Collaborative Uncertainty Benefits Multi-Agent Multi-Modal Trajectory Forecasting

Bohan Tang, Student Member, IEEE, Yiqi Zhong, Chenxin Xu, Wei-Tao Wu, Ulrich Neumann, Member, IEEE, Ya Zhang, Member, IEEE, Siheng Chen, Member, IEEE, and Yanfeng Wang

Abstract—In multi-modal multi-agent trajectory forecasting, two major challenges have not been fully tackled: 1) how to measure the uncertainty brought by the interaction module that causes correlations among the predicted trajectories of multiple agents; 2) how to rank the multiple predictions and select the optimal predicted trajectory. In order to handle the aforementioned challenges, this work first proposes a novel concept, collaborative uncertainty (CU), which models the uncertainty resulting from interaction modules. Then we build a general CU-aware regression framework with an original permutation-equivariant uncertainty estimator to do both tasks of regression and uncertainty estimation. Furthermore, we apply the proposed framework to current SOTA multi-agent multi-modal forecasting systems as a plugin module, which enables the SOTA systems to: 1) estimate the uncertainty in the multi-agent multi-modal trajectory forecasting task; 2) rank the multiple predictions and select the optimal one based on the estimated uncertainty. We conduct extensive experiments on a synthetic dataset and two public large-scale multi-agent trajectory forecasting benchmarks. Experiments show that: 1) on the synthetic dataset, the CU-aware regression framework allows the model to appropriately approximate the ground-truth Laplace distribution; 2) on the multi-agent trajectory forecasting benchmarks, the CU-aware regression framework steadily helps SOTA systems improve their performances. Especially, the proposed framework helps VectorNet improve by 262 cm regarding the Final Displacement Error of the chosen optimal prediction on the nuScenes dataset; 3) in multi-agent multi-modal trajectory forecasting, prediction uncertainty is proportional to future stochasticity; 4) the estimated CU values are highly related to the interactive information among agents. The proposed framework can guide the development of more reliable and safer forecasting systems in the future.

Index Terms—Multi-agent trajectory forecasting, multi-modal trajectory forecasting, uncertainty estimation.

I. INTRODUCTION

MUltI-AGENT multi-modal trajectory forecasting is a task that aims to predict multiple future trajectories of multiple agents based on their observed trajectories and surroundings [1], [2]. Precise trajectory forecasting provides essential information for decision-making and safety in numerous real-world applications such as self-driving cars [3], [4], [5], [6], drones [7], and industrial robotics [8], [9], [10].

As deep learning rapidly advances, a number of deep-learning-based algorithms have been proposed to handle the multi-agent multi-modal trajectory forecasting task [3], [4], [5], [6], [11], [12], [13], [14], [15], [16]. Compared with common trajectory forecasting systems which only predict one future trajectory per agent, a multi-agent multi-modal trajectory forecasting system predicts multiple possible future trajectories for each agent to handle the stochasticity inherent in future forecasting. With state-of-the-art performances, many multi-agent multi-modal trajectory forecasting systems have been widely used in real-world applications. Nevertheless, two major challenges are underexplored in the existing works on deep-learning-based multi-modal forecasting: 1) how to measure the uncertainty over the prediction for each agent under a multi-agent setting; 2) how to rank the multiple predictions for each agent under a multi-modal setting.

Solving the first challenge of uncertainty estimation is practically crucial for the trajectory forecasting task since the task is highly safety-critical while deep-learning-based trajectory forecasting methods do not always come with reliability. Nowadays, most researchers tend to use uncertainty analysis [17], [18], [19] to assess the reliability of deep-learning-based systems. Specifically, in Big Data regimes (e.g., most supervised deep learning systems that have a large number of available data), it is important to model aleatoric uncertainty, namely, the uncertainty regarding information aside from statistic models which data cannot explain [18]. Following this thread of thought, some explorations have been conducted for the task of trajectory forecasting. For example, on the basis of the framework...
proposed in [17], [18], existing works [20], [21], [22], [23] utilize the predictive variance to estimate the uncertainty over each agent’s prediction separately. By doing so, they assume that the prediction of an agent’s future trajectories is independent of the predictions about other agents. However, recent SOTA methods [3], [4], [13], [24], [25], [26] have mentioned that agents’ future trajectories are, in many cases, non-independent because of agent-wise interactions; modelling agent interactions is thus needed and would largely improve forecasting accuracy. When there is an interaction modelling module in the forecasting system, the module will let the prediction of each agent interact with each other, making predictions no longer mutually independent. In this circumstance, the independence assumption held by the previous uncertainty estimation methods will become invalid. As a result, the uncertainty measurements will also become inaccurate, which can lead to safety issues. To tackle this first challenge of uncertainty measurement in the multi-agent setting, we need a more sophisticated and robust measurement to capture the previously neglected uncertainty brought by correlated predictions.

The second challenge (i.e., how to rank multiple predictions) is related to the practical use of forecasting systems. When multiple future trajectories are predicted for an agent, it is helpful for the system to have a ranking strategy that appropriately assigns priorities to the predictions. This ensures the efficiency of the downstream decision-making system to a great extent by informing the system that which predictions should be prioritized during the decision-making procedures. Without a ranking strategy, the downstream system would treat all the predictions equally, which is unideal. Existing works [3], [24], [26], [27], [28] usually adopt a classifier that is trained to assign a higher score to the prediction closer to the ground truth. This strategy, however, may encounter robustness issues during inference because 1) purely data-driven ranking strategy is hard for human to explain; 2) their strategy does not consider the uncertainty of each prediction during the ranking procedure, causing safety concerns. To make the ranking procedure better align with human intuition, an alternative strategy from us, which we detail in this work, is to use the uncertainty estimated for each prediction as the guiding information when assigning the priorities.

Based on the discussions above, we realize that fully addressing the aforementioned two challenges first requires a new uncertainty estimation framework, which better handles the uncertainty analysis when the independence of prediction is not assumed. To this end, we propose a new concept, collaborative uncertainty (CU), to estimate the uncertainty resulting from the use of interaction modules in forecasting systems. We also coin a concept, individual uncertainty (IU), to describe the uncertainty approximated by the predictive variance of a single agent, which is the setting of previous uncertainty estimation methods. We then introduce a novel CU-aware regression framework in Section IV, which describes how to measure CU and IU simultaneously in a general regression task. This framework utilizes the mean $\mu$ and covariance $\Sigma$ of a predictive distribution $p(Y | X)$ to estimate the prediction result and its corresponding uncertainty, where $X$ is the observed trajectory and $Y$ is the ground-truth trajectory. Furthermore, this framework contains a regression model with an original permutation-equivariant uncertainty estimator that learns the values of mean $\mu$ and covariance $\Sigma$.

On the basis of our proposed CU-aware regression framework, we introduce a CU-aware multi-agent multi-modal trajectory forecasting system, which handles the two challenges of the trajectory forecasting task: uncertainty estimation and ranking strategy. The system consists of a novel CU-aware forecasting module and an original CU-based selection module. The CU-aware forecasting module utilizes the regression model of the CU-aware regression framework to learn the mappings that are from input data to i) accurate prediction; ii) individual uncertainty; and iii) collaborative uncertainty. This design leads to more precise uncertainty estimation and prediction in the multi-agent multi-modal trajectory forecasting setting, which aims to solve the first challenge, uncertainty estimation. The CU-based selection module then ranks the multi-modal predictions of each agent according to the uncertainty value estimated by the CU-aware regression framework. It encourages the system to prioritize the predictions with lower uncertainty levels, which better aligns with human intuition, addressing the second challenge. See Fig. 1 for an illustration of uncertainty estimation in forecasting systems.

We conduct extensive experiments to show that: 1) the CU-aware regression framework allows the model to appropriately approximate the ground-truth multivariate Laplace distribution on the synthetic dataset; 2) adding CU estimation benefits accurate predictions; 3) future stochasticity and prediction uncertainty of the multi-agent multi-modal trajectory forecasting system are positively correlated; and 4) CU estimation yields significantly larger performance gains in forecasting systems with
interaction modules (see Fig. 2), confirming that CU is highly related to the usage of the interaction modelling procedure.

The contributions of this work are as follows:

- We propose a novel concept, collaborative uncertainty (CU), and a novel CU-aware regression framework with an original permutation-equivariant uncertainty estimator that models the uncertainty brought by agent-wise interactions.
- We propose a CU-aware multi-agent multi-modal trajectory forecasting system, which leverages collaborative uncertainty to address uncertainty estimation and multi-modal ranking challenges.
- We conduct extensive experiments to validate the CU-aware regression framework on a synthetic dataset and two large-scale real-world datasets.
- On the basis of our proposed CU-aware regression framework, we disclose the positive correlation between the future stochasticity and prediction uncertainty of the multi-agent multi-modal trajectory forecasting system.

A preliminary version of this work is presented in [29]. In comparison, the novelty of the current work is threefold:

- This work leverages the proposed CU-aware regression framework to support the multi-modal trajectory forecasting system. It innovatively designs an uncertainty-based selection module to help the system rank the multiple predicted trajectories of each agent in an easy-to-understand manner. Whereas, our previous work [29] is only compatible with single-modal forecasting systems.
- This work proposes a novel permutation-equivariant uncertainty estimator to estimate collaborative uncertainty, while our previous work [29] cannot guarantee the important property of permutation-equivariance. The uncertainty estimated by the permutation-equivariant uncertainty estimator is independent of the input permutation and is only related to the interactions between agents, which enables this estimator to generate more reasonable results than our previous work [29].
- This work includes new experimental results that demonstrate the competitive performance of our CU-aware forecasting system in multi-agent multi-modal trajectory forecasting. Specially, our CU-based framework helps VectorNet improve by 262 cm regarding the Final Displacement Error (FDE) of the chosen optimal predicted trajectory on the nuScenes dataset. Our previous work [29] only improves VectorNet by 99 cm (FDE) on the same dataset.

