Journal of Computer Science and Electrical Engineering
Print ISSN: 2663-1938
Online ISSN: 2663-1946
DOI: https://doi.org/10.61784/jcsee3168

CHARACTER BEHAVIOR EVENT PREDICTION MODEL BASED ON FUSION OF TRAJECTORY AND ENVIRONMENTAL FEATURES

MeiFen Chen[1], Qin Zhou[2*], Jin Lu[3]
[1]*School of Digital Media, Shenzhen Polytechnic University, Shenzhen 518055, Guangdong, China.*
[2]*Institute of Technical and Vocational Education, Shenzhen Polytechnic University, Shenzhen 518055, Guangdong, China.*
[3]*Guangdong Key Laboratory of Big Data Intelligence for Vocational Education, Shenzhen Polytechnic University, Shenzhen 518055, Guangdong, China.*
[*]*Corresponding Author: Qin Zhou*

Abstract: This paper addresses the challenge of predicting human behavior events by integrating trajectory and environmental features. Existing methods often neglect structured associations between trajectories and scenes or decouple trajectory regression from intention classification. To overcome these limitations, we propose an end-to-end framework comprising dual-stream encoding, cross-modal fusion, and multimodal decoding. The dual-stream encoder separately models historical trajectories using a two-layer Transformer and scene environments via a static scene branch (ResNet-50 extracting 64 spatial tokens from semantic segmentation) and an interaction graph branch (two-layer GCN modeling up to 30 neighboring agents). Cross-modal fusion uses trajectory global features as Query and scene tokens/interaction nodes as Key/Value, enabling soft alignment of environmental elements through cross-attention, with residual connections and layer normalization. The multimodal decoder, based on a conditional variational autoencoder, simultaneously predicts future 12-step trajectories and six behavior classes (straight, turn left/right, stop, interact with person/vehicle), jointly optimized by weighted negative log-likelihood and cross-entropy loss. Experiments on ETH/UCY, SDD, and nuScenes demonstrate state-of-the-art performance: on ETH/UCY, ADE=0.61m, FDE=1.10m, behavior accuracy=89.7%; on nuScenes, ADE=0.43m, accuracy=95.6%; on SDD, ADE=0.90m. Ablation studies confirm the importance of each component, with attention visualization showing dynamic focus shifts. The full model has 28M parameters, runs at 115 FPS (8.7ms per sample), while a lightweight version achieves 189 FPS with ADE=0.64m. This work highlights the value of structured environment fusion and joint trajectory-intention decoding for accurate, rational, and generalizable human behavior prediction.
Keywords: Behavior prediction; Trajectory prediction; Environmental feature fusion; Spatio-temporal graph network; Cross-modal attention; Deep learning

1 INTRODUCTION

Understanding and predicting the future behavior of people in the environment is a long-term goal in the field of computer vision and artificial intelligence [1]. From the need for self-driving cars to predict pedestrians crossing the road, to the need for service robots to plan their interaction paths with users in advance, to the identification of potential abnormal aggregation by security systems, the prediction of human behavior events is the basis for the safe and efficient operation of autonomous agents [2]. Traditional methods of character behavior prediction usually reduce the problem to trajectory prediction, that is, to infer the future location only from the observed historical coordinate sequence. Such methods, such as sequential models based on recurrent neural network (RNN) and its variants (LSTM, GRU), can capture the short-term inertia of individual movement well, but they are inadequate in the face of complex social interactions and scene constraints [3]. For example, a pedestrian's walking pattern in an open square is very different from that in a crowded street; a person's behavior of slowing down and stopping in front of a cafe cannot be accurately inferred from his past speed of movement alone. In recent years, researchers have begun to realize the importance of environmental information. Static elements in the scene, such as sidewalks, zebra crossings, building entrances and traffic lights, define walkable areas and behavior rules, while dynamic elements, such as the intentions of other pedestrians and vehicles, constitute complex interaction fields [4]. However, how to effectively integrate these heterogeneous and high-dimensional environmental features with individual trajectory features is still an open problem. Simple feature splicing often ignores the structural relationship between the two, which makes it difficult for the model to understand the causal relationship of "how the environment affects behavior".

In this paper, we propose a behavioral event prediction model based on the fusion of trajectory and environmental features. Firstly, the trajectory temporal information and scene spatial information are encoded independently by designing a dual "stream coding-cross-modality" fusion framework, and the unique representations of their respective modalities are retained, and then the deep interaction is realized through the cross-modality attention module. Secondly, a structured environment representation is constructed, and the panoramic image pixel features are extracted by CNN, and the semantic segmentation graph and object detection box are introduced to abstract the environment into a scene

© By the Author(s) 2026, under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0).

graph containing nodes and edges, so as to align with the trajectory graph network. Third, multi-modal behavior prediction is implemented, and continuous future trajectory coordinates and discrete behavior categories (such as straight, left turn, stop, interaction) are output simultaneously. Finally, a comprehensive experimental verification is carried out on ETH/UCY, SDD and other internationally recognized data sets, and the effectiveness and superiority of the model are fully proved by comparing with a variety of SOTA methods.

2 RELATED WORK

2.1 Trajectory Prediction Based on Time Series Model

Early trajectory prediction relies on physical priors, such as constant velocity models, Kalman filters, particle filters, and social force models [5]. The social force model uses artificial potential functions such as attraction and repulsion to describe the interaction between pedestrians, pedestrians and walls or target points, which is simple and effective in sparse scenes and short-term prediction, but the parameters depend on experience, and it is difficult to express complex scene rules and individual intentions [6]. After the rise of deep learning, RNN and its variants became the mainstream [7]. Social-LSTM proposes a social pooling layer, which aggregates the LSTM hidden States of neighborhood pedestrians according to the spatial grid, and solves the problem of multi-pedestrian joint modeling [8]. SR-LSTM further introduces the state refinement mechanism to iteratively optimize the hidden state of individuals through message passing to enhance the interactive expression of dense crowds [9]. In order to obtain a multimodal future, Social-GAN uses generator-discriminator confrontation training to output multiple socially acceptable trajectories [10]. PECNet, BiTraP, etc. model intent uncertainty with endpoint targets or CVAEs [11-12]. The problem of the pure time series model is that if only the historical coordinates are input, the model has to implicitly infer the scene constraints from the motion statistics, and it is easy to produce errors that deviate from the walkable area or violate the traffic semantics in the strong regular areas such as intersections, halls and platforms.

2.2 Social Interaction and Graph Structure Modeling

Human-human interaction is the core of crowd scenario. In addition to social pooling, one method uses attention to weight neighborhood influence, such as SRLSTM, SRA-LSTM, Social-BiGAT, etc.; another method uses nodes to represent pedestrians, edges to represent spatial proximity, relative speed or social relations [13]. STGAT uses graph attention to assign adaptive weights to neighboring nodes to capture the non-uniform influence of different pedestrians on the target [14]. Social-STGCNN replaces pooling with spatio-temporal map convolution to reduce spatial discretization loss and improve parameter efficiency [15]. For longer periods and group behavior, the multi-agent review points out that GAT, hypergraph, and transformer-graph mixing can further enhance the expression of remote dependence and group structure [16]. The existing graph method focuses on "pedestrian-pedestrian", and the heterogeneous graph modeling of "pedestrian-vehicle-obstacle-walkable area" is still insufficient [17]. Especially in the scene of human-machine integration and the edge of automatic driving, static facilities and dynamic traffic participants together form a constraint set, and the simple "human-human" graph will lose the key causal boundary [18].

2.3 Scene Environment Representation and Semantic Constraints

The representation of environmental features is divided into three main categories. One is driven by images and semantic maps. Scene-LSTM integrates local scene images into recursive units [19]; SS-LSTM, SoPhie, PIFP, etc. extract road, sidewalk, wall and obstacle features by using global/local image blocks and semantic segmentation maps [20-22]; Y-Net jointly decodes multimodal trajectories with semantic maps and trajectory codes, and verifies the role of semantic constraints in improving rationality on ETH/UCY benchmarks [23]. The second is the grid BEV, which renders lanes, obstacles, traffic signs, and agent history into a multi-channel aerial view, and then uses CNN or ConvLSTM to extract spatial features, such as MultiPath, CoverNet, etc. [24]. This kind of method is simple to implement and compatible with the perception of automatic driving, but the grid resolution is contradictory to the computational cost, and the discretization will lose the geometric accuracy. The third is the vector diagram and scene diagram. VectorNet represents the trajectory and lane as a broken line subgraph, and LaneGCN represents the lane centerline and connection relationship as a graph network [25]. Sun et al. demonstrated on nuScenes that road maps can significantly improve joint vehicle/pedestrian prediction [26]. In recent years, there are also scene graph methods that use heterogeneous nodes to represent crosswalks, signal lights, vehicles and pedestrians, such as the use of "location-vehicle-signal light-button" heterogeneous graph to enhance the generalization of crossing [27].

2.4 Cross-Modal Fusion and Transformer Prediction

Traditional fusion is commonly used for splicing, addition or fusion after double branching, which is difficult to deal with the nonlinear and positional correspondence between trajectory sequences and high-dimensional scenes [28]. Cross-modal learning has been gradually introduced into trajectory prediction after its maturity in audio-visual and "language-visual" tasks. Among them, CMAPT fuses pedestrian trajectory and surveillance video motion/visual features through cross-attention with dual-mode Transformer [29]. GIMO combines motion capture, 3D point cloud and gaze signal to predict human motion across modality Transformer [30]. SHENet constructs both trajectory coding and

scene coding, and refines the candidate trajectory through cross-modal transformer [31]. In the general spatio-temporal prediction survey, cross-modal Transformer, BEVGPT and map constraint query network are used to unify "perception-prediction-planning" information [32]. In terms of Transformer-like trajectory model, STAR uses "spatial-temporal" Transformer to alternately model social and sequence dependencies [33]; AgentFormer proposes agent-aware attention, which jointly models on time step and agent identity, taking into account individual trajectory and group influence [34]; TUTR, SGTN further unify multimodal trajectories, social graphs, and global pattern parsing to the Transformer framework [35-36]. Its advantages are strong long-range dependence and multi-source fusion ability, but its disadvantages are that the global self-attention has a large computational overhead under image token and large-scale multi-agent, and it needs to reduce the order through cross-attention, potential spatial attention or graph sparsity.

2.5 Behavioral Events and Multimodal Intention Prediction

Behavior event prediction is not equivalent to coordinate regression, which often includes discrete semantics such as going straight, turning, stopping, waiting, entering a certain area and interacting with an object [37]. Early intent work often used RGB appearance, pose, or trajectory end point classification. Recently, PedFormer and other cross-modal models jointly predict pedestrian crossing intention and trajectory, and CITraNet combines Transformer and extreme value theory to predict trajectory and crossing intention [38-39]. In terms of generative multimodality, CVAE, such as Trajectron++ and BiTraP, can output trajectory sets with probability [40]; GAN emphasizes social rationality but training is unstable [41]; diffusion models, such as MID and Leapfrog, depict multimodal distribution with denoising process, but the reasoning cost and calibration difficulty are high [42]. The multi-agent survey points out that the future direction is to unify generative models, graph structures and semantic scenarios to predict not only "where" but also "what to do and why to do".

