arXiv:2608.03521v1 [cs.RO] 4 Aug 2026

Pivot-Centric Trajectory Prediction: Bridging Long Horizons via Dynamical Guidance

Xiucong Zhao[1] , **Jindong Tian**[2] , **Hao Miao**[3*]
[1]Xi'an Jiaotong University
[2]East China Normal University
[3]Hong Kong Polytechnic University
zhaoxiucong@stu.xjtu.edu.cn, jdtian@stu.ecnu.edu.cn, hao.miao@polyu.edu.hk

Abstract

Forecasting precise future motion of surrounding agents is essential for reliable autonomous vehicles. However, as the demand for longer prediction horizons increases, existing endpoint-completion or iterative-refine methods increasingly struggle with weak guidance and compounding errors. To tackle the long-horizon prediction challenge, we propose Pivot-Centric Trajectory Prediction (PCTP). By introducing "pivots" and focusing on predicting pivot points along extended trajectories, we divide the long-term prediction task into short-term sub-tasks at various scales. Specifically, PCTP decouples the long-term trajectory predicting process into two processes: pivot prediction and pivot-based trajectory refinement. The pivot prediction process aims to utilize global map context and agent-to-agent interactions to identify these "pivot points", while the pivot-based trajectory refinement process focuses on local map details and refines the short-term trajectory based on predicted "pivot points". Compared with existing methods, PCTP provides more intermediate guidance while reducing compounding errors. Moreover, PCTP is a flexible approach that can be integrated into most state-of-the-art trajectory prediction models. Experimental results show that PCTP improves the prediction accuracy of leading models on both Argoverse I and Argoverse II datasets with minimal impact on model size. Specifically, PCTP combined with QCNet outperforms all published ensemble-free methods on the Argoverse II leaderboard at submission. The code is available at: https://github.com/zxclss/PCNet.

1 Introduction

Spatio-temporal prediction is essential in many domains [Mei *et al.*, 2025; Tian *et al.*, 2026; Xu *et al.*, 2026; Miao *et al.*, 2025b]. In autonomous driving systems, accurately predicting the long-term future trajectory of surrounding agents is of crucial importance to capture others' intentions and make trustworthy decisions [Hagedorn *et al.*, 2023; Wang *et al.*, 2025; Miao *et al.*, 2025a; Liu *et al.*, 2026]. However, as the forecast horizon extends, the behavior uncertainty and ambiguity increase significantly, from an expanding search space, compounding errors, and the complex interaction among heterogeneous information, including dynamic agent history and static map features [Hu *et al.*, 2023; Tolstaya *et al.*, 2021; Salzmann *et al.*, 2020; Ngiam *et al.*, 2021; Zhou *et al.*, 2022].

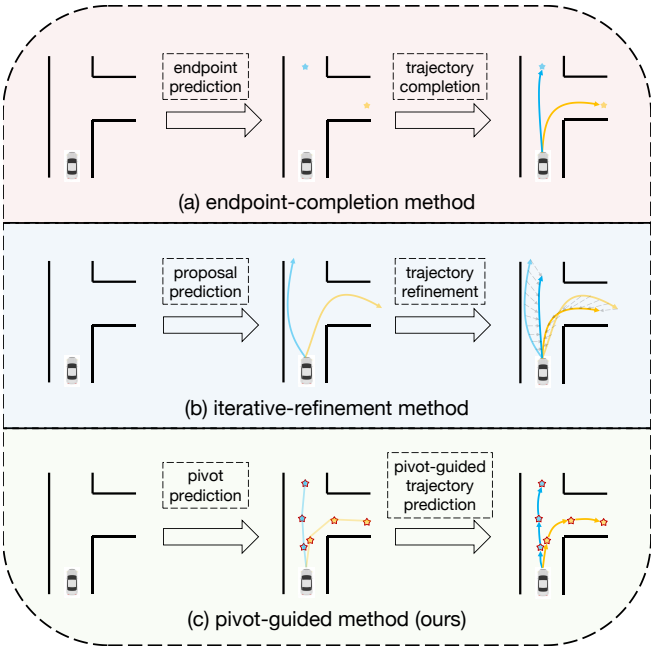

Figure 1: Illustration of various trajectory prediction methods: (a) The endpoint-completion method utilizes a single endpoint as the bridge, which may suffer from weak guidance. (b) The iterative-refinement method treats the entire trajectory as the bridge, leading to the accumulation of compounding errors. (c) Our pivot-guided trajectory prediction method provides sufficient global guidance while simultaneously minimizing compounding errors.

To ease the long-term prediction uncertainty, the research communities have mainly focused on endpoint-completion [Gu *et al.*, 2021; Shi *et al.*, 2022; Cui *et al.*, 2023; Ye *et al.*, 2023; Aydemir *et al.*, 2023] and iterative-refinement methods [Chai *et al.*, 2019; Zhou *et al.*, 2023; Jiang *et al.*, 2023; Liu *et al.*, 2024; Zhou *et al.*, 2024], where the for-

*Corresponding author

mer focuses on utilizing prior-knowledge-based endpoint to guide the trajectory prediction for the purpose of reducing the search space, while the latter prefer to refine trajectory proposals iteratively to utilize detailed context information. (See *Fig*. 1 for more details).

However, the guidance provided by a simple anchor decreases significantly as the horizon length increases for prior-knowledge-based methods. For example, in a scenario where an agent's goal is to bypass an obstacle in front over the next six seconds, using only one endpoint behind the obstacle can cause ambiguity between left and right bypass behaviors. In contrast, iterative-refinement-based methods can avoid this ambiguity because the proposed trajectory predicted in the first stage provides strong guidance—however, the quality of proposals matters [Zhou *et al.*, 2023]. A wrong proposal would lead to further incorrect context for the fine-grained final predictions.

Based on the insight into how humans interact with other traffic participants, we propose a new trajectory decoding scheme called Pivot-Centric Trajectory Prediction (PCTP), trying to combine long-term intention prediction and short-term fine-grained action learning. For this purpose, we propose the concept of *pivots*, which stands for the key points in a trajectory. PCTP consists of the process of Pivot prediction and pivot-guided trajectory prediction. In the first stage, the model predicts several pivots, which represent the long-term goal, considering agent-to-agent interaction and cause map information. In the second stage, the model infers the fine-grained waypoints based on the controlling pivot point to further capture the local kinematics. Moreover, the first stage can be divided into hierarchical pivot prediction processes, where each level focuses on different time scales.