The rest of this paper is structured as follows. In Section II, we introduce previous works in the field of aleatoric uncertainty estimation and multi-agent multi-modal trajectory forecasting. In Section III, we formulate the problem of multi-agent multi-modal trajectory forecasting with uncertainty estimation and demonstrate the necessity of modelling collaborative uncertainty for the task. In Section IV, we propose a novel CU-aware regression framework with an original permutation-equivariant uncertainty estimator and show a special case of it based on the multivariate Laplace distribution. In Section V, we apply our proposed CU-aware regression framework to the multi-agent multi-modal trajectory forecasting system to enable such system to better handle both tasks of uncertainty estimation and multi-modal ranking, on the basis of the special case shown in Section IV. In Section VI, we conduct experiments on a synthetic dataset and two real-world datasets to evaluate the effectiveness of the CU-aware regression framework in distribution estimation and trajectory forecasting. In Section VII, we discuss the causes of both uncertainty and collaborative uncertainty in the task of trajectory forecasting. Finally, we conclude this paper in Section VIII. The important notations used throughout the paper are summarized in Table I.

### II. Related Works

**Aleatoric Uncertainty Estimation in Deep Learning:** There are two main types of uncertainty to model in deep-learning-based algorithms [30]: 1) aleatoric uncertainty, regarding information aside from statistic models, which data cannot explain; 2) epistemic uncertainty, the uncertainty inside a model, when the model lacks knowledge of the system/process being modelled (e.g., due to limited training data). One seminal work [18] points out that it is most effective to model aleatoric uncertainty in Big Data regimes such as those common to supervised deep learning. Since trajectory forecasting is a supervised learning task with numerous training data, our work focuses on aleatoric uncertainty. Moreover, similar to many other papers in trajectory forecasting [20], [21], [22], [23], we use the aleatoric uncertainty estimation framework proposed by [18] as our basis, which is
explicitly representing aleatoric uncertainty by predictive variance. In the literature, many existing works [20], [21], [22], [23], [31], [32] follow [18] and formulate uncertainty as learned loss attenuation. For example, to make predictive-variance-based aleatoric uncertainty measurements more efficient, [32] adds data augmentation during the test time. But, these works only pay attention to individual uncertainty. Other recent works attend to the uncertainty measurement for correlated predictive distributions. In supervised learning, [33] and [34] measure spatially correlated uncertainty in a generative model respectively for image reconstruction and pixel-wise classification. In reinforcement learning, [35] captures joint uncertainty as discrete variables in exploration, but the uncertainty from the collaborative modelling [36] is still ignored. Despite [33], [34], [35], our work is the first to conceptualize and measure collaborative uncertainty in the multi-agent trajectory forecasting task. To the best of our knowledge, there are only two papers [27], [37] that are close to our track. [37] considers the joint uncertainty among different body joints in the human pose estimation task, and [27] measures the joint uncertainty between the two prediction dimensions, \( \mathbf{x} \) and \( \mathbf{y} \), for each individual agent in the trajectory prediction task. However, they present several limitations: 1) they did not provide a permutation-equivariant joint uncertainty estimator; 2) they did not provide a theoretical conceptualization or definition of the uncertainty due to correlated predictive distributions; and 3) they did not analyze the causes of such uncertainty. These limitations are essential problems to tackle. Therefore, in this work, we not only formulate a general framework with a permutation-equivariant uncertainty estimator but also theoretically conceptualize collaborative uncertainty and analyze its causes.

**Multi-Agent Multi-Modal Trajectory Forecasting:** The nature of the trajectory forecasting task is that there is often more than one plausible future trajectory. Recently, multi-modal forecasting has become the dominant research setting in the trajectory forecasting research community [3], [13], [14], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48]. This setting requires models to: i) take the observed trajectories from multiple agents and their surrounding environment (e.g., HD maps) as the inputs, and outputs the multiple possible future trajectories predicted for each agent; ii) select the optimal prediction from the predicted trajectories. To tackle the aforementioned requirements, a typical deep-learning-based multi-agent multi-modal trajectory forecasting system usually consists of a regression module that predicts future trajectories and a selection module that ranks the predictions and selects the optimal future trajectory.

In most state-of-the-art (SOTA) forecasting systems, the regression module is designed based on the encoder-decoder architecture. In the encoding process, like many other sequence prediction tasks, the model used to adopt a recurrent architecture to process the inputs [49], [50], [51]. Later, however, the graph neural networks have become a more common approach as they can significantly assist trajectory forecasting by capturing the interactions among agents [3], [5], [15], [16], [18], [38], [39], [40], [41], [42], [52], [53], [54], [55], [56]. For the decoding phase, multi-modal forecasting methods usually use multi-layer perceptrons (MLPs) to decode the hidden features [4], [57], [58], [59]. Many of these methods choose to adopt multiple MLP-based decoders during the decoding phase [3], [14], [60]. Each decoder individually predicts one possible trajectory. To ensure the prediction diversity, instead of optimizing all the decoders, those methods tend to only optimize the one that is closest to the ground truth.

As for the selection module, most SOTA systems are equipped with an MLP-based selector, which assigns the highest confidence score for the prediction that is the closest to the ground truth, to generate a confidence score for each prediction [3], [14], [24], [26], [60], [61], [62], [63], [64].

For safety reasons, it is necessary to report the uncertainty of each predicted trajectory. Works to date about uncertainty measurements [20], [21], [22], [23] have appropriately modeled the interaction among multi-agent trajectories for boosting performances. But they overlook the uncertainty resulting from the correlations in all predicted trajectories, and their selection modules do not take the uncertainty of predictions into consideration. We seek to fill these gaps by introducing collaborative uncertainty (CU) and designing a CU-based selection module.

### III. Problem Formulation

Consider \( m \) agents in a data sample, and let \( \mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m]^T \in \mathbb{R}^{2T \times m} \). \( \mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m]^T \in \mathbb{R}^{2T+\times m} \) be the past observed and the future trajectories of all agents, where \( \mathbf{x}_i \in \mathbb{R}^{2T} \) and \( \mathbf{y}_i \in \mathbb{R}^{2T+} \) are the past observed and the future trajectories of the \( i \)-th agent. Each \( \mathbf{x}_i, \mathbf{y}_i \) consists of two-dimensional coordinates at different timestamps of \( T_0/T_1 \). A multi-modal trajectory forecasting system usually consists of two modules: the forecasting module and the selection module. The forecasting module is used to model the predictive distribution \( p(\mathbf{Y} | \mathbf{X}) \) whose mean \( \mu \in \mathbb{R}^{K \times 2T \times m} \) is the predicted trajectories and covariance \( \Sigma \in \mathbb{R}^{K \times T \times m \times m} \) is the corresponding uncertainty, where \( K \) is the number of prediction modalities. Similar to many previous works [3], [14], [24], we assume that \( p(\mathbf{Y} | \mathbf{X}) \) consists of \( K \times T_1 \) different predictive distributions and \( p(\mathbf{Y}_{k,i} | \mathbf{X}) \) models the prediction modal \( k \) of each timestamp \( t \). Furthermore, we set element \( \mu_{k,i} \) of \( \mu \) as the predicted geometric location in the \( k \)-th prediction modal of the \( i \)-th agent at timestamp \( t \), and \( \Sigma_{k,i} \) contains the individual and collaborative uncertainty of agents in the \( k \)-th prediction modal at timestamp \( t \). More details of \( \Sigma \) will be specified in Subsection IV-A. Moreover, the selection module aims at selecting the optimal predicted trajectory \( \mu^* \in \mathbb{R}^{2T \times m} \) from \( K \) predicted trajectories \( \mu \).

In the field of uncertainty estimation, previous works such as [17], [18], [65] use the individual distribution to approximate \( p(\mathbf{Y} | \mathbf{X}) \). The assumption behind this approach is that \( p(\mathbf{y}_i | \mathbf{x}_i) \) is independent for every \( i \in \{1, 2, 3, \ldots, m\} \). Mathematically, they set the covariance \( \Sigma \) as a diagonal matrix. This assumption is valid for the regression task that uses the model shown in Fig. 2(a). We refer to the uncertainty under the independence assumption as *individual uncertainty* in this paper. However, most of the cutting-edge multi-agent multi-modal trajectory regression models, such as [3], [4], [5], [6], [11], [12], [13], [60], [61], [62], [63], [64],
[14], [15], [16], adopt the collaborative model to model the interactions among multiple agents (see Fig. 2(b) for an illustration of the collaborative model). In the collaborative model, $y_j$ is no longer solely dependent on $x_j$, but also on other agents $x_j$ where $j \neq i$, which turns $p(Y | X)$ from the individual distribution into the joint distribution of multiple agents and results in a new type of uncertainty. We call this type of uncertainty brought by the interactions as collaborative uncertainty (CU).

In the next section, to delineate the complete landscape of uncertainty in multi-agent multi-modal trajectory forecasting, we introduce a unified CU-aware regression framework with an original permutation-equivariant uncertainty estimator for estimating both individual and collaborative uncertainty. Furthermore, we demonstrate a special case of it: Laplace CU-aware regression, which is used in the multi-agent multi-modal trajectory forecasting system.

IV. COLLABORATIVE UNCERTAINTY

In this section, we first introduce a novel CU-aware regression framework with a probabilistic formulation and an original permutation-equivariant uncertainty estimator. Then we demonstrate a special case of our framework: Laplace CU-aware regression. It specifically models uncertainty when data obey the multivariate Laplace distribution, which is a widely held assumption in the multi-agent trajectory forecasting task [3], [4], [16], [29].

A. CU-Aware Regression Framework

The CU-aware regression framework consists of two key components: a probabilistic formulation and a permutation-equivariant uncertainty estimator. The probabilistic formulation leads to a regression model and a loss function, which are used to estimate uncertainty in the regression task. The permutation-equivariant uncertainty estimator enables our framework to generate uncertainty permutation-equivariant with the input data.

