Y. Wang, B. Gao, and D. Cao, "Driving style recognition for intelligent vehicle control and advanced driver assistance: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 666–676, Mar. 2018.

[3] M. Bansal, A. Krizhevsky, and A. Ogale, "ChauffeurNet: Learning to drive by imitating the best and synthesizing the worst," *Robot.: Sci. Syst.*, 2019.

[4] C. Diehl, T. Sievernich, M. Krüger, F. Hoffmann, and T. Bertram, "Uncertainty-aware model-based offline reinforcement learning for automated driving," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 1167–1174, Feb. 2023.

[5] S. Pini, C. S. Perone, A. Ahuja, A. S. R. Ferreira, M. Niendorf, and S. Zagoruyko, "Safe real-world autonomous driving by learning to predict and plan with a mixture of experts," in *Proc. NeurIPS 2022 Workshop Mach. Learn. Auton. Driving*, 2022.

[6] F.-Y. Wang, X. Wang, L. Li, and P. Mirchandani, "Creating a digital-vehicle proving ground," *IEEE Intell. Syst.*, vol. 18, no. 2, pp. 12–15, Mar./Apr. 2003.

[7] F.-Y. Wang, X. Wang, L. Li, P. Mirchandani, and Z. Wang, "Digital and construction of a digital vehicle proving ground," in *Proc. IEEE IV Intell. Veh. Symp.*, 2003, pp. 533–536.

[8] L. Li, Y. Lin, N. Zheng, and F.-Y. Wang, "Parallel learning: A perspective and a framework," *IEEE/CAA J. Automatica Sinica*, vol. 4, no. 3, pp. 389–395, Jul. 2017.

[9] X. Li, P. Ye, J. Jin, F. Zhu, and F.-Y. Wang, "Data augmented deep behavioral cloning for urban traffic control operations under a parallel learning framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5128–5137, Jun. 2022.

[10] L. Li, Y. Lin, D. Cao, N.-N. Zheng, and F.-Y. Wang, "Parallel learning-a new framework for machine learning," *Acta Automatica Sinica*, vol. 43, no. 1, pp. 1–8, 2017.

[11] Q. H. Miao, Y. S. Lv, M. Huang, X. Wang, and F.-Y. Wang, "Parallel learning: Overview and perspective for computational learning across syn2real and sim2real," *IEEE/CAA J. Automatica Sinica*, vol. 10, no. 3, pp. 599–627, Mar. 2023.

[12] L. Li et al., "Parallel testing of vehicle intelligence via virtual-real interaction," *Sci. Robot.*, vol. 4, no. 28, Mar. 2019, Art. no. eaaw4106.

[13] W. Schwarting, J. Alonso-Mora, L. Pauli, S. Karaman, and D. Rus, "Parallel autonomy in automated vehicles: Safe motion generation with minimal intervention," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 1928–1935.

[14] Y. Xing et al., "Advances in vision-based lane detection: Algorithms, integration, assessment, and perspectives on ACP-based parallel vision," *IEEE/CAA J. Automatica Sinica*, vol. 5, no. 3, pp. 645–661, May 2018.

[15] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[16] E. Gilman, A. Keskinarkaus, S. Tamminen, S. Pirttikangas, J. Röning, and J. Riekki, "Personalised assistance for fuel-efficient driving," *Transp. Res. Part C: Emerg. Technol.*, vol. 58, pp. 681–705, Sep. 2015.

[17] D. Dörr, D. Grabengiesser, and F. Gauterin, "Online driving style recognition using fuzzy logic," in *Proc. IEEE 17th Int. Conf. Intell. Transp. Syst.*, 2014, pp. 1021–1026.

[18] Y. L. Murphey, R. Milton, and L. Kiliaris, "Driver's style classification using jerk analysis," in *Proc. IEEE Workshop Comput. Intell. Veh. Veh. Syst.*, 2009, pp. 23–28.

[19] A. Mudgal, S. Hallmark, A. Carriquiry, and K. Gkritza, "Driving behavior at a roundabout: A hierarchical Bayesian regression analysis," *Transp. Res. Part D: Transport Environ.*, vol. 26, pp. 20–26, Jan. 2014.

[20] Z. Constantinescu, C. Marinoiu, and M. Vladoiu, "Driving style analysis using data mining techniques," *Int. J. Comput. Commun. Control*, vol. 5, no. 5, pp. 654–663, Dec. 2010.

[21] V. Vaitkus, P. Lengvenis, and G. Žylius, "Driving style classification using long-term accelerometer information," in *Proc. IEEE 19th Int. Conf. Methods Models Automat. Robot.*, 2014, pp. 641–644.

[22] A. Augustynowicz, "Preliminary classification of driving style with objective rank method," *Int. J. Automot. Technol.*, vol. 10, no. 5, pp. 607–610, Oct. 2009.

[23] C. Guardiola, B. Pla, D. Blanco-Rodríguez, and A. Reig, "Modelling driving behaviour and its impact on the energy management problem in hybrid electric vehicles," *Int. J. Comput. Math.*, vol. 91, no. 1, pp. 147–156, Jan. 2014.

