基于时空维度分解与异构解码的轻量化轨迹预测模型

林清华[1],王美芳[1],李 辉[1*],葛同澳[2],颜 军[1]
(1. 青岛科技大学数据科学学院,山东青岛 266100;2. 武汉理工大学计算机与人工智能学院,湖北武汉 430070)

摘 要: 高精度与高效率的轨迹预测模型是自动驾驶系统实现安全导航的关键。现有方法多采用图神经网络(Graph Neural Networks,GNNs)或注意力机制联合建模智能体与高精地图之间的时空依赖关系,虽然在预测精度上取得了显著进展,但其耦合式的时空建模方式导致计算复杂度过高,严重限制了模型在车载嵌入式系统等实际平台上的部署能力。此外,目前主流方法在解码阶段普遍采用单一的参数化解码范式处理场景中所有智能体,忽视了不同智能体在预测精度与多模态性方面的差异化需求,造成计算资源的低效分配。为此,本文提出一种基于时空维度分解与异构解码的轻量化轨迹预测模型(lightweight trajectory prediction Model based on spatio-temporal dimension decomposition and heterogeneous decoding,MAM-M),该模型由时空维度分解编码器与异构解码器两大核心模块构成。在编码阶段,针对传统Mamba模型在长序列建模中存在的单向信息衰减问题,设计了时序双流Mamba网络,通过正向与反向双流并行处理历史轨迹,使每个时间步同时包含完整的过去与未来信息,有效缓解了长程遗忘问题,并利用线性复杂度技术实现了时间维度上的高效编码。随后,采用自注意力机制在空间维度上捕捉智能体与高精地图之间的全局交互关系,从而实现"先时间后空间"的解耦编码策略,显著减少了时空维度上的冗余计算。在解码阶段,提出一种异构解码器以适配不同智能体的预测需求:对运动复杂、行为不确定性高的目标智能体,采用基于可学习查询的模式查询解码器,通过交叉注意力机制主动检索场景上下文信息,精确捕捉其未来轨迹的多模态分布;对数量较多且运动相对规则的非目标智能体,引入神经-贝塞尔解码器,通过神经网络预测贝塞尔曲线的控制点并结合数学表达式生成轨迹,在保证精度的同时显著降低推理延迟(latency),实现了高效的端到端训练与推理。在Argoverse与Argoverse 2数据集上的实验结果表明:相较于主流轻量化模型,本文模型以仅2.7×10^6的参数量和18.6 ms的latency,取得了更高的预测精度,在精度、参数量与推理速度三者之间实现了最佳平衡,展现出优异的实际部署潜力。

关键词: 轨迹预测;轻量化;自动驾驶;时空维度;异构解码

基金项目: 国家重点研发计划(No.2023YFF0612100);山东省自然科学基金面上项目(No.ZR2024MF023);山东省重点研发计划(No.2024TSGC0170)

中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112(2026)04-1748-13

电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20260073

A Lightweight Trajectory Prediction Model with a Spatio-Temporal Factorized Encoder and a Heterogeneous Decoder

LIN Qinghua[1], WANG Meifang[1], LI Hui[1*], GE Tongao[2], YAN Jun[1]
(1. *School of Data Science*, *Qingdao University of Science and Technology*, *Qingdao*, *Shandong* 266100, *China*;
2. *School of Computer Science and Artificial Intelligence*, *Wuhan University of Technology*, *Wuhan*, *Hubei* 430070, *China*)

Abstract: Accurate and efficient trajectory prediction is pivotal for safe navigation in autonomous driving systems. Existing methods typically employ graph neural networks (GNNs) or attention mechanisms to jointly model the spatio-temporal dependencies between agents and high-definition maps. Despite notable gains in accuracy, the coupled spatio-temporal modeling incurs excessive computational complexity, severely limiting deployment on embedded vehicular platforms. Moreover, prevailing decoders apply a uniform parameterized paradigm to all agents, overlooking the distinct requirements of different agents in terms of prediction accuracy and multimodality, which leads to inefficient resource allocation. To address these challenges, we propose a lightweight trajectory prediction model based on spatio-temporal dimension decomposition and heterogeneous decoding (MAM-M). In the encoding stage, we design a temporal dual-stream Mamba network that processes historical trajectories through parallel forward and reverse streams, enabling each time step to capture both past and future context. This effectively mitigates the long-range forgetting problem while maintaining linear computational

收稿日期:2026-02-27;录用日期:2026-04-08;责任编委:宋霄罡
*通信作者:李辉

complexity along the temporal dimension. A self-attention mechanism is then employed along the spatial dimension to capture global interactions between agents and map elements, realizing a "temporal-first, spatial-second" decoupled encoding strategy that significantly reduces redundant computations. In the decoding stage, we propose a heterogeneous decoder tailored to different agents: for the target agent with complex motion and high behavioral uncertainty, a mode-query decoder with learnable queries retrieves scene context via cross-attention to accurately capture multimodal trajectory distributions; for the more numerous non-target agents with relatively regular motion, a neural-Bézier decoder predicts Bézier control points through a neural network and generates trajectories via mathematical formulation, substantially reducing inference latency while preserving accuracy and enabling efficient end-to-end training. Experiments on the Argoverse and Argoverse 2 datasets demonstrate that our model, with only 2.7×10^6 parameters and 18.6 ms inference latency, achieves superior prediction accuracy compared to mainstream lightweight models, attaining an optimal balance among accuracy, model size, and inference speed, and exhibiting excellent potential for practical deployment.

Keywords: trajectory prediction; lightweight; self-driving; spatio-temporal dimension; heterogeneous decoder

Foundation Item(s): National Key Research and Development Program of China (No.2023YFF0612100); Natural Science Foundation of Shandong Province (No. ZR2024MF023); Key Research and Development Program of Shandong Province (No.2024TSGC0170)

0 引言

准确预测周围交通参与者的未来轨迹,对自动驾驶系统的决策与规划部分至关重要。精确预测智能体的未来轨迹,关键在于模型能否有效建模场景中智能体历史轨迹与高精地图之间的潜在交互。近年来,基于深度学习的轨迹预测方法取得了显著进展。主流方法多采用单一的图神经网络(Graph Neural Networks,GNNs)[1-2]或Transformer架构,通过联合建模智能体与高精地图之间的时空依赖关系,来提升预测精度。然而,这些轨迹预测方法在兼顾高精度与实时性方面仍存在以下难点问题:

(1)时空耦合的建模方式导致计算复杂度过高。为捕获复杂交互,已有研究通过层次化注意力机制融合多模态场景元素(智能体轨迹和道路)[3-4],或借助GNNs与选择性状态空间模型增强交互建模能力[5-6]。尽管这类方法在精度上表现优异,但其耦合的时空建模方式导致计算复杂度较高,并引入冗余交互,难以满足车载嵌入式系统的实时性要求。同时,此类方法并未重点捕获智能体历史轨迹潜在趋势,其相对于地图结构在预测未来轨迹中具有主导作用[7]。虽然状态空间模型Mamba凭借线性复杂度在效率上表现出显著优势[8],但这种设计引入了固有的信息瓶颈:在单次前向传播中,模型对长序列历史轨迹的"记忆"会随时间衰减,导致"长程遗忘"问题。此外,当仅依赖状态空间结构处理跨模态(轨迹与地图)融合时,其在捕捉复杂空间交互方面的能力仍弱于注意力机制[9-11]。因此,如何改进Mamba类模型以缓解长程遗忘并实现时空维度的高效解耦,是当前面临的一个重要难题。

(2)同质化解码导致资源错配。当前主流的轨迹预测方法在解码阶段普遍采用单一的可学习参数化结构,如直接回归式[4,12-14]或基于锚点[5-6,15-16]的解码器,对场景中所有智能体使用同一可学习参数化范式解码未来轨迹。这种同质化处理忽视了不同智能体在预测任务中对精度、多模态性等方面的差异化需求:目标智能体通常运动情况复杂,需借助参数化网络进行精细建模;而非目标智能体数量较多、运动往往更规则,若仍采用参数化解码器则会导致计算资源浪费。贝塞尔曲线作为一种非参数化方法,无需训练,能够通过数学表达式高效拟合未来轨迹[17],但其依赖预先设定的控制点(通常基于复杂预处理或人工启发式规则),难以嵌入端到端的深度学习框架中进行联合优化。因此,如何根据智能体类型自适应选择解码策略,实现计算资源与预测需求之间的均衡分配,是当前解码设计中的一个难点问题。