1) Probabilistic Framework Overview: Contrary to prior works in uncertainty estimation, to model collaborative uncertainty, we abandon the independence assumption held by previous works [18] and set $p(Y | X)$ as a joint multivariate distribution with a full covariance matrix $\Sigma$. By doing so, we model $p(Y | X)$ more accurately by not setting any restrictions on the form of $\Sigma$. As the diagonal elements of $\Sigma$ are considered individual uncertainty [18], we further let off-diagonal elements describe collaborative uncertainty. Diagonal element $\Sigma_{k,t,i,i}$, which is the variance corresponding to the $\mu_{k,t,i}$, models individual uncertainty in the $k$-th prediction modal of the $i$-th agent at timestamp $t$. Off-diagonal element $\Sigma_{k,t,i,j}$, which is the covariance corresponding to the $\mu_{k,t,i}$ and the $\mu_{k,t,j}$, models collaborative uncertainty in the $k$-th prediction modal between the $i$-th and $j$-th agents at timestamp $t$. In this way, we obtain both individual and collaborative uncertainty by estimating the $\Sigma$ of $p(Y | X)$. Accordingly, we propose a probabilistic CU-aware regression framework with the following steps (see the visualization in Fig. 3).

Step 1: Choose a probability density function for the predictive distribution. The chosen probability density function is formulated as $p(Y | X; \mu, \Sigma, \Phi)$, which includes a mean $\mu \in \mathbb{R}^{K \times 2T \times m}$ used to approximate the future trajectories, a full covariance $\Sigma \in \mathbb{R}^{K \times T \times m \times m}$ used to quantify individual uncertainty and collaborative uncertainty, and some auxiliary parameters $\Phi$ for the predictive distribution.

Step 2: Design a regression model. The regression model is formulated as $F_w(\cdot) = [\mu_w(\cdot), \Phi_w(\cdot), \Sigma_w(\cdot)]$, where $\mu_w(\cdot)$ is a trajectory predictor (T-Predictor) used to approximate the value of mean $\mu$, $\Sigma_w(\cdot)$ is an uncertainty estimator (U-Estimator) estimating the value of covariance $\Sigma$, and $\Phi_w(\cdot)$ is an auxiliary parameter estimator (AP-Estimator) estimating the values of some auxiliary parameters $\Phi$. Here, $w$ only indicates that the parameters of these neural networks are trainable, not that they share the same set of parameters.

Step 3: Derive a loss function. The loss function is derived from the maximum likelihood estimation: $L(w) = -\log p(Y | X; w)$. By minimizing the loss function, we learn the trainable parameters in $F_w(\cdot) = [\mu_w(\cdot), \Phi_w(\cdot), \Sigma_w(\cdot)]$.

To align the permutation of input observed trajectories and predicted trajectories, we implement $\mu_w(\cdot)$ and $\Phi_w(\cdot)$ as two multi-layer perceptrons whose outputs are permutation-equivariant with the input. Further, since we model the individual and collaborative uncertainty via a covariance matrix $\Sigma$, we
To estimate the covariance $\hat{\Sigma}$, which is a positive real number.

Based on $\tau$ and choose the probability density function.

$x_i$ and $x_j$ can be modeled by $\langle e_i', e_j' \rangle$. $\tau$ is a positive real number and $I$ is an identity matrix.

2) **Permutation-Equivalent Uncertainty Estimator:** In this subsection, we illustrate the model structure of the permutation-equivariant uncertainty estimator, $\Sigma_w(\cdot)$. Then, we provide proofs for the permutation equivariance and the positive definiteness of $\Sigma_w(X)$. To emphasize the output of $\Sigma_w(\cdot)$ varies as the input $X$ varies, here, we use $\Sigma_w(X)$ instead of $\hat{\Sigma}$ to represent the output of $\Sigma_w(\cdot)$.

**Model Structure:** The structure of $\Sigma_w(\cdot)$ is illustrated in Fig. 4. Let $E = [e_1, e_2, \ldots, e_m]^T$ represent the input feature, which is permutation-equivariant with the input data $X$, $e_i$ represent the feature generated based on $x_i$, and $f_w(\cdot)$ be a permutation-equivariant neural network (PE-NN). The intuition is that we employ the learning ability of $f_w(\cdot)$ to generate feature $f_w(E) = [e_1', e_2', \ldots, e_m']^T$ based on $E$. $f_w(E)$ is in a feature space where $\langle e_i', e_j' \rangle$ models the individual uncertainty of $y_i$, and where $\langle e_i', e_j' \rangle$ models the collaborative uncertainty of $y_i$ and $y_j$. To make $\Sigma_w(X)$ a positive definite matrix, we generate it by adding a weighted identity matrix to the product of $f_w(E)$ and $f_w'(E)$, which is formulated as:

$$\Sigma_w(X) = f_w(E)f_w'(E) + \tau I,$$

where $\tau$ is a positive real number and $I$ is an identity matrix.

**Theoretical Properties of $\Sigma_w(\cdot)$:** The following two theorems show the uncertainty estimator $\Sigma_w(\cdot)$ promotes permutation-equivariance and positive definite properties.

**Theorem 1:** Given any input data $X$, $\Sigma_w(X)$ is permutation-equivariant; that is,

$$\Sigma_w(PX) = P\Sigma_w(X)P^T,$$

where $P$ is a permutation matrix.

**Proof:** Because the input feature $E$ is permutation-equivariant with the input data $X$, we have:

$$\Sigma_w(PX) = f_w(PE)f_w'(PE) + \tau I.$$  \hspace{1cm} (2)

Moreover, $f_w(\cdot)$ is a permutation-equivariant neural network, $\tau$ is a positive real number and $I$ is an identity matrix, which means

$$f_w(PX) = Pf_w(X), \quad f_w'(PE) = (Pf_w(X))^T \quad \text{and} \quad P\tau P^T = \tau I.$$  \hspace{1cm} (3)

On the basis of (1) and (3), we have:

$$\Sigma_w(PX) = Pf_w(E)f_w'(E)P^T + P\tau P^T.$$  \hspace{1cm} (4)

Thus, $\Sigma_w(X)$ is permutation-equivariant with the input data $X$.

**Theorem 2:** Given any input data $X$, $\Sigma_w(X)$ is a positive definite matrix; that is, $\alpha^T\Sigma_w(X)\alpha \geq 0$ for all $\alpha \in \mathbb{R}^n$; the equality holds if and only if $\alpha = 0$.

**Proof:** Based on (1), we have:

$$\alpha^T\Sigma_w(X)\alpha = \alpha^Tf_w(E)(f_w'(E))^T\alpha + \alpha^T\tau I\alpha$$

$$= \alpha^Tf_w(E)(\alpha^Tf_w'(E))^T + \tau\alpha^T\alpha \geq 0,$$

where the equality holds if and only if $\alpha = 0$. As a result, $\Sigma_w(X)$ is a positive definite matrix.

Compared with our previous work [29], this permutation-equivariant uncertainty estimator benefits uncertainty estimation in 1) making the generated covariance permutation-equivariant with the input data, which is more reasonable for real-world applications; 2) enabling the model to approximate the given distribution more accurately by generating more accurate covariance. See Section VI-A for related empirical results.

### B. Special Case: Laplace CU-Aware Regression

In multi-agent trajectory forecasting, previous methods [3], [4], [16], [29] have found that the $\ell_1$-based loss function derived from the Laplace distribution usually leads to dominant prediction performances, because it is more robust to outliers. Therefore, here we show a special case of the framework proposed in Section IV-A based on the multivariate Laplace distribution: Laplace CU-aware regression.

**Probability Density Function:** We follow the probabilistic formulation proposed in Subsection IV-A and choose the probability density function as the multivariate Laplace distribution:

$$p(Y | X) = \frac{2\det[\Gamma^{-1}]^{\frac{1}{2}}}{(2\pi)^{\frac{n}{2}}\lambda} \frac{K_{\left(\frac{\lambda}{2}\right)}(\sqrt{\frac{1}{2} (Y - \mu)^T \Gamma^{-1} (Y - \mu)})}{(\sqrt{\frac{1}{2} (Y - \mu)^T \Gamma^{-1} (Y - \mu)})^\frac{\lambda}{2}},$$  \hspace{1cm} (4)

where $\Gamma$ is a positive definite matrix, $\Gamma^{-1}$ is the inverse matrix of $\Gamma$, $\det[\Gamma^{-1}]$ denotes the determinant of the matrix $\Gamma^{-1}$, $K_{\left(\frac{\lambda}{2}\right)}(\cdot)$ denotes the modified Bessel function of the second kind with order $\left(\frac{\lambda}{2}\right) - 1$, and $\lambda$ is a positive real number.

According to [66], the covariance $\Sigma$ of the multivariable Laplace distribution is defined as:

$$\Sigma = \lambda \Gamma.$$  \hspace{1cm} (5)

On the basis of (5), for Laplace collaborative uncertainty, generating the covariance $\Sigma$ is equivalent to generating the $\Gamma$ matrix. Therefore, we employ the permutation-equivariant uncertainty estimator $\Sigma_w(\cdot)$ mentioned in Subsection IV-A to generate the $\Gamma$ matrix, and $\hat{\Sigma}$ represents the generated $\Gamma$ matrix.
Model Design: Based on the discussion in Subsection IV-A, we can approximate the value of mean $\mu$ via the T-Predictor $\mu_w(\cdot)$, which is implemented by using a multi-layer perceptron. However, there are still two challenges in designing a model based on (4). First, if we use the neural network $\Sigma_w(\cdot)$ to directly generate $\Sigma \in \mathbb{R}^{K \times T \times m \times m}$, each positive definite matrix $\Sigma_{k,t} \in \mathbb{R}^{m \times m}$ in $\Sigma$ needs to be inverted, which is computationally expensive and numerically unstable. Second, the modified Bessel function is intractable for neural networks to work with [67].