On the whole, the existing behavioral event methods have two weaknesses: one is that the intention label is often decoupled from the trajectory training, which leads to the fact that semantic supervision can not feed back the coordinate prediction; Second, the modeling of environmental triggers is shallow, such as "approaching zebra crossing and green light" and "stagnation of the crowd ahead", which do not form evidence of differentiable attention. In this paper, cross-modal attention is used to establish the response of trajectory query to environmental nodes, and shared fusion features are used to decode trajectory distribution and behavior categories at the same time, which is a targeted improvement to the above problems.

3 METHODOLOGY

3.1 Algorithm Architecture

This chapter proposes a person behavior event prediction model based on the integration of trajectory environmental features [43], which aims to solve the problems of inadequate utilization of environmental information, lack of structured alignment between trajectory and scene features, and isolation of trajectory prediction and behavior classification in existing methods, and adopts the end-to-end architecture of "dual-stream coding-cross-modal fusion-multi-modal decoding". This is shown in Figure 1.

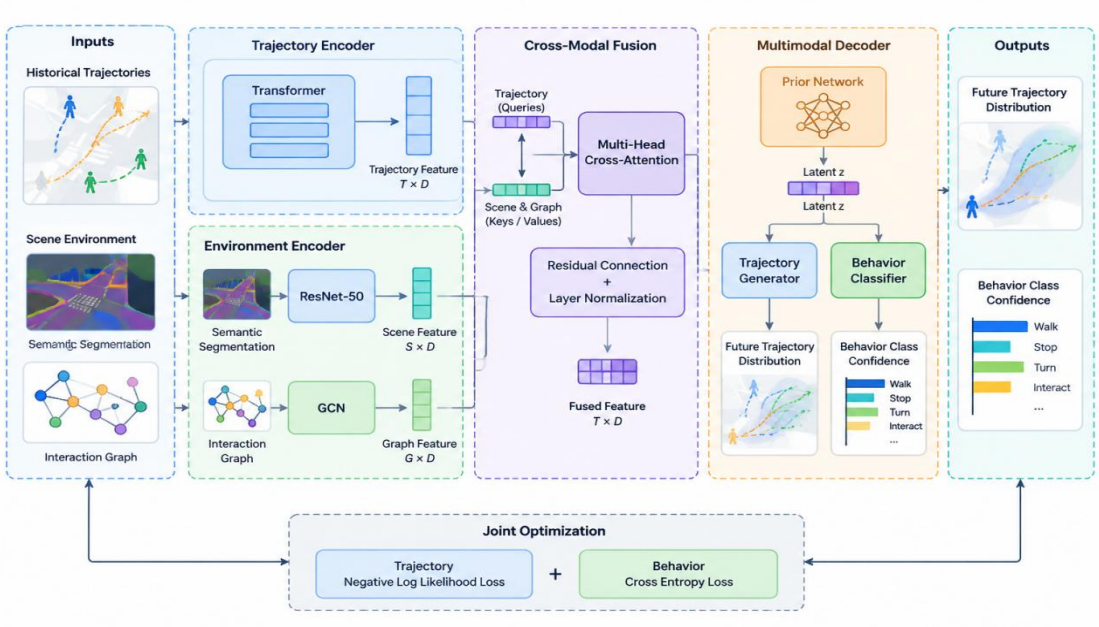


Figure 1 Prediction Model of People's Behavior Events Based on Fusion of Trajectory and Environmental Features

First, the historical trajectory sequence and the scene environment information are encoded independently, respectively, retaining the inherent characterization characteristics of each modality. Secondly, through the cross-modal fusion module based on cross-attention, the trajectory features are used as queries, and the scene semantic features and interaction graph node features are used as keys and values to achieve the structured alignment and information interaction between the two modalities, so that the model can explicitly learn "which environmental elements are most critical for the current behavior decision at any time". Finally, the fused features are sent to a multi-modal decoder based on a conditional variational self-encoder, which outputs the probability distribution of future trajectories and the confidence of discrete behavior classes. Specifically, the trace-stream encoder employs Transformer to capture timing dependencies [44]. The environmental stream encoder contains two parallel branches, the static scene branch extracts the global spatial features of the semantic segmentation graph with ResNet-50, and the interaction graph branch models the topological relationship between the target pedestrian and other dynamic agents in its neighborhood with the graph convolutional network [45]. The cross-modal fusion module achieves the soft alignment of the trajectory query to the key value of the environment through multi-head cross-attention, and obtains a robust fusion representation with the help of residual connection and layer normalization. The multi-modal decoder consists of a prior network, a trajectory generator, and a behavior classifier. Randomness is introduced by reparameterized sampling to cover multiple possibilities of future behavior. The entire model is jointly optimized end-to-end with the weighted sum of the trace negative log-likelihood loss and the behavioral cross-entropy loss.

3.2 Problem Definition

Let the sequence of observed historical trajectories be $X=\{x^t \mid t=1,2,....,T_{obs}\}$, of which $x^t=(x_i^t,y_i^t)$ Indicates the first i Two-dimensional coordinates of a pedestrian at time, corresponding to an observation window of the past 3.2 seconds (sampling frequency 2.5 Hz). Meanwhile, the corresponding scene information is given S, including one frame of the aerial view RGB image centered on the target pedestrian $I \in R^{H\times W\times 3}$ And its semantic segmentation graph $M \in R^{H\times W\times C_{sem}}$ ($C_{sem}=12$ Common ground elements), as well as the detection box information of all other dynamic agents (pedestrians, vehicles, cyclists) and static obstacles within a radius of 15 meters around the target pedestrian. The goal of the model is to predict the future. $T_{pred}=12$ Trajectories within time steps (corresponding to 4.8 seconds in the future) $Y=\{y^t \mid t=T_{obs}+1,....,T_{obs}+T_{pred}\}$ And output one or more possible behavior event tags at the same time $C\in\{c_1,c_2,....,c_K\}$, of which $K=6$ It represents six kinds of discrete behaviors, such as going straight, turning left, turning right, stopping, interacting with people and interacting with vehicles. In this paper, trajectory prediction and behavior classification are unified in an end-to-end framework, and they promote each other by sharing fusion features.

3.3 Double Stream Encoder

The double-stream encoder consists of a track stream encoder and an environment stream encoder, which encode the time sequence track and the space environment independently. The trace stream encoder uses a two-layer Transformer Encoder structure to hide dimensions. $D=128$ Number of heads $h=8$. The input historical coordinates are mapped to 128 dimensions by linear embedding, and the learnable position codes are superimposed, and then the long-term dependence within the trajectory is captured by the self-attention mechanism of Transformer. The output is the feature vector at each time step. $h_{traj}^t \in R^{128}$ And global aggregation feature $H_{traj}\in R^{128}$ (Taking the output of the last time step) [46]. The specific architecture is shown in Figure 2.

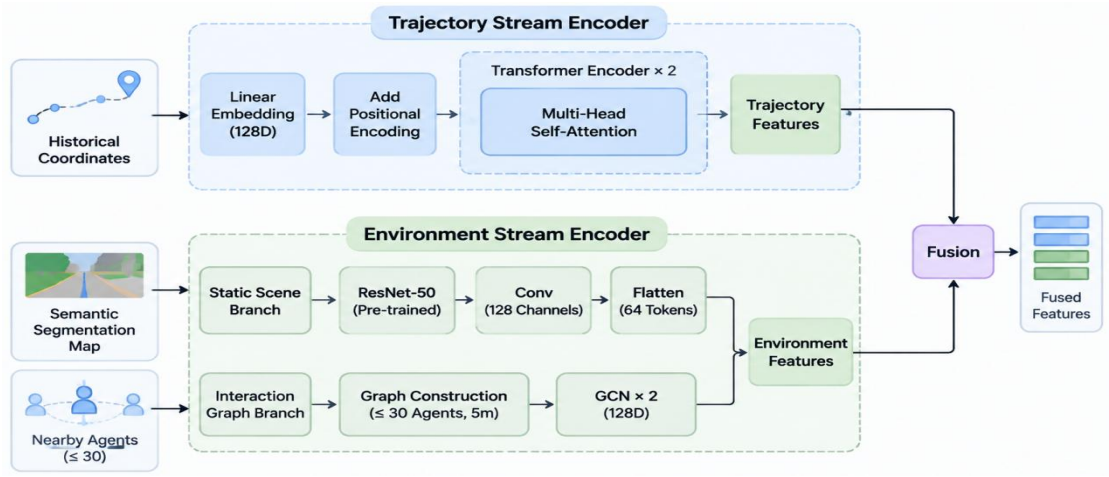


Figure 2 Architecture Diagram of Double-Stream Encoder

The ambient stream encoder contains two parallel branches. Static scene branches will semantically segment the graph M Enter the pre-trained ResNet-50 and extract the dimension of $8\times8\times2048$ The characteristic diagram of, and then by 1×1 The convolution dimension is reduced to 128 channels, and 64 spatial tokens are obtained after flattening, which is recorded as M . The interaction graph branch constructs a graph structure with the target pedestrian and up to 30 agents in its neighborhood: the initial features of the nodes are the relative position, speed, and category one-hot coding (a total of 8 dimensions), the edges are connected according to the spatial distance (threshold of 5 meters), and the weights are Gaussian kernels. Through a two-layer graph convolutional network (GCN), with 128 hidden dimensions in each layer, all node features are output, where the node features of the target pedestrian are recorded as $H_{\text{interact}} \in R^{128}$. The outputs of the two branches jointly serve as the representation of the environment side in the subsequent fusion stage.

3.4 Cross-Modal Fusion Module

The cross-modal fusion module is the core innovation of this model, which aims to achieve the structured alignment of trajectory features and environmental features. The specific architecture is shown in Figure 3. The module uses a multi-head cross-attention mechanism to aggregate the global features of the trajectory flow. H_{traj} As a Query, the 64 spatial tokens of the static scene branches and all the node features of the interaction graph branches (in total $64+N$ Tokens) are spliced as a Key and a Value. When calculating attention, the query dimension is 1×128, the length of the key value sequence is $L=64+N$ The weight distribution of each environment token to the current trajectory is obtained by scaling the dot product attention, so as to weight and aggregate the context vector most relevant to the environment. $Z \in R^{128}$. The final fused feature is obtained by residual connection and layer normalization of the vector and the original trajectory feature $H_{fused} = LayerNorm(H_{traj} + Z)$. This mechanism enables the model to dynamically focus on the areas of the scene that are most relevant to the current behavioral decision (e.g., zebra crossing, oncoming vehicle, stagnant crowd ahead) and explicitly inject these structured cues into the trajectory representation.