Compared to the single endpoint-completion method, PCTP offers more comprehensive intermediate guidance. For instance, these intermediate pivots provide clearer indications on when and where to make a right turn at an intersection, and on when and where to interact with other agents. Compared to the trajectory-refinement method, PCTP offers a much less intermediate process such that the cumulated noise and compounding errors could be minimized. The key insight is that high-quality pivots are easier to learn, even over longer horizons, due to the reduced search space compared to an entire trajectory. In PCTP, the factorization of pivot prediction and pivot-based trajectory refinement not only preserves the diversity and quality of anchors but also ensures that time-adjacent waypoints attend to the same local information. It is also worth noting that PCTP only changes the decoding strategy and does not introduce additional input, making it easy to integrate into most state-of-the-art trajectory prediction models.

In summary, the key contributions of this work are as follows:

- We introduce Pivot-Centric Trajectory Evolution (PCTP), a method that infers trajectory based on predicted pivots, which factorizes global and local attention. Pivot prediction models intention through global interaction, while pivot-guided trajectory prediction focuses on local interaction.
- We propose a trajectory decoding framework based on PCTP, which can be integrated into any trajectory prediction model as a plugin.
- We evaluate the proposed method on the Argoverse I and Argoverse II datasets. Our method combined with QCNet outperforms all published ensemble-free works on the Argoverse II leaderboard.

2 Related Work

Trajectory prediction involves taking scene representations, including surrounding agents' histories and road maps, as inputs to forecast the future motion of agents. There are two major learning-based approaches: dense representation and sparse representation. Dense representation methods [Chai *et al.*, 2019; Phan-Minh *et al.*, 2020] use fixed-resolution grid structures to encode heterogeneous data and aggregate them using Convolutional Neural Networks, which focus on local information rather than global interactions. In contrast, sparse representation methods aggregate vector-based inputs using permutation-invariant set operators [Gao *et al.*, 2020; Varadarajan *et al.*, 2022], graph convolutions [Liang *et al.*, 2020; Zeng *et al.*, 2021; Da and Zhang, 2022; Liu *et al.*, 2024; Liao *et al.*, 2024], and attention mechanisms [Ngiam *et al.*, 2021; Zhou *et al.*, 2022; Nayakanti *et al.*, 2023; Jiang *et al.*, 2023; Philion *et al.*, 2024], allowing for effective global interactions. To address the challenge of long-term prediction uncertainty, recent work has adopted a two-stage scheme that factorizes the task into intention prediction and intention-based trajectory prediction.

Endpoint-based Completion. These methods assume that a trajectory is largely defined by its endpoint. Endpoint can be classified from a pre-defined set [Shi *et al.*, 2022; Cui *et al.*, 2023]. TNT. [Zhao *et al.*, 2021] samples possible endpoints on centerlines in a dynamic scene and then predicts offsets from candidates to construct a discrete set for classification. DenseTNT. [Gu *et al.*, 2021] learns an endpoint distribution heatmap during training. MTR. [Shi *et al.*, 2022] clusters a set of endpoints in the dataset using K-Means before training, then uses these endpoints to query encoded features. Additionally, some methods [Gilles *et al.*, 2021; Gilles *et al.*, 2022] regress the endpoints using an anchor-free scheme.

Trajectory-based Refinement. These methods [Chai *et al.*, 2019; Jiang *et al.*, 2023; Liu *et al.*, 2024] combine refinement networks [Carion *et al.*, 2020] from computer vision with trajectory prediction. They take a proposed trajectory generated in the first stage as input and predict the offset of each waypoint in the proposed trajectory. DCMS [Ye *et al.*, 2022] adds temporal and spatial constraints to the refinement network. QCNet. [Zhou *et al.*, 2023] embeds the proposed trajectory into a Fourier feature to query refinement-related features. SmartRefine. [Zhou *et al.*, 2024] uses an adaptive network to dynamically adjust refinement configurations and the number of iterations.

3 Methodology

PCTP is a two-stage trajectory decoding pipeline designed to alleviate the compounding error of long-term trajectory prediction, in which we follow a long-term pivot prediction and

