ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA241013

Less Is More: Efficient Brain-Inspired Learning for Autonomous Driving Trajectory Prediction

Haicheng Liao[a,1], Yongkang Li[a,1], Zhenning Li[a,*], Chengyue Wang[a], Guofa Li[b], Chunlin Tian[a], Zilin Bian[c], Kaiqun Zhu[a], Zhiyong Cui[d] and Jia Hu[e]

[a]University of Macau
[b]Chongqing University
[c]New York University
[d]Beihang University
[e]Tongji University

Abstract. Accurately and safely predicting the trajectories of surrounding vehicles is essential for fully realizing autonomous driving (AD). This paper presents the Human-Like Trajectory Prediction model (HLTP++), which emulates human cognitive processes to improve trajectory prediction in AD. HLTP++ incorporates a novel teacher-student knowledge distillation framework. The "teacher" model, equipped with an adaptive visual sector, mimics the dynamic allocation of attention human drivers exhibit based on factors like spatial orientation, proximity, and driving speed. On the other hand, the "student" model focuses on real-time interaction and human decision-making, drawing parallels to the human memory storage mechanism. Furthermore, we improve the model's efficiency by introducing a new Fourier Adaptive Spike Neural Network (FA-SNN), allowing for faster and more precise predictions with fewer parameters. Evaluated using the NGSIM, HighD, and MoCAD benchmarks, HLTP++ demonstrates superior performance compared to existing models, which reduces the predicted trajectory error with over 11% on the NGSIM dataset and 25% on the HighD datasets. Moreover, HLTP++ demonstrates strong adaptability in challenging environments with incomplete input data. This marks a significant stride in the journey towards fully AD systems.

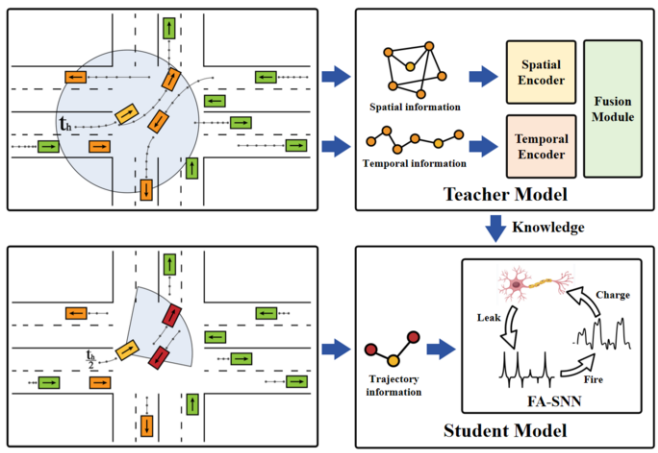

Figure 1. Illustration of our proposed model HLTP++. The "teacher" model is designed to imitate the attention distribution in human driving by capturing contextual data such as surrounding agents and temporal-spatial interactions, generating prior knowledge. The "student" model, guided by this knowledge and leveraging FA-SNN, emulates neural transmission and provides efficient and human-like trajectory prediction with only 50% of the observations provided to the "teacher" model.

1 Introduction

As the field of autonomous vehicles (AVs) approaches a transformative phase, the core challenge extends beyond engineering to endowing these systems with cognitive capabilities that parallel those of human drivers [19]. The central task within this framework is trajectory prediction, which requires a sophisticated understanding of the external driving environment coupled with a comprehension of the cognitive mechanics of human decision-making [13]. While traditional deep learning approaches in AVs have advanced in data processing and pattern recognition, they often falter in complex or incomplete data scenarios. Inspired by the ability of human drivers to navigate complicated environments through adaptability and anticipation, our research aims to replicate these human cognitive processes to improve the predictive accuracy of our trajectory prediction models.

In the intricate process of human driving decision making, a complex network of neural functions involving multiple brain regions plays a critical role. Visual processing is handled by the occipital and temporal lobes, while the prefrontal and parietal cortices handle decision-making and spatial reasoning [26]. This coordinated function allows drivers to anticipate hazards and make decisions by integrating current sensory data with past experience.

Inspired by these neurological processes, our Human-Like Trajectory Prediction (HLTP++) model seeks to mimic human decision making in driving. Leveraging knowledge distillation to balance efficiency with cognitive processing, the HLTP++ framework uses a "teacher" model designed to mimic the brain's method of visual data processing. This model uses advanced visual pooling strategies to analyze and prioritize visual input, similar to the functions of the occipital and temporal lobes. Subsequently, the "student" model incorporates a novel Spike Neural Network, FA-SNN, which simulates the

* Corresponding Author. Email: zhenningli@um.edu.mo
[1] Equal contribution.

decision-making functions similar to those of the prefrontal and parietal cortices. Our goal is to develop a model that not only processes information with brain-like efficiency, but also adapts quickly to new information while maintaining both flexibility and precision, thus overcoming the challenges associated with high parameter counts and less human-like trajectory predictions. Overall, the contributions of HLTP++ are multifaceted:

- We introduce a novel visual pooling mechanism to emulate the dynamic visual sector of human observation and adaptively self-adjust its attention to different agents in central and peripheral vision in real time under different scenes. Additionally, the Fourier Adaptive Spike Neural Network (FA-SNN) is proposed to handle traffic scenes with missing data by mimicking the neuronal pulse propagation process in human brain.
- The HLTP++ presents a heterogeneous teacher-student knowledge distillation framework by using enhanced Knowledge Distillation Modulation (KDM) for multi-level tasks. This approach automatically adjusts the ratio between loss functions, significantly facilitating model training in complex trajectory distillation situations.
- Benchmark tests on the NGSIM, HighD, and MoCAD datasets have shown that the HLTP++ outperforms existing top baselines by considerable margins, demonstrating its superior robustness and accuracy in various traffic conditions, including highways and dense urban environments. Notably, it exhibits remarkable performance even with fewer input observations and missing data.

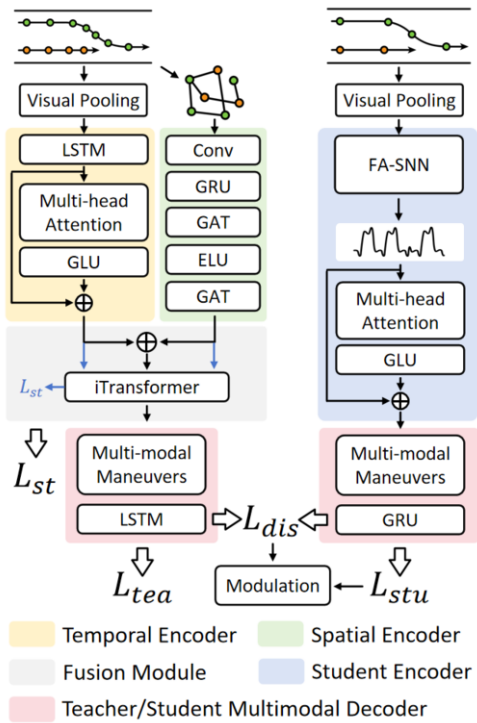

Figure 2. Overall "teacher-student" architecture of the HLTP++. The "teacher" model employs encoders to capture spatio-temporal interactions, followed by a fusion module that integrates features. The "student" model acquires knowledge from the "teacher" model through the KDM training strategy and incorporates FA-SNN for efficient, human-like trajectory prediction.