For the first challenge, we make the permutation-equivariant uncertainty estimator $\Sigma_w(\cdot)$ directly generate $\Sigma^{-1}$. For the second challenge, inspired by [66], we simplify (4) by utilizing the multivariate Gaussian distribution to approximate the multivariate Laplace distribution. We reformulate the multivariate Laplace distribution by introducing auxiliary variables. Let $\Phi$ be a random variable from an exponential distribution:

$$p(\Phi | X; w) = \frac{1}{\lambda} e^{-\frac{\Phi}{\lambda}}. \quad (6)$$

Then (4) can be simplified as:

$$p(Y | \Phi, X; w) = \frac{\det[\Sigma^{-1}]^{\frac{1}{2}}}{(2\pi^{\frac{m}{2}})^{\frac{m}{2}}} e^{-\frac{q(Y)}{\Sigma \Phi}},$$

where $q(Y) = (Y - \hat{\mu}) \Sigma^{-1}(Y - \hat{\mu})^T$. Furthermore, in this work, instead of drawing a value for $\Phi$ from the exponential distribution, we use the AP-Estimator $\Phi_w(\cdot)$ implemented by using a multi-layer perceptron to directly output a value $\Phi^*$ for $\Phi$. The intuition is that, in the training process of the regression model, the value of $p(Y | X; w)$ is the conditional expectation of $\Phi$ given $X$ and $Y$, which makes $p(Y | \Phi, X; w)$ a function of $\Phi$ whose domain is $\mathbb{R^+}$. Thus, there should exist an appropriate $\Phi^*$ to make:

$$p(Y | X; w) = p(Y | \Phi^*, X; w).$$

See the proof for the existence of $\Phi^*$ in Appendix A, available online.

After getting the value of $\Phi^*$ via $\Phi_w(\cdot)$, we can get the parameterized form of $p(Y | X; w)$ as:

$$p(Y | X; w) = \frac{\det[\Sigma^{-1}]^{\frac{1}{2}}}{(2\pi^{\frac{m}{2}})^{\frac{m}{2}}} e^{-\frac{q(Y)}{2\Phi^*}}. \quad (7)$$

Moreover, the value of $\hat{\Phi}$ can be used to approximate the value of $\lambda$, because the expected value of the exponentially distributed random variable $\Phi$ is equal to $\lambda$. Therefore, the $\Sigma$ of chosen multivariate Laplace distribution can be approximated as

$$\Sigma \approx \hat{\Phi} \Sigma. \quad (8)$$

Finally, we can get the regression model $\mathcal{F}_w(\cdot) = [\mu_w(\cdot), \Phi_w(\cdot), \Sigma_w(\cdot)]$. Once $\hat{\Sigma}$ and $\hat{\Phi}$ are fixed, we can use them to compute $\Sigma$ as (8) to get individual uncertainty and collaborative uncertainty.

Loss Function. The log-likelihood function of (7) is:

$$L(w) = \frac{1}{2} \frac{q(Y)}{\Phi} + m \log \hat{\Phi} - \log \det[\Sigma^{-1}]. \quad (9)$$

Since $\Sigma^{-1}$ is a positive definite matrix, according to Hadamard’s inequality [68], we have:

$$\log \det[\Sigma^{-1}] \leq \sum_{i=1}^{m} \log d_{ii},$$

where $d_{ii}$ is the diagonal element of $\Sigma^{-1}$.

Therefore, we have a lower bound of (9) as:

$$L_{\text{Lap-cu}}(w) = \frac{1}{2} \left[ \frac{q(Y)}{\Phi} + m \log \hat{\Phi} - \sum_{i=1}^{m} \log d_{ii} \right]. \quad (10)$$

The regression model $\mathcal{F}_w(\cdot) = [\mu_w(\cdot), \Phi_w(\cdot), \Sigma_w(\cdot)]$ can be trained via minimizing this lower bound (10).

In the next section, we will apply our proposed framework to the multi-modal multi-agent forecasting system based on this Laplace CU-aware regression model.

V. CU-AWARE MULTI-AGENT MULTI-MODAL TRAJECTORY FORECASTING SYSTEM

Multi-modal forecasting models are becoming increasingly important to the multi-agent trajectory forecasting task. They allow the system to predict multiple possible future trajectories for each agent, and thus better handle the stochasticity inherent in this task [3], [13], [69]. However, existing multi-agent multimodal trajectory forecasting systems are facing two challenges: 1) how to measure the uncertainty brought by the interaction module; and 2) how to utilize the uncertainty over each prediction to select the optimal prediction. In this section, in order to solve the above two challenges, we apply our proposed framework to the system based on the Laplace CU-aware regression model, which leads to the CU-aware forecasting system (see Fig. 5 for an illustration). In the following, we show the key component designs and the training strategy of this CU-aware forecasting system.

A. Key Components

A multi-agent multi-modal trajectory forecasting system usually consists of a forecasting module that generates multi-modal predictions and a selection module that selects the optimal prediction. Accordingly, for applying our CU-aware regression framework to the multi-agent multi-modal trajectory forecasting system, we design a CU-aware forecasting module and a CU-based selection module.

1) CU-Aware Forecasting Module: To the best of our knowledge, most SOTA systems are equipped with a forecasting module based on the encoder-decoder architecture [3], [4], [5], [38], [39], [40], [41], [42]. Therefore, we also design our CU-aware forecasting module based on such an architecture (see Fig. 5 for the module illustration). In order to capture the interactions among agents in the input data, while aligning the permutation of input data and the generated feature, the trajectory encoder $E_w(\cdot)$ is designed as a permutation-equivariant neural network with an interaction module. To enable the forecasting system to estimate the uncertainty corresponding to the predicted trajectories, we employ the CU-aware regression model $\mathcal{F}_w(\cdot)$ shown in Subsection IV-B as the encoder. The pipeline of the module
is formulated as:

\[ E = F_w(X), \]
\[ \hat{\mu} = \mu_w(E), \]
\[ \hat{\Phi} = \Phi_w(E, \hat{\mu}), \]
\[ \hat{\Sigma}^{-1} = \Sigma_w(E, \hat{\mu}). \]

The trajectory encoder \( F_w(\cdot) \) generates the latent feature \( E \) containing the information about the historical trajectories of each agent. On the basis of the generated feature \( E \), the regression model \( F_w(\cdot) \) forecasts the future trajectories and estimates the corresponding uncertainty following this process: 1) the T-Predictor \( \mu_w(\cdot) \) in the \( F_w(\cdot) \) utilizes \( E \) to generate the multi-modal prediction \( \hat{\mu} \in \mathbb{R}^{K \times \text{dim}_w}; \) 2) the AP-Estimator \( \Phi_w(\cdot) \) and U-Estimator \( \Sigma_w(\cdot) \) within the \( F_w(\cdot) \) generate \( \hat{\Phi} \in \mathbb{R}^{K \times \text{dim}_w \times m} \) and \( \hat{\Sigma}^{-1} \in \mathbb{R}^{K \times \text{dim}_w \times m \times m} \) to estimate the uncertainty of predictions in each prediction modal based on both \( E \) and \( \hat{\mu} \).

This CU-aware forecasting module enables the system to reflect the complete landscape of uncertainty in multi-agent multi-modal trajectory forecasting, which leads to a more accurate approximation for the predictive distribution and makes the system more robust.

2) CU-Based Selection Module: For a system that predicts multiple future trajectories for each agent, it is natural to have a selection module that ranks those predictions according to their confidence levels. For human understanding, an appropriate selection module should provide an intuitive ranking mechanism. In the previous works [3], [14], [60], [61], [62], [63], the selection module is designed as a classifier which trained to generate the highest confidence score for the prediction that is closest to the ground truth. However, this classifier-based ranking strategy has two drawbacks: i) purely data-driven confidence assignment is hard for human to explain; and ii) it overlooks the model uncertainties during the ranking procedure, which may compromise the robustness of the model and cause safety issues.

To overcome the above two flaws, we propose a novel CU-based selection module \( S(\cdot) \), which does not require training. The proposed selection module will assign the highest ranking to the prediction with the lowest uncertainty as estimated by the CU-based framework, which can be formulated as:

\[ \hat{\mu}^* = S(\hat{\mu}, \hat{\Phi}), \]

where \( \hat{\mu} \) is the prediction and \( \hat{\Phi} \) is the corresponding auxiliary parameter, whose value is positively correlated with the uncertainty of \( \hat{\mu}_k \) according to (8). The CU-based trajectory selection module \( S(\cdot) \) selects the \( \hat{\mu}_k \) corresponding to the lowest \( \hat{\Phi}_k \) as the \( \hat{\mu}^* \) for each agent. Using \( \hat{\Phi}_k \) as an indicator for the uncertainty of \( \hat{\mu}_k \) allows us to avoid computing the inverse of \( \hat{\Sigma}_k^{-1} \) matrix which is computationally expensive and numerically unstable. Using our CU-based selection module leads to four gains: 1) making the selection module better align with human intuition by assigning a higher confidence score for the prediction with a lower uncertainty level; 2) enabling the selection module to utilize the uncertainty of each predicted trajectory; 3) avoiding the redundancy of training an extra neural network to generate the confidence score; 4) in Section VI-B3, we empirically show that the CU-based selection module improves the performance of our forecasting system.

Compared with the system seen in our previous work [29], this CU-aware multi-agent multi-modal forecasting system is different in two perspectives: 1) the forecasting module is able to forecast future trajectories and estimate the corresponding uncertainty for multiple prediction modals instead of only one prediction modal; 2) the CU-based selection module allows this system to select the optimal predicted trajectory based on the estimated uncertainty.

