智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

基于多查询token选择机制的Transformer行为识别模型

刘歆, 曾奎, 陈奉

引用本文:
刘歆, 曾奎, 陈奉. 基于多查询token选择机制的Transformer行为识别模型[J]. 智能系统学报, 2026, 21(2): 410-422.
LIU Xin, ZENG Kui, CHEN Feng. Transformer action recognition model based on multi-query token selection mechanism[J]. *CAAI Transactions on Intelligent Systems*, 2026, 21(2): 410-422.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202503002

DOI: 10.11992/tis.202503002
网络出版地址: https://link.cnki.net/urlid/23.1538.tp.20260131.1316.002

基于多查询 token 选择机制的 Transformer 行为识别模型

刘歆, 曾奎, 陈奉
(重庆邮电大学 计算机科学与技术学院, 重庆 400065)

摘　要: 针对视频行为识别中 ViTs(vision Transformers)模型的空间注意力无法聚焦浅层局部特征、时间注意力无法准确捕捉动态特征等问题, 提出了一种基于多查询 token 选择机制的 Transformer 行为识别模型。该模型构建了由多个时空特征注意力模块组成的局部特征聚合模块, 每个时空特征注意力模块通过 3D 卷积结合通道和空间注意力聚焦浅层局部特征。构建了由多个时空处理单元组成的全局时空特征提取模块, 每个时空处理单元包括: 混合空间感知模块、多查询 token 选择的时间注意力模块和时空特征融合模块。混合空间感知模块在全局空间注意力机制之前引入 3D 深度可分离卷积, 增强对局部邻域的时空特征关注; 多查询 token 选择的时间注意力模块通过多查询 token 选择机制对每一帧的特征筛选, 完成背景的弱化、人体动作的强化; 时空特征融合模块通过顺序融合的方式实现空间与时间特征的高效融合。在不同数据集上的实验结果表明, 该方法的识别效果优于基线模型。

关键词: 行为识别; 注意力机制; 特征融合; 时间注意力; 空间注意力; 动态特征; 局部特征; 时空特征提取模块
中图分类号: TP391.41　　**文献标志码**: A　　**文章编号**: 1673−4785(2026)02−0410−13

中文引用格式: 刘歆, 曾奎, 陈奉. 基于多查询 token 选择机制的 Transformer 行为识别模型 [J]. 智能系统学报, 2026, 21(2): 410–422.
英文引用格式: LIU Xin, ZENG Kui, CHEN Feng. Transformer action recognition model based on multi-query token selection mechanism[J]. CAAI transactions on intelligent systems, 2026, 21(2): 410–422.

Transformer action recognition model based on multi-query token selection mechanism

LIU Xin, ZENG Kui, CHEN Feng
(School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: To address the limitations of ViTs (vision Transformers) in video action recognition, specifically, the inability of spatial attention to focus on shallow local features and the inaccuracy of temporal attention in capturing dynamic information, we propose a Transformer-based action recognition model incorporating a multi-query token selection mechanism. The model introduces a Local Feature Aggregation Module composed of multiple Spatiotemporal Attention Blocks, where each block employs 3D convolutions combined with channel and spatial attention to enhance the focus on shallow local features. Furthermore, a Global Spatiotemporal Feature Perception Module is constructed, consisting of several Spatiotemporal Processing Units. Each unit comprises: (1) a Hybrid Spatial Perception Module, which incorporates 3D depthwise separable convolutions before the global spatial attention mechanism to strengthen attention to local spatiotemporal neighborhoods; (2) a Temporal Attention Module with Multi-Query Token Selection, which filters features of each frame to suppress background noise and emphasize human actions; (3) a Spatiotemporal Feature Fusion Module, which efficiently integrates spatial and temporal features through sequential fusion. Experimental results on different datasets demonstrate that the proposed method outperforms baseline models.

Keywords: action recognition; attention mechanism; feature fusion; temporal attention; spatial attention; dynamic features; local feature; spatio-temporal feature extraction module

收稿日期: 2025−03−03.　**网络出版日期**: 2026−02−02.
基金项目: 重庆市留学人员回国创业创新支持计划项目(CX2024086); 成都市重点研发支撑计划区域科技创新合作项目(2023-YF11-00015-HZ).
通信作者: 陈奉.E-mail: chenfeng@cqupt.edu.cn.

Token 选择机制对每一帧的特征筛选, 完成背景的弱化、人体动作的强化; 时空特征融合模块通过顺序融合的方式实现空间与时间特征的高

效融合。在不同数据集上的实验结果表明, 该方法的识别效果优于基线模型。

表示学习作为计算机视觉领域的核心研究课题[1-2], 在行为识别任务中发挥着关键作用。视频行为识别旨在从视频数据中自动识别和分类特定的行为或动作, 这一任务面临着以下挑战: 在视频的相邻时间帧或空间区域内, 视觉信息高度相似或重复而出现的局部冗余。这种冗余不仅增加了模型的计算负担, 还可能导致模型在相似区域进行大量重复计算, 从而降低整体计算效率; 视频中不同时间帧和空间区域之间存在复杂的关联。这种关联不仅涉及物体的运动, 还包括场景变化和事件发展的多方面因素。这些因素之间具有动态的远程交互作用, 形成复杂的全局依赖关系, 增加了捕捉远程依赖的难度。

为了解决以上挑战, 研究人员在视觉识别领域提出许多模型[2-4]。其主流模型包括基于卷积神经网络(convolutional neural network, CNN)[5-6]的模型和基于视觉 Transformer(vision Transformer, ViT)[1]的模型。与 ViT 的自注意力机制相比, 基于卷积神经网络的模型通过局部特征提取策略和权重共享机制, 减少了参数量, 降低了计算复杂度[5, 7], 但其有限的感受野在全局依赖学习上存在局限[8]。基于 ViT 的模型在视频行为识别任务中表现出色[9-12], 但 ViT 的自注意力机制需要对输入数据中的各个位置进行两两关联计算, 在高冗余的视频数据场景下, 这种计算方式导致大量的计算资源被消耗, 计算效率低下。ViT 在浅层提取特征时, 只有局部邻域的特征有实质的贡献[12], 对全局的空间注意力计算将导致大量计算资源被浪费, 降低了计算效率。TimeSformer(time space Transformer)[10]提出的时间注意力通过处理所有帧中相同位置的特征来捕捉不同帧之间的时序依赖, 降低了计算量。但由于目标人物的运动变化, 时间注意力机制无法有效捕捉运动特征的空间位置变化。

针对 ViT 的空间注意力机制未能有效聚焦浅层局部特征, 导致的全局的自注意力计算资源的浪费和时间注意力机制未能准确捕捉动态特征等问题, 本文提出了一种基于多查询 token 选择机制的 Transformer 行为识别模型, 命名为 MQTSformer。本文的主要贡献如下:

1) 构建了局部特征聚合模块。该模块通过 3D 卷积结合通道和空间注意力, 让空间注意力聚焦于浅层局部特征, 减少模型浅层特征提取中不必要的计算量, 同时保留了有效的空间信息。

2) 提出了多查询 token 选择的时间注意力模块。该模块利用多查询 token 选择机制, 计算当前时序 token 与历史上下文 tokens 之间的注意力分布, 并根据注意力分数筛选出关键 tokens 和冗余 tokens。通过保留关键 tokens 及其时序依赖关系, 去除冗余 tokens 所对应的背景信息, 模块能够有效减少计算复杂度, 专注于提取重要的时序特征, 聚焦关键动作信息的聚合, 避免帧间冗余特征的干扰。

3) 提出了混合空间注意力机制。该机制改进了 ViT 特征提取模块中的空间注意力机制, 保留了全局空间特征的提取方式, 增强了局部邻域的时空特征的关注。

1　相关工作

1.1　基于卷积的行为识别方法

双流网络的行为识别方法　2D 卷积仅能处理单帧图像的空间特征, 难以捕捉视频中帧与帧之间的时序依赖关系。针对这一问题, Simonyan 等[13]提出通过光流[14]来表示运动信息, 并设计了双流网络模型, 融合 RGB(red, green, blue)图像捕捉的帧内静态外观特征和光流捕获的帧间运动特征。Feichtenhofer 等[15]发现早期融合能更好地结合两个流的信息, 学习更丰富的特征, 带来比后期融合更优的性能。基于这一发现, 他们提出了 ConvNets 行为识别模型[16], 为时空特征的深度融合提供了新的思路。