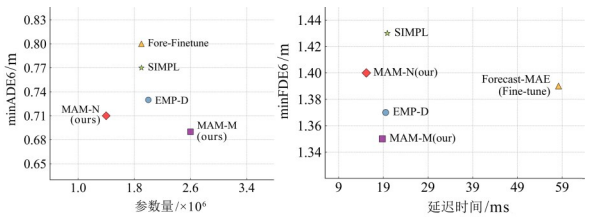
为应对上述挑战,提出一种基于时空维度分解与异构解码的轻量化轨迹预测模型(lightweight trajectory prediction Model based on spatio-temporal dimension decomposition and heterogeneous decoding,MAM-M),为验证非对称解码架构中模式查询解码的有效性,将MAM-M中的模式查询解码替换为多层感知器(MultiLayer Perceptron,MLP),构建了一个名为(lightweight trajectory prediction Model based on spatio-temporal dimension decomposition and MultiLayer perceptron decoding,MAM-N)的变体。这两种模型与主流模型的对比结果如图1所示。在Argoverse系列[18-19]的验证集上,对不同轻量化模型的预测精度与参数量[图1(a)]、预测精度与延迟时间[图1(b)]进行了对比实验(两图中纵坐标数值越低代表预测精度越高)。本文主要贡献如下:

(1)提出一种高效的时空维度分解编码器。首先,通过设计时序双流Mamba网络,利用线性复杂度

技术在时间维度上单独捕捉轨迹点间的时序依赖;其次,通过注意力机制构建轨迹与地图元素间的全局空间关系图。该编码器采用“先时间后空间”的编码逻辑,注重历史轨迹的编码,同时有效减少了在时空维度上的冗余交互。

(2)设计了一种异构解码器。对目标智能体采用可训练的参数化解码结构,精确建模其高度不确定性;对数量较多的非目标智能体采用神经网络与贝塞尔曲线结合的非完全参数化解码结构,以高效且精确地解码其轨迹。该解码器依据不同智能体的预测需求适配不同的解码方式,实现模型性能与计算效率的平衡。

(3)在 Argoverse 系列数据集上进行的消融实验表明:模型在精度、参数量与推理速度三者之间实现了最佳平衡,为自动驾驶轨迹预测的落地应用提供了新的解决方案。



(a) 参数量 (b) 延迟时间

(a) Parameter count (b) Latency time

图 1 训练资源与预测精度对比图

Figure 1 Comparison of training resources and prediction accuracy

1 相关工作

轨迹预测模型通常由编码器与解码器两部分组成。编码器负责挖掘不同智能体之间、智能体与地图之间的潜在交互,并得到编码特征,解码器则基于编码特征生成未来轨迹及其概率分布。

1.1 轨迹预测编码器

基于注意力机制的编码器因其可精确建模不同模态场景元素间的交互,其逐渐成为主流。Ngiam 等人[20]将场景元素统一表示为 token,并通过自注意力机制联合建模其时空关系,Chen 等人[21]与 Liu 等人[22]则进一步通过分层或堆叠结构提升交互建模能力。上述方法采用向量化的方式得到智能体折线和地图折线,维度分别为$[N,T,D]$和$[M,T,D]$,其中N和M分别为智能体折线、地图折线的数量,T代表历史时间步,D代表编码维度。该类方法首先将每个时间步的折线段视作一个基元,这样整个场景就被转换为一个包含$(N+M)T$个基元的集合,然后采用自注意力机制来建模其在时间和空间两个维度上的依赖关系,以提高预测精度,这种耦合式的建模方式会带来$O\left((N+M)^2T^2\right)$的复杂度。同时,注意力机制的计算复杂度随序列度呈平方增长,限制了其在长序列或大规模场景下的应用。为了降低计算复杂度,另一类方法则采用 GNNs 架构[2,5-6],将交通元素抽象为图节点,通过图卷积聚合节点间的交互信息,以建模其时空关联。然而,GNNs 的固有结构限制了其在大规模场景中的推理速度。为了提升效率,Ngiam 等人[20]使用分解注意力对场景元素建模,但仍未从根本上解决耦合建模带来的冗余计算问题。与注意力机制在长序列建模中带来的二次方复杂度不同,Mamba[10]等状态空间模型因其线性的(一次)复杂度优势而备受关注,然而,其在双向依赖建模与多模态融合方面仍存在局限。本文提出双向 Mamba 网络仅对序列化的轨迹点编码,利用结合注意力机制实现时序与空间的解耦编码,在保持全局感知能力的同时显著降低计算负担。

1.2 轨迹预测解码器

基于锚点的解码器[23-24]能够有效建模智能体未来轨迹的不确定性和多模态性。该类解码器通过在未来轨迹可能分布区域预设锚点,并通过 Transformer 架构查询与未来轨迹相关的场景特征,然而预测精度依赖于采集的锚点准确度,且在该类参数化方法中,锚点数量增加带来“模式坍塌”(model collapse)风险。为规避上述问题,Zhou 等人[5]提出基于锚点的两阶段解码策略:先预测初始轨迹,再对其进行精细化优化。然而,这种两阶段模式会显著降低模型的推理速度。

在轨迹规划领域,相关研究[17]采用非参数化的贝塞尔曲线生成未来轨迹。该方法通过伯恩斯坦多项式对智能体周围的控制点(即位置点)进行加权,以产生平滑轨迹。尽管这类非参数化方法凭借其优异的数学特性实现了高效计算,但它们仍需预先获取控制点,因此无法进行端到端训练。

单一的解码范式难以兼顾所有智能体的预测需求。为此,本文提出了一种异构解码架构:将智能体分为目标智能体和非目标智能体,针对需精准捕捉多模态行为的目标智能体,采用完全参数化解码器;而对于作为背景的非目标智能体,则采用高效的非完全参数化方法。这种设计根据预测的优先级智能地分配计算资源,从而在保证精度的同时,极大地提升了模型的整体效率与可扩展性。

1.3 轻量化模型

Zhang 等人[25]尝试通过一种以实例为中心的预处理流程来简化问题。在编码器中,它将相对位置编码与场景特征直接拼接后送入注意力网络。然而,这种简单的特征融合方式不足以捕捉复杂场景下的复

杂交互,导致其在如 Argoverse 2(AV2)这类具有复杂场景的数据集上性能明显下降。因此,Aydemir 等人[26]采用顺序耦合编码[12]方法全面捕获场景下的复杂交互,但为了实现轻量化,在解码阶段舍弃了智能体与全局场景信息的关键交互过程,这种信息的损失同样使其难以应对复杂场景的挑战。与前两者不同,Prutsch 等人[27]完全基于 Transformer 架构搭建模型,不仅在保证交互完整性的同时充分建模了轨迹与地图的关系,还带来源于注意力机制的二次方计算复杂度,使其难以满足实际部署的效率要求。

2　方法

自动驾驶轨迹预测的定义表示。假设智能体(车辆)在 t 时刻的二维坐标为 (x_{ti},y_{ti}),采用 $Z_{ti}=\{(x_{ti},y_{ti})|t=1,2,\cdots,T;i=1,2,\cdots,N\}$ 表示 N 个智能体的历史轨迹坐标,M 表示地图信息。结合历史轨迹和地图信息,预测所有智能体的未来轨迹 $F_{ti}=\{(x_{ti},y_{ti})|t=T+1,T+2,\cdots,T+m;i=1,2,\cdots,N\}$,$m$ 表示未来的时间步。

模型的整体架构如图 2 所示,其中粉色矢量线为预测轨迹,蓝色虚线为历史轨迹。整体架构包含场景预处理、时空因子分解编码器与异构解码器三个核心部分。在场景预处理部分,将场景中的智能体分为目标智能体 $\mathcal{R}$ 和剩余的非目标智能体 $\mathcal{O}$,且所有智能体集合 $\mathcal{N}=\{\mathcal{R}\}\cup\{\mathcal{O}\}$。为了使模型架构更加轻量化且精确捕获目标智能体未来轨迹的多模态性,首先,将目标智能体的数量设置为 1;其次,从原始数据中提取所有智能体历史轨迹和道路信息,并将其表示为向量化的折线,即智能体折线和地图折线;再次,时空因子分解编码器利用提出的时序双流 Mamba 网络对智能体折线进行编码得到智能体特征,同时采用分层道路嵌入对地图折线编码得到道路特征;最后,将得到的场景特征(智能体特征、道路特征)送入异构解码器。该解码器针对不同智能体采用了差异化设计:对于目标智能体 $\mathcal{R}$,设计了全参数化模态查询解码结构,通过可学习的锚点查询并结合跨注意力机制,从场景特征中检索上下文信息,以精确刻画其未来轨迹的多模态与不确定性;对于非目标智能体 $\mathcal{O}$,则采用一种非完全参数化解码结构,通过在贝塞尔曲线中引入神经网络生成控制点,避免了复杂的数据预处理。此种参数化与非完全参数化相结合的异构解码架构,以较低的推理延迟(latency)取得了优异的预测精度。