B. Training Strategy

The loss function used to train the CU-aware multi-agent multi-modal trajectory forecasting system is formulated as:

\[ L_{\text{total}}(w) = L_{\text{Lap-cu}}(w) + \alpha L_{\text{AUTL}}(w), \quad (12) \]

where \( \alpha \) is a hyperparameter, \( L_{\text{Lap-cu}}(w) \) is (10), and \( L_{\text{AUTL}}(w) \) is the auxiliary uncertainty training loss (AUTL),

\[ L_{\text{AUTL}}(w) = \mathbb{E}[\ln(1 + \text{det} \hat{\Sigma}_k)], \]

Note that, for making the predicted trajectories diverse, in (12) only \( \hat{\mu}^* \) and its corresponding \( \hat{\Phi}_k \) and \( \hat{\Sigma}_k \) are used to compute \( L_{\text{Lap-cu}}(w) \).
which is formulated as:

$$\mathcal{L}_{\text{AUTL}}(w) = \sum_{k=1}^{K} ||\hat{\Phi}_k - D\hat{E}_k||_1,$$

where $D\hat{E}_k = ||\hat{\mu}_k - Y||_2$ is the displacement error of the prediction in the $k$-th modal, $\hat{\Phi}_k$ is the auxiliary parameter used to quantify the uncertainty of prediction in the $k$-th modal, and $Y$ is the ground truth future trajectory.

The intuition behind the AUTL is as follows. The values of the estimated uncertainty and the displacement error (DE) are positively correlated. The greater the corresponding uncertainty and DE are, the less reliable the predicted trajectory will be. Therefore, we are able to train our uncertainty estimator by minimizing the AUTL. Section VI-B shows that the AUTL benefits the multi-modal forecasting system with our CU-aware regression framework, especially in terms of selecting the optimal predicted trajectory.

We can train the system in an end-to-end way with (12), as all of its components are differentiable.

VI. EXPERIMENTS

We first use a self-generated synthetic dataset with a limited number of agents as the toy version of the real-world problem in Section VI-A. We use the simplified dataset to test our method’s ability in capturing the distribution information of the input data that obeys the multivariate Laplace distribution and to evaluate our proposed permutation-equivariant uncertainty estimator. We then conduct extensive experiments on two published benchmarks to prove the value of our proposed method in solving real-world problems in Section VI-B.

A. Toy Problem

We define a toy problem to validate the capability of the proposed framework in accurately estimating the probability distribution, and to evaluate our permutation-equivariant uncertainty estimator. The toy problem requires models to take the mutually correlated trajectories sampled from the given multivariate Laplace distribution as the input, and to output the mean and covariance of this distribution.

As far as we know, in real-world datasets, we only have the ground truth for the predicted trajectory, which is the mean of the distribution, but we do not have access to the ground truth of the uncertainty, which is the covariance matrix of the distribution. Thus, we generate a synthetic dataset with the ground truth for both the mean and the covariance matrix of the given multivariate Laplace distributions.

1) Experimental Setup. Datasets: We generate a synthetic dataset for the quaternary Laplace distribution. This dataset contains training, validation, and test sets, which have 36000, 7000, and 7000 instances respectively. Each instance includes the trajectories of four agents consisting of the two-dimensional point coordinates of the four agents at 50 different timestamps. In each instance, the trajectories of the four agents are sampled from a quaternary joint Laplace distribution. Generation details are provided in Appendix B.1, available online.

| Metric: | $\ell_2$ of $\mu$ | $\ell_1$ of $\Sigma^{-1}$ | $\ell_1$ of $\Sigma$ | KL |
|---------|-----------------|-----------------|-----------------|-----|
| IU Only [18] | 0.434 | 0.534 | 0.235 | 4.33 |
| CU-NPE [29] | 0.376 | 0.502 | 0.209 | 3.15 |
| CU | 0.359 | 0.271 | 0.109 | 2.11 |

CU denotes the framework proposed in this paper, CU-NPE denotes the framework proposed in [29], and IU only denotes the framework proposed in [18]. $\hat{\mu}$: Estimated Mean. $\hat{\Sigma}$: Ground Truth Mean. $\hat{\Sigma}$: Estimated Covariance Matrix. $\Sigma_{gt}$: Ground Truth Covariance Matrix. $\ell_2$ of $\mu$: $||\hat{\mu} - \mu_{gt}||_2$. $\ell_1$ of $\Sigma^{-1}$: $||\Sigma^{-1} - \Sigma_{gt}^{-1}||_1$. $\ell_1$ of $\Sigma$: $||\Sigma - \Sigma_{gt}||_1$. KL: KL divergence $D_{KL}(p_{\hat{\mu}}(X)||p_{\mu}(X))$, where $p_{\mu}(X)$ is the estimated distribution and $p_{\hat{\mu}}(X)$ is the ground truth distribution.

Implementation Details: We use the encoder-decoder network architecture for the toy problem. The neural network outputs the predicted mean and covariance matrix of the given quaternary Laplace distribution. Although the ground truth covariance matrix is known in the synthetic dataset, it is not used in training. We train the neural network with the previous uncertainty estimation methods [18], [29] and our proposed method respectively. Please find more details in Appendix B.2, available online.

Metric: We adopt four metrics for evaluation: 1) the $\ell_2$ distances between the estimated mean and the ground truth mean, 2) the $\ell_1$ distances between the estimated $\Sigma$ matrix and the ground truth $\Sigma$ matrix, 3) the $\ell_1$ distances between the inverse of the estimated $\Sigma$ matrix and the inverse of the ground truth $\Sigma$ matrix, and 4) the KL divergence between the ground truth distribution and the estimated distribution. We provide the metrics computing details in Appendix B.3, available online.

2) Results. Quantitative Results: On the test set of our synthetic dataset, we compare our CU-aware regression framework with the permutation-equivariant uncertainty estimator (CU) and two previous uncertainty estimation frameworks: 1) the framework that only estimates individual uncertainty (IU Only) [18]; 2) the CU-NPE (None Permutation Equivariant) framework proposed in [29], which estimates uncertainty by the square-root-free Cholesky decomposition. Table II shows that CU allows the model to more accurately estimate the mean and covariance matrix of the given distribution, which leads to a lower KL divergence between the ground truth distribution and the estimated distribution.

Qualitative Results: Fig. 6 illustrates the outputs of CU and CU-NPE under two different permutations of the same set of inputs, as well as their corresponding ground truths. These visualizations show that CU consistently outputs a more accurate covariance matrix than CU-NPE. Moreover, the collaborative uncertainty estimated by CU is independent of the permutation of the input to the model and is only related to the interactions between agents. This property aligns with the real-world application requirements, which makes the uncertainty estimated by CU more reasonable than the one approximated by CU-NPE.

B. Real-World Problem

1) Experimental Setup. Datasets: Argoverse [70] and nuScenes [71] are two widely used multi-agent trajectory
Fig. 6. Visual comparisons between outputs of the models and ground truth. (a) and (b) show visualizations of the ground truth and the outputs of the models under two different permutations of the same set of inputs. In Fig. 6(a), the input permutation is: the red agent, the green agent, the blue agent and the purple agent. In Fig. 6(b), the input permutation is: the blue agent, the green agent, the purple agent and the red agent. Outputs of the CU model are consistently more accurate than outputs of the CU-NPE model. Further, the outputs of the CU model are permutation-equivariant with the input data, e.g., the covariance between the blue agent and the purple agent does not change with the permutation of the input.

forecasting benchmarks. Argoverse has over 30000 scenarios collected in Pittsburgh and Miami. Each scenario is a sequence of frames sampled at 10 Hz. The sequences are split as training, validation, and test sets, which have 205942, 39472, and 78143 sequences respectively. nuScenes collects 1000 scenes in Boston and Singapore. Each scene is annotated at 2 Hz and is 20s-long. The prediction instances are split as training, validation, and test sets, which have 32186, 8560, and 9041 instances respectively. For Argoverse, we forecast future trajectories for 3 s based on the observed trajectories of 2 s. For nuScenes, we forecast future trajectories for 6 s based on the observed trajectories of 2 s.

**Metric:** We adopt seven widely used multi-agent trajectory forecasting metrics: \( ADE \), \( FDE \), \( ADE_1 \), \( FDE_1 \), \( ADE_k \), \( FDE_k \) and \( Brier-FDE_k \). Firstly, \( ADE \) is the Average Displacement Error, i.e., the average of point-wise \( \ell_2 \) distances between the prediction and the ground truth. Secondly, \( FDE \) is the Final Displacement Error, i.e., the \( \ell_2 \) distance between the final points of the prediction and the ground truth. Thirdly, \( ADE_1/FDE_1 \) is \( ADE/FDE \) of the selected optimal predicted trajectory Fourthly, \( ADE_k/FDE_k \) is the minimum \( ADE/FDE \) of predicted multi-modal trajectories. Finally, \( Brier-FDE_k \) is the weighted \( FDE_k \). (Metrics are in meters. \( K = 10 \) on nuScenes and \( K = 6 \) on Argoverse.)

**Implementation Details:** The model is implemented following the CU-aware multi-agent multi-modal trajectory forecasting system illustrated in Section V. We implement the encoder of LaneGCN [3] and VectorNet [4], which are two cutting-edge multi-agent trajectory forecasting models, as the trajectory encoder in our proposed model. We use multi-layer perceptrons to implement \( \mu_w(\cdot) \) and \( \Phi_w(\cdot) \). Finally, we implement \( \Sigma_w(\cdot) \) as the permutation-equivariant uncertainty estimator introduced in Section IV-A.

2) Results. Evaluation Results on the Benchmark: For evaluation, we implement our proposed framework based on LaneGCN [3] as it is the SOTA model in multi-agent multi-modal trajectory forecasting. We name the modified LaneGCN that estimates collaborative uncertainty with our proposed CU-aware regression framework as CU-aware LaneGCN. We then compare the performance of CU-aware LaneGCN on the Argoverse trajectory forecasting benchmark with six SOTA methods of this benchmark: AutoBot-Ego [25], GOHOME [26],...
Fig. 7. Visualization of predicted trajectories on the Argoverse validation set. These scenarios depict model outputs with blue representing past trajectory, green representing actual future trajectory, and red representing predicted future trajectories. The color bar indicates the reliability of each prediction: 1st stands for the most reliable prediction, 2nd for the second most reliable prediction, and so on. In all four scenarios, predictions generated by the CU-aware LaneGCN are always closer to the ground truth than predictions generated by LaneGCN.