基于分段的行为识别方法　光流[14]在捕捉长期时间依赖关系方面存在明显不足。Wang 等[17]提出 TSN(temporal segment networks)网络, 将视频在每个时间段中随机采样一帧, 利用一致性机制聚合帧间信息, 实现对全局时序特征的捕捉。Zhou 等[18]提出 TRN(temporal relational networks)模型, 同时捕捉短时间局部动态特征和长时间全局动作特征。Lin 等[19]提出的 TSM(temporal shift module)网络通过在时间维度上移动通道特征, 实现帧间信息交互, 有效地进行时序建模。

3D 卷积神经网络的行为识别方法　3D CNN 能够同时捕捉空间和时间信息, 提取视频中的时序特征。Ji 等[20]首次将 3D 卷积用于行为识别任务。TranD 等[21]提出 C3D(3D convolutional neural networks)卷积网络并进行了改进。但是 3D 卷积存在优化难度大和计算成本高的问题。为了解决这一问题, Carreira 等[22]提出 I3D(inflated 3D ConvNet)方法, 通过将图像分类模型的 2D 卷积权重扩展至 3D 卷积, 解决了 3D 卷积从零开始训练的

难题。为进一步降低 3D 网络的训练复杂性，Tran 等[23] 提出的 R2+1D 和 Qiu 等[24]提出的 P3D(pseudo 3D networks)通过将 3D 卷积因式分解为 2D 卷积和 1D 卷积。为了进一步提高 3D CNN 的效率，Feichtenhofer 等提出的 SlowFast (slowFast networks)[5] 和 X3D(expandable 3D networks)[25] 网络尝试在不同维度上对三维卷积核进行因式分解，以减少复杂性。但由于卷积神经网络感受野[26] 有限，三维卷积在捕捉长距离依赖关系方面存在困难。

1.2　基于视觉 Transformer 的行为识别方法

在自然语言处理任务中，Transformer[27] 架构通过自注意力机制进行相似性计算，能够捕捉长距离依赖关系[28]。Bertasius 等[10] 提出的 TimeSfomer 模型和 Arnab 等[9] 提出的 VIVIT(a video vision Transformer)模型首次将 Transformer 的全局建模能力应用于行为识别领域的工作。TimeSformer [10] 提出分离的时空注意力，降低了计算量。但本文发现在该模型的 Layer1 至 Layer3 中，空间注意力机制未能充分聚焦局部特征，如图 1所示。而且时间注意力机制无法有效捕捉由于目标人物运动变化而导致的 token 空间位置变化，难以准确捕获动态特征。Patrick 等[29] 提出一种轨迹注意力机制的网络，提高了对动态场景的理解能力。

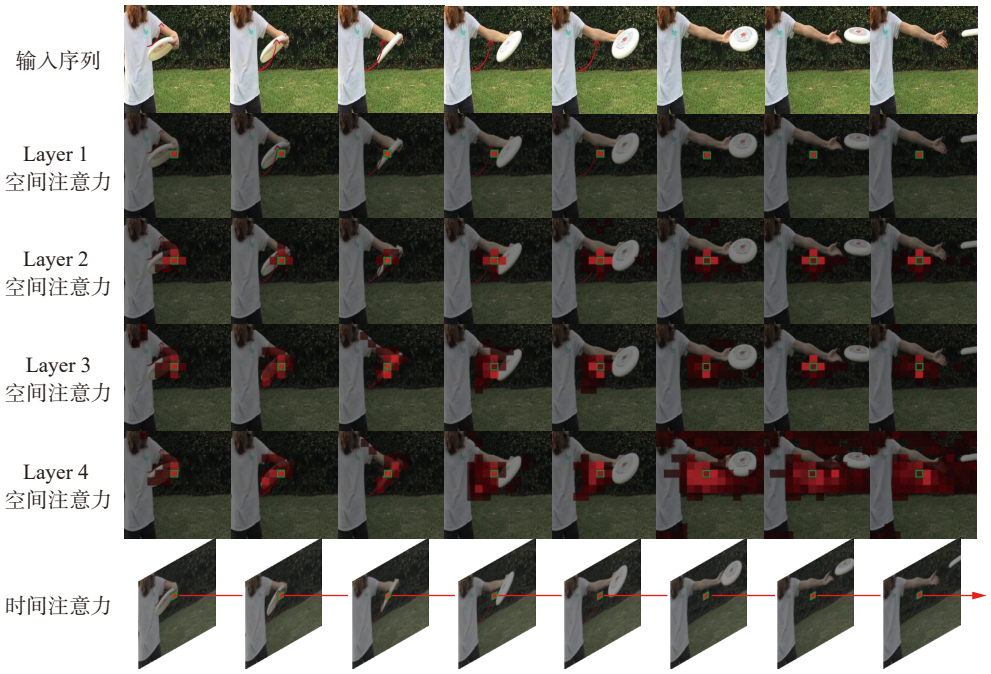

图 1　TimeSformer 的时间空间注意力可视化

Fig. 1　Temporal spatial attention visualization with TimeSformer

1.3　基于 CNN 和 ViT 的行为识别方法

为了充分发挥 CNN 和 ViT 各自的优势，研究者们探索将两者相结合的混合模型研究。Fan 等[3] 提出 MVIT(multiscale vision Transformers)，将卷积的局部特征提取能力与 Transformer 的全局建模能力结合起来。但是 MVIT 的局部窗口注意力无法有效处理图像中大范围的特征。改进版本 MVITv2(improved multiscale vision Transformers)[30] 通过引入改进的池化注意力机制处理高分辨率的视觉输入，并通过相对位置编码的方式捕捉空间特征。Li 等[12] 发现 ViT 浅层特征提取十分低效，提出了 UniFormer(unified Transformer)，并设计了浅层和深层的关系聚合器，以减少冗余并提高全局依赖性捕捉的效率。但 UniFormer 在进行视频行为识别之前需要复杂的图像预训练流程，这限制了其广泛应用。Li 等[31] 提出了 UniFormerV2，通过结合预训练的 ViT 与 UniFormer，构建强大的视频网络模型。Lou 等[32] 提出 TransXNet 通过输入依赖的方式聚合稀疏的全局信息和局部细节。

2　本文方法

为解决空间注意力机制未能有效聚焦浅层局部特征、时间注意力机制未能准确捕捉动态特征等问题，本文提出了一种基于多查询 token 选择机制的 Transformer 行为识别模型，命名为 MQTSformer，整体架构如图 2 所示。本文方法的改进工作如下：为了有效聚合特征并减少相邻帧之间的时空冗余，构建了由多个时空特征注意力模块(spatiotemporal feature attention module, SFAM)组

成的局部特征聚合模块（local feature aggregation module, LFAM），模块通过 3D 卷积结合通道和空间注意力，空间注意力能对浅层局部特征的聚焦。为了高效提取时空特征并增强模型的特征提取能力，本文设计了全局时空特征提取模块（global spatiotemporal feature extraction module, GSFEM）。该模块由 12 个相同的时空处理单元（spatio-temporal processing unit, STPU）组成。每个时空处理单元 STPU 由 3 个模块组成：混合空间感知模块（hybrid spatial perception module, HSPM），该模块通过在全局空间注意力机制之前引入 3D 深度可分离卷积 (depthwise separable Conv3D, DWConv3D)[33]，增强局部邻域特征的关注；多查询 token 选择的时间注意力模块（temporal attention module for multi-query token selection, TAMMQTS），该模块通过关注每一帧关键的 tokens，模型能够关注重要的时间动态信息，忽略背景等冗余信息；时空特征融合模块（spatial-temporal feature fusion module, STFFM），用于实现空间与时间特征的高效融合。