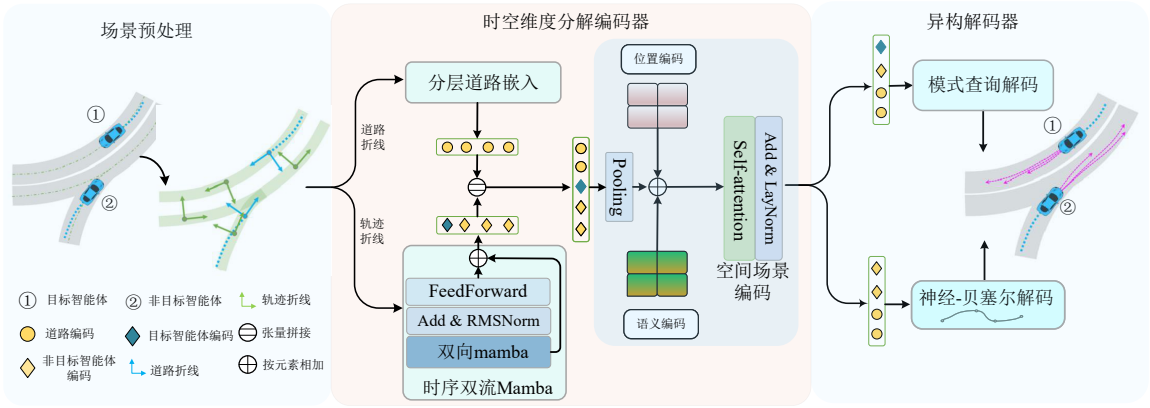


图 2　MAM-M 框架的总体结构

Figure 2　Overall architecture of the MAM-M framework

2.1　场景预处理

本文采用主流的矢量化表示方法,将智能体的历史轨迹和道路分割为折线。具体来讲,选用 $\boldsymbol{Z}_m\in\mathbb{R}^{A\times T\times C}$ 表示场景内所有智能体的历史轨迹,其中 A 为智能体的数量,C 为每个时间步的语义属性,包括每个时间步的二维位置、速度、朝向角和掩码标志。掩码标志用于指示智能体在各时间步是否被观测到。地图信息使用 $\boldsymbol{L}\in\mathbb{R}^{S\times M\times 4}$ 表示,其中 S 表示道路的数量,M 表示每条道路中折线的数量。每条折线由一个四维特征向量表示,该向量包含其中心点的二维位置、朝向角度及一个有效性标志。

2.2　时空维度分解编码器

时序双流 Mamba 利用 Mamba 的线性复杂度优势,在时间维度上建立智能体折线自身的交互关系,复杂度为 $O(N)$。为了捕获所有场景特征 E(包括智能体和道路)之间的空间交互,该部分将它们(共 $N+S$ 个元素)一同输入自注意力层,得到更新后的智能体特征 $\boldsymbol{Z}_{fa}$ 和道路特征 $\boldsymbol{M}_{fa}$,复杂度为 $O\left((N+S)^2\right)$,从而使时空维度分解编码器的复杂度仅为 $O\left((N+S)^2+N\right)$。该编码器不仅减少了时空维度上的冗余交互,还在保证精度的前提下提升了运算效率。

2.2.1 时序双流 Mamba

基于 Attention 机制的网络擅长捕获多模态场景(地图和智能体的历史轨迹)间的交互特征,但会带来 $O(N^2)$ 复杂度;Mamba 作为一种新兴的状态空间模型,凭借其自我回归机制以线性复杂度 $O(N)$ 编码序列,可将其用于历史轨迹的编码。然而,该机制具有单向性,导致先输入的信息相较于后输入的信息更易被遗忘。为此,设计了双向 Mamba 网络:首先,将原始轨迹 $\boldsymbol{Z}$ 通过两组线性层映射后分别得到 $\boldsymbol{Z}_I \in \mathbb{R}^{N\times T\times D}$,$\boldsymbol{Z}_R \in \mathbb{R}^{N\times T\times D}$;其次,将 $\boldsymbol{Z}_I$ 送入 S_2(SSM)得到编码特征;最后,将两种编码特征相加并引入残差。SSM 具体数学表达式如下:

$$\boldsymbol{h}(t)=\bar{\boldsymbol{A}}\boldsymbol{h}(t-1)+\bar{\boldsymbol{B}}\boldsymbol{x}_f(t) \tag{1}$$

$$\boldsymbol{y}_f(t)=\boldsymbol{C}\boldsymbol{h}(t-1)+\boldsymbol{D}\boldsymbol{x}_f(t) \tag{2}$$

$$\boldsymbol{y}_f=[\boldsymbol{y}_{f1},\boldsymbol{y}_{f2},\cdots,\boldsymbol{y}_{fT}] \tag{3}$$

其中,$\bar{\boldsymbol{A}} \in \mathbb{R}^{v\times v}$、$\bar{\boldsymbol{B}} \in \mathbb{R}^{v\times v}$ 为 $\boldsymbol{A}$ 和 $\boldsymbol{B}$ 离散化后的参数,$\boldsymbol{C} \in \mathbb{R}^{p\times v}$ 和 $\boldsymbol{D} \in \mathbb{R}^{p\times t}$ 为参数矩阵,v 为隐藏层维度,t 为某个时间点,$\boldsymbol{h}(t) \in \mathbb{R}^{v\times 1}$ 为每一时间步 t 的隐藏状态,$\boldsymbol{A}$、$\boldsymbol{B}$、$\boldsymbol{C}$、$\boldsymbol{D}$ 为根据输入自适应调整的参数矩阵。由于每个时间步的状态更新和输出均由矩阵乘法得到,因此该方法的计算复杂度是线性的。其最后输出如式(3)所示,$\boldsymbol{y}_f \in \mathbb{R}^{N\times T\times D}$,且 $\boldsymbol{y}_{fi} \in \mathbb{R}^{N\times 1\times D}$ 代表每一个时刻的编码。

进一步地,将 S_1 和 S_2 得到的编码特征相加后通过 RMSNorm 归一化,最后将 $\boldsymbol{Z}_I$ 作为残差与输出相加得到最终的轨迹编码特征 $\boldsymbol{Z}'$。该网络得到更完整、更均衡的轨迹编码,每个时间点均同时包含了其完整的“过去”和“未来”信息。整体架构如图 3 所示,具体表达式为

$$\boldsymbol{Z}_m=\mathrm{RN}\left\{S_1\left(\mathrm{MLP}(\boldsymbol{Z})\right)+S_2\left(\mathrm{MLP}(\boldsymbol{Z})\right)\right\} \tag{4}$$

其中,RN 为 RMSNorm,其为 Mamba 结构中的归一化层,$\boldsymbol{Z}_m \in \mathbb{R}^{N\times T\times D}$。首先,将继续使用 RMSNorm 对每个智能体的特征进行重新缩放,以此稳定训练过程。然后,通过 FeedForward 进一步提高模型的非线性表征能力,表达式为

$$\boldsymbol{Z}_f=\boldsymbol{Z}'+\mathrm{FFN}\left(\boldsymbol{Z}'+\mathrm{RN}(\boldsymbol{Z}')\right) \tag{5}$$

其中,FFN 为 FeedForward;$\boldsymbol{Z}_f \in \mathbb{R}^{N\times T\times D}$。

历史轨迹同样将每条道路分割为折线。由于车道在地理位置上跨度较大,为保证训练过程的稳定性,将道路折线以其中心点为基准进行归一化,并用 $\boldsymbol{L} \in \mathbb{R}^{S\times M\times 4}$ 表示,与轨迹折线不同,道路折线通常包含更多的点,为保留其空间拓扑和局部细节结构,先使用一维卷积网络对道路的局部结构进行编码,再将得到的特征拼接以获取统一的整体道路结构特征,同时进一步通过卷积池化网络得到最后的输出。相关