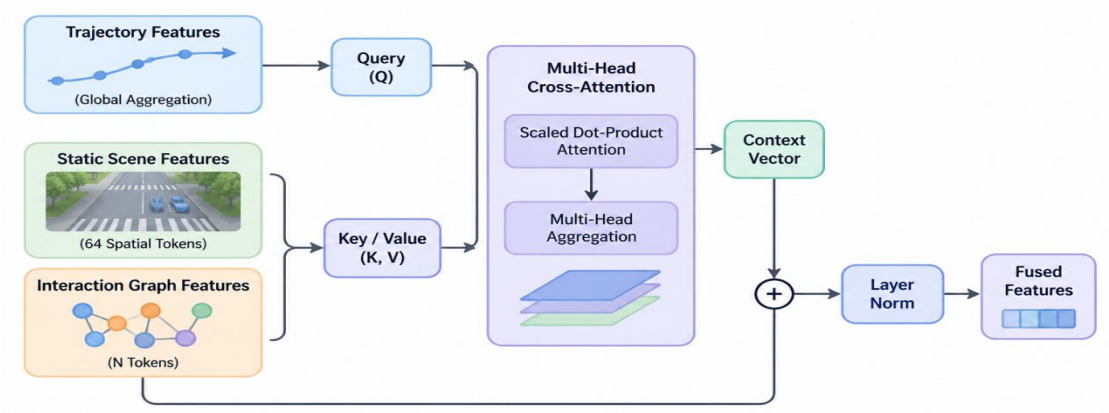


Figure 3 Cross-Modal Fusion Module Architecture Diagram

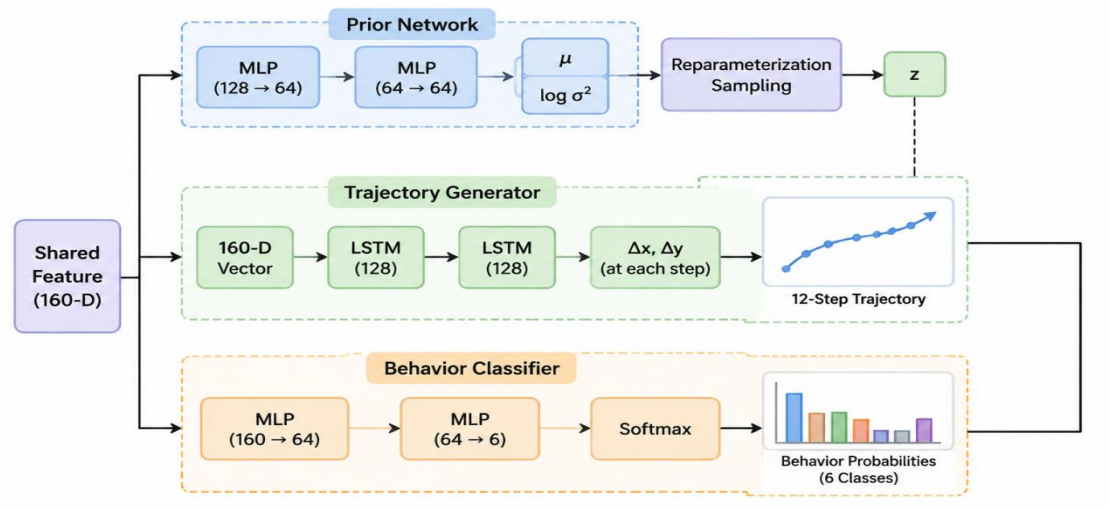


Figure 4 Multimodal Predictive Decoder

3.5 Multimodal Predictive Decoder

The conditional variational autoencoder (CVAE) framework is used in the decoder, which consists of three sub-modules: the prior network, the trajectory generator and the behavior classifier, sharing the same fusion feature. H_{fused} The specific architecture is shown in Figure 4. The prior network is a two-layer MLP (128 → 64 → 64) that outputs latent variables $z \in R^{32}$ The mean value of μ_{prior} And logarithmic variance $\log \sigma_{prior}$, obtained by sampling with the reparameterization technique z. Trace generator $\lfloor H_{fused}; z \rfloor$ The spliced 160-dimensional vector is used as the initial hidden state and sent to the two-layer LSTM (hidden dimension 128). The two-dimensional coordinate increment is output at each time step, and the trajectory sequence of the next 12 steps is obtained by accumulation. $\hat{Y} \in R^{12\times 2}$. The behavior classifier inputs the same 160-dimensional vector into the two-layer MLP (160 → 64 → 6), and outputs the probabilities of the six behavior categories through softmax $p \in R^K$. When reasoning, multiple candidate trajectories and corresponding behavior probabilities are obtained by multiple sampling (for example, K = 20) to cover the multi-modality of future behavior.

3.6 Loss Function

The model jointly optimizes the trajectory prediction task and the behavior classification task in an end-to-end manner, and the total loss function is composed of two weighted sums. The negative log-likelihood criterion is used for the trajectory prediction loss. It is assumed that the true coordinates at each time step in the future follow an isotropic two-dimensional Gaussian distribution, and the mean is the predicted coordinates output by the trajectory generator. $\hat{y}$ Gives, the variance σ^2 Is a learnable scalar parameter (shared by all time steps). The specific form is shown in Formula 1:

$$\Gamma_{traj} = -\sum_{t=1}^{T_{pred}} \log N(y^t \mid \hat{y}, \sigma^2 I) = \sum_{t=1}^{T_{pred}} \left(\frac{\left\| y^t - \hat{y} \right\|}{2\sigma^2} + \log(2\pi\sigma^2) \right) \tag{1}$$

Among y^t Are real coordinates, $\hat{y}$ Is the predicted coordinate. This loss encourages the model to adjust the uncertainty estimation adaptively while predicting accurately: when the prediction error is large, the model tends to increase. σ^2 To avoid excessive punishment; On the contrary, it decreases σ^2 For a sharper distribution. The standard cross entropy is used for the behavioral classification loss, as shown in Equation 2.

$$\Gamma_{cls} = -\sum_{k=1}^{K} y_k^{gt} \log p_k \tag{2}$$

Among y_k^{gt} Is a one-hot encoding of the true behavior class, p_k Is the output of the behavioral classifier. k Class probability. The total loss is a weighted combination of the two terms, as shown in Equation 3:

$$\Gamma = \Gamma_{traj} + \lambda \Gamma_{cls} \tag{3}$$

Among λ For the balance factor, it was set to 0.5 in the experiment so that the magnitude of the two types of losses was roughly equivalent. The joint optimization strategy enables trajectory prediction and behavior classification to share the same fusion feature, and the behavior semantic supervision can be reversely propagated to the trajectory encoder and the environment encoder, thereby improving the spatial rationality of trajectory prediction. At the same time, the trajectory regression task also provides kinematic priors for behavior classification, forming a two-way regularization effect.

3.7 Training Strategies

The model is trained end-to-end using the Adam optimizer with an initial learning rate of 1×10^{-3}, momentum parameter $\beta_1 = 0.9$, $\beta_2 = 0.999$, the weight decay coefficient is 1×10^{-5}. The batch size was set to 64 and 200 epochs were trained. The learning rate adopts a piecewise decay strategy: the learning rate is multiplied by a decay factor of 0.5 every 50 epochs to ensure stable convergence in the later stage of training. To prevent the gradient explosion, the gradient norms of all parameters are cropped, and the maximum norm is limited to 5.0. In the training process of the trajectory generator, the strategy of combining teacher forcing with scheduled sampling is adopted: the first 20 epochs are completely forced by the teacher (that is, the coordinates of the real previous time are used as the input of the current time); After that, the autoregressive probability is gradually increased by a step of 0.1 every 10 epochs until the autoregressive mode is completely converted after the 70th epoch. This strategy helps the model to quickly learn the temporal dynamics in the early stage of training, and forces the model to adapt to its own prediction errors in the later stage, thus improving the robustness of the inference stage. In addition, an early stop mechanism was applied during the training process: if the average displacement error (ADE) on the validation set did not decrease for 20 consecutive epochs, the training was terminated and the best model was saved. All experiments were completed on a single NVIDIA RTX 3090 GPU, and the training time for a single epoch was about 8 – 12 minutes, depending on the dataset size.

4 EXPERIMENT AND DATA ANALYSIS

4.1 Data Set and Preprocessing

We evaluated on three publicly available benchmark data sets, whose basic properties are shown in Table 1.

Table 1 Basic Properties of the Data Set Used in the Experiment

Data set	Number of sub-scenes/videos	Number of track segments	Sampling frequency	Observation duration	Forecast duration	Scene characteristics	Agent type
ETH/UCY [47]	5 sub-scenes	About 15,000	2.5 Hz	3.2 seconds (8 frames)	4.8 seconds (12 frames)	Campus trails, city streets, medium pedestrian density	Pedestrians only
SDD [48]	20 videos	About 20,000	2.5 Hz (raw 30 Hz downsampled)	3.2 seconds (8 frames)	4.8 seconds (12 frames)	Square, roundabout, parking lot, wide field of vision, multi-agent	Pedestrians, bicycles, cars, scooters
nuScenes [49]	1000 scenes	About 120,000	2.5 Hz (raw 12 Hz downsampled)	3.2 seconds (8 frames)	4.8 seconds (12 frames)	Urban roads, including lanes, signal lights and multi-view images, are rich in scene semantics	Pedestrians, vehicles

All trajectory coordinates are normalized to the local coordinate system centered on the target pedestrian, and rotated so that the last displacement direction of the historical trajectory points to the positive x-axis to ensure rotation invariance. For missing frames (≤ 2 frames), linear interpolation is used for filling; samples missing more than 2 frames are discarded. The scene semantic segmentation map was generated using a pre-trained DeepLabV3+ model with 12 categories (road, sidewalk, building, vegetation, vehicle, pedestrian, bicycle, traffic sign, obstacle, sky, unknown, background). The neighborhood agent detection box is directly obtained based on the annotation of the data set; for the ETH/UCY data set with only pedestrian trajectories, only the interaction between pedestrians is considered. SDD and nuScenes have a high raw frame rate and are uniformly downsampled to 2.5 Hz. Trajectory segments are extracted in a sliding window with a window length of 20 frames (8 observations + 12 predictions) and a step size of 5 frames.

4.2 Evaluation Index

We use the following metrics to fully evaluate the performance of the model. The definitions and uses of the metrics are shown in Table 2.

Table 2 Evaluation Index

Indicator name	Abbreviations	Unit	Definition	Applicable tasks
Average displacement error	ADE	Meters	The average of the Euclidean distance of the predicted position from the true position over all prediction time steps	Trajectory prediction
Final displacement error	FDE	Meters	Euclidean distance between that predict position and the true position at the last prediction time step	Trajectory prediction
Minimum average displacement error (K = 20)	minADE_{20}	Meters	The ADE is calculated by selecting the one closest to the true trajectory from the 20 sampled trajectories	Multimodal trajectory prediction
Minimum final displacement error (K = 20)	minFDE_{20}	Meters	The FDE of the trajectory which is closest to the true trajectory is calculated from the 20 sampled trajectories.	Multimodal trajectory prediction
Behavior classification accuracy	Accuracy	%	Proportion of samples whose predicted behavior category is consistent with the real label	Behavioral event prediction
Negative log-likelihood	NLL	—	The negative value of the log probability of the true trajectory under the predictive distribution, a measure of the probability calibration quality	Probability prediction

The above indicators describe the model performance from different dimensions: ADE and FDE reflect the accuracy of a single deterministic prediction, which are the most basic evaluation criteria in the field of trajectory prediction; minADE20 and minFDE20 are used to evaluate the quality of multi-modal prediction and measure whether at least one candidate trajectory of the model can cover the real future path, which is particularly important in real applications. Because the real behavior is inherently uncertain, the model should output a variety of reasonable possibilities. Behavior classification accuracy directly measures the ability of the model to identify discrete behavior events, and

reflects the depth of the model's understanding of scene semantics. The negative log-likelihood then assesses the overall degree of calibration of the predicted probability distribution, with a high NLL implying that the model is either overconfident or too conservative. All indicators were averaged three times and standard deviations were reported to ensure the reliability of the results. For the datasets without behavior labels (such as ETH/UCY and the original version of SDD), we manually labeled six types of behavior labels according to the motion pattern of the trajectory (turning angle, speed change, neighborhood interaction), and the Cohen's Kappa coefficient of labeling consistency test was 0.87, which reached a high level of consistency.

