计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
2025,34(8):159−168 [doi: 10.15888/j.cnki.csa.009944] [CSTR: 32024.14.csa.009944]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

改进 ST-GCN 的人体跌倒检测①

王世刚, 邓珍妮, 饶淼淼

(广西科技大学 自动化学院, 柳州 545616)
通信作者: 王世刚, E-mail: gxwsg@gxust.edu.cn

摘　要: 针对 ST-GCN 算法在动作识别中需要预先定义人体骨架拓扑图及准确率有待提高等问题, 提出了基于 OpenPose 与改进 ST-GCN 结合的跌倒检测算法. 利用 OpenPose 算法提取人体骨骼关键点数据, 将骨骼关键点数据输入改进的 ST-GCN 算法中进行动作识别. 对 ST-GCN 算法进行改进, 引入自适应图卷积模块, 通过动态调整图结构, 增强模型对不同动作类型特征提取的灵活性; 引入注意力机制模块, 进一步提升模型的识别性能. 在公开数据集上验证的结果显示, NTU-RGB+D 60 数据集上, X-Sub 和 X-View 的 top-1 准确率与改进前相比分别提高 2.2% 和 2.5%; Kinetics-Skeleton 数据集上, top-1 和 top-5 准确率分别提高 3.1% 和 4%. 自建数据集上的准确率与改进前相比提高 4.7%. 实验结果表明, 所提出的算法满足实际应用需求.

关键词: 时空图卷积; 人体姿态估计; 跌倒检测; 计算机视觉

引用格式: 王世刚,邓珍妮,饶淼淼.改进 ST-GCN 的人体跌倒检测.计算机系统应用,2025,34(8):159–168. http://www.c-s-a.org.cn/1003-3254/9944.html

Improved ST-GCN for Human Fall Detection

WANG Shi-Gang, DENG Zhen-Ni, RAO Miao-Miao

(School of Automation, Guangxi University of Science and Technology, Liuzhou 545616, China)

Abstract: A fall detection algorithm combining OpenPose with an improved ST-GCN is proposed to address the limitations of low accuracy and the need for pre-defining human skeleton topology graphs of the ST-GCN algorithm in action recognition. The OpenPose algorithm is used to extract the human skeletal keypoint data, which is then input into the improved ST-GCN algorithm for action recognition. The ST-GCN algorithm is improved by introducing an adaptive graph convolution module, which dynamically adjusts the graph structure to enhance the flexibility in feature extraction across different action types; an attention mechanism module is introduced to further improve the recognition performance of the model. Validation on publicly available datasets shows that on the NTU-RGB+D 60 dataset, the top-1 accuracy of X-Sub and X-View is improved by 2.2% and 2.5%, respectively, compared with the baseline; on the Kinetics-Skeleton dataset, the top-1 and top-5 accuracy are improved by 3.1% and 4%, respectively. In addition, the accuracy measured on the self-constructed dataset is improved by 4.7% compared with that before the improvement. The experimental results show that the proposed algorithm meets the requirements of practical applications.

Key words: spatial-temporal graph convolution; human pose estimation; fall detection; computer vision

1　引言

随着人口老龄化的加剧, 老年人的安全问题愈发受到关注和重视. 根据世界卫生组织的数据显示, 全球每年约 64 万起致命跌倒事件, 其中大部分是 65 岁以上的老年人[1]. 此外, 在医院、养老院和康复中心的走廊也常出现意外跌倒的情况. 若未能及时发现跌倒的

① 基金项目: 广西自然科学基金联合专项 (2025GXNSFHA069207, 2025GXNSFHA069265)
收稿时间: 2024-12-22; 修改时间: 2025-02-12; 采用时间: 2025-03-06; csa 在线出版时间: 2025-06-24
CNKI 网络首发时间: 2025-06-25

患者, 将会延误患者的治疗时间[2]. 然而, 当前医疗资源的有限性难以有效应对频繁发生的意外跌倒事件, 这不仅极大地增加了医护人员的工作负担, 还可能对患者的安全构成潜在威胁. 因此, 提出一种能够实时检测人体跌倒的算法尤为重要.

目前的跌倒检测技术主要分为基于可穿戴设备的跌倒检测、基于环境式设备的跌倒检测[3]和基于计算机视觉的跌倒检测[4].