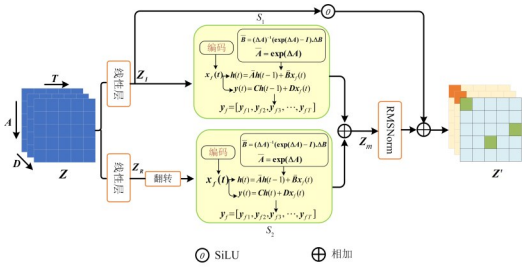


图 3 双向 Mamba 架构示意图

Figure 3 Illustration of the bidirectional Mamba architecture

表达式如下:

$$\boldsymbol{F}_1=\mathrm{Conv1D}\left(\mathrm{RELU}\left(\mathrm{Conv1D}(\boldsymbol{M}^{\mathrm{T}})\right)\right) \tag{6}$$

$$\boldsymbol{F}_2=\mathrm{Concat}\left(\mathrm{MaxPool}(\boldsymbol{F}_1),\boldsymbol{F}_1\right) \tag{7}$$

$$\boldsymbol{M}_f=\mathrm{MaxPool}\left(\mathrm{Conv1D}(\boldsymbol{F}_2^{\mathrm{T}})\right) \tag{8}$$

其中,$\boldsymbol{F}_1 \in \mathbb{R}^{S\times M\times D}$,Conv1D 为一维卷积,$\boldsymbol{M}^{\mathrm{T}}$ 为 $\boldsymbol{M}$ 的转置,MaxPool 为沿着折线数量维度最大池化,Concat 为拼接操作,$\boldsymbol{F}_2 \in \mathbb{R}^{S\times 2D}$,$\boldsymbol{F}_2^{\mathrm{T}}$ 代表 $\boldsymbol{F}_2$ 的转置,输出 $\boldsymbol{M}_f \in \mathbb{R}^{S\times D}$。

2.2.2 空间场景编码

为使模型更好地理解场景间的语义关系,首先将位置编码和智能体类别、地图中道路类型等语义信息编码后加入轨迹特征和地图特征中。表达式如下:

$$\mathrm{PE}=\mathrm{MLP}(x,y,v,\sin\theta,\cos\theta) \tag{9}$$

$$\boldsymbol{Z}_{fa}=\mathrm{MaxPool}(\boldsymbol{Z}_f) \tag{10}$$

$$\boldsymbol{E}=\mathrm{Concat}(\boldsymbol{Z}_f,\boldsymbol{M}_f)+\mathrm{PE}+\mathrm{Cl}_m+\mathrm{Cl}_s \tag{11}$$

其中,D 为隐藏层维度,PE 为位置编码,式(9)中的 (x,y,θ) 为轨迹折线和地图折线中心位置的姿态,v 为智能体的速度,式(10)中得到的 $\boldsymbol{Z}_{fa} \in \mathbb{R}^{N\times D}$,式(11)中 Cl_m 为语义编码,Cl_s 为智能体类别编码(“0”代表目标智能体,“1”代表非目标智能体),$\boldsymbol{E} \in \mathbb{R}^{(N+S)\times D}$ 为场景特征。然后,使用自注意力层对场景特征 $\boldsymbol{E}$ 进行编码。

2.3 异构解码器

2.3.1 模式查询解码器

为了更好地表示目标智能体未来轨迹的多模态性和不确定性,与已有方法[12,17,24]不同,该解码器并未预采集未来区域的位置点作为锚点,而是初始化一组可学习锚点 $\boldsymbol{Q} \in \mathbb{R}^{K\times D}$,其中 K 代表未来可能的多条轨迹,D 代表编码维度。随后,在交叉注意力层中,将 $\boldsymbol{Q}$ 作为 Query,场景特征作为 Key 和 Value,以查询影响目标智能体未来轨迹的信息,该方法加速模型收敛的同时并更新 $\boldsymbol{Q}$ 得到 $\boldsymbol{Q}_n$。接着,通过 Mamba 精确捕捉同一条预测轨迹之间的时序依赖,为了实现多模态间的信息交互,在模态维度 K 上应用一个自注意力层,并回归出目标智能体的初始轨迹 $\boldsymbol{R}_F \in \mathbb{R}^{K\times m\times 2}$ 和每条轨迹的概率 $\boldsymbol{R}_S \in \mathbb{R}^{K\times 1}$,其中“2”代表二维位置点。为进一步提高预测轨迹的精度,将 $\boldsymbol{Q}_n$ 输入到 MLP 中回

归出预测轨迹的误差,并利用偏差矫正原始轨迹得到最终轨迹,如图4所示。

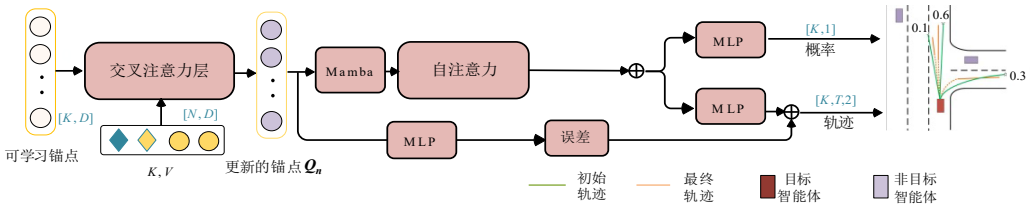


图4　模式查询解码示意图

Figure 4　Schematic diagram of pattern query decoding

2.3.2　神经-贝塞尔解码

基于编码阶段获得的非目标智能体编码,通过贝塞尔曲线进行回归计算,可高效预测其未来轨迹。对于 n 阶贝塞尔曲线,有 n+1 个控制点,控制点 $\boldsymbol{P}_i$ 和控制点的权重 $\boldsymbol{B}_{n,i}(u)\in\mathbb{R}^{m\times(n+1)}$ 加权得到未来 m 个时间步的轨迹,表示为

$$\boldsymbol{C}(u)=\sum_{i=0}^{n}\boldsymbol{B}_{n,i}(u)\boldsymbol{P}_i \tag{12}$$

其中,u 为参数,取值范围[0,1],表示曲线从 P_0 到 P_n 的进度。$\boldsymbol{B}_{n,i}(u)$ 的表达式为

$$\boldsymbol{B}_{n,i}(u)=\left(\frac{n!}{i!(n-i)!}\right)u^i(1-u)^{n-i} \tag{13}$$

其中,$\boldsymbol{B}_{n,i}(u)$ 中第 m 行、第 i 列的数值代表在 m 时刻第 i 个控制点对当前轨迹位置的权重。若直接使用贝塞尔曲线作为解码器,需通过数据预处理以获得与未来轨迹相关的 $\boldsymbol{P}_i$,再通过式(13)生成轨迹。其两阶段的解码方式阻碍了端到端解码流程的实现。为克服上述局限,首先通过MLP回归得到控制点 $\boldsymbol{P}_i$,如式(14)所示。其次,使用贝塞尔曲线拟合控制点得到未来 K 条轨迹,如式(12)~式(13)所示。最后,使用MLP对 $\boldsymbol{Z}_{ff}$ 分类得到每条轨迹的概率分数 $\boldsymbol{C}_S\in\mathbb{R}^{K\times1}$。

$$\boldsymbol{P}_i=\mathrm{MLP}(\boldsymbol{Z}_{ff}) \tag{14}$$

其中,$\boldsymbol{Z}_{ff}\in\mathbb{R}^{(N-1)\times D}$ 输入MLP为每个非目标智能体预测出控制点 $\boldsymbol{P}_i\in\mathbb{R}^{K\times(n+1)\times2}$,$\boldsymbol{Z}_{ff}$ 由 $\boldsymbol{Z}_{fa}$ 去掉目标智能体编码得到。随后将 $\boldsymbol{B}\in\mathbb{R}^{K\times m\times(n+1)}$ 与预测出的2D控制点 $\boldsymbol{P}_i$ 相乘,得到每个非目标智能体在 m 个时间步的 K 条未来轨迹 $\boldsymbol{C}(u)\in\mathbb{R}^{K\times m\times2}$($x$ 和 y 两个维度相互独立)。通过神经网络与数学表达式结合的方式,减少网络训练带来的时间消耗;同时,解码器仅需预测 n+1 个位置点,而非完整的 m 个点,极大地降低了计算复杂度。这使得对数量较多的非目标智能体实行端到端训练和推理更加高效。非目标智能体其中一条轨迹 $\boldsymbol{C}_1(u)$ 的表达式为

$$\boldsymbol{C}_1(u)=\begin{bmatrix}b_n^0(t_1) & \cdots & b_n^n(t_1)\\ \vdots & \vdots & \vdots\\ b_n^0(t_m) & \cdots & b_n^n(t_m)\end{bmatrix}\begin{bmatrix}p_0^x & p_0^y\\ \vdots & \vdots\\ p_n^x & p_n^y\end{bmatrix} \tag{15}$$