TABLE III
COMPARISON WITH SOTA METHODS ON THE ARGOVERSE TEST SET.
COLLABORATIVE UNCERTAINTY ESTIMATION BOOSTS PERFORMANCES

| Methods            | ADE₁ | FDE₁ | ADE₅ | FDE₅ | Brier-FDE₅ |
|--------------------|------|------|------|------|------------|
| AutoBot-Ego [25]   | 1.84 | 4.11 | 0.88 | 1.37 | 2.06       |
| GOHOME [26]        | 1.69 | 3.65 | 0.94 | 1.45 | 1.98       |
| DenseTNT [24]      | 1.68 | 3.63 | 0.88 | 1.28 | 1.98       |
| MMTransformer [72] | 1.77 | 4.00 | 0.84 | 1.34 | 2.03       |
| LaneRCNN [14]      | 1.69 | 3.69 | 0.90 | 1.45 | 2.15       |
| LaneGCN [3]        | 1.71 | 3.78 | 0.87 | 1.36 | 2.06       |
| CU-aware LaneGCN   | 1.62 | 3.55 | 0.83 | 1.26 | 1.95       |

DenseTNT [24], MMTransformer [72], LaneRCNN [14], and LaneGCN [3]. Table III demonstrates that the CU-aware LaneGCN especially outperforms all of the competing methods across all chosen metrics.3 Further, Fig. 7c visually compares the outputs of LaneGCN and CU-aware LaneGCN, which shows that CU-aware LaneGCN provides more accurate predictions and ranking results than LaneGCN. Therefore, estimating collaborative uncertainty enhances the SOTA prediction model.

Evaluation Results for Single-Modal Forecasting: To confirm that our proposed permutation equivariant uncertainty estimation framework (CU) does not compromise the prediction accuracy compared to the previous collaborative uncertainty estimation framework (CU-NPE) proposed in [29], we compare their performances in a single-modal setting. We generalize LaneGCN to the single-modal scenario by letting the forecasting module only predict one possible future trajectory for each agent and removing the selection module. Table IV shows that, on the test set of both Argoverse and nuScenes, the single-modal LaneGCN with CU always outperforms it with CU-NPE. We speculate that this improvement may come from the permutation-equivariant uncertainty estimator, which makes the uncertainty estimation more accurate.

Visualization of Collaborative Uncertainty: To visually understand which factor influences the value of CU in multi-agent multi-modal trajectory forecasting, Fig. 8 illustrates the visualization results generated by our model. We visualize three scenarios. Each scenario includes three agents’ trajectories (i.e., orange, blue, and green dash lines) and their corresponding CU values changing over the last 30 frames (i.e., the heatmap, where \( \sigma_{ij} \) denotes the CU of agent \( i \) and agent \( j \)). The visualization results show that the value of CU in multi-agent multi-modal trajectory forecasting is highly related to the interactive information among agents.7

5These results are from the Argoverse leaderboard on 12/12/2022.
6In order to keep figures concise, we only visualize the predictions and corresponding ground truths of one agent in each scenario.
7Note that to keep the visualization results concise, for each scenario, we only visualize the optimal trajectory and its corresponding CU.
Fig. 8. Visualization of CU on the Argoverse validation set. (a) In scenario 1, Agent 0, Agent 1, and Agent 2 are respectively located in completely different areas and heading towards different directions; \( \sigma_{01}, \sigma_{02} \) and \( \sigma_{12} \) are close to zero. (b) In scenario 2, Agent 0, Agent 1, and Agent 2 are driving in the same direction side by side. This type of scenario may generate complicated interactive information making \( \sigma_{01}, \sigma_{02} \) and \( \sigma_{12} \) show a non-monotonic change over time. (c) In scenario 3, Agent 0 first approaches the road on which Agent 1 is driving, and then moves away from that road, which causes \( \sigma_{01} \) to first increase and then decrease. Agent 2 is far away from Agent 0 and Agent 1. Therefore, little new interactive information between Agent 1 and the other two agents would be generated, which makes \( \sigma_{02} \) and \( \sigma_{12} \) decrease over time.

### Table V

Ablation Studies on the Uncertainty Estimation Approach and the Interaction Module (INT.) in Multi-Agent Multi-Modal Trajectory Forecasting

| Method  | Dataset  | Int. | \( \Delta \) ADE\(_1\) | \( \Delta \) FDE\(_1\) | \( \Delta \) ADE\(_k\) | \( \Delta \) FDE\(_k\) |
|---------|----------|------|----------------|----------------|----------------|----------------|
| LaneGCN | Argoverse | \( \checkmark \) | 1.81 1.75 3.3\% | 4.18 4.02 3.8\% | 0.91 0.90 1.1\% | 1.66 1.65 0.6\% |
|         | nuScenes | \( \checkmark \) | 5.53 5.33 3.4\% | 11.71 10.89 7.0\% | 1.98 1.82 8.1\% | 2.90 2.51 13.4\% |
| VectorNet | Argoverse | \( \checkmark \) | 1.80 1.80 0.0\% | 4.11 4.10 0.2\% | 1.52 1.51 0.7\% | 3.35 3.32 0.8\% |
|         | nuScenes | \( \checkmark \) | 4.94 3.80 21.5\% | 11.71 9.09 22.4\% | 1.77 1.70 4.0\% | 3.50 3.22 8.0\% |

IU denotes the approach only estimating individual uncertainty. IU+CU denotes our proposed approach that estimates both individual and collaborative uncertainty. On argoverse and nuScenes validation sets, both LaneGCN and VectorNet with individual and collaborative uncertainty estimation surpass the ones with individual uncertainty estimation only. Collaborative uncertainty estimation has a larger impact on the performance of the model with an interaction module.

### System Analysis

3) **System Analysis:** We study 1) how different approaches of uncertainty estimation would impact the prediction model; 2) how the proposed CU-based selection module would affect the CU-aware multi-modal trajectory forecasting system; 3) how the proposed auxiliary uncertainty training loss (AUTL) would influence the performance of the CU-aware regression framework in multi-modal trajectory forecasting; 4) how the number of agents in the input would impact the performance of the CU-aware prediction system in multi-modal trajectory forecasting. In this part, we adopt LaneGCN/VectorNet as our trajectory encoder for proving that our proposed method can be used as a plug-in module to improve the performance of existing models for multi-agent multi-modal trajectory forecasting. Here, experiments are conducted on the validation sets of the Argoverse and nuScenes benchmarks.

#### Effect of Different Uncertainty Estimation Approaches:

In the multi-agent multi-modal trajectory forecasting task, on the basis of LaneGCN/VectorNet, we consider two approaches of uncertainty estimation: 1) estimating individual uncertainty only (IU); 2) estimating both collaborative and individual uncertainty (IU + CU). Table V demonstrates that estimating both individual and collaborative uncertainty (IU + CU) is consistently better than estimating individual uncertainty only (IU). These results reflect that our collaborative uncertainty estimation (i.e., the CU-aware regression framework) can function as a plugin module to improve the prediction performance of existing models.

#### Effect of the CU-Based Selection Module:

Here, we empirically study how the proposed CU-based selection module influences the CU-aware multi-agent multi-modal trajectory forecasting system. For the model with the classifier-based selection module, we train an extra neural network to generate the confidence score for each predicted trajectory. Table VI shows that the forecasting model with the CU-based selection module always provides more accurate results than the model with the classifier-based selection module, which indicates that our CU-based selection module improves the performance of the CU-aware forecasting model. Especially, compared to the improvements on ADE\(_k\) and FDE\(_k\), CU-based selection module demonstrates more drastic advantages on ADE\(_1\) and FDE\(_1\) metrics. It further proves that CU-based ranking strategy is not only more intuitive but also has better selection capability compared with the previous purely data-driven classifier-based ranking strategy.

#### Effect of the Auxiliary Uncertainty Training Loss (AUTL):

The ablation study results on the proposed AUTL are shown in Table VII. Results show that CU-aware models (CU-aware...
TABLE VI
ABLATION STUDIES FOR THE SELECTION MODULE ON THE VALIDATION SET OF THE ARGOVERSE/NUSCENES DATASETS. WE COMPARED THE PERFORMANCE OF THE EXISTING CLASSIFIER-BASED SELECTION MODULE AND THE PROPOSED CU-BASED SELECTION MODULE. FOR THE CLASSIFIER-BASED SELECTION MODULE, WE TRAIN AN EXTRA NEURAL NETWORK TO GENERATE THE CONFIDENCE SCORE FOR IT. ON BOTH ARGOVERSE AND NUSCENES VALIDATION SETS, CU-AWARE MODELS (CU-AWARE LANEGCN/VECTORNET) WITH THE CU-BASED SELECTION MODULE ALWAYS OUTPERFORM MODELS WITHOUT IT.

| Method       | Dataset     | Selection Module | ADE₁ | FDE₁ | ADEₖ | FDEₖ |
|--------------|-------------|------------------|------|------|------|------|
| CU-aware     | Argoverse   | Classifier       | 1.45 | 3.26 | 0.70 | 1.03 |
| LaneGCN      |             | CU               | 1.28 | 2.78 | 0.69 | 1.01 |
|              |             |                  | 6.22 | 12.64| 1.83 | 2.61 |
|              | nuscenes    | Classifier       | 5.33 | 10.89| 1.82 | 2.51 |
|              |             | CU               |      |      |      |      |
|              |              |                  | 5.33 | 10.89| 1.82 | 2.51 |

TABLE VII
ABLATION STUDIES ON AUXILIARY UNCERTAINTY TRAINING LOSS (AULT) IN CU-AWARE MULTI-MODAL TRAJECTORY FORECASTING MODELS (CU-AWARE LANEGCN/VECTORNET)

| Method       | Dataset     | Metric          | AULT  | √    | Δ    |
|--------------|-------------|-----------------|-------|------|------|
| CU-aware     | Argoverse   | ADEₖ            | 0.70  | 0.69 | 2.6% |
|              |             | ADE₁            | 1.42  | 1.28 | 9.6% |
|              |             | FDEₖ            | 3.14  | 2.78 | 11.4%|
|              |              |                 | 1.56  | 1.01 | 5.1% |
|              |              | ADEₖ            | 1.84  | 1.82 | 1.4% |
|              |              | ADE₁            | 11.40 | 5.33 | 53.3%|
|              |              | FDEₖ            | 2.57  | 2.31 | 2.2% |
|              |              |                 | 23.39 | 10.89| 53.4%|
|              | nuscenes    | ADEₖ            | 1.30  | 1.27 | 3.0% |
|              |              | ADE₁            | 2.40  | 1.50 | 37.4%|
|              |              | FDEₖ            | 2.75  | 2.66 | 3.3% |
|              |              |                 | 5.50  | 3.31 | 39.8%|
|              |              | ADEₖ            | 1.79  | 1.70 | 3.3% |
|              |              | ADE₁            | 4.86  | 3.80 | 118.1%|
|              |              | FDEₖ            | 3.55  | 3.22 | 9.3% |
|              |              |                 | 11.79 | 9.09 | 22.9%|

Δ Represents the improvement that AUTL brings to the forecasting model. On both Argoverse and nuscenes validation sets, AUTL benefits the prediction model, and it impacts ADE₁/FDE₁ significantly, more so than it impacts ADEₖ/FDEₖ.