● 基于可穿戴设备的跌倒检测. 这类方法主要将倾斜开关、加速度计、陀螺仪等传感器嵌入到可穿戴设备上[5], 将设备佩戴在腰背、手腕、手臂等人体部位, 通过分析设备接收到的数据来判断是否发生跌倒. Montanini 等[6]设计一种基于人体穿戴的鞋子来进行跌倒检测, 通过压力传感器和三轴加速度计收集信息, 从而判断是否跌倒. Lai 等[7]利用数个三轴加速度传感器装置, 对意外跌倒发生时受伤的身体部位进行联合感测, 该系统可以根据加速度超出正常范围的值来判断发生跌倒的可能性, 还能判断受伤程度. Sheikh 等[8]提出一种基于低成本、轻量级惯性传感方法的轮椅跌倒检测系统, 该系统利用混合方案和无监督单类 SVM 来检测导致轮椅操纵过程中“跌倒”异常的情况以及无人协助的转移情况. 该方法具有成本低、易于安装等优点, 但被监测者需要随时随地佩戴, 老年群体的记忆力下降, 会忘记佩戴设备, 且设备穿戴不当也会导致测量值不准确, 从而影响设备的有效性.

● 基于环境式设备的跌倒检测. 这类方法在人体生活区域放置压力传感器、声音传感器等设备, 利用声音、视频数据以及震动信息来进行判断. Mothkari 等[9]使用 UWB 传感器数据应用无监督变化检测方法来检测跌倒. Shao 等[10]利用地板振动来检测跌倒事件并区分不同跌倒姿势. Zhuang 等[11]提出一种仅使用来自单个远场麦克风的音频信号, 即可检测家庭环境中人体跌倒的系统. 该方法虽然无需穿戴设备且不受光线影响, 但成本较高, 容易受到外部因素的干扰.

● 基于计算机视觉的跌倒检测. 这类方法不受环境影响, 也不需要随身携带. Carlier 等[12]通过光流法提取光流图像, 利用卷积神经网络进行分类判断. Fan 等[13]将深度卷积网络应用于跌倒检测, 将一段视频转换成一个动图, 将动图送到深度卷积网络进行训练, 通过检测是否存在一个完整的跌倒流程来判断是否跌倒. 马敬奇等[14]获取人体姿态关节点图像坐标, 结合人体跌倒过程瞬时姿态变化特征和跌倒状态短时持续不变的特征来判断跌倒现象的发生. 骨架点属于非欧数据, 无法使用 CNN 等方法进行处理[15], 研究者提出图卷积网络 (graph convolution network, GCN)[16]用于处理这类数据, 并在动作识别领域取得较好的应用. ST-GCN 算法[17]在空间和时间上交替进行卷积, 可以很好获取动态骨架信息. 但 ST-GCN 算法存在一些不足, 例如预定义邻接矩阵的固定拓扑结构在处理不同动作任务时存在局限性, 整体准确率有待提高.

针对以上问题, 本文提出将 OpenPose 人体姿态估计算法与 ST-GCN 网络相结合, 利用人体骨骼关节的运动信息, 对视频中的人体动作进行分类与识别. 在 ST-GCN 算法中引入自适应图卷积模块, 解决网络中图拓扑结构固定、建模不灵活的问题, 同时加入注意力机制模块, 进一步提升模型的识别性能.

2 基于 OpenPose 的人体姿态估计算法

OpenPose 算法[18]是一种自底向上的关键点检测方法, 能够高效地完成从人体关键点检测到关节连接的整个过程. 首先提取图像中人体的 18 个骨骼关节点信息, 这些关节点包括眼睛、鼻子、脖子和手臂等, 每个关节点具有相应的编号, 从 0–17 进行编号. 其次, 通过部分亲和场 (part affine field, PAF) 将关键点分配给每个人, 并找到当前最合适的连接方式, 对包含人的图像进行人体姿态估计. 18 个骨骼关键点如图 1 所示.

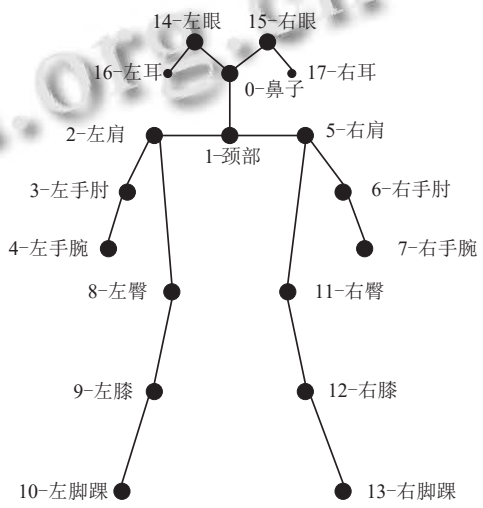

图 1 18 个骨骼关键点

OpenPose 人体姿态估计算法的结构如图 2 所示, 整个网络是多阶段结构, 各阶段之间串行连接, 每个阶段包含两个分支. 使用 VGG-19 的前 10 层卷积网络对输入图片进行特征提取, 得到特征图 F, 然后将提取的