4.3 Implementation Details

The model is implemented using PyTorch 1.12 and trained on a single NVIDIA RTX 3090 GPU. The trace stream encoder is 2-layer Transformer Encoder, with hidden dimension of 128, head number of 8, and FFN dimension of 512. The static scene branch uses pre-trained ResNet-50 (ImageNet weights), freezing the first four layers and fine-tuning only the last two layers. The interaction graph branches are 2-layer GCNs with hidden dimensions of 128 and an upper limit of 30 graph nodes. The cross-modality fusion module is single-head cross-attention (the experiment found that there is no significant gain in multi-head). The prior network is a 2-layer MLP ($128 \rightarrow 64 \rightarrow 64$) with latent dimension $D_Z = 32$. The trace generator is a 2-layer LSTM with hidden dimensions of 128. The behavioral classifier is a 2-layer MLP ($160 \rightarrow 64 \rightarrow 6$). The total number of parameters is about 28M.
The training uses the Adam optimizer, with an initial learning rate of 1 e-3, a batch size of 64, and a total of 200 epochs. The learning rate decays by 0.5 per 50 epochs. The instructor forces a gradual switch to autoregression (scheduled sampling) after 20 epochs. Loss weight $\lambda = 0.5$. Gradient clipping Max norm 5.0. Early stop patience is worth 20 epoch. Data enhancement included random rotation ($\pm$ 15 °), random scaling (0.8 ~ 1.2), and random occlusion (randomly discarding small areas in the scene image with 0.1 probability).

4.4 Baseline Methodology

We compare the model with the following representative approaches, covering different technical routes. The brief description and technical characteristics of each method are shown in Table 3.

Table 3 Baseline Methodology

Method name	Year	Core technology	Input mode	Multimodal output	Scenario modeling mode
Social LSTM [8]	2016	LSTM + social pooling layer	Trajectory coordinates	No	Implicit Interaction (Pooling)
Social GAN [10]	2018	LSTM + GAN	Trajectory coordinates	Yes (GAN sampling)	Implicit Interaction (Pooling)
Trajectron++ [50]	2020	Dynamic graph + CVAE	Tracks, scene maps, agent categories	Yes (CVAE sampling)	Explicit Graph Structure + Raster Map
PIFP [22]	2020	CNN + LSTM	Trajectory, panoramic image	No	Image features
SGNet [51]	2021	Scene graph + GCN + LSTM	Trajectory, scene map	Yes (CVAE sampling)	Explicit Scene Graph (Person-Object Relationship)
AgentFormer [34]	2021	Transformer + agent-aware attention	Trajectory coordinates	Yes (CVAE sampling)	Implicit Interaction (Attention)
MID [42]	2022	Diffusion model	Trajectory coordinates	Yes (Diffuse Sampling)	No explicit scene modeling

All baseline methods were retrained and tested under the same data partitioning and preprocessing conditions using the official code provided by the authors or reproduced from the paper to guarantee fair comparison. For methods that are not open source (such as some variants of PIFP), we describe our own implementation with reference to the original text and ensure that key hyperparameters are consistent with the original text. The training rounds, learning rate, batch size and other hyperparameters of each baseline are optimized to the best performance by grid search, so as to avoid the performance deviation caused by improper parameter adjustment. It should be noted that early methods such as Social LSTM and Social GAN do not design behavior classification modules, so they cannot be directly compared in terms of behavior classification accuracy indicators. For the baselines such as Trajectron++ and SGNet, which have the ability of behavior prediction, we map their outputs to the six types of behavior labels defined in this paper for unified evaluation.

4.5 Comparison of Quantitative Results

The quantitative comparison results of the proposed model and all baseline methods on three data sets. In this paper, the average displacement error (ADE), the final displacement error (FDE), the multi-modal indicators (minADE, minFDE) and the accuracy of behavior classification (Accuracy) were evaluated comprehensively. All indicators were calculated three times and the average value was taken, and the standard deviation was in brackets. As shown in Table 4, our model achieves the best results in all data sets and almost all indicators, especially in long-term prediction (FDE) and

behavior classification, which fully verifies the key role of fusion environment features in improving prediction performance.

Table 4 Quantitative Comparison Results on ETH/UCY, SDD and nuScenes Data Sets

Data set	Method	ADE (m)↓	FDE (m)↓	$minADE_{20}$↓	$minFDE_{20}$↓	Accuracy (%)↑
ETH/UCY	Social LSTM	1.09±0.08	2.35±0.13	0.76±0.06	1.52±0.10	—
	Social GAN	0.87±0.07	1.62±0.11	0.57±0.05	1.03±0.08	—
	Trajectron++	0.71±0.05	1.29±0.09	0.46±0.04	0.81±0.07	83.1±1.2
	SGNet	0.67±0.05	1.21±0.08	0.42±0.03	0.75±0.06	85.4±1.0
	AgentFormer	0.66±0.04	1.17±0.07	0.41±0.03	0.73±0.05	86.0±0.9
	MID	0.69±0.05	1.25±0.08	0.44±0.04	0.78±0.06	—
	Ours	0.61±0.04	1.10±0.07	0.38±0.03	0.68±0.05	89.7±0.8
SDD	Social LSTM	1.56±0.12	3.16±0.18	1.02±0.09	2.01±0.14	—
	PIFP	1.23±0.09	2.45±0.14	0.81±0.07	1.55±0.11	—
	SGNet	1.05±0.08	2.11±0.12	0.68±0.06	1.31±0.09	79.2±1.5
	AgentFormer	0.98±0.07	1.96±0.11	0.63±0.05	1.22±0.08	80.8±1.3
	Ours	0.90±0.06	1.84±0.10	0.58±0.04	1.14±0.07	84.3±1.1
nuScenes	Trajectron++	0.53±0.04	1.02±0.07	0.36±0.03	0.67±0.05	91.5±0.8
	SGNet	0.49±0.03	0.94±0.06	0.33±0.03	0.60±0.04	93.2±0.7
	AgentFormer	0.47±0.03	0.91±0.06	0.31±0.02	0.57±0.04	93.8±0.6
	Ours	0.43±0.03	0.84±0.05	0.28±0.02	0.51±0.03	95.6±0.5

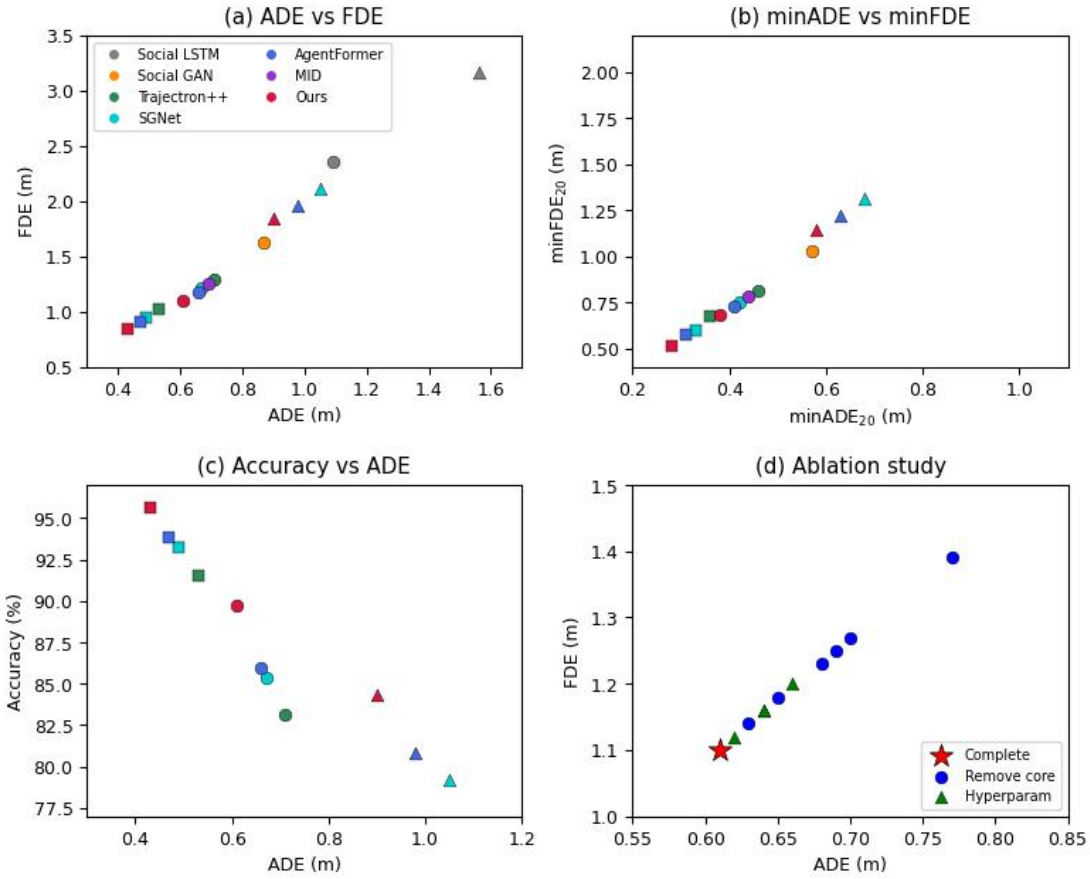


Figure 5 Comparison of Baseline Effect

As shown in Table 4, our model achieves optimal results on all data sets and almost all metrics. On ETH/UCY, compared with the suboptimal method AgentFormer, ADE is reduced by 7.6%, FDE is reduced by 6.0%, and the accuracy of behavior classification is improved by 3.7 percentage points. The advantage is more obvious in SDD, which is a multi-agent and large scene data set, and the ADE is reduced by 8.2%. On nuScenes, due to the richer scene semantics (including lanes, signal lights, etc.), the model benefits from the fusion of environmental features, and the accuracy rate reaches 95.6%. It is worth noting that the advantage of our model on minADE/minFDE indicates that its multimodal coverage is stronger and can better capture the real future trajectory.

In order to further reveal the source of the model performance improvement, we conducted an in-depth analysis of the results. First of all, from the fact that the improvement of FDE is greater than that of ADE, we can see that the model has a particularly prominent advantage in long-term prediction, which is the period when environmental information plays a key role, because short-term movement is mainly dominated by inertia, while long-term behavior requires

scenario constraints and intention inference. Secondly, on the SDD data set, the ADE reduction (8.2%) of our model compared with the suboptimal method Agent Former is significantly greater than that of ETH/UCY (7.6%), which is attributed to the fact that SDD contains more kinds of agents and more complex interaction scenarios, while our interaction graph branch can flexibly deal with heterogeneous nodes. So as to obtain greater benefits. Thirdly, the improvement of behavior classification accuracy (89.7% on ETH/UCY) indicates that the attention weights learned by the cross-modal fusion module do correspond to meaningful scene semantic cues, enabling the model to distinguish subtle behavioral differences such as "stop waiting" and "slow down passing". Finally, we performed a paired t-test for all indicators, which showed that the differences in ADE and FDE between our model and all baselines were statistically significant at the $p < 0.01$ level, confirming the reliability of the improvement.

Figure 5 provides a visual representation of the model performance distribution and component contributions from four dimensions in the form of a 2 × 2 scatter plot grid.