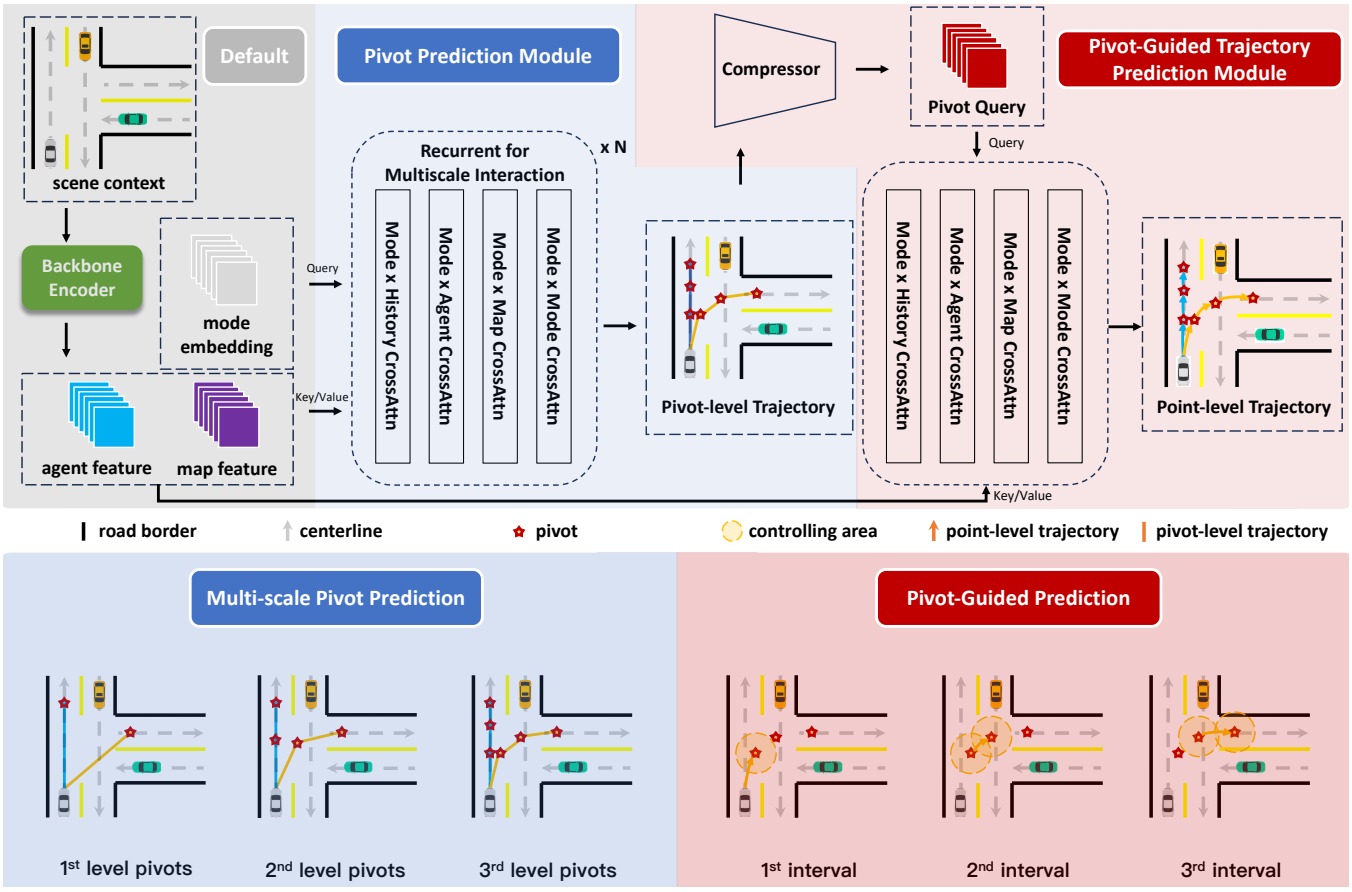

Figure 2: Overview of PCTP. The top section illustrates the overall pipeline, while the lower section details our Multi-Scale Pivot Prediction and Pivot-Guided Trajectory Prediction. First, the backbone encoder processes the HD map and agent-to-agent interaction information into a unified feature space. Next, the Pivot Prediction Module hierarchically predicts the positions of pivot points while simultaneously reducing the uncertainty associated with each pivot. Finally, the Pivot-Guided Trajectory Prediction Module decodes trajectories with the guidance of local pivots.

short-term trajectory refinement scheme. Fig.2 illustrates our framework. In *Sec.* 3.1, we provide a brief overview of the trajectory prediction problem. In *Sec.* 3.2, we delve into the intuition, design, and learning process underlying the pivot mechanism. *Sec.* 3.3 introduces pivot-guided trajectory prediction, while *Sec.* 3.4 presents the whole training details.

3.1 Problem Formulation

Under the context of auto-driving, the goal of trajectory prediction is to predict future trajectories of surrounding agents given agent history and essential map context. Specifically, we have states of N agents in the historic T_h steps $\mathcal{A} = \{a_{-T_h+1}^{1\sim N}, a_{-T_h+2}^{1\sim N}, \cdots, a_0^{1\sim N}\}$, and M static local HD-map information $\mathcal{M} = \{m_1, m_2, \cdots m_M\}$. It is a common practice to encode raw data of various types into high-dimensional features using individual encoders:

$$\mathbf{e_a} = \mathcal{E}_\mathbf{a}(\mathcal{A}) \in \mathbb{R}^{N\times T_h\times H}, \quad (1)$$

$$\mathbf{e_m} = \mathcal{E}_\mathbf{m}(\mathcal{M}) \in \mathbb{R}^{M\times H}, \quad (2)$$

where $\mathcal{E}_\mathbf{a}$ and $\mathcal{E}_\mathbf{m}$ denote the agent encoder and map encoder, respectively. H represents the features dimension after encoding.

Based on the encoder's output, we aim to predict K possible future modalities for N agents, where each modality is represented by the positions over T_f future steps. The output $\mathcal{O}$ is typically generated by a decoder $\mathcal{D}$ as:

$$\mathcal{O} = \mathcal{D}(\mathbf{e_a}, \mathbf{e_m}) \in \mathbb{R}^{N\times K\times T_f\times p}, \quad (3)$$

where p represents the dimension of each prediction, *e.g.*, $p = 2$ representing (x, y). Usually, a $\mathcal{D}$ fuses information from $\mathbf{e}_a$ and $\mathbf{e}_m$, and output each agent' possible trajectories directly.

PCTP, as a decoder plugin, follows the aforementioned overall framework. At the same time, we argue that directly decoding long trajectories from the raw encoded feature may lose many details and lead to significant compounding errors. We modify the decoding process of $\mathcal{D}$ by introducing pivots, which represent these important intermediate key points. Then, the decoding process would be decoupled as Pivot Prediction(*Sec.* 3.2), and Pivot-Guided Trajectory Prediction (*Sec.* 3.3).

3.2 Pivot Prediction

Pivot aims to bridge the gap between raw encoded features and the long-horizon predicted trajectories. Intuitively, we

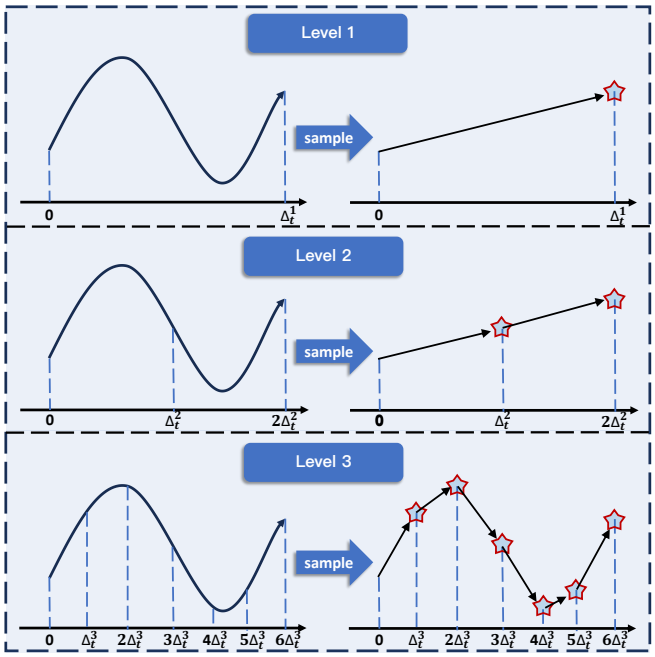

Figure 3: Sampling method for multi-scale pivots. The three point-level trajectories on the left represent the initial ground-truth trajectories, while the other three pivot-level trajectories represent the trajectories after pivot sampling. We use Δ_t^l as the sampling skip interval at level l to sample pivots from the ground truth. The top section illustrates pivot sampling at the highest level, where only the endpoint is set as the pivot. As the number of levels increases, the skip interval decreases, allowing the model to focus on finer-grained features.

seek to identify the "pivot point" within trajectories that predominantly defines the trajectory temporal dynamics, thereby reducing distribution noise in the trajectory decoding process.