2 Related Work

Trajectory Prediction. Deep learning has significantly advanced the use of various neural network frameworks within autonomous vehicle (AV) systems, focusing extensively on improving both spatial and temporal data extraction from trajectories. Techniques such as Recurrent Neural Networks (RNNs) [30], social pooling mechanisms [12], Graph Neural Networks (GNNs) [20], attention schemes [37], generate models [16], and the Transformers architecture [38] have been instrumental. Recent research has concentrated on merging human cognitive science with trajectory prediction tasks. Li et al. [18] were pioneers in analyzing the observation habits of human drivers, recommending a focus on the central vision field, akin to how drivers emphasize their main visual area. Building on this, Liao et al. [24] mimicked the differential allocation of attention to different visual regions during driving, facilitating models to capture essential spatiotemporal relationships. In addition, several significant studies have tackled the task through the lenses of behavioral science and perceived safety [23], adeptly modeling the ongoing driving patterns of human drivers. This strategy aids models in comprehending the inherent uncertainty in traffic conditions, resulting in predictions that are more akin to human drivers. While efforts like these enhance model capabilities in safety and multimodality, the computational demands in existing models often reduce their effectiveness in real-time applications. To address these challenges, this study presents a Brain-Inspired model designed to replicate the structure and functionality of the human brain, aiming to improve the model's reasoning abilities. Compared to our previous work, HLTP [21], the newly proposed HLTP++ model shows differences in several crucial aspects: (1) HLTP++ more effectively manages the trade-off between inference speed and prediction accuracy, delivering performance on par with HLTP while needing significantly less inference time. (2) By refining a teacher-student heterogeneous knowledge distillation network, HLTP++ achieves outstanding results with fewer observation inputs than HLTP. (3) The introduction of an innovative learning strategy allows HLTP++ to attain state-of-the-art (SOTA) performance with shorter training periods, which is not only computationally efficient but also skillfully emulates human driving behavior for improved real-time prediction accuracy.

Knowledge Distillation. Originating from Hinton et al.'s foundational work [11], knowledge distillation facilitates the transfer of complex knowledge from a 'teacher' model to a more streamlined 'student' model. This method, initially developed for model size reduction, now also augments model generalization and accuracy [29], while limiting computational overhead [2]. Despite its sparse application in autonomous driving [21], this study uses knowledge distillation to embed human-like decision-making processes into AVs, ensuring that the 'student' model learns effectively and emulates pragmatic driving intelligence.

Spike Neural Network. The emerging need for edge computing solutions has catalyzed interest in third-generation neural networks, particularly Spike Neural Networks (SNNs) [32], which are inspired by biological neural processes. Known for their efficiency and low power consumption, SNNs offer significant advantages for embedded systems in vehicles [8]. However, conventional SNNs often sacrifice adaptability and temporal accuracy due to their rigid thresholding mechanism. To overcome these limitations, this study introduces the Flexible Adaptability Spike Neural Network (FA-SNN), which improves both adaptability and temporal accuracy in trajectory prediction models.

3 Methodology

3.1 Problem Formulation

The principal objective of this study is to predict the trajectory of the target vehicle, encompassing all traffic agents within the sensing range of the AV in a mixed autonomy environment where AVs coexist with human-driven vehicles. At any time t, the state of the ith traffic agents can be denoted as p_t^i, where $p_t^i = (x_t^i, y_t^i)$ represents the 2D trajectory coordinate. Given the trajectory coordinate data of the target vehicle (superscript 0) and all its observed traffic agents (superscripts from 1 to n) in a traffic scene in the interval $[1, T_{obs}]$, denoted as $\boldsymbol{X} = \{p_t^{0:n}\}_{t=1}^{T_{obs}} \in \mathbb{R}^{(n+1)\times T_{obs}\times 2}$, the model aims to predict a probabilistic multi-modal distribution over the future trajectory of the target vehicle, expressed as $P(\boldsymbol{Y}|\boldsymbol{X})$. Here, $\boldsymbol{Y} = \{\boldsymbol{y}_t^0\}_{t=T_{obs}+1}^{T_f} \in \mathbb{R}^{T_f\times 2}$ is the predictive future trajectory coordinates of the target vehicle over a time horizon T_f, and $\boldsymbol{y}_t^0 = \{(\tilde{p}_t^{0,1}; \tilde{c}_t^{0,1}), (\tilde{p}_t^{0,2}; \tilde{c}_t^{0,2}), \cdots, (\tilde{p}_t^{0,C}; \tilde{c}_t^{0,C})\}$ encompasses both the potential trajectory and its associated maneuver likelihood ($\Sigma_{\boldsymbol{C}}^1 c_t^{0,i} = 1$), with $\boldsymbol{C}$ denoting the total number of potential trajectories predicted. In this study, we concentrate on optimizing the "student" model's output, where the future trajectory coordinates $\boldsymbol{Y}$ are defined as $\boldsymbol{Y}_t^{stu} = \boldsymbol{Y}$.

3.2 Scene Representation

HLTP++ describes scenarios by focusing on the relative positions of traffic agents, aligning with human spatial understanding. It preprocesses historical data into two key spatial forms: 1) visual vectors $\mathcal{S}$ that capture relative position, velocity, and acceleration $\mathcal{S} = \{\mathcal{S}_{\Delta p}, \mathcal{S}_{\Delta s}, \mathcal{S}_{\Delta a}\} \in \mathbb{R}^{(n+1)\times T_{obs}\times 4}$ of the target vehicle relative to its neighbors; 2) context matrices $\mathcal{M}$ describing speed and direction angle differences $\mathcal{M} = \{\mathcal{M}_{\Delta s}, \mathcal{M}_{\Delta\theta}\} \in \mathbb{R}^{(n+1)\times T_{obs}\times 2}$ among the surrounding agents.

3.3 Overall Architecture

Figure 2 illustrates the architecture of HLTP++. We use a novel pooling mechanism with an adaptive visual sector for data preprocessing. This sector dynamically adapts to capture important cues in different traffic situations, which is similar to attention allocation. Additionally, the model also employs a teacher-student heterogeneous network distillation approach for human-like trajectory prediction. i) **The "teacher" model**. The Temporal Encoder and the Spatial Encoder within the "teacher" model process visual vectors and context matrices to produce temporal and spatial feature vectors, respectively. These vectors are then fed into the Fusion Module to fuse two modalities. The output of the Fusion Module is then fed into the Teacher Multimodal Decoder, which enables the prediction of different potential maneuvers for the target vehicle, each with associated probabilities. ii) **The "student" model**. The Fourier Adaptive SNN first processes trajectory temporal information from the Visual Pooling by imitating the transmission of neurons. Then the output matrix is fed into the Student Multimodal Decoder which is similar to the teacher. Besides self-training, the "student" model acquires knowledge from the "teacher" model using a Knowledge Distillation Modulation (KDM) training strategy. This approach ensures accurate trajectory predictions while requiring fewer input observations.