[24] *Markov Chains: Models, Algorithms and Applications*, vol. 83. Boston, MA, USA: Kluwer, 2006.

[25] M. Van Ly, S. Martin, and M. M. Trivedi, "Driver classification and driving style recognition using inertial sensors," in *Proc. IEEE Intell. Veh. Symp.*, 2013, pp. 1040–1045.

[26] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 1, pp. 33–47, Jan. 2022.
[27] Y. Xing et al., "Driver lane change intention inference for intelligent vehicles: Framework, survey, and challenges," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4377–4390, May 2019.
[28] A. Zyner, S. Worrall, J. Ward, and E. Nebot, "Long short term memory for driver intent prediction," in *Proc. IEEE Intell. Veh. Symp.*, 2017, pp. 1484–1489.
[29] D. Lee, Y. P. Kwon, S. McMains, and J. K. Hedrick, "Convolution neural network-based lane change intention prediction of surrounding vehicles for ACC," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst.*, 2017, pp. 1–6.
[30] W. Ding, J. Chen, and S. Shen, "Predicting vehicle behaviors over an extended horizon using behavior interaction network," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 8634–8640.
[31] G. Xie, A. Shangguan, R. Fei, W. Ji, M. Weigang, and X. Hei, "Motion trajectory prediction based on a CNN-LSTM sequential model," *Sci. China Ser. F: Inf. Sci.*, vol. 63, pp. 1–21, 2020.
[32] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," in *Proc. 31st Youth Academic Annu. Conf. Chin. Assoc. Automat.*, 2016, pp. 324–328.
[33] X. Li, X. Ying, and M. C. Chuah, "GRIP: Graph-based interaction-aware trajectory prediction," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2019, pp. 3960–3966.
[34] 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.*, 2018, pp. 2255–2264.
[35] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese, "Social-BiGAT: Multimodal trajectory forecasting using bicycle-GAN and graph attention networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 137–146.
[36] K. Liu, N. Li, H. E. Tseng, I. Kolmanovsky, and A. Girard, "Interaction-aware trajectory prediction and planning for autonomous vehicles in forced merge scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 1, pp. 474–488, Jan. 2023.
[37] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Attention based vehicle trajectory prediction," *IEEE Trans. Intell. Veh.*, vol. 6, no. 1, pp. 175–185, Mar. 2021.
[38] Y. Liu, J. Zhang, L. Fang, Q. Jiang, and B. Zhou, "Multimodal motion prediction with stacked transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 7573–7582.
[39] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4572–4580.
[40] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," in *Proc. IEEE Intell. Veh. Symp.*, 2017, pp. 204–211.
[41] Z. Huang, H. Liu, J. Wu, and C. Lv, "Differentiable integrated motion prediction and planning with learnable cost function for autonomous driving," 2022, *arXiv:2207.10422*.
[42] O. Scheel, L. Bergamini, M. Wolczyk, B. Osiński, and P. Ondruska, "Urban driver: Learning to drive from real-world demonstrations using policy gradients," in *Proc. Annu. Conf. Robot Learn.*, 2022, pp. 718–728.
[43] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.
[44] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.
[45] H. Cui et al., "Multimodal trajectory predictions for autonomous driving using deep convolutional networks," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 2090–2096.
[46] G. Ren, Y. Zhang, H. Liu, K. Zhang, and Y. Hu, "A new lane-changing model with consideration of driving style," *Int. J. Intell. Transp. Syst. Res.*, vol. 17, no. 3, pp. 181–189, Sep. 2019.
[47] Y. Ali, Z. Zheng, M.. M. Haque, M. Yildirimoglu, and S. Washington, "Understanding the discretionary lane-changing behaviour in the connected environment," *Accident Anal. Prevention*, vol. 137, Mar. 2020, Art. no. 105463.
[48] A. Sharma, Z. Zheng, J. Kim, A. Bhaskar, and M. M. Haque, "Estimating and comparing response times in traditional and connected environments," *Transp. Res. Rec.*, vol. 2673, no. 4, pp. 674–684, Apr. 2019.
[49] J. Houston et al., "One thousand and one hours: Self-driving motion prediction dataset," in *Proc. Conf. Robot Learn.*, 2021, pp. 409–418.
[50] F. Tian, Z. Li, F. Wang, and L. Li, "Parallel learning-based steering control for autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 8, no. 1, pp. 379–389, Jan. 2023.
[51] J. Wang et al., "Parallel vision for long-tail regularization: Initial results from IVFC autonomous driving testing," *IEEE Trans. Intell. Veh.*, vol. 7, no. 2, pp. 286–299, Jun. 2022.
[52] J. Lu, Q. Wei, and F.-Y. Wang, "Parallel control for optimal tracking via adaptive dynamic programming," *IEEE/CAA J. Automatica Sinica*, vol. 7, no. 6, pp. 1662–1674, Nov. 2020.
[53] X. Wang, J. Yang, J. Han, W. Wang, and F.-Y. Wang, "Metaverses and DeMetaverses: From digital twins in CPS to parallel intelligence in CPSS," *IEEE Intell. Syst.*, vol. 37, no. 4, pp. 97–102, Jul./Aug. 2022.
[54] F.-Y. Wang, "Parallel intelligence in metaverses: Welcome to Hanoi!," *IEEE Intell. Syst.*, vol. 37, no. 1, pp. 16–20, Jan./Feb. 2022.
[55] P. Ye and F.-Y. Wang, "Parallel population and parallel human—A cyber-physical social approach," *IEEE Intell. Syst.*, vol. 37, no. 5, pp. 19–27, Sep./Oct. 2022.
[56] H. Mo, Y. Meng, F.-Y. Wang, and D. Wu, "Interval type-2 fuzzy hierarchical adaptive cruise following-control for intelligent vehicles," *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 9, pp. 1658–1672, Sep. 2022.
[57] X. Zhu et al., "Interaction-aware cut-in trajectory prediction and risk assessment in mixed traffic," *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 10, pp. 1752–1762, Oct. 2022.
[58] C. Zhao, Y. Lv, J. Jin, Y. Tian, J. Wang, and F.-Y. Wang, "DeCAST in TransVerse for parallel intelligent transportation systems and smart cities: Three decades and beyonds," *IEEE Intell. Transp. Syst. Mag.*, vol. 14, no. 6, pp. 6–17, Nov./Dec. 2022.
[59] L. Li et al., "Three principles to determine the right-of-way for AVs: Safe interaction with humans," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7759–7774, Jul. 2022.
[60] P. Ye, X. Wang, W. Zheng, Q. Wei, and F.-Y. Wang, "Parallel cognition: Hybrid intelligence for human-machine interaction and management," *Front. Inf. Technol. Electron. Eng.*, vol. 23, no. 12, pp. 1765–1779, Dec. 2022.