Pivot Definition. Intuitively, there are several ways to define pivots. Ideal pivots should consider map geometry, temporal dynamics, and agent interactions. However, it would be quite complicated and error-prone to design such complex pivots. In this paper, we propose that pivots focusing solely on temporal dynamics are sufficient for most scenarios. By definition, temporal-dynamic pivots are sampled from the raw trajectory, which could be defined by sampling from the ground truth as:

$$\{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_{T_f}\} \xrightarrow{\Delta_t \text{ sample}} \{\mathbf{y}_{\Delta_t}, \mathbf{y}_{2\Delta_t}, \cdots, \mathbf{y}_{T_f}\}, \quad (4)$$

where $\mathbf{y}_t$ represents the ground truth waypoint at t time step. By introducing a time skip interval Δ_t, pivots are defined by sampling from the raw ground truth. Fig 3 illustrates how we sample pivots from the initial trajectory. Using these pivots as the bridge between the raw feature and the final trajectory would significantly reduce unnecessary redundancy. Without loss of generality, we denote the sampled pivot points as $\mathcal{P} = \{p_0, p_1, \cdots, p_{T_f/\Delta_t}\}$.

From End-Point To Pivot-Points. In previous works [Gu *et al.*, 2021; Shi *et al.*, 2022; Cui *et al.*, 2023; Ye *et al.*, 2023; Aydemir *et al.*, 2023], to reduce the trajectory uncertainty, a common practice involves predicting the endpoint of an entire trajectory in the first stage, with the intention of dividing the whole trajectory action space into several endpoint-defined subspaces. This approach performs well for short-term trajectory predictions, where complex temporal dynamics are minimal. However, as the prediction horizon extends, the guidance provided by a single endpoint diminishes significantly, offering little improvement for long-term prediction tasks due to the lack of intermediate guidance. The endpoint acts like a distant destination in a fog: while the model knows where it needs to reach, there are no concrete steps to guide it along the way. We contend that, in the first stage, "pivot points" would aggregate more intermediate information and provide clear space-time transition paths.

From Trajectory-Refine To Pivot-Refine. In the long-term trajectory prediction task, there usually involves an extra stage as trajectory refinement that iteratively refines given trajectories to capture more fine-grained motion details. This extra refinement stage guidance significantly improves the model's ability to tackle long-term prediction tasks, which, at the same time, would suffer from more compounding errors when a longer time horizon is involved. In this paper, we argue that, in the second stage, a whole-size proposal trajectory is not necessary for the refinement, while pivots are all we need to conduct the refinement, as these pivots already contain all the necessary information for a target trajectory.

Hierarchical Multi-Scale Pivots. Humans typically begin with a high-level goal, then establish intermediate sub-goals, progressively decomposing these sub-goals into smaller, hierarchical objectives. Each level of goals focuses on different contextual aspects of the task. Drawing inspiration from this hierarchical cognitive process, PCTP further sets hierarchical multi-scale pivots as goals of different levels. We establish L levels of pivots, where each level uses progressive sampling intervals. At the highest level, the interval is set to T_f, which means only the endpoint is considered as a *goal pivot*. Following, the second highest level interval is set to $T_f/2$, which means the endpoint and middle-point are considered as pivots, and this pattern continues downward. Formally, the hierarchical multi-scale pivots are defined by a set of hierarchical sampling intervals $\{\Delta_t^{(l)}\}$:

$$\Delta_t^{(l+1)} = \alpha_l \times \Delta_t^{(l)}, \quad (5)$$

where $\Delta_t^{(l)}$ represents skip interval at the l-th level, α_l is interval growth factor from l-th level to $l+1$-th level. Fig 3 illustrates how we sample pivots from different scales. Pivots sampled from different skip intervals certainly focus on different scale scene contexts. It is worth noting that pivots are predicted in agent-centric coordinates which can also be regarded as multi-scale relative spatial positional embedding.

Pivot Learning. Pivot learning leverages information from $\mathbf{e_a}$ and $\mathbf{e_m}$ to decode pivots. Consistent with previous work [Nayakanti *et al.*, 2023; Varadarajan *et al.*, 2022; Tang *et al.*, 2024], we employ a cross-attention decoder to predict multiple groups of pivots with multiple learnable queries. Each group of pivots is decoded into a final single trajectory. Moreover, pivots are predicted using a hierarchical multi-scale schema that enables the decoder to progressively refine pivots based on high-level intentions and various scales of global context.

Learnable Pivot. Each query begins as a learnable embedding and corresponds to a final decoded trajectory in the interactive decoding process. In each level, queries cross-attend globally to the scene context, and output pivots at the corresponding scale, representing the current level of intention. This mode-to-scene cross-attention involves three types of queries: mode-to-history, mode-to-map, and mode-to-agent queries. Through these specialized queries, each query retrieves essential features from the corresponding scene context, enhancing the overall trajectory prediction accuracy. Formally, given the encoded agent feature $\mathbf{e_a}$, map feature $\mathbf{e_m}$, and K initial queries $\mathbf{q} \in \mathbb{R}^{K\times H}$, pivots are learned as:

$$\mathbf{e}_q = \mathrm{DAttn}\Big(\mathbf{q}, [\mathbf{e_a}, \mathbf{e_m}], [\mathbf{e_a}, \mathbf{e_m}]\Big), \tag{6}$$

$$\mathcal{P} = \mathcal{D}_{\text{pivot}}(\mathbf{e}_q, \mathbf{e_a}, \mathbf{e_m}), \tag{7}$$

where DAttn, named Decoupled Attention, consists of mode-to-history attention, mode-to-agent attention, mode-to-map attention and mode-to-mode attention with $\mathbf{q}$ as the queries, $[\mathbf{e_a}, \mathbf{e_m}]$ as the keys and values. $\mathcal{P}$ represents the predicted pivots generated by the pivot decoder $\mathcal{D}_{\text{pivot}}$.