3.4 Visual Pooling Mechanism

Research [33] shows that human drivers, constrained by brain's working memory, focus mainly on a few external agents in their central visual field, especially in high-risk situations. The driver's visual sector, influenced by speed, narrows at higher speeds for focused attention and widens at lower speeds for broader awareness.

HLTP++ introduces a visual pooling mechanism that emulates this adaptive visual attention. It features an adaptive visual sector that adjusts the field of view based on vehicle speed. Specifically, in contrast to models with uniform attention distribution, we propose a visual weight matrix H that adapts to changing focus in the central visual field at different speeds. Speed thresholds at $0km/h$, $30km/h$, $60km/h$, and $90km/h$ define distinct values of the visual sectors. This approach refines the understanding of attention during driving. The visual weight matrix H_{vision} is then integrated by the input visual vectors $\mathcal{S}$. Formally,

$$\tilde{\mathcal{S}} = H_{vision} \odot \mathcal{S}, \tag{1}$$

This equation produces visual vectors $\tilde{\mathcal{S}}$ that encapsulate human drivers' varying attention patterns.

3.5 Teacher Model

The "teacher" model integrates Temporal and Spacial encoders to closely emulate human visual perception, mirroring the retinal processing of the human drivers. As an enhancement, it further employs the iTransformer framework in the decoder to effectively extract spatio-temporal interactions.

Temporal Encoder. In real-world driving, the human brain, with its limited processing capacity, prioritizes information to facilitate efficient decision-making. Therefore, allocating distinct attention to different features is essential, as it reduces the cognitive load on the brain in processing non-essential information. We employ a LSTM layer to process temporal information, followed by a multi-head attention mechanism for attention allocation.

Spatial Encoder. Human drivers focus on their central vision while continuously monitoring their peripheral vision through side and rear-view mirrors to understand their surroundings, including nearby vehicles, pedestrians, and road conditions. To replicate this peripheral monitoring, especially during maneuvers, we introduce the Spatial Encoder. It processes a quarter of the time-segmented matrices $\mathcal{M} \in \mathbb{R}^{(n+1)\times \frac{T_{obs}}{4}\times 2}$ with a 1×1 convolutional layer for channel expansion, followed by a 3×3 layer for specific feature extraction, incorporating batch normalization and dropout for robustness. Enhanced with Graph Attention Networks and ELU activation, it produces spatial vectors $\mathbf{O}_s$.

Fusion Module. The combined outputs of the Spatial Encoder $\mathbf{O}_s$ and the Temporal Encoder $\mathbf{O}_t$ are fused and then fed into the iTransformer architecture [25] for advanced spatio-temporal interaction analysis, generating hidden states $\mathbf{I}$. Furthermore, we use the disparities between temporal and spatial features to generate a loss, denoted as L_{st}, which serves as one of the loss functions for training the "teacher" model.

Teacher Multimodal Decoder. The decoder of the "teacher" model, based on a Gaussian Mixture Model (GMM), accounts for uncertainty in trajectory prediction by evaluating multiple possible maneuvers and their probabilities. Specifically, built on the base layer of estimated maneuvers $\boldsymbol{C}$, the model assumes that the probability distribution for trajectory predictions follows a Gaussian framework:

$$P_{\boldsymbol{\Omega}}(\boldsymbol{Y}|\boldsymbol{C}, \boldsymbol{X}) = N(\boldsymbol{Y}|\mu(\boldsymbol{X}), \Sigma(\boldsymbol{X})) \tag{2}$$

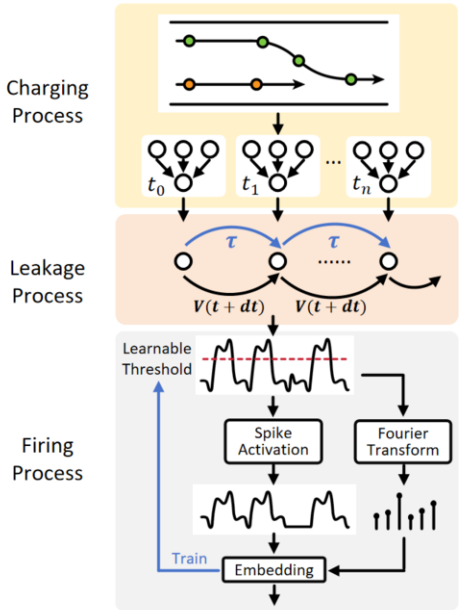

Figure 3. Overall architecture of the FA-SNN.

where $\boldsymbol{X}$ represents the input to our model, and $\boldsymbol{\Omega} = \left[\Omega^{t+1}, \ldots, \Omega^{t+t_f}\right]$ symbolizes the estimable parameters of the distribution. Each $\Omega^t = [\mu^t, \Sigma^t]$ represents the mean and variance for the predicted trajectory at time t. Then, in the next layer, the multimodal predictions are conceptualized as a GMM:

$$P(\boldsymbol{Y}|\boldsymbol{X}) = \sum_{\forall i} P\left(c_i|\boldsymbol{X}\right) P_{\boldsymbol{\Omega}}\left(\boldsymbol{Y}|c_i, \boldsymbol{X}\right) \tag{3}$$

where c_i denotes the i-th maneuver in $\boldsymbol{C}$. Then, the hidden states are processed through softmax activation and a MLP layer, forming a probability distribution $\mathbf{O}_{tea}$ over potential future trajectories. This approach balances the precision and validity, which is critical for dynamic and uncertain driving environments.

3.6 Student Model

The FA-SNN is introduced in the "student" model, which emphasizes the short-term and fewer observations, with visual vectors $\tilde{\boldsymbol{S}}$ ($T_{obs} = 8$), utilizing visual vectors and a lightweight architectural design for efficient learning. In a departure from the complex framework of the "teacher" model, the "student" model employs a more lightweight architecture to produce a multimodal prediction distribution $\mathbf{O}_{stu}$. By learning the behavioral paradigm from the "teacher" model, the "student" model is able to make human-like predictions even when constrained by limited observations.

Specifically, The FA-SNN proposed in this study is an enhanced version of the traditional SNN model. It addresses the challenge of fitting difficulties during training by introducing adaptive adjustments and optimizing temporal feature extraction. The approach is based on the idea that neurons in an SNN should adapt to different scenarios, which is mainly reflected in the adjustment of the threshold magnitude. Inspired by the Leaky Integrate-and-Fire (LIF) mechanism, the Fourier Transform (FT) is used to extract specific features. More specifically, the FA-SNN's forward propagation involves three essential processes: Charging, Leakage, and Firing Processes.

Charging Process. Similar to traditional perceptron neurons, the current neuron is charged by aggregating input spike sequences from previous neurons through varying weights at discrete time steps $\{t_0, t_1, ..., t_n, ..., t_N\}$.