LaneGCN/VECTORNET) trained with AUTL always outperform CU-aware models trained without it on all chosen metrics and that AUTL makes a larger improvement in ADE₁/FDE₁ than in ADEₖ/FDEₖ. These results show proof that the proposed AUTL benefits CU-aware multi-agent multi-modal trajectory forecasting models, especially in terms of choosing the optimal predicted trajectory.

Effect of the Number of Agents: Based on LaneGCN/VECTORNET, we study the effectiveness of the CU-Aware prediction system for samples with different agent numbers on both the Argoverse dataset and the nuscenes dataset. We use the improvement rate to compare the performance gain on samples that contain different numbers of agents. The improvement rate of a certain metric ΔM is defined as:

$$\Delta M = \frac{M_{\text{base}} - M_{\text{CU}}}{M_{\text{base}}} \times 100\%,$$  \hspace{1cm} (13)

where $\mathcal{M} \in \{\text{ADE}_1, \text{FDE}_1, \text{FDE}_k, \text{FDE}_k\}$, $M_{\text{base}}$ is the metric value from the baseline model and $M_{\text{CU}}$ is the metric value from the CU-aware model.

As indicated in Fig. 9, the performance of the CU-Aware LaneGCN/VECTORNET is consistently better than LaneGCN/VECTORNET no matter how many agents are in the sample. On the validation set of Argoverse/nuscenes 37%/9% samples contain between 0 and 10 agents, 40%/16% samples contain between 10 and 20 agents, 19%/15% samples contain between 20 and 30 agents, and 4%/60% samples contain more than 30 agents.

VII. DISCUSSION: WHAT CAUSES THE UNCERTAINTY IN MULTI-AGENT MULTI-MODAL TRAJECTORY FORECASTING

In Section VI, our experimental results show that the CU-aware regression framework can be used to aid models in approximating the distribution information of given data and in improving prediction performance. In this section, we leverage the proposed CU-aware regression framework to study the cause of the uncertainty in multi-agent multi-modal trajectory forecasting, particularly the cause of our proposed collaborative uncertainty.

A. Cause of Uncertainty

As mentioned in Sections II and IV, potential future trajectories of multiple agents are inherently stochastic. We argue that future stochasticity contributes to the uncertainty in multi-agent multi-modal trajectory forecasting.

To support this argument, we empirically study the relationship between future stochasticity and the prediction uncertainty (i.e., the stochasticity-uncertainty relationship) of multi-agent multi-modal trajectory forecasting models. In our experiments, the prediction uncertainty is quantified by our proposed Laplace...
CU-aware regression framework. For the scale of future stochasticity, according to [73] and [74], the prediction diversity of a forecasting model is positively correlated with the stochasticity of given agents’ potential future trajectories. Therefore, we use the prediction diversity of multi-agent multi-modal trajectory forecasting systems to estimate the future stochasticity scale, which can be quantified by (14).

\[
\text{Stochasticity} = m \left( \frac{\sum_{i=1}^{K} (\mu_i - \hat{\mu}_i)^2}{K-1} \right),
\]

where \( \hat{\mu} = [\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_K] \) is a multi-modal prediction result, \( K \) is the number of the prediction modal, \( \hat{\mu}_k \in \mathbb{R}^{2T^+} \), \( \hat{\mu} \in \mathbb{R}^{2T^+} \) is the mean of elements in \( \hat{\mu} \), and \( m(\cdot) \) symbolizes the computation of element-wise average of a vector.

We visually analyze the stochasticity-uncertainty relationship for LaneGCN as well as VectorNet on the Argoverse and nuScenes datasets. See visualization results in Fig. 10. These results show a positive correlation between future stochasticity and prediction uncertainty in multi-agent multi-modal trajectory forecasting models. Therefore, the stochasticity of given agents’ potential future trajectories contributes to the existence of uncertainty in multi-agent multi-modal trajectory forecasting.

**B. Cause of Collaborative Uncertainty**

In this subsection, we focus on the cause of our proposed collaborative uncertainty. As mentioned in Section III, we can divide multi-agent multi-modal trajectory forecasting models into two types: the individual model and the collaborative model. The individual model predicts the future trajectory and the corresponding uncertainty for each agent independently, while the collaborative model leverages an interaction module to explicitly capture the interactions among multiple agents, which makes all the predicted trajectories correlated. The correlations among predicted trajectories can bring extra uncertainty to the model; in other words, we consider that the interaction module in a forecasting model leads to collaborative uncertainty.

To validate this, we empirically compare the impact of collaborative uncertainty estimation on an individual model versus a collaborative model. In our experiments, we use two cutting-edge multi-agent trajectory forecasting models, LaneGCN [3] and VectorNet [4]. Furthermore, the improvement rate of \( ADE_1, FDE_1, ADE_k, \) and \( FDE_k \) between models with/without an interaction module are represented by \( \Delta ADE_1, \Delta FDE_1, \Delta ADE_k, \) and \( \Delta FDE_k \) respectively. The definition of the improvement rate is shown in (13). As Fig. 11 illustrates, when we remove the interaction module from LaneGCN/VectorNet, collaborative uncertainty estimation brings much less gain to LaneGCN/VectorNet (see Table V for more results of this experiment). As a result, the interaction module causes the collaborative uncertainty in multi-agent multi-modal trajectory forecasting.

**VIII. CONCLUSION**

This work proposes a novel collaborative uncertainty (CU) aware regression framework for multi-agent multi-modal trajectory forecasting, which contains a novel permutation-equivariant uncertainty estimator. The framework’s key novelty is twofold. Firstly, it conceptualizes and models the collaborative uncertainty introduced by interaction modules. Secondly, it enables the multi-agent multi-modal trajectory forecasting system to rank multi-modal predictions based on the uncertainty over each prediction. Results of extensive experiments demonstrate the ability of this CU-aware regression framework in boosting the performance of SOTA forecasting systems, especially collaborative-model-based systems. With further experiments, we reveal a positive correlation between future stochasticity and prediction uncertainty in multi-agent multi-modal trajectory forecasting systems. We believe that this work, as well as the framework it presents, will guide the development of more reliable and safer forecasting systems in the near future. Furthermore, we also think that the collaborative uncertainty estimation method proposed in our work could also be applied to other machine learning tasks that care about the interaction between different agents, for example, multi-agent reinforcement learning systems using the centralised training and decentralised execution paradigm.
REFERENCES

[1] P. Stahl, B. Donmez, and G. A. Jamieson, “Anticipation in driving: The role of experience in the efficacy of pre-event conflict cues,” IEEE Trans. Hum. Mach. Syst., vol. 44, no. 5, pp. 603–613, Oct. 2014.

[2] S. Patrick, D. Birsen, and G. A. Jamieson, “Supporting anticipation in driving through attentional and interpretational in-vehicle displays,” Accident Anal. Prevention, vol. 91, pp. 103–113, 2016.

[3] M. Liang et al., “Learning lane graph representations for motion forecasting,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 541–556.

[4] J. Gao et al., “VectorNet: Encoding HD maps and agent dynamics from vectorized representation,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit., 2020, pp. 11522–11530.

[5] M. Ye, T. Cao, and Q. Chen, “TPCN: Temporal point cloud networks for motion forecasting,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit., 2021, pp. 11 318–11 327.

[6] T. Gilles, S. Sabatinari, D. Tsishkou, B. Stanciulescu, and F. Moutarde, “HOME: Heatmap output for future motion estimation,” 2021, arXiv:2105.10968.

[7] K. Xiao, J. Zhao, Y. He, and S. Yu, “Trajectory prediction of UAV in smart city using recurrent neural networks,” in Proc. IEEE Int. Conf. Commun., Shanghai, China, 2019, pp. 1–6.

[8] N. Jetchew and M. Toussaint, “Trajectory prediction: Learning to map situations to robot trajectories,” in Proc. 26th Annu. Int. Conf. Mach. Learn., Montréal, Québec, Canada, 2009, pp. 449–456.

[9] C. Rösmann, M. Oeljeklaus, F. Hoffmann, and T. Bertram, “Online trajectory prediction and planning for social robot navigation,” in Proc. IEEE Int. Conf. Adv. Intell. Mechatronics, Munich, Germany, 2017, pp. 1255–1260.

[10] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, “Symbolic graph neural networks for 3D skeleton-based human action recognition and motion prediction,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 6, pp. 3316–3333, Jun. 2022.

[11] T. Zhao et al., “Multi-agent tensor fusion for contextual trajectory prediction,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit., 2019, pp. 12118–12126.

[12] C. Choi and B. Dariush, “Looking to relations for future trajectory forecast,” in Proc. IEEE/CONF Int. Conf. Comput. Vis., 2019, pp. 921–930.

[13] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, “Trajectron: Dynamically-feasible trajectory forecasting with heterogeneous data,” 2020, arXiv:2001.03093.