Multi-Scale Pivot Prediction. Instead of introducing additional modules and parameters, we reuse the mode-to-scene cross-attention module to gradually refine the mode query from high-level intention to low-level pivots in an iterative manner. In each iteration, mode queries perform mode-to-scene cross-attention, outputting pivots at the current scale. Before entering the next iteration, the current output pivots are transformed to Fourier features, then embedded into the query feature space, and finally fused with the original query to achieve high-level intention embedding. Therefore, as iterations proceed, the pivot scale becomes finer, allowing for increasingly precise pivot refinement. Formally:

$$\mathbf{q}^{(0)} = \mathbf{q}, \;\; \mathbf{q}^{(l)} = \mathcal{E}_{pivot}(\mathcal{P}^{(l-1)}), \; l > 0, \tag{8}$$

$$\mathbf{e}_q^{(l)} = \mathrm{MHA}\Big(\mathbf{q}^{(l)}, [\mathbf{e_a}, \mathbf{e_m}], [\mathbf{e_a}, \mathbf{e_m}]\Big), \tag{9}$$

$$\mathcal{P}^{(l)} = \mathcal{D}_{\text{pivot}}(\mathbf{e}_q^{(l)}, \mathbf{e_a}, \mathbf{e_m}), \tag{10}$$

where we reuse the same pivot decoder $\mathcal{D}_{\text{pivot}}$, and output multi-level pivots.

3.3 Pivot-Guided Trajectory Prediction

In the second stage, we employ a Pivot-Guided Trajectory Prediction scheme to transform pivots into a fine-grained trajectory. Intuitively, each pivot offers substantial information within its neighborhood. Consequently, the Pivot-Guided Trajectory Prediction process operates by predicting the offset of each trajectory point from its associated pivot. This approach facilitates a hierarchical learning process, where the pivot captures macro dynamics while the trajectory learns the fine-grained temporal structure.

The process begins by using predicted pivots to query the essential local context around each pivot by pivot-to-context cross-attention. Given the predicted pivot $\mathcal{P} = \{p_0, p_1, \cdots, p_{T_f/\Delta_t}\}$, along with the encoded feature $\mathbf{e_a}$ and $\mathbf{e_m}$, the pivot embeddings are obtained by DAttn as follows:

$$\mathbf{e}_{p_i} = \mathrm{DAttn}\Big(\mathcal{E}_{pivot}(p_i), [\mathbf{e_a}, \mathbf{e_m}], [\mathbf{e_a}, \mathbf{e_m}]\Big). \tag{11}$$

Next, $\mathbf{e}_{\mathcal{P}_i}$ is used to predict the local offsets $\boldsymbol{\delta}_{p_i:p_{i+1}}$:

$$\boldsymbol{\delta}_{p_i:p_{i+1}} = \mathcal{D}_{\text{traj}}(\mathbf{e}_{\mathcal{P}_i}, \mathbf{e_a}, \mathbf{e_m}), \tag{12}$$

$$\mathbf{o}_{p_i:p_{i+1}} = \boldsymbol{\delta}_{p_i:p_{i+1}} + p_i, \tag{13}$$

where $\mathcal{D}_{\text{traj}}$ is our Pivot-Guided Trajectory Decoding Module, p_i denotes the i-th pivot and $\mathbf{o}_{p_i:p_{i+1}}$ is final trajectory between p_i and p_{i+1}. Under this framework, we reframe the initial long-term prediction task as a series of short-term subtasks, where each subtask operates within a more manageable temporal window.

In order to facilitate the integration of our model into most SOTA models, we provide two solutions corresponding to the two current mainstream trajectory decoding solutions.

Pivot-Guided One-Shot Trajectory Decoding. For one-shot trajectory decoders[Gu *et al.*, 2021; Cui *et al.*, 2023; Shi *et al.*, 2022; Tang *et al.*, 2024] which don't need iterative operations, PCTP only learns top-level pivots without iterative operations and then replaces mode-to-context cross-attention with pivot-to-context cross-attention, guiding the model to focus on relevant local context information. After generating the trajectory, each waypoint is added to its assigned pivot, effectively performing short-term trajectory prediction at each pivot and then producing the final trajectory.

Pivot-Guided k-Shot Trajectory Decoding. For decoders with refinement [Jiang *et al.*, 2023; Zhou *et al.*, 2023; Liu *et al.*, 2024; Zhou *et al.*, 2024], often referred to as k-shot trajectory decoding, PCTP also replaces mode-to-context cross-attention with pivot-to-context cross-attention, while retaining the iterative refinement module to achieve a more detailed trajectory. However, experimental results show that, with PCTP, the improvement gained from additional refinement becomes minimal. Consequently, the number of refinement iterations can be reduced to as few as one or even eliminated altogether.

3.4 Training Loss

Following common practice [Zhou *et al.*, 2022; Zhou *et al.*, 2023], we parameterize the pivots as a mixture of Laplace distributions:

$$f(\mathcal{P}) = \sum_{k=1}^{K} \pi_k \prod_{i=1}^{T_f/\Delta_t} \mathrm{Laplace}(p_{i,k} | \mu_{i,k}, \mathbf{b}_{i,k}), \tag{14}$$

where $\{\pi_k\}_{k=1}^{K}$ are the mixing coefficients, and the Laplace density of the k-th mixture component for the i-th pivot is parameterized by location $\mu_{i,k}$ and scale $\mathbf{b}_{i,k}$. We optimize the mixing coefficients using a classification loss $\mathcal{L}_{\text{cls}}$.