特征图 F 送入 Branch1 和 Branch2 这两个并列的卷积神经网络中. Branch1 用于预测关节点的置信图 (part confidence map, PCM); Branch2 用于预测部分亲和域. 将两个分支的输出和特征图 F 进行融合后传入下一个阶段, 以此类推直到最后一个阶段. 在最后一个阶段, 整合所有阶段的输出结果, 得到精确的关节点置信图和部分亲和域, 利用这些信息进行关节点的匹配和连接, 从而实现对人体姿态的准确估计.

置信图和部分亲和域分别映射关节点置信度 (S) 和连接关系 (L), 这两部分共同作用可以准确描述人体关节点位置及其连线关系. 在模型的第 1 阶段, VGG-19 网络的输出特征图被用作输入层, 用于生成初步的关节点置信图和亲和域. 在随后的各个阶段, 输入层由 VGG-19 网络的输出特征图与前一阶段的关节点置信图和亲和域的组合向量共同构成, 通过逐阶段优化结果, 不断提高预测的精度和鲁棒性, 即:

$$S_1 = \rho_1(F),\ L_1 = \phi_1(F) \tag{1}$$

$$S_t = \rho_t(F, S_{t-1}, L_{t-1}),\ L_t = \phi_t(F, S_{t-1}, L_{t-1}),\ \text{if}\ \forall t \geqslant 2 \tag{2}$$

其中, S_1 与 S_t 表示第 1 阶段和第 t 阶段的 2D 置信度; L_1 和 L_t 表示第 1 阶段和第 t 阶段的 2D 矢量场; $\rho_t(\cdot)$ 表示 PCM 预测网络; $\phi_t(\cdot)$ 表示 PAF 预测网络.

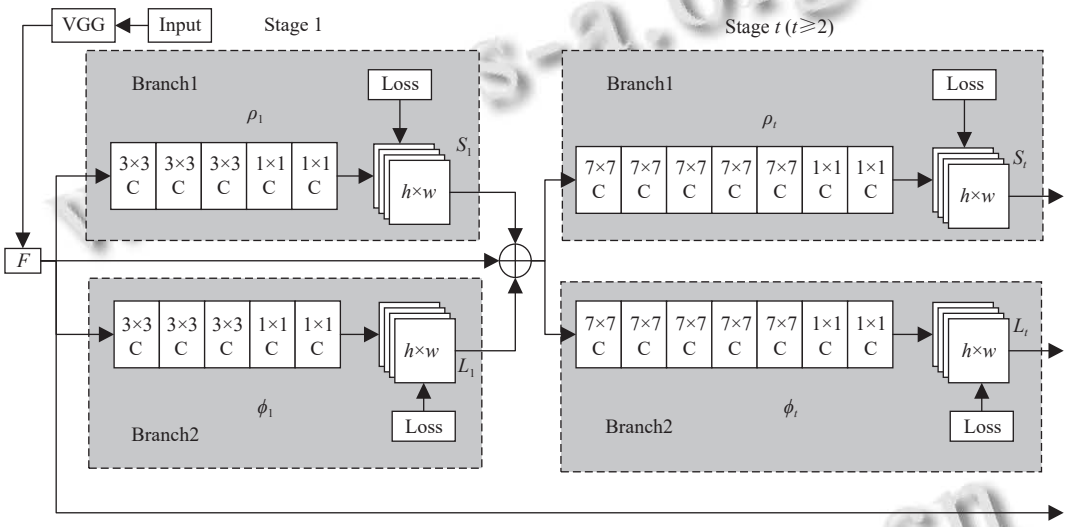

图 2 OpenPose 网络结构

部分亲和场网络描述关键点之间的联系, 保存位置和方向信息, 部分亲和场估计原理图如图 3 所示. 计算公式如式 (3) 和式 (4) 所示.

$$L_{c,k}^*(p) = \begin{cases} v, \text{若第}k\text{人的肢体}c\text{位于点}p \\ 0, \text{其他} \end{cases} \tag{3}$$

$$v = \frac{x_{j_2,k} - x_{j_1,k}}{\left\| x_{j_2,k} - x_{j_1,k} \right\|_2} \tag{4}$$

其中, $L_{c,k}^*(p)$ 表示第 k 个人的肢体 c 是否在像素点 p 上; v 表示肢体方向上的单位向量; $x_{j_1,k}$ 和 $x_{j_2,k}$ 表示第 k 个人的第 j_1 和 j_2 个关节点的真实位置.