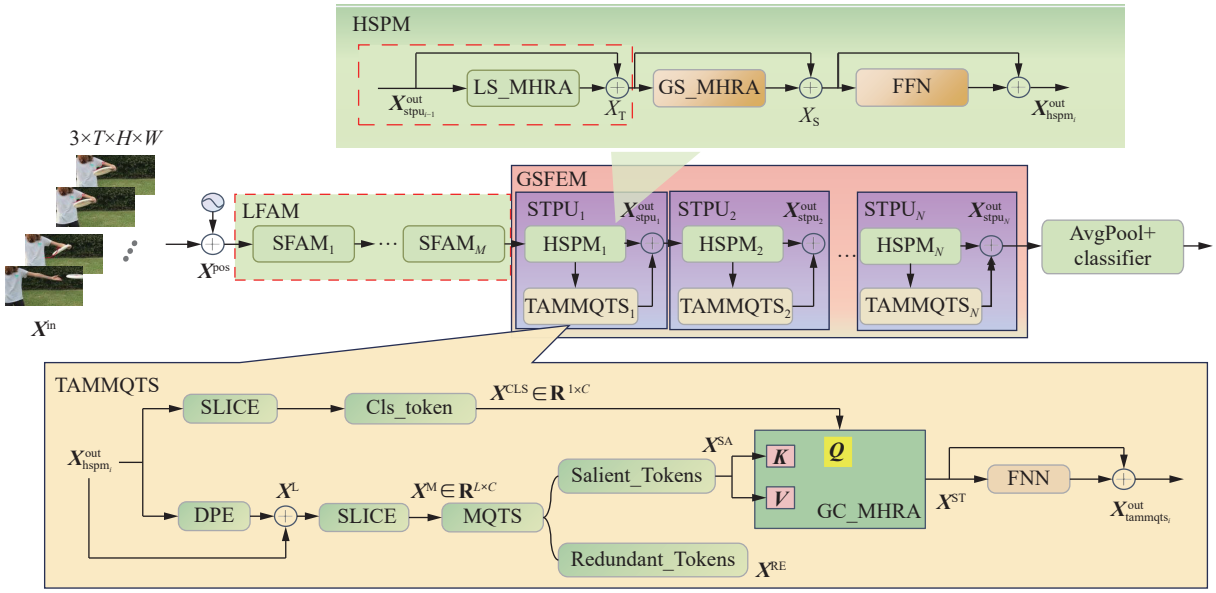

图 2　MQTSformer 架构

Fig. 2　Framework of MQTSformer

2.1　局部特征聚合模块

局部特征聚合模块 LFAM 旨在解决 ViT 空间注意力机制在特征提取前期未能有效聚焦局部特征，全局自注意力计算资源浪费的问题。该模块由多个相同的时空特征注意力模块 SFAM 组成。每个 SFAM 的设计理念遵循 Transformer 中前馈神经网络（feed forward network, FFN）的设计思路。

设输入视频数据为 $3\times T\times H\times W$，$T$ 表示视频的采样频率，H 和 W 表示每一帧的高度和宽度。通过三维卷积将输入视频映射为 L 个 token，它们被表示为 $\boldsymbol{X}^{\text{in}}\in\mathbf{R}^{L\times C}$，其中 $L=\dfrac{T}{D}\times\dfrac{H}{K_{\text{h}}}\times\dfrac{W}{K_{\text{w}}}$，表示为映射后的 token 数量，$C$ 表示 token 的维度，D 是卷积核在时间方向上的大小，K_{h} 和 K_{w} 分别表示卷积核在空间上的高度和宽度。对 $\boldsymbol{X}^{\text{in}}$ 中的 L 个 token 使用可学习的位置编码，并在生成一个分类 token[1]，得到 $\boldsymbol{X}^{\text{in}}$ 编码后的表示 $\boldsymbol{X}^{\text{pos}}\in\mathbf{R}^{K\times C}$，其中 $K=L+1$，表示在加入分类 token 后的 token 总数。

将 $\boldsymbol{X}^{\text{pos}}$ 输入局部特征聚合模块 LFAM 的时空特征注意力模块 SFAM 结构中。如图 3 所示，SFAM 包含以下关键结构：3D 卷积、3D 深度可分离卷积 DWConv3D、通道注意力模块（channel attention module, CAM）[34]、空间注意力模块（spatial attention module, SAM）[35]。SFAM 的计算过程如下。

1) 通过批归一化[27] 对输入信息进行归一化，稳定特征分布，避免训练过程中出现梯度爆炸或梯度消失。采用 3D 卷积对归一化后的特征进行通道维度的压缩，以降低计算复杂度，得到 $\boldsymbol{X}^{\text{L}}_{\text{SFAM}_i}\in\mathbf{R}^{K\times C_{\text{d}}}$，其中，$C_{\text{d}}$ 是卷积后的通道数，$C_{\text{d}}=C/\text{dw\_reduction}$，dw_reduction 是深度可分离卷积中的缩减因子。

$$\boldsymbol{X}^{\text{L}}_{\text{SFAM}_i}=\text{Conv3D}\left(\text{bn}\left(\boldsymbol{X}^{\text{OUT}}_{\text{SFAM}_{i-1}}\right)\right)$$

式中 bn(·) 表示批归一化。当 $i=1$ 时，输入的数据为 $\boldsymbol{X}^{\text{pos}}$，其他层的输入为上一层 SFAM 的输出 $\boldsymbol{X}^{\text{OUT}}_{\text{SFAM}_{i-1}}$。$i$ 的范围为 1~M，M 是 LFAM 中 SFAM 的层数。

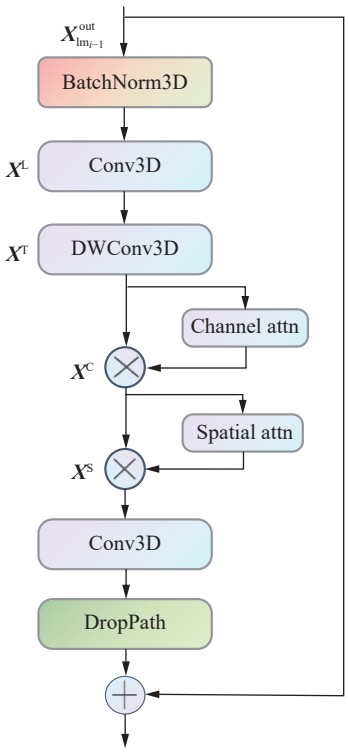

图 3　时空特征注意力模块

Fig. 3　Spatiotemporal feature attention module

2) 采用 3D 深度可分离卷积 DWConv3D[33] 进一步对 $\boldsymbol{X}^{\mathrm{L}}_{\mathrm{SFAM}_i}$ 进行特征提取，得到输出特征 $\boldsymbol{X}^{\mathrm{T}}_{\mathrm{SFAM}_i}\in\mathbf{R}^{K\times C_{\mathrm{d}}}$。该卷积操作通过将标准卷积分解为逐通道的空间卷积和逐点的 1×1×1 卷积，在显著降低模型参数量和计算复杂度的同时，有效保留了特征的关键信息。

$$\boldsymbol{X}^{\mathrm{T}}_{\mathrm{SFAM}_i}=\mathrm{DWConv3D}(\boldsymbol{X}^{\mathrm{L}}_{\mathrm{SFAM}_i})$$

3) 引入通道注意力模块 CAM[34] 对 $\boldsymbol{X}^{\mathrm{T}}_{\mathrm{SFAM}_i}$ 进行处理，生成通道注意力图，并与 $\boldsymbol{X}^{\mathrm{T}}_{\mathrm{SFAM}_i}$ 进行逐通道点乘，得到 $\boldsymbol{X}^{\mathrm{C}}_{\mathrm{SFAM}_i}\in\mathbf{R}^{K\times C_{\mathrm{d}}}$：

$$\boldsymbol{X}^{\mathrm{C}}_{\mathrm{SPAM}_i}=\mathrm{Ch\_Attn}(\boldsymbol{X}^{\mathrm{T}}_{\mathrm{SPAM}_i})\odot_{\mathrm{C}}\boldsymbol{X}^{\mathrm{T}}_{\mathrm{SPAM}_i}$$

式中：$\odot_{\mathrm{C}}$代表逐通道相乘，Ch_Attn代表通过注意力模块。CAM 能够自适应调整通道权重，使模型更好地聚焦关键通道信息。

4）引入空间注意力模块 SAM[35] 对 $\boldsymbol{X}^{\mathrm{C}}_{\mathrm{SFAM}_i}$ 进行处理，生成空间注意力权重图，并与 $\boldsymbol{X}^{\mathrm{C}}_{\mathrm{SFAM}_i}$ 进行逐元素点乘，得到 $\boldsymbol{X}^{\mathrm{S}}_{\mathrm{SFAM}_i}$：

$$\boldsymbol{X}^{\mathrm{S}}_{\mathrm{SFAM}_i}=\mathrm{Sp\_Attn}(\boldsymbol{X}^{\mathrm{S}}_{\mathrm{SFAM}_i})\odot_{\mathrm{E}}\boldsymbol{X}^{\mathrm{S}}_{\mathrm{SFAM}_i}$$

式中：$\odot_{\mathrm{E}}$代表逐元素相乘，Sp_Attn(·)代表空间注意力。空间注意力模块 SAM 能够帮助模型有效聚焦帧内的关键区域信息。

5）将经过 SAM 处理后获得的特征 $\boldsymbol{X}^{\mathrm{S}}_{\mathrm{SFAM}_i}$ 通过三维卷积提升通道维度，并与输入特征 $\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{SFAM}_{i-1}}$ 进行残差连接，生成最终的输出 $\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{SFAM}_i}\in\mathbf{R}^{K\times C}$：

$$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{SFAM}_i}=\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{SFAM}_{i-1}}+\mathrm{Conv3D}\left(\boldsymbol{X}^{\mathrm{S}}_{\mathrm{SFAM}_{i-1}}\right)$$