Leakage Process. In the LIF mechanism, neurons experience leakage due to voltage differences in their surroundings. The internal voltage V of the spiking neuron tends towards an equilibrium voltage U over time t, adhering to the differential equation $U - V = -\eta \frac{dV}{dt}$, where $\eta = 1$ denote the leakage decay rate, indicating the magnitude of voltage decay. This dynamic is pivotal for the neuron's voltage stabilization, allowing the calculation of future voltage states. After solving the above equation, we can obtain:

$$V(t_n) = U - Ce^{-\frac{t_n}{\eta}} \tag{4}$$

where C is a constant value. This allows us to calculate the voltage at the next moment t_{n+1}:

$$V(t_{n+1}) = V(t_n + dt) = e^{-\frac{dt}{\eta}}(V(t) - U) + U \tag{5}$$

Firing Process. The firing process is activated based on the spike magnitude through an activation function. Given a spike threshold U_0, the voltage $V'(t_{n+1})$ can be defined as follows:

$$V'(t_{n+1}) = \begin{cases} V(t_{n+1}) - U_0, & V(t_{n+1}) > U_0 \\ V(t_{n+1}), & V(t_{n+1}) \leq U_0 \end{cases} \tag{6}$$

Unlike traditional models with a fixed spike threshold, the proposed FA-SNN employs a learnable threshold, adjusting dynamically to preserve temporal features. Importantly, a Fast Fourier Transform (FFT) is applied to the pre-firing spike value $V(t_{n+1})$ to incorporate frequency information, enhancing the representation capability:

$$F[V(t_{n+1})] = \sum_{\tau=0}^{N} V(t_{n+1}) \cdot e^{-\frac{2\pi i}{N+1}(n+1)\tau} \tag{7}$$

where i is the imaginary unit. According to Euler's formula:

$$e^{-\frac{2\pi i}{N+1}(n+1)\tau} = cos[-\frac{2\pi(n+1)}{N+1}\tau] + isin[-\frac{2\pi(n+1)}{N+1}\tau], \tag{8}$$

where $A = cos[-\frac{2\pi(n+1)}{N+1}\tau]$ and $B = sin[-\frac{2\pi(n+1)}{N+1}\tau]$ respectively represent the real and imaginary parts of $e^{-\frac{2\pi i}{N+1}(n+1)\tau}$. We then compute the power spectrum $W = (|A| + |B|)^2$ to serve as the output feature, where "||" represent the absolute value symbol.

Backpropagation. Due to the discontinuity of the activation function used during SNN firing, conventional chain-rule differentiation is infeasible. To circumvent this, the gradient G is redefined, factoring in the spike threshold and introducing parameters like the absolute width w_a, gradient width w_g and gradient scale s:

$$G(V'(t_{n+1})) = \frac{s}{w_a} \times \exp\left(-\frac{|V'(t_{n+1}) - U_0|}{w_a}\right) \tag{9}$$

where $w_a = U_0 \cdot w_g$, and $w_g = 0.5,\ s = 1.0$.

4 Training

4.1 Teacher Training

For the "teacher" model, we follow a standard protocol, using 3 seconds of observed trajectory for input ($T_{obs} = 16$) and predicting a 5-second future trajectory ($T_f = 25$). To extract complex knowledge from the dataset, we allow slight overfitting during training. The

training loss function of the teacher model consists of three parts: trajectory loss $\mathcal{L}_{traj}^{tea}$, maneuver loss $\mathcal{L}_{man}^{tea}$ and temporal-spacial loss L_{st} from iTransformer.

Trajectory Loss. In our trajectory prediction process, we treat the output 2D trajectory coordinates $P = (x, y)$ as a bivariate Gaussian distribution. Therefore, we could use Negative Log-Likelihood (NLL) loss $\mathcal{L}_{traj}^{tea}$ to measure the disparity between the prediction and the ground truth. Considering the total number C of potential trajectories predicted, with P_{pred}^{tea} and P_{gt} representing the teacher model's predicted trajectory coordinates and the ground truth coordinates respectively, the trajectory loss $\mathcal{L}_{traj}^{tea}$ for the teacher model can be formulated as: $\mathcal{L}_{traj}^{tea} = \sum_{t}^{T_f} \sum_{c}^{C} \mathcal{L}^{N}(P_{pred}^{tea}, P_{gt})$. Mathematically, for a specific data point, given the model's predicted probability distribution $P(\boldsymbol{Y}|\boldsymbol{X})$, where $\boldsymbol{Y}$ represents the output future trajectory and $\boldsymbol{X}$ denotes the input features, the NLL Loss $\mathcal{L}^{N}$ is defined as follows:

$$\mathcal{L}^{N} = -log(P(\boldsymbol{Y}|\boldsymbol{X})), \tag{10}$$

Maneuver Loss. To address the potentially detrimental effects of mis-classifying maneuver types on trajectory prediction accuracy and robustness, we adopt the Mean Squared Error (MSE) loss $\mathcal{L}_{man}^{tea}$ to measure the disparity between the predicted maneuver types M_{pred}^{tea} and the ground truth maneuver types M_{gt}^{tea}, which is formulated as: $\mathcal{L}_{man}^{tea} = \sum_{t}^{T_f} \sum_{c}^{C} \mathcal{L}^{M}(M_{pred}^{tea}, M_{gt})$, so the total loss function of teacher model is formulated as follows:

$$\mathcal{L}^{tea} = \mathcal{L}_{traj}^{tea} + \mathcal{L}_{man}^{tea} + \mathcal{L}_{st}, \tag{11}$$

4.2 Student Training

The "student" model is trained to predict 5-second future trajectories with fewer input observations. To improve its predictive performance with limited observations, we decouple the total loss function of the student model $\mathcal{L}$ as the student loss (especially refers to the loss function exclusively associated with the "student" model) $\mathcal{L}^{stu}$ and distillation loss $\mathcal{L}^{dis}$. The student loss, similar to that of the "teacher" model, quantifies the discrepancy between the model's predicted trajectories, maneuvers and their ground truths. Formally,

$$\begin{aligned}\mathcal{L}^{stu} &= \mathcal{L}_{traj}^{stu} + \mathcal{L}_{man}^{stu} \\ &= \sum_{t}^{T_f} \sum_{c}^{C} \left(\mathcal{L}^{N}(P_{pred}^{stu}, P_{gt}) + \mathcal{L}^{M}(M_{pred}^{stu}, M_{gt}) \right),\end{aligned} \tag{12}$$

where P_{pred}^{stu} and M_{pred}^{stu} represent the predicted 2D coordinates and maneuvers of the "student" model. Moreover, we apply the MSE loss to measure the disparity between the outputs of the teacher and the "student" model:

$$\begin{aligned}\mathcal{L}^{dis} &= \mathcal{L}_{traj}^{dis} + \mathcal{L}_{man}^{dis} \\ &= \sum_{t}^{T_f} \sum_{c}^{C} \left(\mathcal{L}^{M}(P_{pred}^{stu}, P_{pred}^{tea}) + \mathcal{L}^{M}(M_{pred}^{stu}, M_{pred}^{tea}) \right),\end{aligned} \tag{13}$$

Hence, the total loss function of the "student" model is formulated as $\mathcal{L} = \mathcal{L}^{stu} + \mathcal{L}^{dis}$. Then, we propose a method for tuning multiple tasks that evaluates the importance of different loss functions and automatically adjusts the weights between them for efficient training.