其中,m 为预测轨迹所包含的时间步数量,t_i 为归一化的时间点,(p_i^x,p_i^y) 为第 i 个控制点的二维位置。

3　实验

3.1　实验步骤

3.1.1　数据集

为了评估模型的性能,本文选用Argoverse(AV1)和AV2两个主流自动驾驶运动预测数据集。AV1提供了20.5万个驾驶场景,要求模型依据2 s的10 Hz历史轨迹,预测未来3 s的轨迹。AV2数据集规模更庞大,包含25万个场景和更丰富的地图信息,其任务也更具挑战性:观测5 s,预测未来6 s。依照标准设定,分别将AV1和AV2的智能体感知范围设置为65 m与150 m。

3.1.2　评价指标

根据Argoverse竞赛使用的评估协议,使用以下指标:minFDEk,K 条轨迹中最优轨迹的终点与真实轨迹终点间的 L_2 距离;minADEk,K 条轨迹中最优轨迹的终点与真实轨迹之间的平均 L_2 距离;Miss rate(MR6),误差超过2 m阈值的场景占比b-minFDEk,minFDEk加上修正项$(1-p)^2$,其中 p 代表在模型预测的所有轨迹中,与真实未来轨迹(Ground truth)误差最小的一条轨迹所对应的概率。

3.1.3　实验细节

在模式查询解码中,将Cross-Attention的层数设置为3,并将注意力的头数设置为8。为更方便地在AV1和AV2数据集上评估,可将学习锚点 $\boldsymbol{Q}\in\mathbb{R}^{K\times D}$ 中的 K 设置为6,隐藏层维度为128。模型为所有智能体生成6条未来轨迹。本文使用AdamW优化器在4张NVIDIA A10 GPU(24 GB VRAM)上训练模型60轮,学习率设置为0.001,batch size设置为128。模型的损失函数由Huber Loss[28-29](计算回归损失)和Cross-Entropy Loss[30](计算分类损失)构成。

3.2　实验结果

如表1所示,将本文提出的模型与SOTA(State-Of-The-Art)方法在预测精度和参数量方面进行了对比测试。

表1 AV2预测数据集轨迹预测结果

Table 1 Trajectory prediction results on the Argoverse 2 test dataset

模型	出版物	minADE6↓	minFDE6↓	MR6↓	minADE1↓	minFDE1↓	b-minFDE6↓	Para/×10^6
SIMPL[25]	IEEE RA-L	0.72	1.43	0.19	2.03	5.50	2.05	2.6
EMP-M[27]	IROS	0.72	1.43	0.19	1.80	4.53	2.07	2.0
EMP-D[27]	IROS	0.71	1.37	0.17	1.75	4.35	1.98	3.2
THOMAS[7]	ICLR	0.88	1.51	0.20	1.95	4.71	2.16	—
HPTR[15]	NeurIPS	0.73	1.43	0.19	1.84	4.61	2.03	15.0
MTR[16]	NeurIPS	0.73	1.44	**0.15**	1.74	4.39	1.98	65.8
QCNet[5]	CVPR	**0.65**	**1.29**	0.16	**1.69**	4.30	**1.91**	7.7
Fore[13]-Scratch	ICCV	0.73	1.43	0.19	1.85	4.60	2.06	1.9
Fore[13]-Finetuned	ICCV	0.71	1.39	0.17	1.74	4.36	2.03	1.9
BANet[31]	Arvix	0.72	1.43	0.19	1.84	4.70	2.03	9.5
GANet[32]	ICRA	0.71	1.34	0.17	1.77	4.48	1.96	61.7
GoRela[33]	ICRA	0.76	1.48	0.22	1.84	4.62	2.01	—
PerReg[34]	CVPR	0.77	1.48	0.21	1.84	4.62	2.07	—
MAM-M(ours)		0.70	1.32	0.17	1.71	**4.23**	1.94	2.7
MAM-N(ours)		0.71	1.39	0.18	1.79	4.44	2.05	1.6

表1中带↓的指标表示数值越低精度越高，加粗字体表示最优，下划线表示次优，“—”表示无法复现，Para代表参数量，单位为×10^6。MAM-M在预测精度上优于高效运动预测-类DETR解码器(Efficient Motion Prediction-DETR-like decoder，EMP-D)[27]，同时实现了更少的参数量和更低的latency。MAM-M(2.7×10^6)的参数量不仅远少于包括基于相对姿态编码的异构折线Transformer(Heterogeneous Polyline Transformer with Relative pose encoding，HPTR)[15](采用耦合时空编码结构)、PerReg(Perceiver with Register queries)[33](基于GNNs)、运动Transformer(Motion TRansformer，MTR)[16](65.8×10^6)和边界感知网络(Boundary-Aware Network，BANet)[31](9.5×10^6)在内的SOTA模型，而且预测精度仍超越了它们，这充分说明性能的关键在于架构设计，而非参数堆砌。

此外，相较于EMP-D[27]、Fore-Finetuned[13]和Fore-Scratch[13]等轻量化预测模型，MAM-M具有更少的参数和更优越的性能。尽管高效运动预测-基于MLP的解码器(Efficient Motion Prediction-MLP-based decoder，EMP-M)[27]和简单高效的运动预测基线(Simple and effIcient Motion Prediction baseLine，SIMPL)[25]的参数更少，但它们的MR6和minADE6指标(值越高，性能越差)分别比MAM-M高11.8%和2.9%，且SIMPL[25]的minFDE1指标也比MAM-M高23.0%。

特别地，仅有1.6×10^6参数量和简洁解码结构的MAM-N，其精度和推理速度已超越了所有轻量级模型。这些结果强有力地证明了解耦时空编码器设计的合理性。“先时间后空间”的分解编码策略通过对场景元素在时间和空间分解编码，在实现对历史轨迹单独编码的同时，不仅简化了模型架构，还减少了冗余的场景交互，从而在轻量级轨迹预测任务中实现了精度与效率的有效提升。

为评估模型的计算效率与实际部署潜力，引入了两项关键指标：浮点运算次数(FLOating Point operations，FLOPs)和latency，结果如表2所示。“—”表示无法复现，加粗字体表示最优，下划线表示次优，FLOPs(以GFLOPs计)反映了理论计算复杂度，而latency(以ms计)则衡量了将batchsize设置为1时，在单张NVIDIA A10 GPU上的实际推理耗时。

表2 MAM-M模型推理

Table 2 Inference performance of the MAM-M model

模型	minFDE6↓	FLOPs/GFLOPs	Latency/ms
MAM-M	1.32	0.212	18.6
MAM-N	1.39	**0.208**	**16.5**
EMP-D[27]	1.40	0.952	20.0
SIMPL[25]	1.43	1.235	19.7
QCNet[5]	1.29	13.357	82.9
Fore[13]-Scratch	1.44	0.548	58.3
GANet[32]	1.34	4.650	—
Tamba[6]	**1.24**	11.204	68.7

与EMP-D[27]、Fore-Scratch[13]及SIMPL[25]等轻量化模型相比，MAM-N模型在获得更高预测精度的同时，实现了显著更低的计算复杂度和推理时间。相较于MAM-N，MAM-M的FLOPs仅增加了1.8%，却将minFDE6降低了5.7%，展示了效率与性能之间的理想权衡。此外，与EMP-D[27]相比，MAM-M不仅延迟更低，

其 minFDE6 和 FLOPs 还分别降低了 6.1% 和 33.6%。QCNet[5]通过时空耦合编码器在时空维度上为场景元素建立语义连接,导致其计算成本是MAM-M的63倍,latency 是 MAM-M 的 4.4 倍。这进一步证实了时空维度分解编码器有效减少场景中的冗余交互。同样,Tamba[6]仅取得了略高于 MAM-M 的精度,但计算复杂度却极高(FLOP 是 MAM-M 的 52.8 倍),目前主流的自动驾驶部署平台算力水平与 NVIDIA A10 GPU 相似,MAM-N 仅有 18.6 ms 的 latency 和 0.212 GFLOPs,因此,相对于以查询为中心的网络(Query-Centric Network,QCNet)[5]和 Tamba[6]的高昂计算需求,其更适用于真实的自动驾驶系统部署。

结合表 1 和表 2 可得,与 MAM-N 相比,MAM-M 中的模式查询解码模块显著提升了对多模态和不确定未来轨迹的建模能力。

为进一步检验模型的泛化能力,仅使用一个 NVIDIA A10 GPU 在 AV1 验证集上进行了评估。该数据集包含较短的轨迹序列和相对简单的道路结构。如表 3 所示,本文将预测精度、训练时间与当前的轻量级模型进行了比较,其中加粗字体表示最优,下划线表示次优。实验结果表明:模型在所有对比方法中取得了最高的预测精度,更重要的是,该高精度是以相对较低的训练时间成本实现的,这证实了提出的模型成功地在性能与效率之间达到了平衡,且即便在较简单的场景中仍展现出强大的泛化能力。