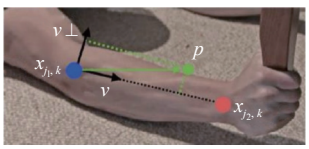

图 3 部分亲和场估计原理图

计算出肢体上一个点的亲和度无法判断骨骼关键点连接的有效性, 通过计算相应的关键点连线上的积分来表示其取值, 通过评估关节点之间的 PAF 值, 判断关键点之间连接的正确性. 当计算得到的 PAF 值越大时, 说明关键点连线方向的一致性越高, 即连接的正确性越强; 反之, 若 PAF 值较小, 表示连接的方向不一致或连接无效[19], 如式 (5) 所示:

$$E = \int_{u=0}^{u=1} L_c(p(u)) \times \frac{d_{j_1} - d_{j_2}}{\left| d_{j_2} - d_{j_1} \right|_2} \mathrm{d}u \tag{5}$$

其中, d_{j_1} 和 d_{j_2} 表示两个连续的像素点; $p(u)$ 表示连续像素点之间的像素点.

3 基于改进的 ST-GCN 人体动作识别算法

3.1 ST-GCN 算法

ST-GCN 算法应用于骨骼序列的动作识别任务中,

与早期的动作识别算法相比, 该方法显著提高了动作识别的效果.

通过 OpenPose 得到视频帧中人体的骨架图序列, 骨架序列只表示空间维度上人体关节点信息之间的关联性. 视频中人体的动作不是由某一帧来判断的, 而是通过连续帧来进行判断. 为了能够充分利用时间信息, 对得到的骨架序列进行时空建模. 时空骨架图的构建由两个步骤组成: 首先将每一帧图像上人体的每一个关键点用边连接起来构造空间骨架图; 其次对不同帧之间得到的空间骨架图, 按照时间顺序将每一帧中相同的关节点连接起来, 此时的骨架包含时间信息, 以此构建时空骨架图, 如图 4 所示. 时空骨架图包含有空间边和时序边, 空间边表示同一帧中不同节点之间的自然连通, 时间边表示不同帧之间相同节点的连接.

ST-GCN 网络结合空间卷积和时间卷积从数据中学习骨架序列的信息特征, 特征表达效果更加突出. ST-GCN 网络的数据处理流程如图 5 所示. 首先, 利用人体姿态估计算法 OpenPose 提取视频中人体的骨骼关键点数据, 将获得的骨骼关键点数据输入到时空图卷积模型中. 其次, 基于分区策略对连接完成的骨架进行分组. ST-GCN 有 3 种分区策略: 1) 单标签分区策略, 将根节点及其所有相邻节点划分为一个子集, 输出的邻接矩阵是一维的. 2) 距离分区策略, 该分区被分为两个子集, 一个是根节点本身, 另一个是与根节点直接相邻的节点, 输出的邻接矩阵是二维的. 3) 空间结构分区策略, 该分区被分为 3 个子集, 分别为根节点、相比于根节点更接近骨架重心的节点以及比根节点更远离骨架重心的节点, 输出的邻接矩阵是三维的. 接着, 将分区完成的数据放入时空图卷积网络, 分别进行空间卷积和时间卷积处理, 以提取更高层次的特征信息. 空间卷积用于捕捉骨骼关节点之间的空间关系, 时间卷积用于建模动作在时间维度上的动态变化. 最后, 通过 *Softmax* 分类器对提取的特征进行分类, 输出人体动作的最终分类结果.