4.3 Knowledge Distillation Modulation

Given that $\mathcal{L}$ is composed of sub-loss functions from multiple tasks, determining the proportionality relationships between them poses a challenging problem. Both $\mathcal{L}^{stu}$ and $\mathcal{L}^{dis}$ are further decomposed into maneuver loss function $\mathcal{L}_{traj}$ and trajectory coordinate loss function $\mathcal{L}_{man}$:

$$\mathcal{L}^{stu} = \mathcal{L}_{traj}^{stu} + \mathcal{L}_{man}^{stu}, \qquad \mathcal{L}^{dis} = \mathcal{L}_{traj}^{dis} + \mathcal{L}_{man}^{dis}, \tag{14}$$

Drawing the inspiration from the notable work [14], we incorporate the KDM to weight the *trajectory* loss and the *distillation* loss, with homoscedastic uncertainty. To the best of our knowledge, we are the first to propose a multi-level, multi-task hyperparameter tuning approach to autonomously adjust knowledge distillation hyperparameters during training in this field. Our approach defines a multi-level task where the overarching training loss function is composed of an ensemble of sub-loss functions. Each of these sub-loss functions is further composed of additional sub-loss functions that share some degree of similarity.

Following the approach outlined in Kendall et al. [14] for the first level of the loss function, we obtain the following equation:

$$\begin{aligned}\mathcal{L}^{stu} &= \frac{1}{2\sigma_t^2}\mathcal{L}_{traj}^{stu}(\mathbf{W}) + \frac{1}{2\sigma_m^2}\mathcal{L}_{man}^{stu}(\mathbf{W}) + log\sigma_t\sigma_m, \\ \mathcal{L}^{dis} &= \frac{1}{2\sigma_t^2}\mathcal{L}_{traj}^{dis}(\mathbf{W}) + \frac{1}{2\sigma_m^2}\mathcal{L}_{man}^{dis}(\mathbf{W}) + log\sigma_t\sigma_m,\end{aligned} \tag{15}$$

where σ_t, σ_m are the learnable uncertainty variances, $\mathbf{W}$ represents trainable parameters of the model. Since $\mathcal{L}$ is composed of $\mathcal{L}^{stu}$ and $\mathcal{L}^{dis}$, we use the multi-task tuning approach for the second level:

$$\mathcal{L} = \frac{1}{2\sigma_s^2}\mathcal{L}^{stu}(\mathbf{W}) + \frac{1}{2\sigma_d^2}\mathcal{L}^{dis}(\mathbf{W}) + log\sigma_s\sigma_d, \tag{16}$$

Combining Eq. 15 and Eq. 16, we obtain the following formulation:

$$\begin{aligned}\mathcal{L}(W, \sigma_t, \sigma_m, \sigma_s, \sigma_d) &= \frac{1}{2\sigma_s^2}\left(\frac{1}{2\sigma_t^2}\mathcal{L}_{traj}^{stu} + \frac{1}{2\sigma_m^2}\mathcal{L}_{man}^{stu}\right) \\ &+ \frac{1}{2\sigma_d^2}\left(\frac{1}{2\sigma_t^2}\mathcal{L}_{traj}^{dis} + \frac{1}{2\sigma_m^2}\mathcal{L}_{man}^{dis}\right) + F(\sigma_t, \sigma_m, \sigma_s, \sigma_d),\end{aligned} \tag{17}$$

where F equals to $log\sigma_t\sigma_m(\frac{1}{2\sigma_s^2} + \frac{1}{2\sigma_d^2}) + log\sigma_s\sigma_d$. To ensure uniformity in the derived equations from different level-segmenting approach, we modify F as $F = log(\sigma_t\sigma_m\sigma_s\sigma_d)$.

5 Experiment

5.1 Experimental Setup

Datasets. We conduct experiments using three esteemed datasets: NGSIM [7], HighD [15], and MoCAD [22]. These three datasets cover various traffic conditions, including on highways and in dense urban environments.

Metric. Root Mean Square Error (RMSE) and average RMSE is applied as our primary evaluation metric, which is commonly used as a measure to calculate the square root of the average squared prediction error in autonomous driving.

Implementation Details. HLTP++ is developed using PyTorch and trained on an A40 48G GPU. We use the Adam optimizer along with CosineAnnealingWarmRestarts for scheduling, with a training batch size of 256 and learning rates ranging from 10^{-3} to 10^{-5}. Unless specified, all evaluation results are based on the "student" model.

5.2 Experiment Results

Comparison with the State-of-the-art Baselines. Our comprehensive evaluation demonstrates HLTP++'s superior performance compared to SOTA baselines, as detailed in Table 1. It notably achieves

Table 1. Evaluation results for our proposed model and the other SOTA baselines in the NGSIM, HighD, and MoCAD datasets. **Bold** and underlined values represent the best and second-best performance in each category. "AVG" is the average value of the RMSE.