[14] W. Zeng, M. Liang, R. Liao, and R. Urtsun, “LaneRCNN: Distributed representations for graph-centric motion forecasting,” 2021, arXiv:2101.06653.

[15] J. Li, F. Yang, M. Tomizuka, and C. Choi, “EvolveGraph: Multi-agent trajectory prediction with dynamic relational reasoning,” in Proc. Neural Inf. Process. Syst. 2020, pp. 19783–19794.

[16] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. D. Reid, H. Rezatofighi, and W. Zeng, M. Liang, R. Liao, and R. Urtasun, “Gohome: Graph-oriented heatmap output for future motion estimation,” in Proc. Int. Conf. Robot. Automat., 2022, pp. 9107–9114.

[17] R. Girigi et al., “Latent variable sequential set transformers for joint multi-agent motion prediction,” in Proc. Int. Conf. Learn. Representations, 2021.

[18] T. Gilles, S. Sabatinari, D. Tsishkou, B. Stanciulescu, and F. Moutarde, “Gohome: Graph-oriented heatmap output for future motion estimation,” in Proc. Int. Conf. Robot. Automat., 2022, pp. 9107–9114.

[19] J. Ngiam et al., “Scene transformer: A unified multi-task model for behavior prediction and planning,” 2021, arXiv:2106.08417.

[20] B. Tang, Y. Zhong, U. Neumann, G. Wang, S. Chen, and Y. Zhang, “Collaborative uncertainty in multi-agent trajectory forecasting,” in Proc. Adv. Neural Inf. Process. Syst., 2021, pp. 6328–6340.

[21] A. Der Kiureghian and O. Ditlevsen, “Aleatory or epistemic? Does it matter?” Struct. Saf., vol. 31, no. 2, pp. 105–112, 2009.

[22] A. Kendall, Y. Gal, and R. Cipolla, “Multi-task learning using uncertainty to weigh losses for scene geometry and semantics,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Salt Lake City, UT, USA, 2018, pp. 7482–7491.

[23] M. S. Ayhan and P. Berens, “Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks,” in Proc. Int. Conf. Med. Imag. Deep Learn., 2018.

[24] G. Dorta, S. Vicente, L. Agapito, N. D. F. Campbell, and I. Simpson, “Structured uncertainty prediction networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 5477–5485.

[25] A. Smilkov et al., “Stochastic segmentation networks: Modelling spatially correlated aleatoric uncertainty,” in Proc. Annu. Conf. Neural Inf. Process. Syst., 2020, pp. 12756–12767.

[26] D. Hafner, T. Lillincrap, M. Norouzi, and J. Ba, “Mastering atari with discrete world models,” 2020, arXiv:2010.02193.

[27] W. Böhm, V. Kurin, and S. Whiteson, “Deep coordination graphs,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 980–991.

[28] N. B. Gundavaru, D. Srivastava, R. Mitra, A. Sharma, and A. Jain, “Structured aleatoric uncertainty in human pose estimation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Long Beach, CA, USA, 2019, pp. 50–53.

[29] C. Xu, W. Mao, W. Zhang, and S. Chen, “Remember intentions: Retrospective-memory-based trajectory prediction,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit., 2022, pp. 6478–6487.

[30] C. Xu, M. Li, Z. Ni, Y. Zhang, and S. Chen, “GroupNet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit., 2022, pp. 6488–6497.

[31] I. Bae, J.-H. Park, and H.-G. Jeon, “Non-probability sampling network for stochastic hidden-hexbin trajectory prediction,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit., 2022, pp. 6467–6477.

[32] L. Zhou, D. Yang, X. Zhai, S. Wu, Z. Hu, and J. Liu, “GA-STT: Human trajectory prediction with group aware spatial-temporal transformer,” IEEE Robot. Automat. Lett., vol. 7, no. 3, pp. 7660–7667, Jul. 2022.

[33] T. Gu et al., “Stochastic trajectory prediction via motion indeterminacy diffusion,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit. Workshops, 2022, pp. 17 113–17 122.

[34] N. Deo, E. Wolff, and O. Beijbom, “Multimodal trajectory prediction conditioned on lane-graph traversals,” in Proc. Conf. Robot Learn., 2022, pp. 203–212.

[35] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov, “Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction,” 2019, arXiv:1910.05449.

[36] P. Dendorfer, S. Elflein, and L. Leal-Taixé, “MG-GAN: A multi-generator transformer for multi-agent motion prediction,” in Proc. IEEE/CONF Comput. Vis. Pattern Recognit., 2022, pp. 8823–8833.

[37] C. Xu, Y. Wei, B. Tang, S. Yin, Y. Zhang, and S. Chen, “Dynamic-group-aware networks for multi-agent trajectory prediction with relational reasoning,” 2022, arXiv:2206.13114.

[38] Y. Zhong, Z. Ni, S. Chen, and U. Neumann, “Aware of the history: Trajectory forecasting with the local behavior data,” in Proc. 17th Eur. Conf. Comput. Vis., Tel Aviv, Israel, 2022, pp. 393–409.
Yiqi Zhong received the bachelor’s degree from Dalian Maritime University, China, in 2016, and the MS degree in computer science from the University of Southern California, in 2018. She is currently working toward the PhD degree in computer science with the University of Southern California. Affiliated to the USC Computer Graphics and Immersive Technologies (CGIT) laboratory. Her research interests include focuses on computer vision, multimodal learning, and multi-agent trajectory prediction.

Wei-Tao Wu received the BS degree from Xi’an Jiaotong University, in 2011, and the PhD degree from the Department of Mechanical Engineering of Carnegie Mellon University, in 2015. He is a professor with the School of Mechanical Engineering, Nanjing University of Science and Technology. Before joining Nanjing University of Science and Technology, he was a postdoc research associate with the Department of Biomedical Engineering, Carnegie Mellon University, in 2015. His research interests include fluid mechanics, heat transfer, and physics-informed machine learning. Within recent 5 years, he has authored or co-authored more than 50 peer-reviewed journal papers, including 3 journal cover/featured papers. He has also served as a member of Physics and Aerodynamics Committee of Chinese Aerodynamics Research Society.

Bohan Tang (Student Member, IEEE) received the BE degree in information engineering from Shanghai Jiao Tong University, Shanghai, China, in 2018. He is currently working toward the PhD degree with the Department of Engineering Science, University of Oxford since 2021. His research interests include graph machine learning, hypergraph machine learning, and uncertainty estimation.

Chenxin Xu received the BE degree in information engineering from Shanghai Jiao Tong University, Shanghai, China, in 2019. He is currently working toward the joint PhD degree with the Cooperative Mediation Innovation Center, Shanghai Jiao Tong University, and in electrical and computer engineering with the National University of Singapore since 2019. His research interests include computer vision, machine learning, graph neural network, and multi-agent prediction.

Wei-Bing Ding received the BE degree in information engineering from Shanghai Jiao Tong University, Shanghai, China, in 2016. He is currently working toward the PhD degree with the Department of Electrical and Computer Engineering, The University of Texas at Austin and the Department of Engineering Science, University of Oxford since 2021. His research interests include fluid mechanics, heat transfer, and physics-informed machine learning.
Ulrich Neumann (Member, IEEE) received the MSEE degree from the State University of New York at Buffalo, and the PhD degree in computer science from the University of North Carolina at Chapel Hill. He is a professor with the Computer Science Department, University of Southern California with a joint appointment in the Ming Hsieh Department of Electrical and Computer Engineering. His research interests include the intersection of computer vision and computer graphics.

Ya Zhang (Member, IEEE) received the bachelor’s degree from Tsinghua University, in China, and the PhD degree in information sciences and technology from Pennsylvania State University. She is currently a professor with the Cooperative Medianet Innovation Center, Shanghai Jiao Tong University. Her research interest includes mainly in machine learning with applications to multimedia and healthcare. Before joining Shanghai Jiao Tong University, she was a research manager with Yahoo! Labs, where she led an R&D team of researchers with strong backgrounds in data mining and machine learning to improve the web search quality of Yahoo international markets. Prior to joining Yahoo, she was an assistant professor with the University of Kansas with a research focus on machine learning applications in bioinformatics and information retrieval. She has published more than 150 refereed papers in prestigious international conferences and journals, including IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Neural Networks and Learning Systems, ICDM, CVPR, ICCV, ECCV, and ECML. She currently holds 5 US patents and 4 Chinese patents and has 9 pending patents in the areas of multimedia analysis. She was appointed the chief expert for the ‘Research of Key Technologies and Demonstration for Digital Media Self-organizing’ project under the 863 program by the Ministry of Science and Technology of China.

Siheng Chen (Member, IEEE) received the two master’s degrees in electrical and computer engineering from the College of Engineering and machine learning from the School of Computer Science, and the PhD degree in electrical and computer engineering from Carnegie Mellon University. He is a tenure-track associate professor with Shanghai Jiao Tong University. Before joining Shanghai Jiao Tong University, he was a research scientist with Mitsubishi Electric Research Laboratories (MERL), and an autonomy engineer with Uber Advanced Technologies Group (ATG), working on the perception and prediction systems of self-driving cars. Before joining industry, he was a postdoctoral research associate with Carnegie Mellon University. His work on sampling theory of graph data received the 2018 IEEE Signal Processing Society Young Author Best Paper Award. His co-authored paper on structural health monitoring received ASME SHM/NDE 2020 Best Journal Paper Runner-Up Award and another paper on 3D point cloud processing received the Best Student Paper Award at 2018 IEEE Global Conference on Signal and Information Processing. He contributed to the project of scene-aware interaction, winning MERL President’s Award. His research interests include graph signal processing, autonomous driving, and collective intelligence.

Yanfeng Wang received the BE degree in information engineering from the University of PLA, Beijing, China, and the MS and PhD degrees in business management from the Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai, China. He is currently the vice director with the Cooperative Medianet Innovation Center and also the vice dean with the School of Electrical and Information Engineering, Shanghai Jiao Tong University. His research interests mainly include media Big Data and emerging commercial applications of information technology.