SFAM_i表示局部特征聚合模块 LFAM 中的第i层 SFAM。

SFAM 按照特征降维、特征提取、特征升维与融合的设计思路构建。采用 3D 卷积压缩通道维度，通过深度可分离卷积对空间局部特征提取。考虑到视频的时间和空间的关系，引入了通道和空间注意力，排列顺序参考了 Woo 等[35] 的工作。最后通过 3D 卷积恢复通道维度并与输入数据残差连接防止过拟合。

2.2　全局时空特征提取模块

为了让时间注意力机制能捕捉动态特征，本文构建了全局时空特征提取模块 GSFEM，并在该特征提取过程中平衡计算复杂度与精度。该模块设计参考了 ViT[1] 的层次化架构思想，由 12 层相同的时空处理单元 STPU 堆叠而成。每个 STPU 由 3 个模块组成：混合空间感知模块 HSPM、多查询 token 选择的时间注意力模块 TAMMQTS 和时空特征融合模块 STFFM。HSPM 对 ViT 的空间注意力机制进行改进，增强局部邻域的时空特征关注，保留全局空间特征的提取方式。TAMMQTS 是本文设计的一种时间注意力机制，采用多查询策略以实现时间维度上关键动态信息的选择性捕捉。STFFM 将 HSPM 与 TAMMQTS 的输出特征进行融合，确保不同维度信息的有效整合，并优化特征传递效率。将 LFAM 输出的 $\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{SFAM}_M}$ 输入到时空处理单元 STPU 中，第 i 层 STPU 经计算得到输出 $\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_i}$。

2.2.1　混合空间感知模块

通过观察图 1 中 TimeSformer 的特征提取过程，相邻的 token 始终表现出较高的相关性。为了增强局部邻域内对 token 的关注，构建混合空间感知模块 HSPM。在该模块中，将改进的局部时空关联多头关系聚合器（local spatiotemporal multi-head relation aggregator, LS_MHRA）加入全局空间关联多头关系聚合器[31] 之前，用于捕捉局部时空域内的高相关性 token，强化相邻块之间的特征表达。HSPM 模块的计算过程如下。

1) 通过批归一化[27] 对输入数据 $\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}$ 进行归一化处理后，将采用 LS_MHRA 增强后的特征与输入特征进行残差连接，得到局部增强特征 $\boldsymbol{X}^{\mathrm{T}}_{\mathrm{BSPM}_i}\in\mathbf{R}^{K\times C}$。

LS_MHRA 基于局部时间关联的多头关系聚合器（local temporal multi-head relation aggregator, LT_MHRA）[31] 改进，它通过定义亲和力矩阵 $\boldsymbol{a}^{\mathrm{LT}}_{n,k}$来描述 tokens 之间的局部时空关联性。为了增强局部邻域内 token 的关注度，LS_MHRA 的亲

和力矩阵被限定在局部范围, 包含可学习的参数矩阵 $\boldsymbol{a}_{\mathrm{ls}} \in \mathbf{R}^{t\times3\times3}$。

$$\boldsymbol{a}^{\mathrm{LT}}_{n,k}(\boldsymbol{x}_k,\boldsymbol{x}_j)=\boldsymbol{a}^{k-j}_{\mathrm{ls}},\ j\in\Omega^{t\times3\times3}_k$$

式中: $\boldsymbol{a}^{\mathrm{LT}}_{n,k}$代表第$n$个头中第$k$个 token 的亲和力矩阵; $n=1,2,\cdots,N$, N代表 LS_MHRA 的数量; $k=1,2,\cdots,K$, K代表 token 的总数; $\boldsymbol{x}_k$表示输入数据 $\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}$ 中的第 k 个 token; $\boldsymbol{x}_j$表示$\boldsymbol{x}_k$在$\boldsymbol{a}_{\mathrm{ls}}$范围中的某一个 token。通过对该范围内所有$\boldsymbol{x}_j$进行计算比较, 可使得$\boldsymbol{x}_k$学习到在亲和力限定范围内与所有$\boldsymbol{x}_j$之间的局部时空关系。输入数据$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}$中的每个 token 都有各自对应亲和力矩阵。全局亲和力矩阵$\boldsymbol{A}^{\mathrm{LT}}_n$由$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}$中所有 token 的亲和力矩阵拼接而成, 其计算公式为

$$\boldsymbol{A}^{\mathrm{LT}}_n=\mathrm{Concat}(\boldsymbol{a}^{\mathrm{LT}}_{n,1},\boldsymbol{a}^{\mathrm{LT}}_{n,2},\cdots,\boldsymbol{a}^{\mathrm{LT}}_{n,K})$$

式中$\boldsymbol{A}^{\mathrm{LT}}_n$表示 LS_MHRA 中第$n$个头的全局亲和力矩阵。为了进一步建模 token 之间的关系, 第n个头的关系聚合器$R_n(\cdot)$定义为

$$R_n\left(\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}\right)=\boldsymbol{A}^{\mathrm{LT}}_n\boldsymbol{V}_n\left(\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}\right)$$

式中$V_n(\cdot)$表示对该函数输入信息的线性投影。

LS_MHRA 拼接 N 个头的关系聚合器结果, 并进行线性变换, 其计算公式为

$$\boldsymbol{X}^{\mathrm{TMP}}_{\mathrm{HSPM}_i}=\mathrm{CT}(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_N)$$

$$\boldsymbol{X}^{\mathrm{T}}_{\mathrm{HSPM}_i}=\boldsymbol{X}^{\mathrm{TMP}}_{\mathrm{HSPM}_i}\boldsymbol{U}+\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}$$

式中: $\boldsymbol{U}\in\mathbf{R}^{C\times C}$是可学习的融合矩阵, $\mathrm{CT}(\cdot)$代表拼接操作, $\boldsymbol{K}_i$代表$\boldsymbol{R}_i\left(\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}\right)$。

2) 在局部特征增强之后, 遵循 ViT 的设计范式, 采用层归一化对输入数据特征$\boldsymbol{X}^{\mathrm{T}}_{\mathrm{HSPM}_i}$进行处理。引入全局空间关联的多头关系聚合器 GS_MHRA(global cross MHRA)[31], 该聚合器通过自注意力机制捕捉帧内全局空间依赖关系, 得到全局空间关联后的特征表示$\boldsymbol{X}^{\mathrm{S}}_{\mathrm{HSPM}_i}\in\mathbf{R}^{K\times C}$。该过程能够增强单帧图像的全局空间感知能力, 使得模型能够更有效地建模远程空间依赖关系。

$$\boldsymbol{X}^{\mathrm{S}}_{\mathrm{HSPM}_i}=\mathrm{GS\_MHRA}\left(\boldsymbol{X}^{\mathrm{T}}_{\mathrm{HSPM}_i}\right)$$

3) 通过 FFN 对$\boldsymbol{X}^{\mathrm{S}}_{\mathrm{HSPM}_i}$进行处理, 得到输出特征$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{HSPM}_i}\in\mathbf{R}^{K\times C}$。该过程保留了局部细节与全局上下文信息, 进一步增强网络的特征表达能力。

$$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{HSPM}_i}=\mathrm{FFN}(\boldsymbol{X}^{\mathrm{S}}_{\mathrm{HSPM}_i})+\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{STPU}_{i-1}}$$

式中: HSPM_i表示 GSFEM 中的STPU_i的 HSPM; STPU_i表示 GSFEM 中第i个 STPU。

在特征提取过程中, 先进行全局空间特征提取, 再增强局部邻域特征, 这种方式将会导致全局特征的过度平滑, 削弱局部区域的显著性, 导致关键细节信息的丢失。在本文的 HSPM 特征提取过程中, 采用“先关注局部邻域特征, 再进行全局空间特征处理”的策略, 以更好地保留细节信息。

2.2.2　多查询 token 选择的时间注意力模块

多查询 token 选择的时间注意力模块 TAMMQTS 是本文设计的一种时间注意力机制, 采用多查询策略以实现时间维度上关键动态信息的选择性捕捉。如图 4 所示, TAMMQTS 包含以下关键部分: 动态位置编码(dynamic positional encoding, DPE)[36], 多查询 token 选择机制 (multi query token selection, MQTS), 交叉注意力机制 (global cross mHRA, GC_MHRA)[31], 前馈神经网络 FFN。