Dataset	Model	Prediction Horizon (s)					
		1	2	3	4	5	AVG
NGSIM	S-LSTM [1]	0.65	1.31	2.16	3.25	4.55	2.38
	S-GAN [10]	0.57	1.32	2.22	3.26	4.40	2.35
	CS-LSTM [7]	0.61	1.27	2.09	3.10	4.37	2.29
	MATF-GAN [36]	0.66	1.34	2.08	2.97	4.13	2.22
	NLS-LSTM [27]	0.56	1.22	2.02	3.03	4.30	2.23
	IMM-KF [17]	0.58	1.36	2.28	3.37	4.55	2.43
	MFP [31]	0.54	1.16	1.89	2.75	3.78	2.02
	DRBP [9]	1.18	2.83	4.22	5.82	-	3.51
	WSiP [34]	0.56	1.23	2.05	3.08	4.34	2.25
	CF-LSTM [35]	0.55	1.10	1.78	2.73	3.82	1.99
	MHA-LSTM [28]	0.41	1.01	1.74	2.67	3.83	1.91
	STDAN [5]	**0.39**	**0.96**	1.61	2.56	3.67	1.84
	iNATran [4]	0.39	0.96	1.61	2.42	3.43	1.76
	DACR-AMTP [6]	0.57	1.07	1.68	2.53	3.40	1.85
	FHIF [39]	0.40	0.98	1.66	2.52	3.63	1.84
	BAT [22]	**0.23**	**0.81**	1.54	2.52	3.62	1.74
	HLTP++	0.46	0.98	**1.52**	**2.17**	**3.02**	**1.63**
	HLTP++ (h)	0.49	1.05	1.64	2.34	3.27	1.76
HighD	S-LSTM [1]	0.22	0.62	1.27	2.15	3.41	1.53
	S-GAN [10]	0.30	0.78	1.46	2.34	3.41	1.69
	WSiP [34]	0.20	0.60	1.21	2.07	3.14	1.44
	CS-LSTM [7]	0.22	0.61	1.24	2.10	3.27	1.48
	MHA-LSTM [28]	0.19	0.55	1.10	1.84	2.78	1.29
	NLS-LSTM [27]	0.20	0.57	1.14	1.90	2.91	1.34
	DRBP[9]	0.41	0.79	1.11	1.40	-	0.92
	EA-Net [3]	0.15	0.26	0.43	0.78	1.32	0.59
	CF-LSTM [35]	0.18	0.42	1.07	1.72	2.44	1.17
	STDAN [5]	0.19	0.27	0.48	0.91	1.66	0.70
	DACR-AMTP [6]	**0.10**	**0.17**	0.31	0.54	1.01	0.42
	GaVa [24]	0.17	0.24	0.42	0.86	1.31	0.60
	HLTP++	0.11	**0.17**	**0.30**	**0.47**	**0.75**	**0.36**
	HLTP++ (h)	0.12	0.18	0.32	0.52	0.89	0.41
MoCAD	S-LSTM [1]	1.73	2.46	3.39	4.01	4.93	3.30
	S-GAN [10]	1.69	2.25	3.30	3.89	4.69	3.16
	CS-LSTM [7]	1.45	1.98	2.94	3.56	4.49	2.88
	MHA-LSTM [28]	1.25	1.48	2.57	3.22	4.20	2.54
	NLS-LSTM [27]	0.96	1.27	2.08	2.86	3.93	2.22
	WSiP [34]	0.70	0.87	1.70	2.56	3.47	1.86
	CF-LSTM [35]	0.72	0.91	1.73	2.59	3.44	1.87
	STDAN [5]	0.62	0.85	1.62	2.51	3.32	1.78
	HLTP++	**0.60**	**0.81**	**1.56**	**2.40**	**3.19**	**1.71**
	HLTP++ (h)	0.64	0.86	1.62	2.52	3.35	1.80

gains of 11.2% for long-term (5s) and 11.4% for average predictions on the NGSIM dataset. The corresponding outstanding performance is also evident on the HighD dataset and the MACAD dataset. It is noteworthy that our model HLTP++(h), despite utilizing only 1.5 seconds of input data (half of the input of other baselines), achieves comparable prediction accuracy. This highlights the adaptability and robustness of HLTP++.

Comparing Model Performance and Complexity. As detailed in Table 2, our benchmarking against SOTA baselines reveals that HLTP++ models outperform in all metrics while maintaining a minimal parameter count. Specifically, HLTP++ reduce parameters by 56.91% and 33.51% compared to WSiP and CS-LSTM, respectively. Compared to HLTP++(SM), the "teacher" model of HLTP++, HLTP++(TM), achieve the second best score in three datasets, while maintaining a larger number of parameters and slower inference speed. However, HLTP++ maintain the lowest inference time while achieve the best accuracy in trajectory prediction. Utilizing the Knowledge Distillation Module (KDM), HLTP++ retains the lightweight advantages of the HLTP++(SM), while concurrently enhancing its predictive capabilities by assimilating knowledge gleaned

Table 2. Comparative evaluation of our model with SOTA baselines. Emphasizing model complexity via parameter count (Param.). "Avg. IT" represents the average inference time of the model. HLTP++(TM) is the teacher model of the HLTP++. HLTP++(SM) is the student model of the HLTP++ without KDM.

Model	Param. (K)	Average RMSE (m)			Avg. IT (s)
		NGSIM	HighD	MoCAD	
CS-LSTM	194.92	2.29	1.49	2.88	0.0259
CF-LSTM	387.10	1.99	1.17	1.88	0.4565
WSiP	300.76	2.25	1.44	1.86	0.3292
STDAN	486.82	1.87	0.70	1.78	0.0670
HLTP++	**129.60**	**1.63**	**0.36**	**1.62**	**0.0214**
HLTP++(SM)	**129.60**	1.76	0.52	1.75	**0.0214**
HLTP++(TM)	453.45	1.74	0.47	1.72	0.0726

Table 3. Different methods and components of ablation study.

Components	Ablation Methods						
	A	B	C	D	E	F	G
Visual Pooling Mechanism	✘	✔	✔	✔	✔	✔	✔
Spatial Encoder	✔	✘	✔	✔	✔	✔	✔
Fusion Module	✔	✔	✘	✔	✔	✔	✔
FA-SNN	✔	✔	✔	✘	✔	✔	✔
Multimodal Decoder	✔	✔	✔	✔	✘	✔	✔
KDM	✔	✔	✔	✔	✔	✘	✔

from the teacher model, thereby surpassing the performance of the teacher model itself. This highlights the efficiency and adaptability of our lightweight "teacher-student" knowledge distillation framework, offering a balance between accuracy and computational resources.

Qualitative Results. Figure 5 showcase the multimodal probabilistic prediction performance of HLTP++ on the NGSIM dataset. The heat maps shown represent the Gaussian Mixture Model of predictions in challenging scenes. These visualizations show that the highest probability predictions of our model are very close to the ground truth, indicating its impressive performance. Figure 6 visually demonstrates our model's ability to accurately predict complex scenarios such as merging and lane changing, confirming its effectiveness and safety in various traffic situations. Interestingly, in certain complex scenarios, the trajectory predictions of the "student" model exceed the accuracy of the "teacher" model. This result illustrates the ability of the "student" model to selectively assimilate and refine the knowledge acquired from the "teacher" model, effectively "extracting the essence and discarding the dross". Moreover, we observed that vehicles in closer proximity to the target vehicle received higher attention values. This observation aligns with the driving behavior of human operators who primarily focus on the vehicle ahead, as it has the most significant influence on the driving trajectory. This also substantiates the utility of vision pooling in reducing the perturbations caused by neighboring vehicles, thereby prioritizing the significance of the leading vehicle.

5.3 *Ablation Studies*

Ablation Study for Core Components. Table 3 shows that our ablation study evaluates the performance of HLTP++ using six model variations, each omitting different components. The data in Table 4 clearly indicate that the performance of all models degrades when components are removed, as compared to the baseline model. Notably, integrating the iTransformer and a multimodal probabilistic maneuvering module significantly improves the accuracy. This underscores their vital function in encapsulating the spatio-temporal dynamics among vehicles. Furthermore, ablation studies A and B

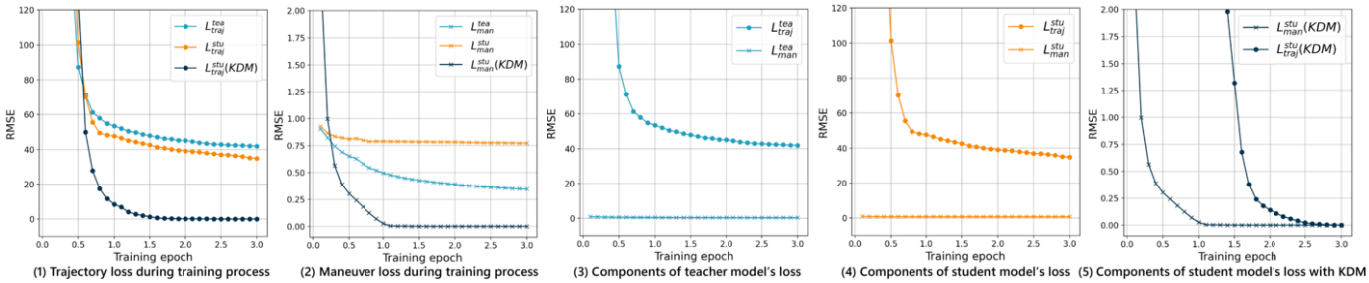

Figure 4. Visualizations of the (1) trajectory loss, (2) maneuver loss, (3) teacher model's loss, (4) student model's loss without KDM, (5) student model's loss with KDM during training.