表 3　AV1 验证数据集评测结果

Table 3　Evaluation results on the Argoverse validation dataset

模型	MR6 ↓	minADE6 ↓	minFDE6 ↓	Training time/h
Fore[13]-Scratch	0.10	0.74	1.10	~21.7
Fore[13]-Finetune	0.095	0.71	1.05	~71.0
EMP-M[27]	0.096	0.63	1.04	~11.8
EMP-D[27]	0.090	0.63	1.02	~15.9
MAM-M	**0.075**	**0.60**	**0.90**	~12.3
MAM-N	<u>0.080</u>	<u>0.61</u>	<u>1.01</u>	~11.1

3.3　消融实验

为评估模型中各组成部分的有效性,进行了一系列消融实验研究,结果如表 4 所示,其中加粗字体表示最优,下划线表示次优。模型一至模型四针对 MAM-M 结构进行消融实验。实验表明:随着时序双流 Mamba、空间场景编码及神经-贝塞尔解码模块的引入,模型的预测精度逐步提升,且 Latency 逐渐降低。与模型三相比,模型四(MAM-M)采用模式查询解码替代 MLP 来预测目标智能体的未来轨迹,尽管 latency 增加了 4 ms(如表 2 所示,仍优于轻量级 SOTA 模型 EMP-D),但其 minFDE6 和 MR6 指标分别显著下降 5.8% 和 5.9%,从而在精度与效率之间实现了更优的平衡。

表 4　AV2 预测数据集消融实验

Table 4　Ablation study on the Argoverse 2 validation dataset

模型	编码器		解码器		评价指标			
	时序双流 Mamba	空间场景编码	目标智能体	非目标智能体	latency/ms ↓	minADE6 ↓	minFDE6 ↓	MR6 ↓
模型一	×	×	MLP	MLP	17.5	0.734	1.463	0.194
模型二	√	√	MLP	MLP	**15.9**	0.718	1.412	0.186
模型三	√	√	MLP	神经-贝塞尔解码	<u>16.5</u>	0.710	1.392	0.180
模型四	√	√	模式查询解码	神经-贝塞尔解码	19.6	**0.698**	**1.333**	**0.170**
模型五	√	√	模式查询解码	MLP	20.9	0.712	1.344	0.175
模型六	√	√	神经-贝塞尔解码	神经-贝塞尔解码	17.1	0.707	1.406	0.176
模型七	√	√	模式查询解码	模式查询解码	29.8	<u>0.702</u>	<u>1.340</u>	<u>0.171</u>
模型八	Mamba	√	模式查询解码	神经-贝塞尔解码	18.5	0.709	1.360	0.176
模型九	时空耦合编码器[5]		模式查询解码	神经-贝塞尔解码	56.7	0.703	1.349	0.175

模型五至模型七对异构解码器进行消融实验,并与模型四对比。模型四的 latency 和 minADE6 相对于模型五分别降低 6.6% 和 2.0%,相对于模型七分别降低 34.2% 和 0.5%,进一步说明针对不同智能体的预测需求采用差异化解码方式的合理性。

由模型四和模型八的实验结果可知,模型四(MAM-M)使用时序双流 Mamba 对历史轨迹编码,使 minFDE6 和 MR6 分别降低 2.0% 和 3.5%,在仅增加 1.1 ms 的 latency 的情况下,有效缓解 Mamba 存在的“长程遗忘”问题。

由模型四和模型九的实验结果可知,模型四的 latency 降低了 65.4%,同时使 minFDE6 和 MR6 分别降低了 1.2% 和 3.0%,说明时空维度分解编码器(相较于模型九的时空耦合编码器[5])独立编码历史轨迹的同时有效减少场景冗余交互,在预测精度和效率上均展现出更优的整体性能。

为了验证时序双流 Mamba 缓解“长程遗忘”的优势,本文使用 Bidirectional GRU、Transformer Encoder、Bi-

directional LSTM 分别代替时序双流 Mamba 去做消融实验。Bi-GRU 代表 Bidirectional GRU，T-En 代表 Transformer Encoder，Bi-LSTM 代表 Bidirectional LSTM，Bi-Mamba 代表时序双流 Mamba，实验结果如表 5 所示。

表 5 时序双流 Mamba 消融实验结果

Table 5 Ablation results of the temporal dual-stream Mamba

模型	minADE6 ↓	minFDE6 ↓	MR6 ↓	latency/ms	Para/×10^6
Bi-GRU[35]	0.715	1.362	0.174	19.9	3.2
T-En[36]	0.720	1.370	0.178	21.6	2.9
Bi-LSTM[37]	0.719	1.379	0.177	20.3	3.4
Bi-Mamba	0.698	1.333	0.170	19.6	2.7

Bi-GRU 和 Bi-LSTM 虽然引入双向门控机制，但受限于串行计算模式，且容易遗忘序列早期的关键信息。T-Encoder 则受制于注意力机制的二次方复杂度，难以平衡长序列建模与计算效率。相较而言时序双流 Mamba 利用 SSM 的线性复杂度优势，通过双向流结构对历史轨迹进行全局编码，更有效地捕获轨迹点间的潜在交互，并保持较低的 latency。

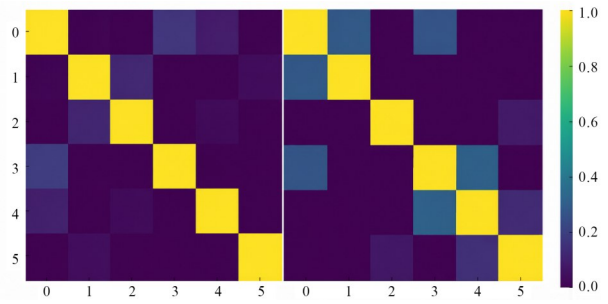
3.4 正交可视化

模式崩塌是指模型预测的未来轨迹丧失了多样性。为降低因锚点数量冗余而带来的模式崩塌风险，并保留多模态不确定性，仅为目标智能体设置可学习的锚点（每个锚点代表一条预测轨迹）。随后，将这组锚点进行随机初始化，以确保多条未来轨迹在初始状态便具有差异性。

在图 5 中，横轴与纵轴均表示 6 个锚点的索引。颜色代表余弦相似度数值，黄色越深表示锚点间余弦相似度越高，蓝色越深表示相似度越低。矩阵中第 i 行第 j 列方格的颜色反映了第 i 个锚点与第 j 个锚点的余弦相似度。如图 5(a)所示，不同可学习锚点之间的低余弦相似度与锚点自身的高余弦相似度，共同证实了该初始化策略的有效性。经训练之后，如图 5(b)所示，部分锚点之间的相似性有所增加，但这并不表示发生了模式崩塌，反而表明：在向真实未来轨迹收敛的同时，这些模式依旧保留了各自的模态特征，从而共同捕捉了轨迹的不确定性。

3.5 训练资源对比

如图 6 所示，横坐标表示模型名称，左半部分纵坐标表示达到最佳性能所需的最短训练时间（单位为 h），右半部分纵坐标为训练过程中 GPU 显存利用率的最大值，所有模型的 batchsize 均设为 128。结合图 6 与表 1 综合分析可知，提出的模型在训练效率与预测精度上优势显著。MAM-M 得益于“先时序后空间”的解耦编码与非对称解码策略，在降低训练时间的同时显



（a）训练前 （b）训练后

（a）Before training （b）After training

图 5 可学习锚点在训练前后的正交性图示

Figure 5 Orthogonality illustration of learnable anchors before and after training

著提升预测精度。相比之下，SIMPL[25]虽 GPU 内存占用最低，但其训练时间分别为 MAM-M 的 3.4 倍、MAM-N 的 5.0 倍，且预测精度远不及提出的模型。同样，Fore Scratch[27]参数量更少，但完全基于 Transformer 的架构导致模型复杂度高，不仅预测精度低于 MAM-N，训练时间也为后者的 2.5 倍。综上，MAM-M 与 MAM-N 凭借高效架构设计，在训练时间、latency 及预测精度间实现更优权衡，证明其在真实自动驾驶系统中具有更高部署价值。