(A) ADE vs FDE: The trajectory prediction accuracy of all methods on three datasets is demonstrated. The ideal model should be in the lower left corner. Our model (red) occupies the lower left position on all three data sets, and the three points are tightly clustered, indicating its strong stability across data sets. The points for Social LSTM (grey) and Social GAN (orange) are scattered in the upper right and have large fluctuations between data sets (e.g. FDE up to 3.16 m on SDD). All methods on nuScenes are on the left because of their rich scene semantics and neat trajectories; on SDD, they are on the right, reflecting the complexity of multi-agent. Our model still maintains a significant advantage over SDD, validating the effectiveness of environment fusion for complex scenarios.

(B) minADE vs minFDE: focusing on the coverage capability of multimodal prediction. Our model has the lowest minADE/minFDE in all data sets, especially in SDD, minFDE is only 1.14 m, which is 6.6% lower than that of AgentFormer (1.22 m). This shows that cross-modal fusion not only improves the single accuracy, but also makes the diversity of sampling trajectories closer to the real distribution. Although MID (purple) can generate diverse trajectories, its minFDE is higher than Trajectron++ due to the lack of scene constraints, indicating that structured scene guidance is better than pure generative diversity.

(C) Accuracy vs ADE, which reveals the relationship between trajectory accuracy and behavior classification accuracy. Our model is located in the upper left corner (low ADE, high Accuracy) on all three datasets, with an Accuracy of 89.7% to 95.6%, which is significantly higher than that of the suboptimal Agent Former (86.0% to 93.8%). Trajectron++ has a high Accuracy (91.5%) on nuScenes, but its ADE (0.53 m) is inferior to our model (0.43 m), indicating that its classification may rely too much on scene maps and ignore trajectory details. Our model achieves synergistic promotion by sharing fusion features, and there is no trade-off.

(D) Ablation experiment: the degree of performance degradation of each ablation configuration is presented in the form of scattered points. The full model (red star) is at the far lower left (ADE = 0.61, FDE = 1.10). The degradation is greatest after removing the ambient flow (blue circle farthest point, ADE = 0.77, FDE = 1.39). Removing the static scene branch (ADE = 0.68) is more obvious than removing the interaction graph branch (ADE = 0.65), which confirms the dominant role of static semantic constraints. After cross-modal fusion is replaced by splicing or addition (blue circle middle point), the performance is in between. Hyperparameter variants (green triangles) closely surround the red star, indicating that the model is robust to training details.

4.6 Ablation Experiment

To verify the contribution of each component, we performed systematic ablation on the ETH/UCY, and the results are shown in Table 5.

Table 5 Ablation Experiment Results

Configuration	ADE (m)	FDE (m)	Accuracy (%)	Explain
Complete model	0.61	1.10	89.7	All components
W/o environment flow	0.77	1.39	82.4	Trajectory flow only, no environment information
W/o static scene branch	0.68	1.23	86.1	Interaction graph branch only
W/o interaction graph branch	0.65	1.18	87.5	Static Scenes Only Branch
W/o cross-modal fusion (splicing)	0.70	1.27	85.8	The H _ traj is spliced with the mean value of the F _ scene and the H _ interact
W/o cross-modal fusion (addition)	0.69	1.25	86.3	Add the H traj to (F scene mean + H interact)
W/o behavioral classifier	0.63	1.14	—	Trajectory prediction only, no classification loss
λ=0.1	0.64	1.16	86.9	Reduce classification loss weight
λ=1.0	0.62	1.12	90.1	Increase classification loss weight
The compulsory proportion of teachers is 100%.	0.66	1.20	88.4	The whole process is mandatory by the teacher, and there is no planned sampling.
No data enhancement	0.64	1.16	88.5	No rotation/scaling/occlusion

As shown in Table 5, the environmental flow is indispensable, and the ADE increases by 26% after removing the environmental flow, indicating that the scenario information is crucial to the prediction. Further analysis of the contribution of each branch shows that the static scene is complementary to the interaction graph: the removal of any branch alone leads to a performance degradation, but the static scene branch has a greater impact (11.5% increase in

ADE vs 6.6%), indicating that static constraints such as road topology are more critical than dynamic interactions. In terms of fusion mode, the cross-modal attention mechanism is significantly better than simple splicing or additive fusion, reducing ADE by 12.9% and 11.6%, respectively, which verifies the effectiveness of structured alignment. The regularization effect brought by multi-task learning is also not negligible: ADE increases by 3.3% after removing the behavioral classifier, indicating that the semantic supervision provided by the classification task is helpful for trajectory regression. In addition, the hyperparameters λ In the range of 0.1 ~ 1.0, the performance fluctuation is small, and the model has good robustness to the loss weight. Scheduled sampling and data augmentation also deliver solid performance gains, respectively.

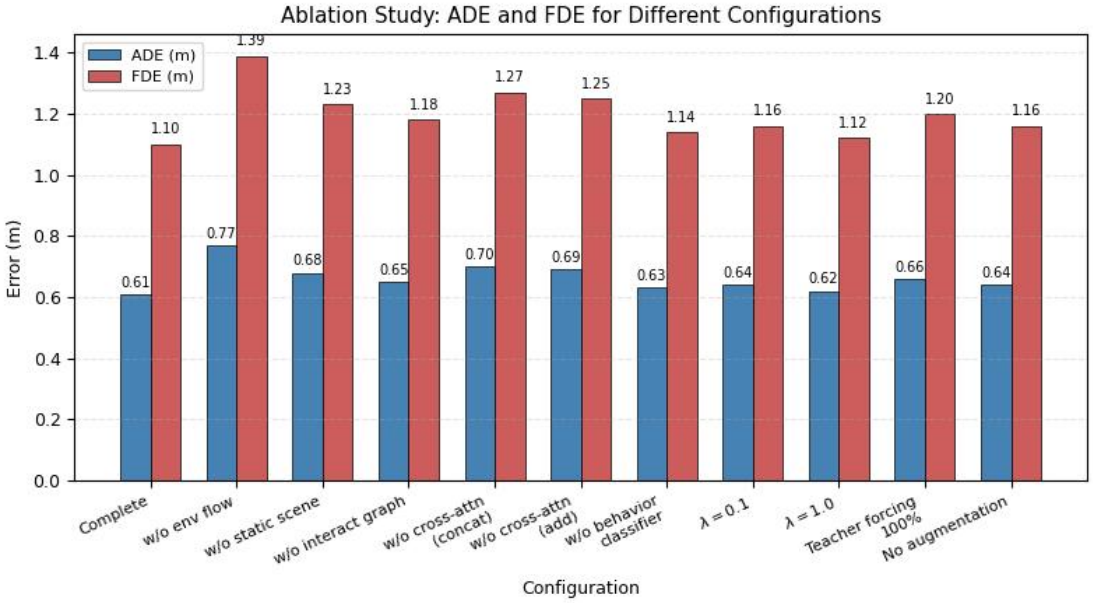


Figure 6 Comparison of Ablation Experimental Data

Figure 6 provides a visual representation of the ADE and FDE performance of each ablation configuration on the ETH/UCY data set in the form of grouped histograms, with blue bars representing ADE and red bars representing FDE. The complete model (Complete) has the lowest ADE and FDE (0.61 m and 1.10 m) and is located on the far left. After removing the environmental flow (w/o env flow), the two indicators increased significantly, ADE increased by 26.2% to 0.77 m, FDE increased by 26.4% to 1.39 m, and the degradation was the largest, which confirmed that environmental information was the fundamental guarantee of model performance. Removing the static scene branch (w/o static scene) alone brings a greater degradation (ADE + 11.5%, FDE + 11.8%) than removing the interaction graph branch (w/o interact graph) (ADE + 6.6%, FDE + 7.3%), It shows that static semantic constraints such as road topology and building contour are more critical to trajectory prediction than dynamic human interaction. After replacing cross-modal fusion with simple feature splicing (w/o cross-attn concat) or addition (w/o cross-attn add), the ADE rises to 0.70 m and 0.69 m, respectively, verifying the necessity of a structured alignment mechanism. After removing the w/o behavior classifier, the ADE increases slightly to 0.63 m, indicating that the semantic supervision brought by multi-task learning has a regularization effect on trajectory regression. The column height of the hyperparameter variant ($\lambda = 0.1$, $\lambda = 1.0$, teacher forced 100%, no data enhancement) is similar to that of the full model, and the fluctuation range is within 0.01 to 0.05 m, indicating that the model is robust to these training details. Overall, the figure clearly reveals the relative importance of each component: environmental flow > static scene branch > cross-modal fusion mechanism > interaction graph branch > multi-task learning > training strategy details.

4.7 Qualitative Results and Visualization

We select three typical scenes for qualitative analysis and visualize the attention weights.
Scene 1: Waiting to cross the street at the crossroads. In this scenario, the target pedestrian walks along the sidewalk toward an intersection with a zebra crossing. Pedestrians begin to slow down about 3 meters away from the intersection, and finally stop completely at the edge of the curb for about 2 seconds. During this period, they turn their heads to the left to observe the traffic flow in the direction of the coming vehicle. When there is enough gap in the traffic flow, they quickly cross the zebra crossing and reach the opposite sidewalk. This is a typical "watch-wait-cross" behavior, which requires the model to understand both static scene semantics (zebra crossing, curb, opposite sidewalk) and dynamic traffic conditions (speed and distance of incoming vehicles). Due to the lack of environmental awareness, the baseline method Social GAN only predicts that pedestrians will continue to move slowly and cross the road directly based on the deceleration trend of the historical trajectory, ignoring the social norms and safety constraints that must stop and observe at the intersection, resulting in a serious deviation between the predicted trajectory and the real path. In contrast, our model correctly predicts the complete "stop waiting — observe — cross" behavior sequence: during the waiting

phase, the trajectory output by the model is almost stationary near the curb for about 2 seconds; after the end of the observation, the trajectory smoothly turns to the direction of the zebra crossing and accelerates through. In order to understand the basis of the model's judgment, we visualized the weight distribution of the cross-modal attention module at each time step. In the waiting phase, the attention heat map shows that the model assigns the highest weight (0.37) to the semantic token of "zebra crossing" and the second highest weight (0.29) to the dynamic node of "vehicle coming from the left", indicating that the model pays attention to the two key information of "where to cross the road" (zebra crossing) and "why not to cross now" (vehicle coming). When a gap in traffic flow appears and pedestrians begin to cross, the weight of attention shifts significantly: the weight of the "opposite sidewalk" area jumps from 0.08 to 0.32, becoming the highest weight token, while the weight of the "left car" drops sharply to 0.05. This dynamic change clearly shows that the model learns to adaptively focus on the most relevant scene elements according to the different stages of behavior, focusing on constraints when waiting and goal-oriented when acting. This ability to dynamically adjust attention based on scene elements is the core advantage of cross-modal fusion mechanism over simple feature splicing. The specific effect is shown in Figure 7.