Table 4. Ablation results for different models. (NGSIM/HighD)

Time (s)	Ablation Methods					
	A	B	C	D	E	F
1	0.52/0.11	0.47/0.11	0.53/0.12	0.47/0.11	0.49/0.20	0.50/0.16
2	1.07/0.18	1.03/0.20	1.09/0.23	1.10/0.17	1.10/0.26	1.06/0.32
3	1.68/0.33	1.71/0.34	1.74/0.34	1.67/0.32	1.81/0.44	1.64/0.45
4	2.35/0.49	2.40/0.51	2.41/0.49	2.34/0.50	2.63/0.72	2.33/0.69
5	3.24/0.78	3.39/0.80	3.32/0.79	3.27/0.79	3.67/0.97	3.26/0.98
AVG	1.77/0.38	1.80/0.39	1.82/0.39	1.77/0.38	1.94/0.52	1.76/0.52

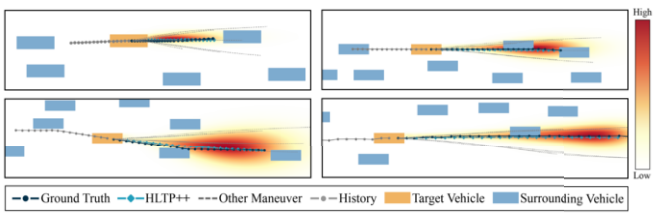

Figure 5. Visualizations of the multi-modal probabilistic prediction of HLTP++ on NGSIM. Heat maps illustrate the GMM of predictions: brighter areas denote higher probabilities.

provide empirical evidence supporting the utility of incorporating cognitive mechanisms similar to human brain processes.

Ablation Study for FA-SNN. To further showcase the effectiveness of our proposed FA-SNN, we conducted an ablation study in HLTP++ by replacing the FA-SNN with the standard SNN, SNN only with the Fourier Transform (FT), and the SNN only with the adaptive spike threshold (AST). Table 5 demonstrates that incorporating the AST and FT in SNN can significantly improve the predictive performance of the model. This underscores the FA-SNN's ability to capture and extract spatio-temporal interactions in complex scenes.

Ablation Study for KDM. To illustrate the impact of KDM on HLTP++, we present the loss function curves during training in Figures 4 (1) and 4 (2). Specifically, L^{tea}_{traj}, L^{stu}_{traj}, L^{stu}_{traj}(KDM), and L^{stu}_{traj}(KDM) represent the trajectory losses for the teacher model, the student model without KDM, and the student model with KDM, respectively. Similarly, L^{tea}_{man}, L^{stu}_{man}, and L^{stu}_{man}(KDM) denote the maneuver losses. As shown in Figure 4, L^{stu}_{traj}(KDM) and L^{stu}_{man}(KDM) initially start higher but rapidly decrease as training progresses, indicating efficient early training and finer adjustments in later stages. Figures 4 (3)-(5) compare the trajectory and maneuver losses between models. Figures 4 (3) and 4 (4) show a significant difference in loss values, with a stabilization around a factor of 40 as training progresses. The lack of maneuver loss reduction suggests that, without KDM, there is an imbalance favoring trajectory fitting over maneuver prediction. In contrast, Figure 4 (5) demonstrates that KDM facilitates convergence between L^{stu}_{traj}(KDM) and L^{stu}_{man}(KDM), enabling balanced optimization across tasks.

Ablation Study for Missing Data. We introduce a missing test set on the NGSIM dataset, focusing on scenarios where part of the historical data is missing. The set is divided into five subsets based on varying durations of data absence. For example, subset t_m=0.4 indicates a missing trajectory data duration of 0.4 seconds, which was imputed using linear interpolation. The results in Table 6 show that HLTP++ outperforms all baselines even with 1.6s missing data, highlighting its adaptability and deep understanding of traffic dynamics.

Table 5. Ablation study on FA-SNN.

Models	Components		Metrics	
	AST	FT	RMSE (5s)	AVG
FA-SNN	✔	✔	3.02	1.63
F-SNN	✘	✔	3.13	1.70
A-SNN	✔	✘	3.11	1.69
SNN	✘	✘	3.21	1.73

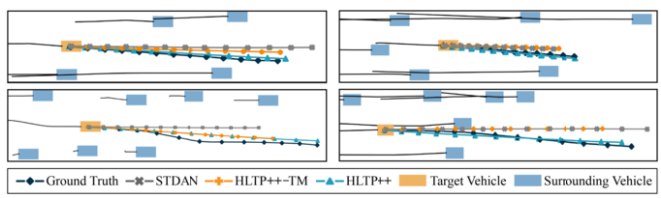

Figure 6. Visualizations of HLTP++ and SOTA baseline on NGSIM. HLTP++(TM) denotes the "teacher" model of HLTP++.

Table 6. Ablation results for missing data.

Time (s)	t_m=0.4	t_m=0.8	t_m=1.2	t_m=1.6	t_m=2.0	t_m=2.4
1	0.46	0.46	0.47	0.48	0.50	0.56
2	0.98	0.99	1.00	1.03	1.07	1.17
3	1.53	1.54	1.57	1.61	1.67	1.79
4	2.20	2.21	2.25	2.31	2.39	2.54
5	3.07	3.10	3.15	3.23	3.32	3.51

6 Conclusion

This study presents a novel trajectory prediction model (HLTP++) for AVs. It addresses the limitations of previous models in terms of parameter heaviness and applicability. HLTP++ is based on a multi-level task knowledge distillation network, providing a lightweight yet efficient framework that maintains prediction accuracy. Importantly, HLTP++ adapts to scenarios with missing data and reduced inputs by simulating human observation and making human-like predictions. The empirical results indicate that HLTP++ excels in complex traffic scenarios and achieves SOTA performance. In future work, we plan to feed multimodal data, such as Bird's Eye View (BEV), multi-view camera images and Lidar, into the HLTP++ model to further enhance the scene understanding of the model.

Acknowledgements

This research is supported by Science and Technology Development Fund of Macau SAR (File no. 0021/2022/ITP, 0081/2022/A2, 001/2024/SKL), Shenzhen-Hong Kong-Macau Science and Technology Program Category C (SGDX20230821095159012), State Key Lab of Intelligent Transportation System (2024-B001), Jiangsu Provincial Science and Technology Program (BZ2024055), and University of Macau (SRG2023-00037-IOTSC).