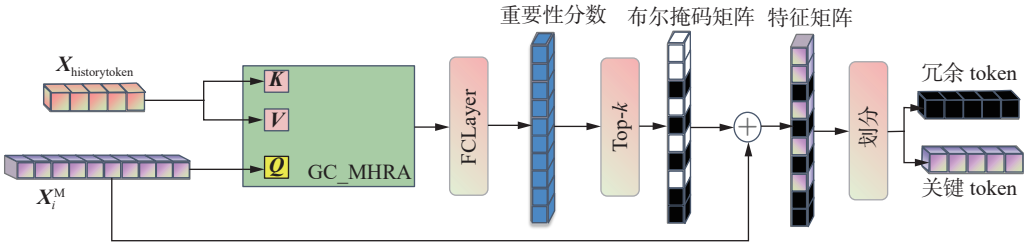

图 4　多查询 token 选择机制

Fig. 4　Multi query token selection

TAMMQTS 设计两条处理路径: 一条路径是分类 token 提取路径, 输入数据$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{HSPM}_i}$由分类 token 和时空特征 tokens 组成。分类 token[1] 是一个特殊的 token, 通常被放置在序列的最前面。通过对输入数据切分, 将分类 token 切分出来, 记作$\boldsymbol{X}^{\mathrm{CLS}}_i\in\mathbf{R}^{l\times C}$。另一条路径是时空特征 token 的筛选路径, 该路径通过动态位置编码 DPE[12] 对输入数据$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{HSPM}_i}$引入时序信息, 使用 MQTS 筛选关键 token, 并通过 GC_MHRA[31] 融合时空特征。时空特征 token 的筛选路径的计算过程如下。

1) 通过 DPE[12] 为输入数据$\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{HSPM}_i}$注入相对位置信息, 增强模型对时序特征的感知能力, 得到输出特征 $\boldsymbol{X}^{\mathrm{L}}_{\mathrm{TAMMQTS}_i}\in\mathbf{R}^{K\times C}$。该过程增强模型对时序特征的感知能力。

$$\boldsymbol{X}^{\mathrm{L}}_{\mathrm{TAMMQTS}_i}=\mathrm{DPE}\left(\boldsymbol{X}^{\mathrm{OUT}}_{\mathrm{HSPM}_i}\right)+\mathrm{X}^{\mathrm{OUT}}_{\mathrm{HSPM}_i}$$

2) 为了在数据$\boldsymbol{X}^{\mathrm{L}}_{\mathrm{TAMMQTS}_i}$中筛选出运动特征, 本文对输入数据采用分段切割的方法提取时空 token, 表示为$\boldsymbol{X}^{\mathrm{M}}_i\in\mathbf{R}^{L\times C}$。通过 MQTS 对$\boldsymbol{X}^{\mathrm{M}}_i$的 token

筛选，得到关键 token（key tokens）和冗余 token（redundant tokens）并分别表示为 $\boldsymbol{X}_i^{\mathrm{KEY}}$,$\mathrm{X}_i^{\mathrm{RE}}$。如图 4 所示，MQTS 包含以下关键部分：GC_MHRA[31]、全连接层（fully connected layer, FCLayer）[27]、Top-k 算法。在 MQTS 中，$\boldsymbol{X}_{\mathrm{historytoken}}$代表上下文信息，用于引导交叉注意力计算。它不是来自实际的输入数据，而是为注意力机制提供固定参考的向量，在实验中初始化为全零向量。

① 将 $\boldsymbol{X}_i^{\mathrm{M}}$和 $\boldsymbol{X}_{\mathrm{historytoken}}$进行交叉注意力计算得到增强特征 $\boldsymbol{X}_{\mathrm{MQTS}_i}^{\mathrm{ST}} \in \mathbf{R}^{L\times C}$。该过程将全局上下文信息融入输入数据 X_i^{M}，增强时序特征表示。

$$\boldsymbol{X}_{\mathrm{MQTS}_i}^{\mathrm{ST}} = \mathrm{GC\_MHRA}\left(\boldsymbol{X}_i^{\mathrm{M}}, \boldsymbol{X}_{\mathrm{historytoken}}\right)$$

② 通过全连接层对 $\boldsymbol{X}_{\mathrm{MQTS}_i}^{\mathrm{ST}}$进行线性变化和激活函数引入非线性变换后得到每个 token 的重要性分数。

$$X_{\mathrm{attnweight}_i} = \delta\left(\mathrm{FCLayer}\left(\boldsymbol{X}_{\mathrm{MQTS}_i}^{\mathrm{ST}}\right)\right)$$

式中：$X_{\mathrm{attnweight}_i}$表示每个 token 的重要性分数，$\delta$表示 Quick_GeLU 激活函数。

③ 通过 Top-k 筛选算法取出 $X_{\mathrm{attnweight}_i}$的前 k 个值，并记录这 k 个值对应的索引。根据这些索引构建了一个布尔掩码矩阵区分关键 token 和冗余 token。在实验中，通过调整 k 值，可以动态控制关键 token 与冗余 token 的比例，探究不同比例对模型性能的影响。

$$\boldsymbol{X}_{\mathrm{maskweight}_i} = \text{TOP-}k(\boldsymbol{X}_{\mathrm{attnweight}_i})$$

式中 $\boldsymbol{X}_{\mathrm{maskweight}_i}$表示通过 k 个索引构建的布尔掩码矩阵。

④ $\boldsymbol{X}_{\mathrm{maskweight}_i}$与 $\boldsymbol{X}_i^{\mathrm{M}}$逐元素相乘，得到特征矩阵 $\boldsymbol{X}_{\mathrm{featurematrix}_i}$。

$$\boldsymbol{X}_{\mathrm{featurematrix}_i} = \boldsymbol{X}_i^{\mathrm{M}} \odot \boldsymbol{X}_{\mathrm{maskweight}_i}$$

式中⊙表示逐元素相乘。

⑤ 通过布尔掩码矩阵将 $\boldsymbol{X}_{\mathrm{featurematrix}_i}$划分为两部分：关键 token 和冗余 token。

$$\boldsymbol{X}_i^{\mathrm{KEY}},\ \boldsymbol{X}_i^{\mathrm{RE}} = \mathrm{Divide}(\boldsymbol{X}_{\mathrm{featurematrix}_i})$$

分类 token 提取路径和时空特征 token 筛选路径处理结束之后。TAMMQTS 关注关键 tokens 的变化。通过 GC_MHRA[31] 对 $\boldsymbol{X}_i^{\mathrm{CLS}}$和 $\boldsymbol{X}_i^{\mathrm{KEY}}$融合，得到输出特征 $\boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{ST}} \in \mathbf{R}^{K\times C}$。该过程融合分类 token 和时空特征 token 的信息，捕获更丰富的语义和时空依赖关系，并通过关注关键 tokens 的变化，减少对冗余 tokens 的计算。

$$\boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{ST}} = \mathrm{GC\_MHRA}\left(\boldsymbol{X}_i^{\mathrm{CLS}},\ \boldsymbol{X}_i^{\mathrm{KEY}}\right)$$

通过 FFN 对 $\boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{ST}}$进行处理，得到 TAMMQTS 的输出特征 $\boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{OUT}} \in \mathbf{R}^{K\times C}$。

$$\boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{OUT}} = \mathrm{FFN}\left(\boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{ST}}\right) + \mathrm{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{ST}}$$

式中 $\mathrm{TAMMQTS}_i$表示 GSFEM 中的 STPU_i的 TAMMQTS。

TAMMQTS 整体按照筛选关键特征和冗余特征的步骤设计。这种设计基于视频数据的特点：关键信息通常在每一帧的局部区域内，背景信息通常在多帧之间重复且冗余。

2.2.3　时空特征融合模块

本文设计了一种顺序融合的时空特征融合模块 STFFM，用于对每个 STPU 的 HSPM 和 TAMMQTS 的输出特征进行融合，具体结构如图 2 所示。

$$\boldsymbol{X}_{\mathrm{STPU}_i}^{\mathrm{OUT}} = \boldsymbol{X}_{\mathrm{HSFM}_i}^{\mathrm{OUT}} + \boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{OUT}}$$

式中：$\boldsymbol{X}_{\mathrm{STPU}_i}^{\mathrm{OUT}}$表示第 STPU_i 的 HSPM 的输出特征，$\boldsymbol{X}_{\mathrm{TAMMQTS}_i}^{\mathrm{OUT}}$表示 STPU_i的 TAMMQTS 的输出特征，$\boldsymbol{X}_{\mathrm{STPU}_i}^{\mathrm{OUT}} \in \mathbf{R}^{K\times C}$表示 STPU_i的输出特征。

3　实验与分析

3.1　数据集介绍

为了验证 MQTSformer 用于视频行为识别的效果，本文在多个公开数据集上进行了实验，主要包括 Kinetics 数据集（Kinetics-400、Kinetics-600）和 Something-something 数据集（Something-somethingV1、V2）。