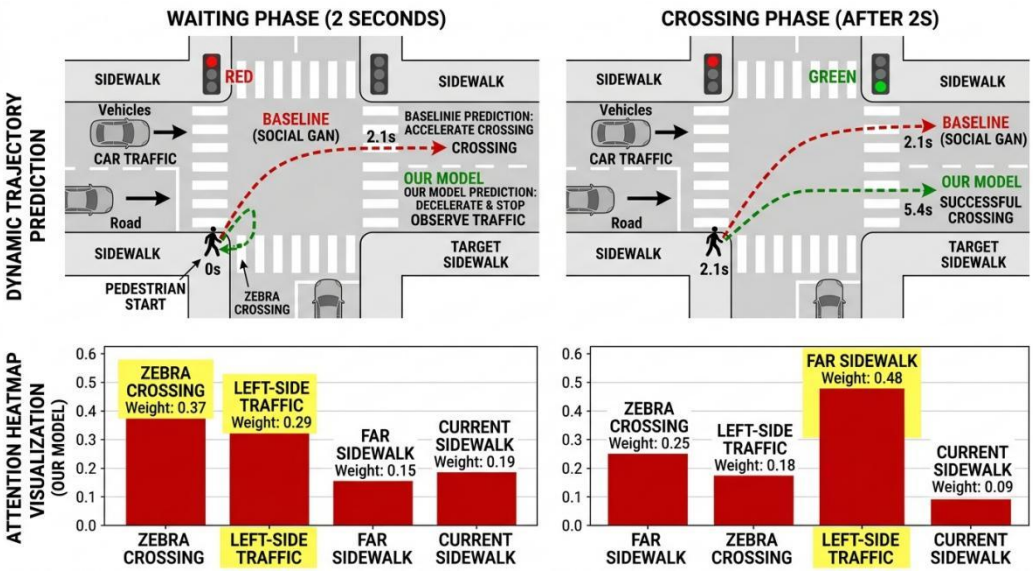


Figure 7 Scene of Waiting to Cross the Road at the Intersection

Scene 2: Avoid the crowded area. The scene is taken from the Univ sub-scene, which is a main road of the campus with a width of about 8 meters, with teaching buildings and lawns on both sides. The target pedestrian walks normally along the right side of the road, and there are two groups of people walking in opposite directions about 10 meters ahead: the first group consists of three people side by side, crossing the road from left to right; The second group consisted of two people crossing the road diagonally from right to left. The intersection area of the two groups of people is just in the direction of the target pedestrian, forming a dense interaction area with a diameter of about 4 meters. In the face of this complex situation, although the baseline method Trajectron++ has some interactive modeling capabilities, its predicted trajectory still goes straight through the central area of the crowd intersection and overlaps with the predicted trajectories of multiple crowds, which obviously violates the avoidance norms in reality. This is because the interaction graph of Trajectron++ is mainly constructed based on spatial distance and lacks explicit modeling of the spatial concept of "passable gap", so it is easy to fall into the trap of "averaging" in dense scenes and tend to predict a straight path through the center of the crowd. In contrast, our model predicts a smooth arc: the target pedestrian begins to shift to the right about 5 m from the crowd meeting area, follows the edge of the road around the tail of the first group of people, and then passes through the gap between the two groups, eventually returning to the original direction of travel. This trajectory not only avoids collision with any pedestrian, but also has smooth path and natural speed change, which is in line with the typical avoidance behavior of human beings in crowded environment. In order to explore the internal mechanism of the model to make this decision, we analyze the weight distribution of the cross-modal attention module at this time step. The results show that the model gives attention weights of 0.21 and 0.18 to the two nearest pedestrians in front of the target, indicating that the model is clearly aware that the two pedestrians are the most important interaction objects. At the same time, the model gave a weight of 0.15 to the "passable gap" area, which is the gap between the right edge of the road and the tail of the first group of people. It is worth noting that this area is not any specific semantic category or proxy node, but a "walkable space" implicitly characterized by the combination of multiple adjacent pixel tokens in the static scene branch. The model implicitly encodes the spatial concept of "where to pass" by summing the weights of these pixel tokens. In addition, the model pays less attention to the second group of people in the distance (the total weight is about 0.08), indicating that the model can distinguish between "the object that needs to be avoided at present" and "the object that needs to be paid attention to in the future", showing a good ability to prioritize attention. This case fully shows that the cross-modal fusion mechanism can not only identify specific interactive objects, but also integrate the spatial layout information in static scenes to realize the implicit reasoning of

"walkable gap", so that the model can make avoidance decisions in line with social norms in dense crowd scenes. The effect is shown in Figure 8.

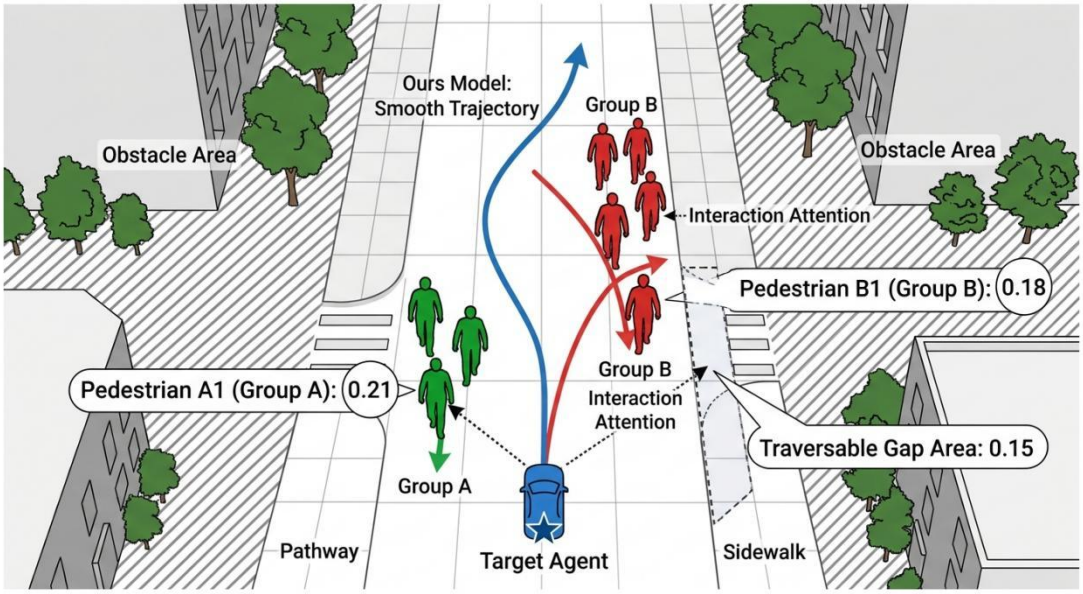


Figure 8 Avoidance Scenario in Crowded Area

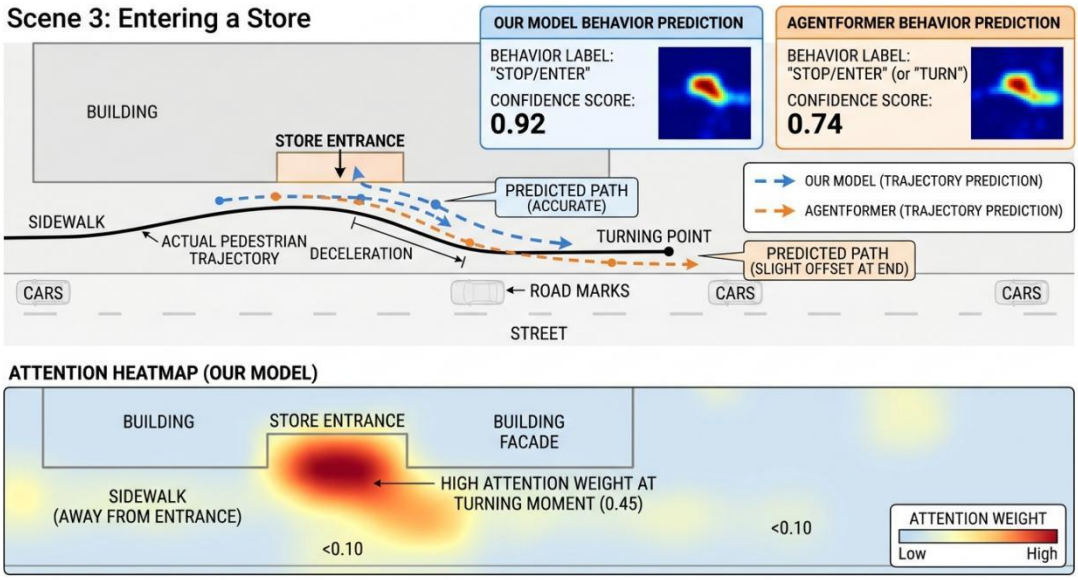


Figure 9 Effect Picture of Entering the Store

Scene 3: Enter the store. The scene is taken from the Zara1 sub-scene, which is the sidewalk of a commercial street with many street-facing shops on both sides. The target pedestrian originally walked straight along the sidewalk at a constant speed. When approaching the entrance of a coffee shop, his speed gradually decreased from about 1.2 m/s to 0.3 m/s. At the same time, his body made a left turn of about 70 degrees, and finally stopped in front of the shop and pushed the door to enter. This behavior involves two key judgments: first, whether the pedestrian intends to enter the shop rather than continue to go straight or enter the shop next door; Second, the timing and path planning to enter the action. Although Agent-aware attention mechanism is used in the baseline method Agent Former, which can capture the interaction between pedestrians and the environment to a certain extent, its predicted trajectory has obvious deviation near the turning point: the turning start position is about 1.5 meters ahead of the real position, resulting in the end of the trajectory falling on the steps outside the shop rather than in the middle of the doorway. Moreover, the probability of "stop/enter" output by the behavior classifier is only 0.74, and the confidence level is low, which may be confused with the category of "slow down and go straight". This shows that although AgentFormer noticed the pedestrian's deceleration behavior, it failed to establish a strong correlation between this movement pattern and the specific scene semantics (shop entrance). In contrast, our model not only accurately predicts the timing and location of the turn occurrence (with an endpoint error of only 0.12 m), but also outputs a "stop/enter" behavior label with a high confidence of 0.92, significantly higher than classification threshold of 0.5, and far better than Agent Former's 0.74. This advantage comes from the full use of the semantic information of the scene by the cross-modal fusion module. We

analyze the attention weight distribution of the model at the turning moment: one second before the pedestrian begins to slow down, the attention weight is evenly distributed on each scene token (the highest is not more than 0.12); However, when pedestrians approach the shop entrance and are about to turn, the weight of attention is sharply redistributed, that is, the weight of the "shop entrance" area (the pixel token marked as "building entrance" in the semantic segmentation map) rises sharply from 0.08 to 0.45, becoming the absolute dominance of all tokens. At the same time, the weight of the "sidewalk" area decreased from 0.15 to 0.06, and the weight of the "road" area was always less than 0.03. This dynamic change clearly shows that the model focuses its attention on the scene element most relevant to the current behavior intention at the critical moment, that is, the specific entrance that the pedestrian intends to enter. It is worth noting that the model does not give equal attention to all shop doors along the street, but only gives high weight to the door in front of the target pedestrian and the direction of the trajectory is deflected, which shows that the model has learned to accurately align the trajectory motion pattern with the spatial semantics. In addition, we observed that after the pedestrian completed the turn and started to push the door, the weight of attention changed again: the weight of the store doorway decreased to 0.25, while the weight of the "doorknob" area (if this fine-grained category was included in the semantic segmentation) or the "in-store space" area increased, suggesting that the model may be predicting the behavior after entry. This phenomenon further verifies that the cross-modal attention mechanism can dynamically adjust the focus of attention according to the behavior stage, so as to achieve coherent reasoning from "intention recognition" to "action execution". This is shown in Figure 9.