In addition, we apply the Winner-Takes-All[Lee *et al.*, 2016] strategy to optimize PCTP. Specifically, our matching operation occurs during the pivot prediction stage to ensure intention consistency between pivot-level trajectory and point-level trajectory. This process is defined as follows:

$$k^* = \underset{k\in[1,K]}{\mathrm{argmin}} \sum_{i=1}^{T_f/\Delta_t} \mathrm{L2}(\mathbf{y}_{i\times\Delta_t}, \mathcal{P}_i), \tag{15}$$

where $\mathbf{y}_t$ denotes the ground truth at time step t and $\mathcal{P}_i$ represents the i-th pivot. This formulation enables us to identify

Dataset	Method	**b-minFDE$_6$** ↓	minADE$_6$ ↓	minFDE$_6$ ↓	MR$_6$ ↓	minADE$_1$ ↓	minFDE$_1$ ↓	MR$_1$ ↓
Argoverse I	LaFormer	1.759	0.918	1.091	0.096	1.473	2.810	0.474
	LaFormer w/ Ours	**1.696**	**0.717**	**1.033**	0.099	**1.293**	**2.792**	0.477
	HPNet (no ref)	1.580	0.663	0.933	0.078	1.370	2.940	0.465
	HPNet (no ref) w/ Ours	**1.560**	**0.657**	**0.928**	0.081	**1.368**	**2.920**	**0.460**
Argoverse II	DenseTNT	2.424	0.992	1.749	0.221	2.086	4.972	0.661
	DenseTNT w/ Ours	**2.377**	**0.934**	**1.708**	**0.217**	**2.040**	**4.947**	**0.657**
	QCNet (no ref)	1.928	0.729	1.292	0.164	1.680	4.348	0.590
	QCNet (no ref) w/ Ours	**1.852**	**0.708**	**1.236**	**0.157**	**1.645**	**4.222**	**0.575**
	QCNet	1.874	0.720	1.253	0.157	1.687	4.316	0.579
	QCNet w/ Ours	**1.847**	**0.700**	**1.226**	**0.152**	**1.674**	**4.306**	0.584

Table 1: Performance on Argoverse I and Argoverse II validation set. The sign (no ref) represents we implement the version that removes the refinement module based on the initial network since we try to prove that PCTP can be regarded as a lightweight refinement network, which leads to the effect of refinement being greatly reduced. PCTP improves most metrics of all state-of-the-art methods.

the most similar mode k^* by minimizing the sum of L2 distances.

For stabilization, the pivot-guided trajectory decoding module stops the gradient flow through the pivots. The final loss function combines pivot-level trajectory loss $\mathcal{L}_{\text{pivot}}$, point-level trajectory loss $\mathcal{L}_{\text{traj}}$ and the classification loss $\mathcal{L}_{\text{cls}}$ for end-to-end training:

$$\mathcal{L}_{\text{pivot}} = \frac{\Delta_t^l}{T_f} \sum_{l=1}^{L} \mathcal{L}_{\text{preg}}^l, \tag{16}$$

$$\mathcal{L} = \mathcal{L}_{\text{pivot}} + \mathcal{L}_{\text{traj}} + \beta \cdot \mathcal{L}_{\text{cls}}, \tag{17}$$

where $\mathcal{L}_{\text{preg}}^l$ is the pivot's regression loss at l level, Δ_t^l is the skip interval of l level and β is a hyper-parameter that balances regression and classification. It is noted that the normalization factor Δ_t^l/T_f ensures that the learning rate for each pivot aligns with that for each point.

4 Experiments

We show our result on validation set and test set of Argoverse I and Argoverse II. As shown in Table 1, PCTP can improve the accuracy of all considered state-of-the-art methods.

4.1 Experimental Setting

Datasets. We evaluate our approach on two large-scale autonomous driving datasets: Argoverse I and Argoverse II. Both datasets capture real-world driving scenarios and include high-definition maps annotated with detailed motion data sampled at 10 Hz. Argoverse I contains 323,557 sequences collected from Miami and Pittsburgh. The prediction task requires forecasting 3-second future trajectories based on 2 seconds of historical observations. Argoverse II, comprising 250,000 scenarios across six cities, offers enhanced data quality and presents a more challenging task: predicting 6-second future trajectories given 5 seconds of observation history. We follow the official dataset guidelines for partitioning both datasets into training, validation, and test sets.

Metrics. Following the official evaluation protocols, we assess our approach using standard motion prediction metrics, including Brier minimum Final Displacement Error (b-minFDE$_K$), minimum Average Displacement Error (minADE$_K$), minimum Final Displacement Error (minFDE$_K$), and Miss Rate (MR$_K$). The minADE$_K$ metric calculates the l_2 distance between the ground truth trajectory and the best of K predicted trajectories, averaged over all future steps, while minFDE$_K$ computes only the best prediction error among the K predicted endpoints. For b-minFDE$_K$, we add $(1-\hat{\pi})^2$ to minFDE$_6$ to evaluate classifier performance, where $\hat{\pi}$ denotes the highest predicted probability score output by the classifier. Additionally, MR$_K$ represents the proportion of predictions where minFDE$_K$ exceeds 2 meters. Typically, K is set to 1 or 6; for cases where the number of trajectories exceeds K, we reduce the count by selecting those with the top-K probability scores.

Baselines. The modular design of PCTP enables integration with most existing trajectory prediction frameworks. We demonstrate this versatility by evaluating PCTP with four state-of-the-art prediction models as backbones: HPNet [Tang *et al.*, 2024], LaFormer [Liu *et al.*, 2024], DenseTNT [Gu *et al.*, 2021], and QCNet [Zhou *et al.*, 2023].

4.2 Quantitative Result

Performance on Val Set. We present qualitative results on the Argoverse I and Argoverse II validation set in Table. 1. We integrate PCTP into each state-of-the-art prediction model and achieve good improvement. For instance, PCTP can reduce b-minFDE of LaFormer, HPNet, DenseTNT, and QCNet by 3.5%, 1.2%, 1.9%, and 3.9% respectively. For the implementation of HPNet and QCNet, we provide the result of both the original version and the version without the refinement module, since the pivot-guided module also has a refinement function.

Performance on Test Set. We also present some qualitative results on the Argoverse II test set in Table. 2. It shows that PCTP improves the results of QCNet by 0.3% on b-minFDE, 0.9% on minADE, 0.4% on minFDE, and 1.2% on MR. Specifically, PCTP based on QCNet outperforms all published ensemble-free works on the Argoverse 2 leaderboard.