Xiao Wang (Senior Member, IEEE) received the bachelor's degree in network engineering from the Dalian University of Technology, Dalian, China, in 2011, and the Ph.D. degree in social computing from the University of Chinese Academy of Sciences, Beijing, China, in 2016. She is currently a Professor with the School of Artificial Intelligence, Anhui University, Hefei, China, and the President with the Qingdao Academy of Intelligent Industries, Qingdao, China. Her research interests include social network analysis, social transportation, cybermovement organizations, and multiagent modeling.

Ke Tang received the graduation degree from the Beijing University of Aeronautics and Astronautics and the master's degree in instrument science and technology from the Beijing University of Aeronautics and Astronautics in 2018. He is currently the Tech Leader of PnC algorithm with Haomo.ai.

Xingyuan Dai received the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2022. He is currently an Assistant Professor with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include intelligent transportation systems, machine learning, and deep learning.

Jintao Xu received the Ph.D. degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China, in 2016. He is currently the Director of algorithm with Haomo.AI.

Jinhao Xi received the B.E. degree from Chongqing University, Chongqing, China, in 2020. He is currently a Postgraduate with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include intelligent transportation systems and reinforcement learning.

Rui Ai received the Ph.D. degree in pattern recognition and intelligent systems from the Chinese Academy of Sciences, Beijing, China, in 2013. He is currently the Vice President of Haomo Technology Company, Ltd., and leads the AI Center of Haomo. He is responsible for the research and development of high-level autonomous driving systems, unmanned delivery vehicle, and data intelligence.

Yuxiao Wang received the B.Eng. degree in automation from Xi'an Jiao Tong University, Xi'an, China, in 2021. He is currently working toward the MA.Eng. degree in artificial intelligence with the Institute of Automation, University of Chinese Academy of Sciences, Beijing, China. His research interests include autonomous driving and deep reinforcement learning.

Weihao Gu received the graduation degree major in computer science from Beijing Jiaotong University, Beijing, China. He is the Co-founder and CEO of HAOMO.AI, a top-tier startup company in the field of autonomous driving. His research interests include autonomous driving, artificial intelligence, and Big Data. In 2003, he joined Baidu, and held roles, including the Chief Architect for MP3 and video search, Head of speech recognition technology, deputy General Manager of Baidu map, and General Manager of the company's intelligent car division. He led his team to create the first self-driving map in China. He has also developed the first low-cost assisted driving solution and low-cost automatic parking solution in China. He is a Pioneer and Innovator of autonomous driving in China.

Changyin Sun (Senior Member, IEEE) received the bachelor's degree from the College of Mathematics, Sichuan University, Chengdu, China, in 1996, and the M.S. and Ph.D. degrees in electrical engineering from Southeast University, Nanjing, China, in 2001 and 2004, respectively. He is currently a Professor with the School of Artificial Intelligence, Anhui University, Hefei, China. His research interests include intelligent control, flight control, pattern recognition, and optimal theory. Dr. Sun is an Associate Editor for the IEEE Transactions on Neural Networks and Learning Systems, IEEE Neural Processing Letters, and IEEE/CAA Journal of Automatica Sinica.