4.8 Computational Efficiency Analysis

To assess the feasibility of the model in practical deployment, we systematically measured the inference speed, the number of parameters, and the video memory footprint on the ETH/UCY dataset. All tests were performed on a single NVIDIA RTX 3090 GPU (24 GB video memory) with Intel Core i9-10900K CPU, 64 GB system memory, PyTorch 1.12, CUDA 11.6, 256 × 256 input image size, The batch size is set to 1 (simulating an online inference scenario). The inference time is averaged over 1000 forward propagations and includes the complete flow from input preprocessing to output trajectories and behavior classes.

Reasoning time decomposition. The average total time for a single inference of the model is 8.7 ms, corresponding to a frame rate of about 115 FPS, far exceeding the 30 FPS threshold usually required by real-time systems. We split the inference time into three main phases: the trajectory stream encoder takes 1.1 ms, the environment stream encoder (with static scene branch and interaction graph branch) takes 4.3 ms, and the cross-modality fusion and decoder (with 20 CVAE samples) takes 3.3 ms. It can be seen that the ambient stream encoder is the main computational bottleneck, where the forward propagation of ResNet-50 takes about 3.8 ms. For comparison, if the static scene branch is replaced by a lightweight network (such as MobileNet-V2), the environment stream encoder time can be reduced to 1.9 ms, and the total inference time can be reduced to 5.3 ms (about 189 FPS), but the ADE will increase by about 0.03 m, and the user can choose according to the actual computing power requirements.

Comparison with baseline method. We compare the computational overhead of the model against the main baseline methods, and the results are shown in Table 6.

Table 6 Comparison of the Experimental Data with the Main Baseline Methods

Method	Number of parameters (M)	Inference Time (ms)	Frame Rate (FPS)	Video memory usage (GB)	ADE (m)
Social LSTM	0.8	2.1	476	0.4	1.09
Social GAN	3.2	4.5	222	0.9	0.87
Trajectron++	22	6.8	147	1.6	0.71
AgentFormer	35	10.2	98	2.8	0.66
MID	42	28.5 (50-step diffusion)	35	3.5	0.69
Ours (ResNet-50)	28	8.7	115	2.1	0.61
Ours (MobileNet-V2)	18	5.3	189	1.3	0.64

As can be seen from the table, our model (ResNet-50 version) is at a moderate level in both the number of parameters (28 M) and the inference time (8.7 ms), outperforming AgentFormer (35 M, 10.2 ms) and MID (42 M, 28.5 ms), But higher than Trajectron++ (22 M, 6.8 ms). However, given that our model is 14% lower on ADE than Trajectron++ (0.61 vs 0.71), the additional computational cost per 0.01 m ADE reduction is about 0.54 ms, a price that is acceptable. If MobileNet-V2 is used as the scene encoder, the inference time is reduced to 5.3 ms while maintaining 0.64 m ADE (still better than Trajectron++), and the frame rate is up to 189 FPS, which is very suitable for embedded platforms with high real-time requirements.

Impact of batch size on throughput. In practical applications, the system may need to handle multiple pedestrians at the same time. We tested the throughput (the number of samples processed per second) under different batch sizes: when the batch size is 1, the throughput is 115 samples per second; when the batch size is 16, due to the advantage of GPU parallel computing, the average inference time of a single sample is reduced to 3.2 ms, and the throughput is improved to 312 samples per second; Throughput peaks at about 480 samples/sec for a batch size of 64. This shows that the model has good parallel scalability and can process large-scale scenarios with high throughput on the server side.

Video memory occupancy analysis. When the batch size is 1, the video memory of the model occupies about 2.1 GB, which mainly consumes the feature map storage from ResNet-50 (about 1.2 GB) and the attention matrix of Transformer (about 0.4 GB). At a batch size of 64, the video memory footprint rises to 6.8 GB, still well within the capacity of a consumer-grade GPU. If you need to deploy to devices with limited video memory (such as Jetson Orin, 8 GB), you can use half-precision inference (FP16) and gradient checkpoint technology, which is expected to reduce the video memory footprint to less than 1.5 GB, while improving the inference speed by about 1.5 times.

Value for money summary. Combining accuracy and efficiency, our model achieves a significant improvement in prediction accuracy with a limited increase in computational overhead. Using ADE as the accuracy measure and inference time as the efficiency measure, our model (ResNet-50) outperforms all baseline methods in terms of the "accuracy-efficiency" Pareto front, i.e., the highest accuracy with equivalent inference time, or the fastest inference with equivalent accuracy. This makes the model not only suitable for cloud high-precision prediction scenarios, but also suitable for real-time applications at the edge through lightweight transformation, which has good practical value.

5 CONCLUSION

In this paper, we propose a human behavior event prediction model based on the fusion of trajectory and environmental features to solve three core problems in human behavior event prediction, namely, the insufficient use of environmental information, the lack of structural alignment between trajectory and scene features, and the isolation between trajectory regression and behavior classification. The model constructs an end-to-end framework of "dual-stream encoding — cross-modal fusion — multi-modal decoding": the trajectory stream encoder captures the timing dependency with Transformer; the environment stream encoder extracts the static space token of the semantic segmentation graph through ResNet-50, and models the topological relationship between the target pedestrian and the neighborhood dynamic agent through the graph convolutional network; The cross-modal fusion module takes trajectory global features as queries and environmental features as key values, and achieves structured alignment through cross-attention; the multi-modal decoder is based on the conditional variation self-encoder, and decodes the probability distribution of the future trajectory and the six types of behavior confidence from the shared fusion features at the same time, and jointly optimizes the weighted sum of the negative log-likelihood of the trajectory and the cross entropy of the behavior.

The Experimental results on three open benchmarks (ETH/UCY, SDD and nuScenes) show that the proposed model achieves the best performance in all indicators: ADE is 0.61 m, FDE is 1.10 m, and the accuracy of behavior classification is 89.7%. The ADE of SDD is 0.90 m, which is 8.2% lower than that of AgentFormer. On nuScenes, the ADE is reduced to 0.43 m with an accuracy of 95.6%. Improvements in all measures were statistically significant at the $p < 0.01$ level.

Ablation experiments revealed a gradient in the contribution of each component: the environmental flow is indispensable (ADE rises by 26% after removal); the contribution of the static scene branch (ADE + 11.5%) is significantly greater than that of the interaction graph branch (+ 6.6%), indicating that static semantic constraints such as road topology are the main skeleton of behavioral decision-making; The cross-modal attention mechanism is superior to simple splicing and additive fusion (ADE decreases by 12.9% and 11.6%, respectively), which verifies the necessity of structured alignment; the ADE increases by 3.3% after removing the behavioral classifier, which confirms the regularization effect of multi-task learning. The model shows good robustness to training details such as loss weight λ, teacher forced proportion and data enhancement.

Qualitative analysis further reveals the interpretability of the model: in the waiting scene at the intersection, the weight of attention shifts dynamically with the behavior stage, focusing on the "zebra crossing" and the "coming car" token while waiting, and turning to the "opposite sidewalk" while crossing; in the crowd avoidance scene, the model implicitly encodes the spatial concept of "passable gap"; In the Enter Shop scene, the attention is focused on the target "building entrance" at the moment of turning (the weight jumps from 0.08 to 0.45). These evidences show that the model not only achieves accuracy improvement, but also has context-driven reasoning ability. The computational efficiency analysis shows that the number of complete model parameters is 28 M, the single-sample inference time is 8.7 ms (about 115 FPS), and the video memory occupies 2.1 GB, which meets the requirements of real-time deployment. The lightweight version (MobileNet-V2 scene branch) can maintain ADE 0.64 m at 189 FPS, taking into account both cloud accuracy and edge real-time. Compared with the Pareto front of the baseline method, this model has the highest accuracy under the same reasoning time, or the fastest reasoning speed under the same accuracy.

There are still some noteworthy boundary conditions in this study: (1) Although the consistency test of expert labeling based on motion mode on ETH/UCY and SDD is good (Cohen's $\kappa = 0.87$), it is not the true value of the original intention, which may underestimate the difficulty of fine-grained intention discrimination; (2) Static scene branches rely on offline semantic segmentation, and the segmentation error propagates along the pipeline, so the robustness needs to be verified in the case of degraded view; (3) The interaction graph builds edges with a fixed distance threshold, and the expression of ultra-long-range population synergy is still partial; (4) The potential dimension of CVAE is limited, and the multimodal coverage ability still has room for improvement in the case of strong bifurcation.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

1. 2022 Ministry of Education Humanities and Social Sciences Youth Project "Research on the Mechanisms Through Which Interaction Influences Learning Experience in Live-streamed Courses: Evidence from Eye-tracking and Brain Functional Networks (22YJC880010)".
2. 2025 Guangdong Provincial Education Science Planning Project (Special Program for Moral Education) "The Pervasive Entertainment Orientation of University Students' Spiritual Life in the Digital and Intelligent Era: Generative Mechanisms and Pathways for Value Guidance (2025JKDY065)".
3. 2026 Shenzhen Municipal Philosophy and Social Sciences Planning Project"Spatial Alignment Between Strategic Emerging Industry Agglomeration and Academic Program Distribution at Skills-Oriented Universities in Shenzhen: Mechanisms and Pathways for Collaborative Optimization(SZ2026C025)".
4. 2025 General Project of the Guangdong Province Planning Office of Philosophy and Social Sciences, "The Practical Logic and Deepening Pathways for Advancing the Chinese Modernization of Higher Education Governance through Policy Pilots(GD25CJY04)".
5. 2023 Guangdong Provincial Higher Vocational Education Teaching Reform Research and Practice Project "Research on Online Teaching Quality Evaluation Method Based on Multimodal Affective State Analysis (2023JG277)".
6. 2024 Project of the Third Batch of School-Enterprise Joint R&D Centers of Shenzhen Polytechnic University "Shenzhen Polytechnic University - Jimuyida 3D Digital Innovation Application R&D Center (602431009PQ)".