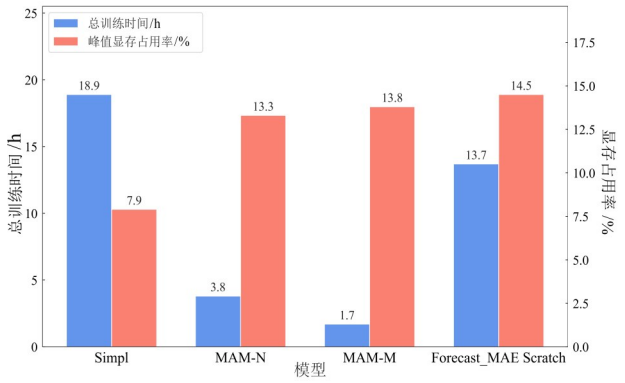


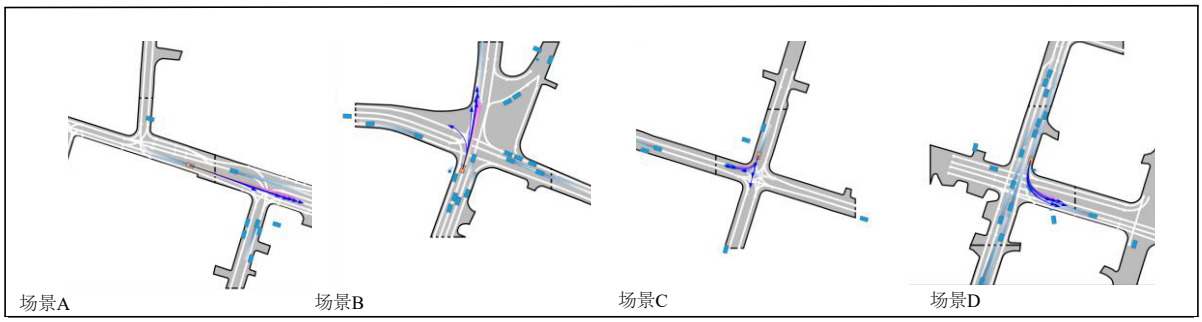
图 6 轻量化模型训练过程对比图

Figure 6 Comparison of the training process of lightweight models

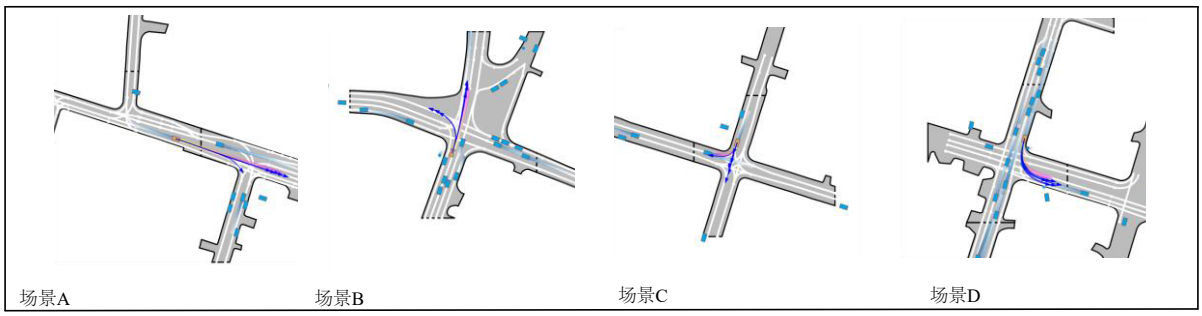
3.6 可视化结果

图 7 展示了 MAM-M 模型与精度最高的轻量化模型 EMP-D 在同一场景下的对比结果。图 7(a)为 MAM-M 模型的可视化结果，图 7(b)为 EMP-D 模型的可视化结果。粉色渐变线段为真实的未来轨迹，蓝色矢量线为预测轨迹，黄色方形为目标智能体，蓝色渐变代表历史轨迹，绿色方形代表非目标智能体。

场景 B、C、D 为高密度交叉路口，涉及复杂交互；场景 A 则为长直道。此类驾驶情境对模型建模长距离时序依赖的能力提出了更高要求，即模型必须有效



(a) MAM-M

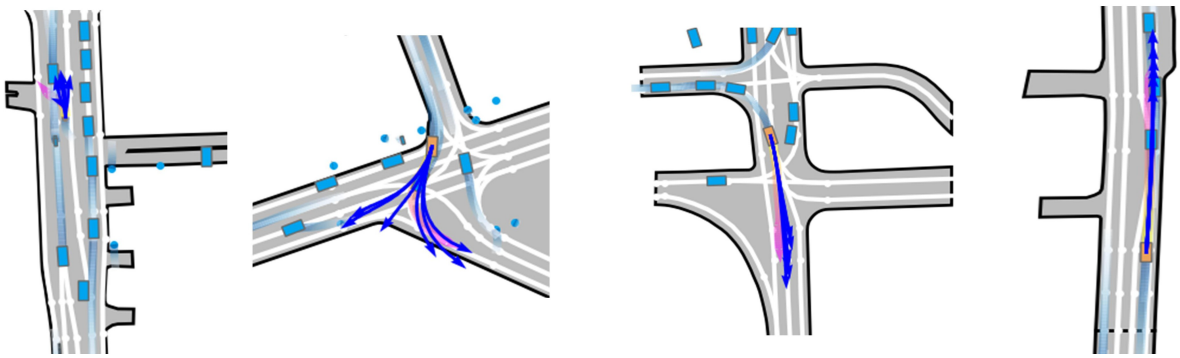


(b) EMP-D

图7　可视化轨迹对比图

Figure 7　Qualitative comparison of trajectory predictions

捕捉并利用早期的历史轨迹信息,以提高预测精度。得益于编码器设计,MAM-M能够有效建立这种长距离依赖性,因此减少了对初始轨迹状态的遗忘。在此类场景中,尽管EMP-D能够生成相对准确的多模态轨迹,但仍有部分预测轨迹明显偏离真值。为进一步地展现本文模型(MAM-M)的泛化性,可视化了其在极端路况下的轨迹预测图,如图8所示。图8(a)展示了低速行驶状态下的突发变道场景。受限于稀疏和较大的不确定性,模型预测出的多模态轨迹中仍然有一条接近于真实轨迹;图8(b)为复杂的"S"型轨迹场景,得益于将神经网络和贝塞尔曲线结合,充分发挥神经网络挖掘潜在交互的能力,回归出精确的控制点(位置点),相较于直接使用贝塞尔曲线准确度较高,解码出的轨迹也能较好地与真实未来轨迹重合;图8(c)与图8(d)分别展示了十字路口处的转向变道及道路拓扑结构的突变场景。即便在上述极端情况下,模型依然表现出鲁棒的预测性能,能够输出精确的未来轨迹。



(a) 场景A　(b) 场景B　(c) 场景C　(d) 场景D

(a) Scenario A　(b) Scenario B　(c) Scenario C　(d) Scenario D

图8　极端驾驶路况可视化图

Figure 8　Visualization in extreme driving conditions

4 结束语

本文提出了一种基于时空维度分解与异构解码的轻量化轨迹预测模型,在保证预测精度的同时提升计算效率,较好地兼顾了车载场景对高精度与实时性的双重需求。模型核心由时空维度分解编码器与异构解码器构成:在编码器阶段,通过分离时空维度的建模方式,利用时序双流 Mamba 模块有效捕捉长程轨迹趋势,并借助注意力机制融合道路与交互信息,从而在降低冗余计算的同时增强了关键特征的提取能力;在解码器阶段,针对非目标智能体设计了神经-贝塞尔解码器,实现高效、免人工规则的端到端轨迹生成,而对目标智能体则采用可学习的模式查询解码器,通过锚点机制主动检索场景上下文信息,以精确捕捉其未来轨迹的多模态分布。在 AV1 与 AV2 数据集上的实验表明:该模型在训练与推理时间方面均表现出显著优势,证明了其在资源受限的嵌入式系统上的实用价值。

当前的异构解码器依据目标与非目标智能体进行二元划分。未来可以探索更动态、更精细的智能体划分策略,例如,根据智能体在场景中的实时交互复杂度或行为不确定性,自适应地为其分配不同计算开销的解码器[38],以实现更优的全局性能与效率平衡。

参考文献

[1] Tang Xiaolong, Kan Meina, Shan Shiguang, et al. HPNet: Dynamic trajectory forecasting with historical prediction attention[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2024: 15261-15270.

[2] 孔玮, 刘云, 李辉, 等. 基于全局自适应有向图的行人轨迹预测[J]. 电子学报, 2022, 50(8): 1905-1916.
Kong Wei, Liu Yun, Li Hui, et al. Pedestrian trajectory prediction based on global adaptive directed graph[J]. Acta Electronica Sinica, 2022, 50(8): 1905-1916. (in Chinese)

[3] Nayakanti N, Al-Rfou R, Zhou A, et al. Wayformer: Motion forecasting via simple & efficient attention networks[C]//IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2023: 2980-2987.

[4] Jia Xiaosong, Wu Penghao, Chen Li, et al. HDGT: Heterogeneous driving graph transformer for multi-agent trajectory prediction via scene encoding[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(11): 13860-13875.