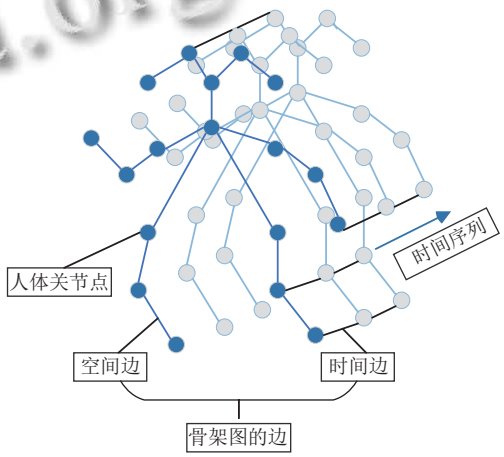

图 4 时空骨架图

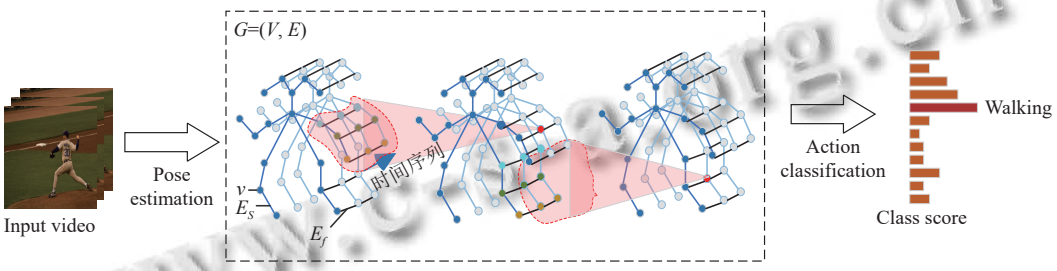

图 5 ST-GCN 网络数据处理流程图

3.2 引入自适应图卷积模块

ST-GCN 算法对于所有的动作识别都使用相同的拓扑结构, 动作不相同, 就会使得关节点之间的依赖程度和重要程度不同, 所以固定的图拓扑结构并不适用于所有的动作样本. 例如“弯腰”动作中, 腰部和背部关节点之间的依赖关系较强, 采用固定结构可能无法达到最佳效果, 且在特征提取过程中无法充分利用深度学习的自学习能力. 针对以上问题, 在 ST-GCN 算法的基础上, 加入自适应图卷积模块, 不需要预定义固定的邻接矩阵, 而是根据数据分布自适应调整节点之间的连接关系, 使神经网络能够动态学习并优化图的拓扑结构. ST-GCN 算法中的空间图卷积运算如式 (6) 所示:

$$f_{\text{out}}(v_{ti}) = \sum_{v_{tj} \in B(v_{ti})} \frac{1}{Z_{ti}(v_{tj})} f_{\text{in}}(v_{tj}) \cdot w(l_{ti}(v_{tj})) \tag{6}$$

其中, w 是权重矩阵; f_{in} 是输入的节点特征; Z_{ti} 是对应的节点数量; B 是采样函数, 定义为目标节点 v_{ti} 到相邻节点 v_{tj} 的距离.

将式 (6) 进行变换, 得到图卷积在空间维度上的公式, 如式 (7) 所示:

$$f_{\text{out}} = \sum_{k}^{K_v} W_k (f_{\text{in}} A_k) \odot M_k \tag{7}$$

其中, W_k 是可学习权重矩阵; M_k 是掩码矩阵; K_v 是空间维数的卷积核大小; A_k 是对邻接矩阵进行归一化; $\odot$ 表示矩阵点积运算.

为使图卷积具备自适应的特性, 自适应图卷积模块的公式如式 (8) 所示:

$$f_{\text{out}} = \sum_{k} W_k f_{\text{in}} (A_k + B_k + C_k) \tag{8}$$

其中, A_k 与原始的归一化邻接矩阵相同, 表示人体关节点的物理构造; B_k 也是一个邻接矩阵, 该矩阵没有受到归一化等约束条件的限制, 可以进行训练, 不仅能表明两个关节点之间存在联系, 还表明连接的强弱. 原矩阵 M_k 与 A_k 进行点乘, 当 A_k 中的一个值为 0, 不管 M_k 中的值如何, 结果都始终为 0, 因此, 它无法生成原骨架图表示中不存在的新连接. 而 B_k 可以产生先前没有的联系, 进而对相隔较远的关节点之间的联系进行建模, 比 M_k 更加灵活. C_k 是一个数据相关图, 为每个样本学习唯一的图, 通过归一化嵌入式高斯函数计算两个顶点之间的交互来获得连接强度, 如式 (9) 所示:

$$f(v_{ti}, v_{tj}) = \frac{e^{\theta(v_{ti})^{\mathrm{T}} \phi(v_{tj})}}{\sum_{j=1}^{J} e^{\theta(v_{ti})^{\mathrm{T}} \phi(v_{tj})}} \tag{9}$$

其中, $f(v_{ti}, v_{tj})$ 是输入的映射; J 是顶点的总数; v_{ti} 和 v_{tj} 代表两个任意顶点; θ 和 ϕ 是两个嵌入函数, 通过 1×1 卷积学习; $e^{\theta(v_{ti})^{\mathrm{T}} \phi(v_{tj})}$ 是相似度函数.