4.3 Ablation Study

To thoroughly evaluate the impact of each component of PCTP, we conducted ablation studies across three dimensions: module effectiveness, prediction performance across varying future time steps, and the effect of different pivot

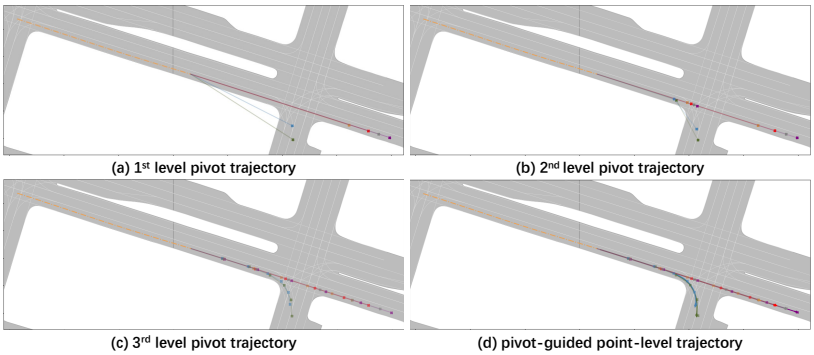

Figure 4: Visualization of PCTP learning process. Orange polyline denotes the agent history, while the other 6 different colors represent 6 modal trajectories respectively. As the pivot level increases from (a) to (d), the *right-turn* trajectory is gradually refined.

Method	b-minFDE$_6$	minADE$_6$	minFDE$_6$	MR$_6$
QCNet	1.861	0.636	1.241	0.154
QCNet w/ Ours	**1.854**	**0.630**	**1.235**	**0.152**

Table 2: Performance on Argoverse II test set. We choose QCNet as our baseline and submit the initial version and the version with PCTP.

Method	b-minFDE$_6$ ↓	NLL ↓
QCNet	1.874 ± 0.001	1.050 ± 0.012
QCNet + PCTP	$\mathbf{1.847 \pm 0.001}$	$\mathbf{1.040 \pm 0.010}$

Table 3: Statistical robustness over 5 independent runs on Argoverse 2 validation set. We report mean ± standard deviation. ↓ indicates lower is better.

sampling intervals. We only show the performance of QCNet with PCTP, while the other results of baseline with PCTP can be found in the supplement materials.

Component of PCTP. We analyze the contribution of each core module in PCTP: Pivot Prediction (PP), Multi-Scale Pivot prediction (MSP), and Pivot-guided Trajectory Prediction (PTP) by progressively removing them from the full model. PP provides foundational intermediate anchors for capturing motion dynamics, MSP enables adaptive multi-scale intention interaction, and PTP maintains temporal consistency via local pivot context. As shown in Table
reftable3, each module contributes meaningfully to overall performance, with the full PCTP configuration achieving the best accuracy.

Statistical Robustness. To confirm that our improvements are not artifacts of training stochasticity, we report the mean and standard deviation over 5 independent runs in Table 4. The consistently low variance and the improvement in Negative Log-Likelihood (NLL) demonstrate that PCTP yields statistically robust gains in both point-wise accuracy and probabilistic calibration.

4.4 Case Illustration

We present a case illustration of PCTP integrated with QCNet in Fig 4. The PCTP framework generates 6 distinct trajectory modalities at the top level, each accompanied by a 6-second pivot prediction that explicitly specifies destination goals. As illustrated in Fig 4 (a), PCTP produces 4 straight-path modalities and 2 right-turn modalities. The straight modes include 1 deceleration mode (orange '×') and 3 differentiated acceleration modes (red, gray, and purple '×'), while the right-turn modes encode lane-specific intentions - first-lane (green '×') and second-lane (blue '×') targeting respectively. In subsequent stages, full trajectories are incrementally constructed through hierarchical refinement guided by these top-level pivot objectives in Fig 4 (b,c,d).

PP.	*MSP.*	*PTP.*	**b-minFDE$_6$**	minADE$_6$	minFDE$_6$	MR$_6$
✓			1.890	0.720	1.290	0.170
✓	✓		1.870	0.710	1.260	0.162
✓	✓	✓	**1.847**	**0.700**	**1.226**	**0.152**

Table 4: Ablation study results examining the contribution of three key modules: PP(Pivot Prediction), MSP (Multi-Scale Pivot prediction), and PTP(Pivot-guided Trajectory Prediction). The checkmark ✓ indicates the presence of a module. Experimental results are based on the Argoverse II validation set.

5 Conclusion

This paper introduces PCTP, a novel trajectory prediction framework, designed to address the limitations of traditional endpoint-based and iterative refinement methods, particularly for long-term prediction tasks. Unlike these two approaches, PCTP decomposes long-term trajectory prediction into a series of short-term sub-tasks. Considering hierarchical agent interactions and adaptive trajectory refinement, PCTP introduces pivot-level anchors that balance global intention with fine-grained local prediction. Ultimately, it significantly improves model accuracy and consistency over extended horizons. We believe that PCTP will serve as a foundation for future innovations in reliable trajectory prediction [Miao *et al.*, 2025c].

Acknowledgements

This work is partially supported by SCRI, The Hong Kong Polytechnic University (No. Q-CDDG). Hao Miao is the corresponding author.

References

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

[Carion *et al.*, 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision (ECCV)*, pages 213–229. Springer, 2020.

[Chai *et al.*, 2019] Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. *arXiv preprint arXiv:1910.05449*, 2019.

[Cui *et al.*, 2023] Alexander Cui, Sergio Casas, Kelvin Wong, Simon Suo, and Raquel Urtasun. Gorela: Go relative for viewpoint-invariant motion forecasting. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7801–7807. IEEE, 2023.

[Da and Zhang, 2022] Fang Da and Yu Zhang. Path-aware graph attention for hd maps in motion prediction. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6430–6436. IEEE, 2022.

[Gao *et al.*, 2020] Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11525–11533, 2020.

[Gilles *et al.*, 2021] Thomas Gilles, Stefano Sabatini, Dzmitry Tsishkou, Bogdan Stanciulescu, and Fabien Moutarde. Home: Heatmap output for future motion estimation. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 500–507. IEEE, 2021.

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

[Gu *et al.*, 2021] Junru Gu, Chen Sun, and Hang Zhao. Densetnt: End-to-end trajectory prediction from dense goal sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15303–15312, 2021.

[Hagedorn *et al.*, 2023] Steffen Hagedorn, Marcel Hallgarten, Martin Stoll, and Alexandru Condurache. Rethinking integration of prediction and planning in deep learning-based automated driving systems: a review. *arXiv preprint*, 2023.

[Hu *et al.*, 2023] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17853–17862, 2023.

[Jiang *et al.*, 2023] Chiyu Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhou, Dragomir Anguelov, et al. Motiondiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9644–9653, 2023.

[Lee *et al.*, 2016] Stefan Lee, Senthil Purushwalkam Shiva Prakash, Michael Cogswell, Viresh Ranjan, David Crandall, and Dhruv Batra. Stochastic multiple choice learning for training diverse deep ensembles. *Advances in Neural Information Processing Systems*, 29, 2016.