References

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016.

[2] S. Bhardwaj, M. Srinivasan, and M. M. Khapra. Efficient video classification using fewer frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 354–363, 2019.

[3] Y. Cai, Z. Wang, H. Wang, L. Chen, Y. Li, M. A. Sotelo, and Z. Li. Environment-attention network for vehicle trajectory prediction. *IEEE Transactions on Vehicular Technology*, 70(11):11216–11227, 2021.

[4] X. Chen, H. Zhang, F. Zhao, Y. Cai, H. Wang, and Q. Ye. Vehicle trajectory prediction based on intention-aware non-autoregressive transformer with multi-attention learning for internet of vehicles. *IEEE Transactions on Instrumentation and Measurement*, 71:1–12, 2022.

[5] X. Chen, H. Zhang, F. Zhao, Y. Hu, C. Tan, and J. Yang. Intention-aware vehicle trajectory prediction based on spatial-temporal dynamic attention network for internet of vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):19471–19483, 2022.

[6] P. Cong, Y. Xiao, X. Wan, M. Deng, J. Li, and X. Zhang. Dacr-amtp: Adaptive multi-modal vehicle trajectory prediction for dynamic drivable areas based on collision risk. *IEEE Transactions on Intelligent Vehicles*, 2023.

[7] N. Deo and M. M. Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1468–1476, 2018.

[8] P. U. Diehl and M. Cook. Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Frontiers in Computational Neuroscience*, 9, 2015. URL https://api.semanticscholar.org/CorpusID:3637557.

[9] K. Gao, X. Li, B. Chen, L. Hu, J. Liu, R. Du, and Y. Li. Dual transformer based prediction for lane change intentions and trajectories in mixed traffic environment. *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[10] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2255–2264, 2018.

[11] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[12] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, L. Lu, X. Jia, Q. Liu, J. Dai, Y. Qiao, and H. Li. Planning-oriented autonomous driving, 2023.

[13] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang, and H. Chen. A survey on trajectory-prediction methods for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 7(3):652–674, 2022.

[14] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, 2018.

[15] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125, 2018. doi: 10.1109/ITSC.2018.8569552.

[16] S. Lai, L. Hu, J. Wang, L. Berti-Equille, and D. Wang. Faithful vision-language interpretation via concept bottleneck models. In *The Twelfth International Conference on Learning Representations*, 2023.

[17] V. Lefkopoulos, M. Menner, A. Domahidi, and M. N. Zeilinger. Interaction-aware motion prediction for autonomous driving: A multiple model kalman filtering scheme. *IEEE Robotics and Automation Letters*, 6(1):80–87, 2020.

[18] Z. Li, Z. Chen, Y. Li, and C. Xu. Context-aware trajectory prediction for autonomous driving in heterogeneous environments. *Computer-Aided Civil and Infrastructure Engineering*, 2023.

[19] Z. Li, Z. Cui, H. Liao, J. Ash, G. Zhang, C. Xu, and Y. Wang. Steering the future: Redefining intelligent transportation systems with foundation models. *CHAIN*, 1(1):46–53, 2024.

[20] H. Liao, X. Li, Y. Li, H. Kong, C. Wang, B. Wang, Y. Guan, K. Tam, Z. Li, and C. Xu. Characterized diffusion and spatial-temporal interaction network for trajectory prediction in autonomous driving. *arXiv preprint arXiv:2405.02145*, 2024.

[21] H. Liao, Y. Li, Z. Li, C. Wang, Z. Cui, S. E. Li, and C. Xu. A cognitive-based trajectory prediction approach for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 9(4):4632–4643, 2024. doi: 10.1109/TIV.2024.3376074.

[22] H. Liao, Z. Li, H. Shen, W. Zeng, D. Liao, G. Li, and C. 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.

[23] H. Liao, Z. Li, C. Wang, B. Wang, H. Kong, Y. Guan, G. Li, Z. Cui, and C. Xu. A cognitive-driven trajectory prediction model for autonomous driving in mixed autonomy environment. *arXiv preprint arXiv:2404.17520*, 2024.

[24] H. Liao, S. Liu, Y. Li, Z. Li, C. Wang, B. Wang, Y. Guan, and C. Xu. Human observation-inspired trajectory prediction for autonomous driving in mixed-autonomy traffic environments. *arXiv preprint arXiv:2402.04318*, 2024.

[25] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long. itransformer: Inverted transformers are effective for time series forecasting, 2023.

[26] J. F. Louie and M. Mouloua. Executive attention as a predictor of distracted driving performance. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 61(1):1436–1440, 2017. doi: 10.1177/1541931213601844. URL https://doi.org/10.1177/1541931213601844.

[27] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi. Non-local social pooling for vehicle trajectory prediction. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 975–980, 2019. doi: 10.1109/IVS.2019.8813829.

[28] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi. Attention based vehicle trajectory prediction. *IEEE Transactions on Intelligent Vehicles*, 6(1):175–185, 2021.

[29] A. Porrello, L. Bergamini, and S. Calderara. Robust re-identification by multiple views knowledge distillation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 93–110. Springer, 2020.

[30] R. Quan, L. Zhu, Y. Wu, and Y. Yang. Holistic lstm for pedestrian trajectory prediction. *IEEE transactions on image processing*, 30:3229–3239, 2021.

[31] C. Tang and R. R. Salakhutdinov. Multiple futures prediction. *Advances in neural information processing systems*, 32, 2019.

[32] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida. Deep learning in spiking neural networks. *Neural networks*, 111:47–63, 2019.

[33] A. Tucker and K. Marsh. Speeding through the pandemic: Perceptual and psychological factors associated with speeding during the covid-19 stay-at-home period. *Accident Analysis & Prevention*, 159:106225, 2021.

[34] R. Wang, S. Wang, H. Yan, and X. Wang. Wsip: wave superposition inspired pooling for dynamic interactions-aware trajectory prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4685–4692, 2023.

[35] Y. Xu, J. Yang, and S. Du. Cf-lstm: Cascaded feature-based long short-term networks for predicting pedestrian trajectory. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):12541–12548, Apr. 2020. doi: 10.1609/aaai.v34i07.6943. URL https://ojs.aaai.org/index.php/AAAI/article/view/6943.

[36] T. Zhao, Y. Xu, M. Monfort, W. Choi, C. Baker, Y. Zhao, Y. Wang, and Y. N. Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12126–12134, 2019.

[37] Z. Zhou, L. Ye, J. Wang, K. Wu, and K. 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.

[38] Z. Zhou, J. Wang, Y.-H. Li, and Y.-K. Huang. Query-centric trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17863–17873, 2023.

[39] Z. Zuo, X. Wang, S. Guo, Z. Liu, Z. Li, and Y. Wang. Trajectory prediction network of autonomous vehicles with fusion of historical interactive features. *IEEE Transactions on Intelligent Vehicles*, 2023.