1) Kinetics-400[22]：Kinetics-400 是由 Google DeepMind 提供的广泛使用的行为识别数据集，包含约 260 232 个视频片段，涵盖 400 个动作类别，涉及日常活动、体育和娱乐等领域。每个视频约 10 s，每个类别有约 400 个样本。

2) Kinetics-600[37]：Kinetics-600 扩展了 Kinetics-400，涵盖 600 个动作类别，总计约 48 万个视频片段，每个视频约 10 s，每类约 600 个样本，涉及日常生活、体育和职业等领域。

3) Something-somethingV1[38]：Something-somethingV1 是一个用于视频理解和动作识别的标准数据集，包含 174 个动作类别，每个视频时长为 2~3 s，涵盖日常生活中的短暂行为。

4) Something-somethingV2[38]：Something-somethingV2 是一个大型视频动作识别数据集，包含约 22 万个视频片段，涵盖 170 种日常动作类别。

3.2　实验设计

实验 1　为了验证本文提出的 MQTSformer 中局部特征聚合模块 LFAM、混合空间感知模块 HSPM 和多查询 token 选择的时间注意力模块 TAMMQTS 的有效性，实验 1 在 Kinetics-400 数据集上进行了实验验证和对比分析。以 TimeSformer

作为基线模型, 表示为 TS。在 TimeSformer 中增加 LFAM 的方法表示为 TS+ LFAM。改进 ViT 空间注意力模块的方法为 TS+HSPM。增加了 TAMMQTS 的实验方法 TS+TAMMQTS。本实验中模型的训练总轮数为 55 轮, 每个视频的裁剪帧数为 8, Clip 数量为 3, Crop 数量为 4。基础学习率为 1×10^{-5}, 最小学习率为 1×10^{-6}, 并采用余弦退火学习率调整策略[39]。Kinetics 数据集在进行行为识别时, 会关注场景相关信息。如图 1 所示的 TimeSformer 可视化效果, 前 3 层的空间注意力只关注于局部范围, 从而导致计算效率低。4 层之后的特征提取贡献范围为全局, 将引入场景信息。为了在提高计算效率的同时保证识别准确率, 在 Kinetics 数据集的实验中, 消融实验设置 SFAM 的层数为 3。TAMMQTS 的多查询 token 选择机制 MQTS 中关键 tokens 与冗余 tokens 的选择比例为 1∶1。

实验 2　为了验证时空特征注意力模块 SFAM 的层数和多查询 token 选择机制 MQTS 中关键 token 和冗余 token 的比例对模型的影响。实验 2 在 Something-somethingV1 数据集上进行了实验验证和对比分析。SFAM 的层数设置为 2、3、4、5、6。MQTS 中关键 tokens 和冗余 tokens 的比例设置为 1∶4, 2∶3, 1∶1, 3∶2, 4∶1。本实验中模型的训练总轮数为 30 轮, 每个视频的裁剪帧数为 16, 基础学习率为 4×10^{-5}, 最小学习率为 1×10^{-6}, 并采用余弦退火学习率调整策略[39]。

实验 3　与其他基线模型的对比实验。在本实验中, 针对场景相关的数据集, 如 Kinetics-400 和 Kinetics-600, 每个视频提取的帧数为 8; 在时序关系较强的数据集, 如 Something-somethingV1 和 Something-somethingV2 中, 每个视频提取的帧数为 16。在 Kinetics-400 和 Kinetics-600 数据集的实验中, 学习率、训练轮次、多查询 token 选择机制 MQTS 中关键 tokens 和冗余 tokens 的比例和局部特征聚合模块 LFAM 中时空特征注意力模块 SFAM 的层数的设置与实验 1 保持一致; 在 Something-somethingV1 和 Something-somethingV2 数据集的实验中, 学习率、训练轮次和 LFAM 中 SFAM 的层数的设置则与实验 2 一致, MQTS 中关键 tokens 和冗余 tokens 的比例为 1∶1。

所有实验均在运行 Ubuntu 11.4.0 操作系统的计算平台上, 该平台配备了 8 张 NVIDIA GeForce RTX 4090D 显卡, 每张显卡具有 24 GB 显存。实验环境采用 PyTorch 1.11.0 深度学习框架, 并基于 Python 3.9.18 编程语言实现。视频数据处理过程中, 所有输入帧均统一裁剪为 224×224 的分辨率。模型训练采用交叉熵损失函数(cross-entropy loss), 并使用 AdamW 优化器进行参数更新, 其中动量参数设置为 0.9。

3.3　实验分析

3.3.1　实验 1 的结果及分析

如表 1 所示, TimeSformer 在 Top-1 和 Top-5 的评价指标上表现最差。在 TimeSformer 中加入本文改进和提出的模块后, 在准确率上呈现出不同程度的提升。

表 1　在 Kinetics-400 数据集上的消融实验结果

Table 1　Ablation experiment results on the Kinetics-400 dataset

方法	FLOPs/10^9	Top-1/%	Top-5/%
TS	140	78.8	93.2
TS+ LFAM	143	80.2	94.4
TS+HSPM	151	82.3	95.9
TS+ TAMMQTS	152	82.9	95.8
TS+TAMMQTS + LFAM	155	83.1	95.6
MQTSformer	**166**	**83.7**	**96.1**

注: 加粗表示最优结果。

3.3.2　实验 2 的结果及分析

表 2 给出了在 LFAM 中设置不同的 SFAM 层数的实验结果。其中, M 代表 SFAM 的层数, $M=2,3,\cdots,6$。随着 SFAM 层数从 2 增加到 6, 模型在 Top-1 准确率指标上呈现出明显的上升趋势, 这表明更深层的 SFAM 能够建立更丰富的时空特征交互, 增强模型的特征表达能力。Top-5 准确率在 SFAM 层数为 4 时达到最高, 随后出现下降。这表明过深的网络结构可能导致特征过度细化, 影响多候选预测的泛化能力。如表 2 的 FLOPs 结果所示, SFAM 层数的增加与计算量呈现线性关系。而就 Something-something 数据集而言, 它具有强时序相关的特性。相对于 Kinetics 数据集, 该数据集在关注全局时序特征变化的同时, 更关注局部特征信息的变化。因此, 在 Something-something 数据集的实验中, SFAM 的层数设置为 4。

表 3 给出了在 MQTS 中设置不同关键 tokens 和冗余 tokens 的比例的实验结果。在表 3 中, Key∶Redundant 表示关键 token 和冗余 token 的选择比例。如表 3 所示, 即关键 tokens 与冗余 tokens 的比例为 1∶4 时, 关键 token 占比仅为 20% 时, Key∶Redundant 的参数比例过大表明过高的冗余 token 比例导致模型聚焦于非判别性特征, 削弱了时空关键信息的捕获能力。比例为

2∶3 时，模型的计算量相对较低，冗余 token 占比较高，模型对关键特征的捕捉能力不足，准确率没有达到最优。当比例为 1∶1 时，模型的表现达到了较好的平衡，此时虽然计算量略有增加，但模型能够更有效地捕捉视频中的关键信息，准确率显著提升。当关键 token 占比超过 60% 后，模型表现下降，这表明过高的比例反而引入了不必要的冗余，导致特征选择的判别能力降低，准确率下降。

表 2　SFAM 层数实验

Table 2　Experiment of SFAM counts

M	FLOPs/10^9	Top-1/%	Top-5/%
2	331	57.3	86.1
3	332	57.8	86.5
4	**333**	58.5	**87.8**
5	334	58.1	87.3
6	335	**58.9**	87.2

注：加粗表示最优结果。

表 3　关键 token 和冗余 token 选择比例实验

Table 3　Experiment on the selection ratio of key tokens and redundant tokens

Key∶Redundant	FLOPs/10^9	Top-1/%	Top-5/%
1∶4	330	57.4	86.9
2∶3	332	57.8	87.1
1∶1	334	**58.5**	**87.8**
3∶2	336	58.1	87.4
4∶1	**338**	58.3	87.5

注：加粗表示最优结果。

3.3.3　实验 3 的结果及分析

表 4 给出了 MQTSFormer 在 Kinetics-400 和 Kinetics-600 数据集上的实验结果，并与多个基线模型进行了对比。表 4 列出了各模型的预训练信息、视频帧输入设置、FLOPs、Top-1 和 Top-5 准确率。表中实验数据均来自所引用论文，且输入尺寸都是 224×224 的情况下，“—”表示原论文模型没有在该数据集上进行实验。