[Liang *et al.*, 2020] Ming Liang, Bin Yang, Rui Hu, Yun Chen, Renjie Liao, Song Feng, and Raquel Urtasun. Learning lane graph representations for motion forecasting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 541–556. Springer, 2020.

[Liao *et al.*, 2024] Haicheng Liao, Zhenning Li, Huanming Shen, Wenxuan Zeng, Dongping Liao, Guofa Li, and Chengzhong Xu. Bat: Behavior-aware human-like trajectory prediction for autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 10332–10340, 2024.

[Liu *et al.*, 2024] Mengmeng Liu, Hao Cheng, Lin Chen, Hellward Broszio, Jiangtao Li, Runjiang Zhao, Monika Sester, and Michael Ying Yang. Laformer: Trajectory prediction for autonomous driving with lane-aware scene constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2039–2049, 2024.

[Liu *et al.*, 2026] Shuai Liu, Ning Cao, Yile Chen, Yue Jiang, and Gao Cong. Bus-conditioned zero-shot trajectory generation via task arithmetic. *arXiv preprint arXiv:2602.13071*, 2026.

[Mei *et al.*, 2025] Jieyuan Mei, Jindong Tian, Ronghui Xu, Hanyue Wei, Chenjuan Guo, and Bin Yang. Find3: A dual 3d state space model with dynamic hypergraph for financial stock prediction. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pages 2084–2094, 2025.

[Miao *et al.*, 2025a] Hao Miao, Ziqiao Liu, Yan Zhao, Chenxi Liu, Chenjuan Guo, Bin Yang, Kai Zheng, Huan Li, and Christian S Jensen. Lighttr+: A lightweight incremental framework for federated trajectory recovery. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 38(2):1174–1188, 2025.

[Miao *et al.*, 2025b] Hao Miao, Ziqiao Liu, Yan Zhao, Kai Zheng, Yupu Zhang, and Christian S Jensen. Federated trajectory similarity learning with privacy-preserving clustering. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, pages 959–972. IEEE, 2025.

[Miao *et al.*, 2025c] Hao Miao, Yan Zhao, Chenjuan Guo, Bin Yang, Kai Zheng, and Christian S Jensen. Spatio-

temporal prediction on streaming data: A unified federated continuous learning framework. *IEEE Transactions on Knowledge and Data Engineering*, 37(4):2126–2140, 2025.

[Nayakanti *et al.*, 2023] Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2980–2987. IEEE, 2023.

[Ngiam *et al.*, 2021] Jiquan Ngiam, Benjamin Caine, Vijay Vasudevan, Zhengdong Zhang, Hao-Tien Lewis Chiang, Jeffrey Ling, Rebecca Roelofs, Alex Bewley, Chenxi Liu, Ashish Venugopal, et al. Scene transformer: A unified architecture for predicting multiple agent trajectories. *arXiv preprint arXiv:2106.08417*, 2021.

[Phan-Minh *et al.*, 2020] Tung Phan-Minh, Elena Corina Grigore, Freddy A Boulton, Oscar Beijbom, and Eric M Wolff. Covernet: Multimodal behavior prediction using trajectory sets. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 14074–14083, 2020.

[Philion *et al.*, 2024] Jonah Philion, Xue Bin Peng, and Sanja Fidler. Trajeglish: Traffic modeling as next-token prediction. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[Salzmann *et al.*, 2020] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 683–700. Springer, 2020.

[Shi *et al.*, 2022] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. *Advances in Neural Information Processing Systems*, 35:6531–6543, 2022.

[Tang *et al.*, 2024] Xiaolong Tang, Meina Kan, Shiguang Shan, Zhilong Ji, Jinfeng Bai, and Xilin Chen. Hpnet: Dynamic trajectory forecasting with historical prediction attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15261–15270, 2024.

[Tian *et al.*, 2026] Jindong Tian, Yifei Ding, Ronghui Xu, Hao Miao, Chenjuan Guo, and Bin Yang. Arrow: An adaptive rollout and routing method for global weather forecasting. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026.

[Tolstaya *et al.*, 2021] Ekaterina Tolstaya, Reza Mahjourian, Carlton Downey, Balakrishnan Vadarajan, Benjamin Sapp, and Dragomir Anguelov. Identifying driver interactions via conditional behavior prediction. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3473–3479. IEEE, 2021.

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

[Wang *et al.*, 2025] Zichen Wang, Hao Miao, Senzhang Wang, Renzhi Wang, Jianxin Wang, and Jian Zhang. C2ftp: A coarse-to-fine denoising framework for uncertainty-aware trajectory prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pages 12810–12817, 2025.

[Xu *et al.*, 2026] Ronghui Xu, Jihao Chen, Jindong Tian, Chenjuan Guo, and Bin Yang. Most: A foundation model for multi-modality spatio-temporal traffic prediction. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1 (KDD)*, pages 1716–1727, 2026.

[Ye *et al.*, 2022] Maosheng Ye, Jiamiao Xu, Xunnong Xu, Tengfei Wang, Tongyi Cao, and Qifeng Chen. Dcms: Motion forecasting with dual consistency and multi-pseudo-target supervision. *arXiv preprint arXiv:2204.05859*, 2022.

[Ye *et al.*, 2023] Maosheng Ye, Jiamiao Xu, Xunnong Xu, Tengfei Wang, Tongyi Cao, and Qifeng Chen. Bootstrap motion forecasting with self-consistent constraints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8504–8514, 2023.

[Zeng *et al.*, 2021] Wenyuan Zeng, Ming Liang, Renjie Liao, and Raquel Urtasun. Lanercnn: Distributed representations for graph-centric motion forecasting. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 532–539. IEEE, 2021.

[Zhao *et al.*, 2021] Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Ben Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, et al. Tnt: Target-driven trajectory prediction. In *Conference on Robot Learning*, pages 895–904. PMLR, 2021.

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

[Zhou *et al.*, 2023] Zikang Zhou, Jianping Wang, Yung-Hui Li, and Yu-Kai Huang. Query-centric trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17863–17873, 2023.

[Zhou *et al.*, 2024] Yang Zhou, Hao Shao, Letian Wang, Steven L Waslander, Hongsheng Li, and Yu Liu. Smartrefine: A scenario-adaptive refinement framework for efficient motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15281–15290, 2024.