如图 6 所示, 首先将大小为 $C_{\text{in}} \times T \times J$ 的输入特征分别通过两个 1×1 卷积层的嵌入函数, 并分别被重新排列为一个 $J \times C_m \times T$ 矩阵和一个 $C_m \times T \times J$ 矩阵, 然后将它们相乘得到一个 $J \times J$ 大小的矩阵 C_k, 其元素 C_{ij} 表示顶点 v_{ti} 和顶点 v_{tj} 的连接强度. 将矩阵的值进行归一化, 作为两个顶点的虚拟连接, 归一化具有 *Softmax* 操作, C_k 的表达式如式 (10) 所示:

$$C_k = Softmax(f_{\text{in}}^{\mathrm{T}} W_{\theta k}^{\mathrm{T}} W_{\phi k} f_{\text{in}}) \tag{10}$$

其中, $W_{\theta k}$ 和 $W_{\phi k}$ 分别表示嵌入函数 θ 和 ϕ 的参数.

3.3 引入注意力机制模块

注意力机制能够根据输入数据的特征自适应地调整权重分配方式, 使得模型更关注于有用的信息. 从空间角度来看, 人的某个动作可能只需要移动部分关键点; 从时间角度来看, 一个包含多帧的动作流, 可能存在多个不同的关键阶段, 对于最后的识别具有不同的重要性; 从特征角度来看, 卷积的多个通道通常包含不同层次的语义信息, 对于不同动作的识别具有不同的价值[20]. 将空间注意力、时间注意力和通道注意力机制按照该顺序嵌入到 ST-GCN 算法中.

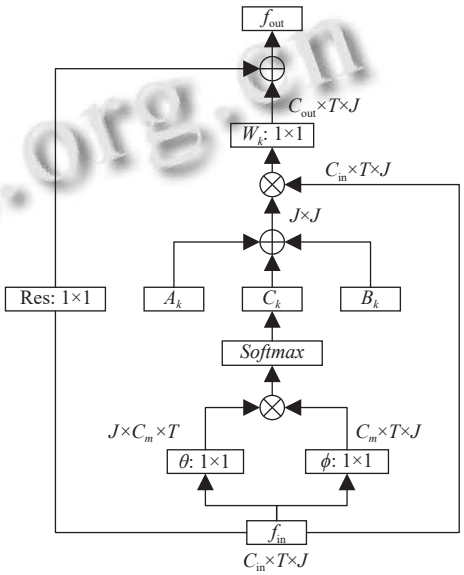

图 6 自适应图卷积模块

空间注意力机制 (spatial attention mechanism, SAM) 模块. 该模块可以使网络自动调整关注的区域, 精确地捕捉与动作相关的空间信息, 提升空间特征的表达能力, 进而提高动作识别的准确性. 空间注意力模块的表达式如式 (11) 所示:

$$s = \sigma(m_s(AvgPool(X))) \tag{11}$$

其中, $X \in R^{C \times T \times N}$ 表示输入模块; *AvgPool* 表示在时间维度上进行全局平均池化, 池化完成后, 特征图维度变为 $C \times 1 \times N$; m_s 是在空间维度上的一维卷积操作, 输出通道数为 1; σ 表示 Sigmoid 激活函数; $s \in R^{1 \times 1 \times N}$ 表示输出特征图. 输出与输入点乘后并相加, 得到最终的结果. 空间注意力的模块结构如图 7 所示.

时间注意力机制 (temporal attention mechanism, TAM) 模块: 该模块根据每一帧在时间序列中的重要性, 动态地分配不同的权重, 自动识别并关注对动作识别最为关键的帧, 而忽略对分类结果影响较小的帧. 有助于提高时序特征的表达能力, 使模型更聚焦于有意义的时间段. 时间注意力模块的表达式如式 (12)

所示:

$$s=\sigma(m_t(AvgPool(X))) \tag{12}$$

其中, $AvgPool$表示在空间维度上进行全局平均池化; m_t是在时间维度上的一维卷积操作; 最终得到$s\in R^{1\times T\times 1}$的输出特征, 与空间注意力模块相同, 将输出与输入进行点乘后相加. 时间注意力模块结构如图 8 所示.

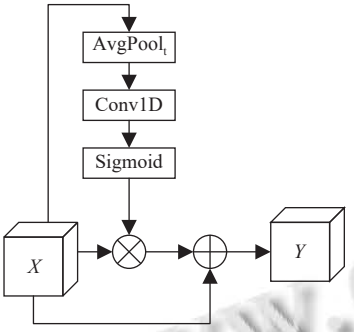

图 7 空间注意力模块

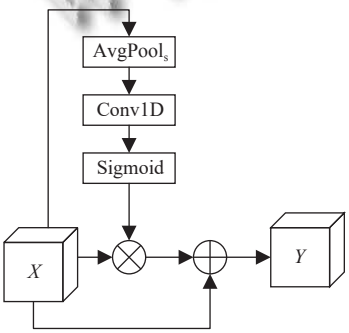

图 8 时间注意力模块

通道注意力机制 (channel attention mechanism, CAM) 模块: 该模块根据每个通道的特征重要性动态地调整其权重, 从而增强对关键通道的关注, 抑制冗余或不相关的通道特征. 有助于提升动作识别的准确性. 通道注意力包含压缩和激励两部分. 首先, 将输入大小为$C\times H\times N$的特征图进行压缩, 压缩后得到f_1, 特征图f_1的大小为$1\times 1\times C$. 具体来说就是进行全局平均池化的操作, 将全局空间和时间信息压缩到通道描述符中, 如式 (13) 所示:

$$f_1=\frac{1}{H\times N}\sum_{i=1}^{H}\sum_{j=1}^{N}u_c(i,j) \tag{13}$$

接下来是激励部分, 对f_1进行变换, 如式 (14) 所示:

$$f_2=\sigma(l_2\delta(l_1f_1)) \tag{14}$$

其中, $l_1\in R^{\frac{C}{r}\times C}$和$l_2\in R^{C\times\frac{C}{r}}$表示两个权重矩阵, 与两个全连接层相对应; σ表示 Sigmoid 激活函数; δ表示 ReLU 激活函数. 通道注意力模块结构如图 9 所示.

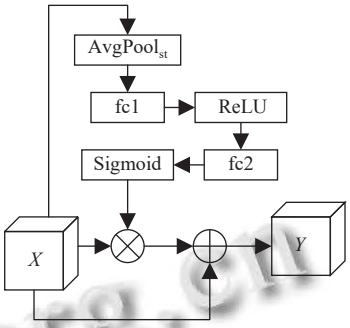

图 9 通道注意力模块

3.4 改进后的 ST-GCN 网络结构

ST-GCN 网络结构主要由 GCN 模块和 TCN 模块组成, GCN 模块主要用于提取骨架序列的空间特征, TCN 模块主要用于对时序特征进行提取. 在 GCN 模块中引入自适应图卷积模块, 以更好地处理图结构数据. 空间注意力机制、时间注意力机制和通道注意力机制依次插入两个模块之间, 简称为 STC 注意力模块. 为保持训练过程稳定, 减缓梯度传播问题, 在每个基本单元中加入残差连接. 基本单元的组成由图 10 所示.

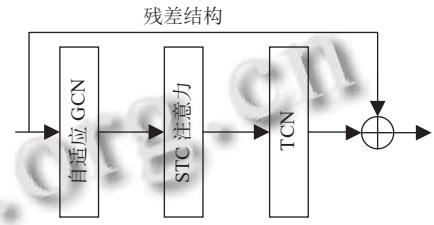

图 10 网络基本单元结构

网络整体结构如图 11 所示. 整个网络结构由 9 个基本单元组成, 每个单元的输出通道数依次为 64、64、64、128、128、128、256、256、256. 将输出的特征图送入平均池化层和一个全连接层, 最后经过 *Softmax* 函数处理得到预测结果.

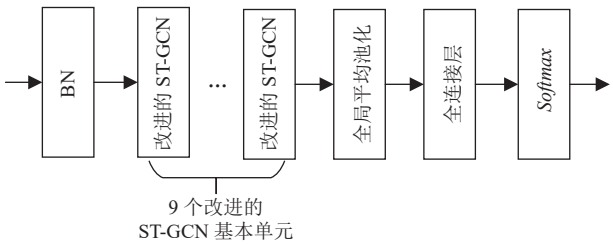

图 11 网络整体结构

4 实验

4.1 实验环境

实验处理器采用 Intel i7 13700F, 使用 Windows 10 操作系统, 内存为 64 GB, 显卡为索泰 RTX 4090, 显存为 24 GB. 以 Python 作为编程语言, 使用 PyTorch 作为深度学习框架. 在训练过程中, 全程通过 GPU 加速.

4.2 实验数据集

使用 NTU-RGB+D 60 和 Kinetics-Skeleton 两个公开数据集进行验证. 同时使用自建跌倒数据集对改进的算法进行训练和验证.

● NTU-RGB+D 60 数据集. 为包含 60 类动作的大规模人体动作识别数据集, 总计 56 000 个动作视频片段. 每个视频片段最多包含 2 个人, 视频由 3 台摄像头从不同角度进行拍摄. 使用三维坐标 (x, y, z) 表示视频中每个关节点的位置, 每个人体的骨架由 25 个关节点标注. 数据集划分为两种评估基准: X-Sub 和 X-View, 分别表示跨动作和跨视角. X-Sub 数据子集使用 40 320 个视频进行训练, 16 560 个视频进行测试. X-View 数据子集使用 37 920 个视频进行训练, 18 960 个视频进行验证.