表 4　在 Kinetics-400/ Kinetics-600 数据集上的对比实验结果

Table 4　Comparative experimental results on the Kinetics-400/ Kinetics-600 datasets

方法	预训练	帧×片段×剪裁	FLOPs/10^{12}	Kinetics-400		Kinetics-600	
				Top-1/%	Top-5/%	Top-1/%	Top-5/%
LGD[40]	IN-1K	128 × N/A	N/A	79.4	94.4	81.5	95.6
SlowFast+NL[5]	None	16 × 3 × 10	7.00	79.8	93.9	81.8	95.1
X3D-XL[25]	None	16 × 1 × 4	1.50	79.1	93.9	81.9	94.5
UniFormerV1[12]	IN-1K	16× 3 × 4	0.40	82.0	95.1	84.0	*96.4*
TimeSformer-L[10]	IN-21K	96 × 3 × 1	7.10	76.1	92.6	82.2	95.5
Mformer -HR[29]	IN-21K	16 × 3 × 10	28.80	81.1	95.2	82.7	96.1
ip-CSN[41]	Sports1M	32 × 3 × 10	3.30	79.2	93.8	—	—
CorrNet[41]	Sports1M	32 × 3 × 10	6.70	81.0	94.2	—	—
MoViNet-A6[42]	None	120 × 1 × 1	0.39	81.5	95.3	83.5	96.2
ViT-B-VTN[43]	IN-21K	250 × 1 × 1	4.00	78.6	93.7	—	—
STAM[44]	IN-21K	64 × 1 × 16	1.10	79.2	93.2	—	—
X-ViT[45]	IN-21K	8 × 3 × 1	0.40	78.5	93.7	<u>84.5</u>	96.3
MViT-B[3]	None	16 × 1 × 5	0.40	78.4	93.5	82.1	95.7
ViViT-L[9]	JFT-300M	16 × 3 × 4	17.40	<u>82.8</u>	95.3	*84.3*	96.3
Swin-B[11]	IN-21K	32 × 3 × 4	3.40	*82.7*	<u>95.5</u>	84.0	<u>96.5</u>
VideoMamba-M[46]	IN-1K	16 × 3 × 4	2.40	81.9	*95.4*	—	—
MQTSFormer	CLIP400M	8 × 1 × 3	0.20	**83.7**	**96.1**	**84.7**	**96.6**

注：加粗字体代表最优准确率，下划线代表次优准确率，斜体字体代表第三准确率。

表 4 中，在输入帧率为 8 帧/s 时，MQTSFormer 在 Kinetics-400 和 Kinetics-600 数据集上均取得了最优准确率，展现了其高效的时空建模能力。与卷积神经网络的方法（如 LGD、SlowFast、X3D）相

比, MQTSFormer 解决了 CNN 因感受野小而难以捕捉全局运动信息的问题; 与基于 ViT 的方法(如 TimeSformer、VIVIT、Swin-Transformer)相比, 解决了 ViT 的空间机制在浅层无法聚焦局部特征, 造成计算资源浪费的问题。

MQTSFormer 对时间注意力机制进行了改进, 通过聚焦关键特征并舍弃冗余特征的方式, 显著降低了计算量。通过对 FLOPs 指标的分析, MQTSFormer 的计算复杂度仅为 0.2×10^{12}, 为所有对比方法中最低。

为了比较和分析本文方法的推理速度, 选取 UniFormerV1 作为比较对象。UniFormerV1 的 FLOPs 为 0.4×10^{12}, 高于本文提出的 MQTSformer, 但低于其他基线模型。并且, UniFormerV1 是所有 FLOPs 为 0.4×10^{12} 方法中准确率最高的。实验中, 用 FPS(frame per second)表示推理速度。得到 UniFormerV1 的 FPS 为 13, MQTSFormer 的 FPS 为 12, 两者性能差距不大。此结果表明了 MQTSFormer 在推理速度、准确度和计算复杂度方面的优势。

使用 MQTSFormer 在 Something-SomethingV1(SSv1)和 V2(SSv2)数据集上进行实验, 并与多个基线模型进行对比。表 5 给出了不同对比模型的输入尺寸、预训练信息以及 Top-1 和 Top-5 准确率。表中实验数据均来自所引用的论文, "—"表示原论文模型没有在该数据集上进行实验。

表 5　在 Something-somethingV1/ Something-somethingV2 数据集上的对比实验结果

Table 5　Comparative experimental results on the Something-somethingV1/ Something-somethingV2 datasets　%

方法	预训练	输入大小	SSv1		SSv2	
			Top-1	Top-5	Top-1	Top-5
TSN[17]	IN-1K	16×224^2	19.9	47.3	30.0	60.5
TSM[19]	IN-1K	16×224^2	47.2	77.1	—	—
TEA[47]	IN-1K	16×224^2	51.9	80.3	—	—
CT-Net[48]	IN-1K	16×224^2	52.5	80.9	64.5	89.3
TDN[49]	IN-1K	16×224^2	55.3	**88.3**	65.3	89.5
SlowFast[5]	K400	32×224^2	—	—	63.1	87.6
ViViT-L[9]	—	16×224^2	—	—	65.4	89.8
MViTv1-B[3]	K400	16×224^2	—	—	64.7	89.2
TimeSformer-HR[10]	IN-1K	16×224^2	—	—	62.4	81.0
Mformer-HR[29]	—	16×224^2	—	—	67.1	90.6
UniFormerV1-B[12]	K400	16×224^2	**59.1**	*86.2*	**70.4**	**92.8**
UniFormerV2[31]	CLIP-400M	16×224^2	*56.8*	84.2	69.5	92.3
VideoMamba-Ti[46]	IN-1K	16×224^2	—	—	66.0	89.6
MQTSFormer	CLIP-400M	16×224^2	58.5	87.8	*67.9*	*91.4*

注: 加粗字体代表最优准确率, 下划线代表次优准确率, 斜体字体代表第三准确率。

表 5 中, 在输入帧率为 8 帧/s 时, MQTSFormer 在 SSv1 数据集的 Top-1 和 Top-5 准确率达到次优, 在 SSv2 数据集的 Top-1 和 Top-5 的准确率位列前三。与卷积神经网络方法(如 TSN、SlowFast、TSM)相比, MQTSFormer 克服了 CNN 因感受野受限而难以捕捉全局运动信息的固有缺陷, 在 Top-1 准确率上实现了最优表现, Top-5 准确率上取得了次优的结果。与基于 ViT的方法(如 TimeSformer、VIVIT、MViTv1)相比, MQTSFormer 成功解决了 ViT 空间机制在浅层网络难以聚焦局部特征而导致计算资源浪费的问题, 其 Top-1 和 Top-5 准确率均稳定在前三。虽然在 Something-Something 数据上, MQTSFormer 在基于 ViT 的模型中尚未达到最优识别效果, 但如 Kinetics 数据集实验中关于 FLOPs 的比较和分析表明, MQTSFormer 改进的时间注意力机制通过精准聚焦关键特征并有效舍弃冗余特征, 显著降低了计算复杂度。

4　结束语

本文提出了一种名为 MQTSformer 的行为识别模型, 设计了浅层局部聚合模块 LFAM、多查

询 token 选择的时间注意力模块 TAMMQTS 和混合空间感知模块 HSPM。实验结果表明，MQTSFormer 在 Kinetics-400、Kinetics-600、SomethingSomethingV1 和 V2 等多个公开数据集上均表现出色。尽管 MQTSFormer 在实验中取得了较好的性能，但仍存在一些局限性。未来的研究可以重点关注设计一种动态变化的层数方案，根据不同数据集的特性对层数进行自适应调整。在 MQTSFormer 的时间注意力中，多查询 token 选择机制 MQTS 筛选关键 token，舍弃冗余的 token 来提高计算效率，这种做法虽然有效减少了计算资源的消耗，但也可能导致信息的丢失。未来可以研究自适应 token 保留策略。

参考文献：

[1] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16×16 words: Transformers for image recognition at scale [EB/OL]. (2020−10−22)[2025−03−03]. https://arxiv.org/abs/2010.11929.

[2] LIU Ze, LIN Yutong, CAO Yue, et al. Swin transformer: hierarchical vision transformer using shifted windows [C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 10012−10022.

[3] FAN Haoqi, XIONG Bo, MANGALAM K, et al. Multiscale vision transformers[C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 6824−6835.

[4] ZHANG D J, LI Kunchang, CHEN Yunpeng, et al. MorphMLP: a self-attention-free, MLP-like backbone for image and video [EB/OL]. (2021−11−24)[2025−03−03]. https://arxiv.org/abs/2111.12527v1.