[5] Zhou Zikang, Wang Jianping, Li Y H, et al. Query-centric trajectory prediction[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2023: 17863-17873.

[6] Huang Yizhou, Cheng Yihua, Wang Kezhi. Trajectory Mamba: Efficient attention-mamba forecasting model based on selective SSM[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2025: 12058-12067.

[7] Gilles T, Sabatini S, Tsishkou D, et al. THOMAS: Trajectory heatmap output with learned multi-agent sampling[C/OL]// Proceedings of the 10th International Conference on Learning Representations, 2022: 1-18. https://openreview.net/forum id=QDdJhACYrlX.

[8] Gu A, Dao T. Mamba: Linear-time sequence modeling with selective state spaces[PP/OL]. V1. arXiv (2023-12-01)[2025-11-01]. https://arxiv.org/abs/2312.00752.

[9] Dao T, Gu A. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality[C]//Proceedings of the 41st International Conference on Machine Learning. Vienna: PMLR, 2024: 10041-10071.

[10] Li Wenbing, Zhou Hang, Yu Junqing, et al. Coupled Mamba: Enhanced multimodal fusion with coupled state space model[PP/OL]. V1. arXiv (2024-05-29) [2025-11-01]. https://arxiv.org/abs/2405.18014.

[11] Waleffe R, Byeon W, Riach D, et al. An empirical study of Mamba-based language models[EB/OL]. (2024-06-12)[2025-11-01]. https://arxiv.org/abs/2406.07887.

[12] Liang Ming, Yang Bin, Hu Rui, et al. Learning lane graph representations for motion forecasting[C]//Proceedings of the 16th European Conference on Computer Vision. Berlin: Springer, 2020: 541-556.

[13] Cheng Jie, Mei Xiaodong, Liu Ming. Forecast-MAE: Self-supervised pre-training for motion forecasting with masked autoencoders[C]//IEEE/CVF International Conference on Computer Vision. Piscataway: IEEE, 2023: 8645-8655.

[14] Zhou Zikang, Ye Luyao, Wang Jianping, et al. HiVT: Hierarchical vector transformer for multi-agent motion prediction[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2022: 8813-8823.

[15] Zhang Zhejun, Liniger A, Sakaridis C, et al. Real-time motion prediction via heterogeneous polyline transformer with relative pose encoding[C]//Proceedings of the 37th International Conference on Neural Information Processing Systems. New York: Curran Associates Inc., 2023: 2507.

[16] Shi Shaoshuai, Jiang Li, Dai Dengxin, et al. Motion trans-

former with global intention localization and local movement refinement[C]//Proceedings of the 36th International Conference on Neural Information Processing Systems. New York: Curran Associates Inc., 2022: 473.

[17] Choi J W, Curry R, Elkaim G. Path planning based on Bézier curve for autonomous ground vehicles[C]//Advances in Electrical and Electronics Engineering - IAENG Special Edition of the World Congress on Engineering and Computer Science. Piscataway: IEEE, 2008: 158-166.

[18] Wilson B, Qi W, Agarwal T, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting[C]//Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks. New York: Curran Associates Inc., 2021.

[19] Chang Mingfang, Lambert J, Sangkloy P, et al. Argoverse: 3D tracking and forecasting with rich maps[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2019: 8740-8749.

[20] Ngiam J, Caine B, Vasudevan V, et al. Scene transformer: A unified architecture for predicting future trajectories of multiple agents[C/OL]//Proceedings of the 10th International Conference on Learning Representations, 2022. https://openreview.net/pdf id=Wm3EA5OlHsG.

[21] Chen Weihuang, Wang Fangfang, Sun Hongbin. S2TNet: Spatio-temporal transformer networks for trajectory prediction in autonomous driving[C]//Proceedings of the 13th Asian Conference on Machine Learning. PMLR, 2021: 454-469.

[22] Liu Yicheng, Zhang Jinghuai, Fang Liangji, et al. Multimodal motion prediction with stacked transformers[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2021: 7573-7582.

[23] Cui Henggang, Radosavljevic V, Chou F C, et al. Multimodal trajectory predictions for autonomous driving using deep convolutional networks[C]//IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2019: 2090-2096.

[24] Varadarajan B, Hefny A, Srivastava A, et al. MultiPath++: Efficient information fusion and trajectory aggregation for behavior prediction[C]//IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2022: 7814-7821.

[25] Zhang Lu, Li Peiliang, Liu Sikang, et al. SIMPL: A simple and efficient multi-agent motion prediction baseline for autonomous driving[J]. IEEE Robotics and Automation Letters, 2024, 9(4): 3767-3774.

[26] Aydemir G, Akan A K, Güney F. ADAPT: Efficient multi-agent trajectory prediction with adaptation[C]//IEEE/CVF International Conference on Computer Vision. Piscataway: IEEE, 2023: 8261-8271.

[27] Prutsch A, Bischof H, Possegger H. Efficient motion prediction: A lightweight & accurate trajectory prediction model with fast training and inference speed[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway: IEEE, 2024: 9411-9417.

[28] Wagner R, Tas Ö S, Klemp M, et al. JointMotion: Joint self-supervision for joint motion prediction[C]//Proceedings of the 8th Conference on Robot Learning. PMLR, 2025: 3395-3406.

[29] Peng Ling, Liu Xiaohui, Lian Heng. Functional adaptive Huber linear regression[PP/OL]. V1. arXiv (2024-09-17)[2025-11-01]. https://arxiv.org/abs/2409.11053.

[30] 桑海峰, 陈旺兴, 王海峰, 等. 基于多模式时空交互的行人轨迹预测模型[J]. 电子学报, 2022, 50(11): 2806-2812. Sang Haifeng, Chen Wangxing, Wang Haifeng, et al. Pedestrian trajectory prediction model based on multi-model space-time interaction[J]. Acta Electronica Sinica, 2022, 50(11): 2806-2812. (in Chinese)

[31] Wang Zhepeng, Chen Feng, Lertniphonphan K, et al. Technical report for argoverse challenges on unified sensor-based detection, tracking, and forecasting[PP/OL]. V1.arXiv (2023-11-27)[2025-11-01]. https://arxiv.org/abs/2311.15615.

[32] Wang Mingkun, Zhu Xinge, Yu Changqian, et al. GANet: Goal area network for motion forecasting[C]//IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2023: 1609-1615.

[33] Cui A, Casas S, Wong K, et al. GoRela: Go relative for viewpoint-invariant motion forecasting[C]//IEEE International Conference on Robotics and Automation. Piscataway: IEEE, 2023: 7801-7807.

[34] Messaoud K, Cord M, Alahi A. Towards generalizable trajectory prediction using dual-level representation learning and adaptive prompting[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2025: 27564-27574.

[35] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Doha: Association for Computational Linguis-

tics, 2014: 1724-1734.

[36] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. New York: Curran Associates Inc., 2017: 6000-6010.

[37] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM networks[C]//IEEE International Joint Conference on Neural Networks. Piscataway: IEEE, 2005, 4: 2047-2052.

[38] 王中天, 吴一全. 基于视觉与深度学习的无人机自主着陆场景感知方法研究进展[J]. 电子学报, 2025, 53(11): 4171-4198.
Wang Zhongtian, Wu Yiquan. Research progress of UAV autonomous landing scene perception methods based on vision and deep learning[J]. Acta Electronica Sinica, 2025, 53(11): 4171-4198. (in Chinese)

作者简介

林清华 男,1998年2月出生于山东省潍坊市。现为青岛科技大学数据科学学院硕士研究生。主要研究方向为多目标跟踪、轨迹预测。
E-mail: f023111001@mails.qust.edu.cn

王美芳 女,2001年2月出生于山东省东营市。现为青岛科技大学数据科学学院硕士研究生。主要研究方向为多目标跟踪、轨迹预测。
E-mail: wangmeifang@mails.qust.edu.cn

李 辉 男,1984年3月出生于河南省平顶山市。现为青岛科技大学数据科学学院副教授、硕士生导师。主要研究方向为计算机视觉、3D目标检测及跟踪。
E-mail: lihui@qust.edu.cn

葛同澳 男,1999年6月出生于山东省菏泽市。2024年毕业于青岛科技大学数据科学学院。现为武汉理工大学博士研究生。主要研究方向为自动驾驶、目标检测。
E-mail: getongao@whut.edu.cn

颜 军 男,1962年12月出生于山东省高密市。现为青岛科技大学数据科学学院院长、俄罗斯工程院外籍院士、俄罗斯自然科学院外籍院士。主要研究方向为自动驾驶、遥感图像处理。
E-mail: yanjun@qust.edu.cn