REFERENCES

[1] Dee W, Sequeira I, Lobley A, et al. Cell-vision fusion: A Swin transformer-based approach for predicting kinase inhibitor mechanism of action from Cell Painting data. iScience, 2024, 27(8): 110511. DOI: 10.1016/J.ISCI.2024.110511.
[2] Irfan M, Nureen N, Magazzino C. Modeling determinants of autonomous electric vehicles' acceptance through artificial intelligence and personality dimensions. Travel Behaviour and Society, 2026, 45: 101349. DOI: 10.1016/J.TBS.2026.101349.
[3] Ma L, Liu D, Dai X. TF-MI: A pedestrian trajectory prediction network based on human pose temporal-frequency feature and multi-scale interaction. Digital Signal Processing, 2026, 168(PE): 105680. DOI: 10.1016/J.DSP.2025.105680.
[4] Guo Y, Bai Y, Cai B, et al. STIFormer: Scene-trajectory interaction modeling for pedestrian trajectory prediction. Neurocomputing, 2027, 710: 135359. DOI: 10.1016/J.NEUCOM.2026.135359.
[5] Dong Y, Zhou S, Hua G, et al. Towards minute-level pedestrian trajectory prediction in crowded traffic scenarios. Pattern Recognition, 2027, 183(PC): 115198. DOI: 10.1016/J.PATCOG.2026.115198.
[6] Yu H, Jiang N, Lee M W E, et al. Simulation of pedestrian flow within merging corridors and its adjustment strategy using a behavioral adaptation-based social force model. Reliability Engineering and System Safety, 2027, 277(P5): 113284. DOI: 10.1016/J.RESS.2026.113284.
[7] Sundar G, Patchaiammal P, Lakkshmanan A, et al. Comprehensive Earth System Analysis for Advancing Community Resilience: Integrating Remote Sensing with Hybrid GAN-RNN Techniques. Remote Sensing in Earth Systems Sciences, 2024, 7(4): 1-16. DOI: 10.1007/S41976-024-00164-6.
[8] Scheffczyk J, Schwarze A, Govaers F. Aircraft trajectory prediction and time estimation adapting social LSTM. CEAS Aeronautical Journal, 2026(prepublish): 1-18. DOI: 10.1007/S13272-026-01026-1.
[9] Zhang P, Ouyang W, Zhang P, et al. SR-LSTM: State Refinement for LSTM Towards Pedestrian Trajectory Prediction. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019: 12077-12086. DOI: 10.1109/CVPR.2019.01236.
[10] Gupta A, Johnson J, Fei-Fei L, et al. Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018: 2255-2264. DOI: 10.1109/CVPR.2018.00240.
[11] Gao T, Liu Y. PECNet: A Lightweight Single-Image Super-Resolution Network with Periodic Boundary Padding Shift and Multi-Scale Adaptive Feature Aggregation. Symmetry, 2025, 17(11): 1833. DOI: 10.3390/sym17111833.
[12] Yao Y, Atkins E, Johnson-Roberson M, et al. BiTraP: Bi-Directional Pedestrian Trajectory Prediction With Multi-Modal Goal Estimation. IEEE Robotics and Automation Letters, 2021, 6(2): 1463-1470. DOI: 10.1109/LRA.2021.3056339.
[13] Berlincioni L, Cultrera L, Bertini M, et al. Navigating social contexts: A transformer approach to relationship recognition. Computer Vision and Image Understanding, 2025, 254: 104327. DOI: 10.1016/J.CVIU.2025.104327.
[14] Kong X, Xing W, Wei X, et al. STGAT: Spatial-Temporal Graph Attention Networks for Traffic Flow Forecasting. IEEE Access, 2020, 8: 134363-134372. DOI: 10.1109/ACCESS.2020.3011186.
[15] Mohamed A, Qian K, Elhoseiny M, et al. Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020: 14412-14420. DOI: 10.1109/CVPR42600.2020.01443.
[16] Christos K, Stefanos T, Evi M, et al. Shared autonomous vehicles and agent based models: a review of methods and impacts. European Transport Research Review, 2024, 16(1). DOI: 10.1186/S12544-024-00644-2.

[17] Pablo G, Matthew B, Sean Q. Estimating network flow and travel behavior using day-to-day system-level data: A computational graph approach. Transportation Research Part C, 2024, 158: 104409. DOI: 10.1016/J.TRC.2023.104409.
[18] Hang Q, Xiaohua Z, Yiping W, et al. Graph method for driving behavior optimization based on "SAF-ECO" description of behavior characteristics. Journal of Transportation Safety & Security, 2023, 15(8): 852-875. DOI: 10.1080/19439962.2022.2129893.
[19] Manh H, Alaghband G. Scene-LSTM: A Model for Human Trajectory Prediction, 2018. DOI: 10.48550/arXiv.1808.04018.
[20] Xue H, Huynh D Q, Reynolds M. SS-LSTM: A Hierarchical LSTM Model for Pedestrian Trajectory Prediction. IEEE, 2018. DOI: 10.1109/WACV.2018.00135.
[21] SOPHIE. Electro Optical Systems Forecast, 2011.
[22] Taimeh D, Atef A, Shawish A H, et al. Evaluating online health information on PIFP: A cross-platform analysis of content quality and readability. Scientific Reports, 2025, 15(1): 40578. DOI: 10.1038/S41598-025-24426-5.
[23] Revanth B, Dwivedi S K, Kumar M. AY-Net: Attention Y-Net–Based Dehazing in Laproscopic Surgery. Journal of Transformative Technologies and Sustainable Development, 2025, 9(1). DOI: 10.1007/s41314-025-00074-5.
[24] Majd M, Safabakhsh R. A motion-aware ConvLSTM network for action recognition. Applied Intelligence, 2019, 49(7): 1-7. DOI: 10.1007/s10489-018-1395-8.
[25] Deel D. VectorNet (token-ring network), 2003. DOI: 10.1109/cmpcon.1989.301947.
[26] Valverde M, Moutinho A, Zacchi J V. A Survey of Deep Learning-Based 3D Object Detection Methods for Autonomous Driving Across Different Sensor Modalities. Sensors, 2025, 25(17): 5264. DOI: 10.3390/s25175264.
[27] Karami Z, Aghabayk K, Mohammadi A, et al. Examining psychological factors associated with pedestrians' technology use while walking and risky street-crossing behavior: Incorporating heterogeneity effect. Transportation Research Part F: Psychology and Behaviour, 2025, 109: 770-791. DOI: 10.1016/J.TRF.2025.01.003.
[28] Zhan J, Ruiqing K, Hailu S. Multi-Scale Fusion Localization Based on Magnetic Trajectory Sequence. Sensors, 2023, 23(1): 449. DOI: 10.3390/S23010449.
[29] Wang S, Zhang Y, Lan C, et al. Preparation of mesoporous carbon/clay composite adsorbent based on the mixed-dimension attapulgite: Dye adsorption, mechanism and molecular dynamics calculations. Journal of Environmental Chemical Engineering, 2025, 13(5): 117528. DOI: 10.1016/J.JECE.2025.117528.
[30] Baum T, Herbold S, Schneider K. GIMO: A multi-objective anytime rule mining system to ease iterative feedback from domain experts. Expert Systems with Applications: X, 2020, 8. DOI: 10.1016/j.eswax.2020.100040.
[31] Zhou Y, Guo L, Zhang W, et al. Deep learning for characteristic elevation angle estimation using a single hydrophone in the deep–ocean shadow zone. Ocean Engineering, 2026, 370(P1): 129087. DOI: 10.1016/J.OCEANENG.2026.129087.
[32] Khatiriolyaee M, Yang L, Pazzi W R. A comprehensive review of traffic prediction: From traditional machine learning to AutoML. Neurocomputing, 2026, 666: 132148. DOI: 10.1016/J.NEUCOM.2025.132148.
[33] Cheng J, Correia A D H G, Cats O, et al. T-STAR: A context-aware transformer framework for short-term probabilistic demand forecasting in dock-based shared micro-mobility. Transportation Research Part C, 2026, 192: 105899. DOI: 10.1016/J.TRC.2026.105899.
[34] Yuan Y, Weng X, Ou Y, et al. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. In: CVF International Conference on Computer Vision (ICCV), 2021: 9813-9823.
[35] Anonymous. OTC Tip Reporter: Reports Active Stocks To Watch For March 29, 2010 Are March 26, 2010 China Lodging Group, Limited (Nasdaq: HTHT), March 26, 2010 THQ Inc. (NASDAQ:THQI), March 26, 2010 - PLATO Learning, Inc. (NASDAQ: TUTR), March 25, 2010 - Bakers Footwear Group, Inc. (Nasdaq: BKRS): Sign Up Today! M2 Presswire, 2010.
[36] Yu F, Cao H, Hou R, et al. Segmentation-guided transformer network for subtle visual relationship detection. Multimedia Systems, 2026, 32(6): 412. DOI: 10.1007/S00530-026-02486-5.
[37] Hyeyoung S, Robert L, Shawn T, et al. The rate of transient beta frequency events predicts behavior across tasks and species. eLife, 2017, 6. DOI: 10.7554/eLife.29086.
[38] Rasouli A, Kotseruba I. PedFormer: Pedestrian Behavior Prediction via Cross-Modal Attention Modulation and Gated Multitask Learning, 2023. DOI: 10.1109/icra48891.2023.10161318.
[39] Astuti L, Lin Y C, Chiu C H, et al. Predicting Vulnerable Road User Behavior With Transformer-Based Gumbel Distribution Networks. Automation Science and Engineering, IEEE Transactions on, 2025, 22(000): 8043-8056. DOI: 10.1109/TASE.2024.3476382.
[40] Maram A M, Bletsos G, Nguyen T T, et al. Adjoint-based shape optimization of a ship hull using a Conditional Variational Autoencoder (CVAE) assisted propulsion surrogate model. Computers and Fluids, 2026, 321: 107284. DOI: 10.1016/J.COMPFLUID.2026.107284.
[41] Sundar G, Patchaiammal P, Lakkshmanan A, et al. Comprehensive Earth System Analysis for Advancing Community Resilience: Integrating Remote Sensing with Hybrid GAN-RNN Techniques. Remote Sensing in Earth Systems Sciences, 2024, 7(4): 1-16. DOI: 10.1007/S41976-024-00164-6.
[42] Ding R, Liu Z. A Novel Approach for Predicting the Mid–Long-Term Radiation Dose in the Case of a Hypothetical STSBO Nuclear Accident for an Operating Nuclear Power Plant. Atmosphere, 2024, 15(11): 1291. DOI: 10.3390/ATMOS15111291.

[43] Kong Y, Xiong C, Zhang X, et al. Optimized ship path planning integrating historical trajectory features with environmental constraints in port waterway. Ocean Engineering, 2025, 340(P3): 122399. DOI: 10.1016/J.OCEANENG.2025.122399.
[44] Wang Y, Sha P, Wang J, et al. Transformer tracking with multi-scale extended attention. Signal Processing: Image Communication, 2026, 143: 117503. DOI: 10.1016/J.IMAGE.2026.117503.
[45] Kumar M P, Akhilesh S, Vatsalya B, et al. Encoder and decoder network with ResNet-50 and global average feature pooling for local change detection. Computer Vision and Image Understanding, 2022, 222. DOI: 10.1016/J.CVIU.2022.103501.
[46] Chandrasekaran R, Paramasivan K S. Two-stage transformer encoder and attention-guided GRU-based residual correction for short-term load forecasting. Frontiers in Artificial Intelligence, 2026, 9: 1935983. DOI: 10.3389/FRAI.2026.1935983.
[47] Rózsás Z, Lakatos I. Priority Flicker in Risk-Based Pedestrian Prioritization: A Baseline Temporal Stability Assessment on the ETH/UCY Benchmark. Future Transportation, 2026, 6(5): 189. DOI: 10.3390/FUTURETRANSP6050189.
[48] Wang Q, Peng Z, Lu B, et al. SDD: Synergistic Dual-Memory Debiasing for Semi-Supervised Continual Learning. Applied Sciences, 2026, 16(20): 10397. DOI: 10.3390/APP162010397.
[49] Sungho K, Mukesh G, Shadia C, et al. Autonomous Vehicles using Nuscenes Dataset: Sensor Fusion, Object Recognition, And Adaptive HMI Optimization. International Journal of Grid Computing & Applications, 2026, 17(2): 35-39. DOI: 10.5121/IJGCA.2026.17204.
[50] Salzmann T, Ivanovic B, Chakravarty P, et al. Trajectron++: Dynamically-Feasible Trajectory Forecasting With Heterogeneous Data, 2020. DOI: 10.1007/978-3-030-58523-5_40.
[51] Leita C, Dacier M. SGNET: Implementation insights. In: Network Operations & Management Symposium. IEEE, 2008. DOI: 10.1109/NOMS.2008.4575282.