[5] FEICHTENHOFER C, FAN Haoqi, MALIK J, et al. SlowFast networks for video recognition[C]//2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019: 6201−6210.

[6] HE Kaiming, ZHANG Xiangyu, REN Shaoqing, et al. Deep residual learning for image recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 770−778.

[7] HOWARD A G, ZHU Menglong, CHEN Bo, et al. MobileNets: efficient convolutional neural networks for mobile vision applications[EB/OL]. (2017−04−17)[2025−03−03]. https://arxiv.org/abs/1704.04861.

[8] LI Xianhang, WANG Yali, ZHOU Zhipeng, et al. SmallBigNet: integrating core and contextual views for video classification[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 1089−1098.

[9] ARNAB A, DEHGHANI M, HEIGOLD G, et al. ViViT: a video vision transformer[C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2022: 6816−6826.

[10] BERTASIUS G, WANG Heng, TORRESANI L. Is space-time attention all you need for video understanding? [EB/OL]. (2021−02−09)[2025−03−03]. https://arxiv.org/abs/2102.05095.

[11] LIU Ze, NING Jia, CAO Yue, et al. Video swin Transformer[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 3192−3201.

[12] LI Kunchang, WANG Yali, GAO Peng, et al. UniFormer: unified Transformer for efficient spatiotemporal representation learning[EB/OL]. (2022−01−12)[2025−03−03]. https://arxiv.org/abs/2201.04676.

[13] SIMONYAN K, ZISSERMAN A. Two-stream convolutional networks for action recognition in videos[J]. Advances in neural information processing systems, 2014, 27: 568−576.

[14] HORN B K P, SCHUNCK B G. Determining optical flow[J]. Artificial intelligence, 1981, 17(1-3): 185−203.

[15] FEICHTENHOFER C, PINZ A, ZISSERMAN A. Convolutional two-stream network fusion for video action recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 1933−1941.

[16] FEICHTENHOFER C, PINZ A, WILDES R P. Spatiotemporal residual networks for video action recognition[EB/OL]. (2016−11−07)[2025−03−03]. https://arxiv.org/abs/1611.02155.

[17] WANG Limin, XIONG Yuanjun, WANG Zhe, et al. Temporal segment networks: towards good practices for deep action recognition[C]//Computer Vision–ECCV 2016. Cham: Springer International Publishing, 2016: 20−36.

[18] ZHOU Bolei, ANDONIAN A, OLIVA A, et al. Temporal relational reasoning in videos[C]//Computer Vision–ECCV 2018. Cham: Springer International Publishing, 2018: 831−846.

[19] LIN Ji, GAN Chuang, HAN Song. TSM: temporal shift module for efficient video understanding[C]//2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2020: 7082−7092.

[20] JI Shuiwang, XU Wei, YANG Ming, et al. 3D convolutional neural networks for human action recognition[J]. IEEE transactions on pattern analysis and machine intelli-

gence, 2013, 35(1): 221−231.

[21] TRAN D, BOURDEV L, FERGUS R, et al. Learning spatiotemporal features with 3D convolutional networks [C]//2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2016: 4489−4497.

[22] CARREIRA J, ZISSERMAN A. Quo vadis, action recognition? a new model and the kinetics dataset[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 4724−4733.

[23] TRAN D, WANG Heng, TORRESANI L, et al. A closer look at spatiotemporal convolutions for action recognition[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 6450−6459.

[24] QIU Zhaofan, YAO Ting, MEI Tao. Learning spatio-temporal representation with pseudo-3D residual networks [C]//2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017: 5534−5542.

[25] FEICHTENHOFER C. X3D: expanding architectures for efficient video recognition[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 200−210.

[26] 田枫, 卫宁彬, 刘芳, 等. 基于时空-动作自适应融合网络的油田作业行为识别[J]. 智能系统学报, 2024, 19(6): 1407−1418.
TIAN Feng, WEI Ningbin, LIU Fang, et al. Oilfield operation behavior recognition based on spatio-temporal and action adaptive fusion network[J]. CAAI transactions on intelligent systems, 2024, 19(6): 1407−1418.

[27] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30: 5998−6008.

[28] 陈卓超. 基于 Transformer 模型的行为识别研究及系统实现[D]. 北京: 北京邮电大学, 2024.
CHEN Zhuochao. Research and system implementation of behavior recognition based on Transformer model[D]. Beijing: Beijing University of Posts and Telecommunications, 2024.

[29] PATRICK M, CAMPBELL D, ASANO Y M, et al. Keeping your eye on the ball: trajectory attention in video Transformers[EB/OL]. (2021−06−09)[2025−03−03]. https://arxiv.org/abs/2106.05392.

[30] LI Yanghao, WU Chaoyuan, FAN Haoqi, et al. Mvitv2: improved multiscale vision Transformers for classification and detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. New Orleans: IEEE, 2022: 4804−4814.

[31] LI Kuchang, WANG Yali, HE Yinan, et al. Uniformerv2: spatiotemporal learning by arming image vits with video uniformer[C]//2023 IEEE/CVF International Conference on Computer Vision. Paris: IEEE, 2023: 1632−1643.

[32] LOU Meng, ZHANG Shu, ZHOU Hongyu, et al. TransXNet: learning both global and local dynamics with a dual dynamic token mixer for visual recognition [EB/OL]. (2023−10−30)[2025−03−03]. https://arxiv.org/abs/2310.19380.

[33] CHOLLET F. Xception: deep learning with depthwise separable convolutions[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 1800−1807.

[34] HU Jie, SHEN Li, SUN Gang. Squeeze-and-excitation networks[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 7132−7141.

[35] WOO S, PARK J, LEE J Y, et al. CBAM: convolutional block attention module[C]//Computer Vision–ECCV 2018. Cham: Springer International Publishing, 2018: 3−19.

[36] ZHENG J, REZAGHOLIZADEH M, PASSBAN P. Dynamic position encoding for Transformers[EB/OL]. (2022−04−18)[2025−03−03]. https://arxiv.org/abs/2204.08142.

[37] DONG Xiaoyi, BAO Jianmin, CHEN Dongdong, et al. CSWin transformer: a general vision transformer backbone with cross-shaped windows[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 12124−12134.

[38] GOYAL R, KAHOU S E, MICHALSKI V, et al. The “something something” video database for learning and evaluating visual common sense[C]//2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017: 5843−5851.

[39] LOSHCHILOV I, HUTTER F. Stochastic gradient descent with warm restarts[C]//Proceedings of the 5th International Conference on Learning Representations. Toulon: OpenReview. net, 2017: 1−16.

[40] QIU Zhaofan, YAO Ting, NGO C W, et al. Learning spatio-temporal representation with local and global diffusion[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. Long Beach: IEEE, 2019: 12056−12065.

[41] TRAN D, WANG Heng, FEISZLI M, et al. Video classification with channel-separated convolutional networks [C]//2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019: 352−361.

[42] KONDRATYUK D, YUAN Liangzhe, LI Yandong, et al.

MoViNets: mobile video networks for efficient video recognition[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021: 16020−16030.

[43] NEIMARK D, BAR O, ZOHAR M, et al. Video transformer network[C]//2021 IEEE/CVF International Conference on Computer Vision Workshops. Montreal: IEEE, 2021: 3163−3172.

[44] SRINIVAS A, LIN T Y, PARMAR N, et al. Bottleneck transformers for visual recognition[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021: 16514−16524.

[45] BULAT A, PEREZ RUA J M, SUDHAKARAN S, et al. Space-time mixing attention for video Transformer[J]. Advances in neural information processing systems, 2021, 34: 19594−19607.

[46] LI Kunchang, LI Xinhao, WANG Yi, et al. VideoMamba: state space model for efficient video understanding[C]// Computer Vision – ECCV 2024. Cham: Springer Nature Switzerland, 2024: 237−255.

[47] LI Yan, JI Bin, SHI Xintian, et al. TEA: temporal excitation and aggregation for action recognition[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 909−918.

[48] LI Kunchang, LI Xianhang, WANG Yali, et al. Ct-Net: channel tensorization network for video classification [EB/OL]. (2021−06−03)[2025−03−03]. https://arxiv.org/abs/2106.01603.

[49] WANG Limin, TONG Zhan, JI Bin, et al. TDN: temporal difference networks for efficient action recognition[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021: 1895−1904.

作者简介：

刘歆，副教授，博士，主要研究方向为机器学习、数据分析、行为识别以及图像处理。E-mail：liuxin@cqupt.edu.cn。

曾奎，硕士研究生，主要研究方向为视频行为识别。E-mail：a13350326994@163.com。

陈奉，讲师，博士，主要研究方向为智能数据分析。E-mail：chenfeng@cqupt.edu.cn。