● Kinetics-Skeleton 数据集. 该数据集从 YouTube 上下载剪辑, 共 30 万段视频, 动作类型丰富, 包含有 400 种人类动作. 训练集包含 24 万个片段, 验证集包含两万个片段, 每个视频片段的时长为 10 s 左右, 帧率为 30 f/s. Kinetics-Skeleton 数据集都是未经处理的视频片段, 不包含骨骼信息, 使用 OpenPose 算法对视频中的人体进行骨骼信息提取. 通过 OpenPose 算法获取人体的 18 个关节点信息, 每个关节点的数据包括 (X, Y, C), 其中 (X, Y) 为关键点位置坐标, C 为对应的置信度.

● 自建跌倒数据集 IBFD. 为验证算法的实际效果, 按照 Kinetics-Skeleton 数据集的格式制作跌倒数据集 IBFD, 该数据集共包含 6 种动作, 分别为跌倒、站立、坐下、弯腰、蹲下和走路, 每类动作共有 400 段视频, 每段视频长度至少 2 s, 总共有 2 400 个动作样本. 将动作样本分为跌倒行为和正常行为, 正常行为包括站立、坐下、弯腰、蹲下和走路. 由于视频格式的数据集不能直接用于训练和验证, 因此使用 OpenPose 算法提取骨骼信息, 大致的流程为: 首先, 将所有视频样本统一处理为不超过 300 帧的长度, 规范数据格式; 其次, 使用 OpenPose 算法对视频中的人体骨骼信息进行提取, 并将提取结果保存为 JSON 文件; 再次, 将数据集划分为训练集和验证集, 生成相应的标签文件; 最后, 将数据转换为 data.npy 和 label.pkl 文件.

4.3 实验结果与分析

(1) NTU-RGB+D 60 数据集训练及验证结果

在 NTU-RGB+D 60 数据集的 X-Sub 数据子集和 X-View 数据子集上对改进的算法进行训练和验证, 使用 SGD 优化训练过程, 迭代训练 50 个 epoch, 设置批次大小为 8, 初始学习率为 0.1, 在第 30 和 40 轮时进行学习率衰减, 每次将学习率降低至原来的 1/10. 训练过程中损失值的变化曲线如图 12 和图 13 所示.

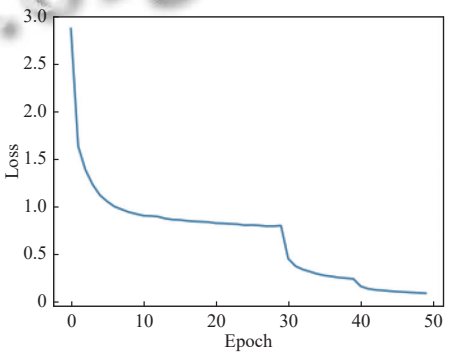

图 12 X-Sub 上训练损失变化曲线

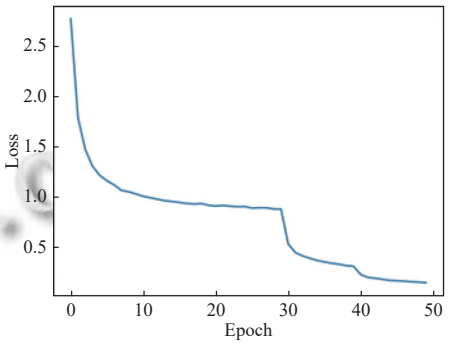

图 13 X-View 上训练损失变化曲线

在 NTU-RGB+D 60 的 X-Sub 数据子集上进行动作识别实验, 检验算法引入不同模块的有效性, ST-GCN 为基线网络; ST-AGCN 表示单独加入自适应图卷积模块; AST-GCN 表示单独加入注意力机制模块; AAST-GCN 表示同时加入自适应图卷积模块和注意力机制模块. 采用 top-1 和 top-5 准确率来评估模型性能. 实验结果如表 1 所示.

从表 1 中可知, 单独加入自适应图卷积模块的 ST-AGCN 算法与 ST-GCN 算法相比 top-1 和 top-5 准确率分别提升 1.7% 和 9%. 单独加入注意力机制的 AST-

GCN 与 ST-GCN 算法相比 top-1 和 top-5 准确率分别提升 0.9% 和 9.1%. 同时加入自适应图卷积模块和注意力机制模块的 AAST-GCN 算法在动作识别任务上 top-1 和 top-5 准确率分别达到 83.7% 和 97.4%, 相比于 ST-GCN 算法分别提高 2.2% 和 9.1%, AAST-GCN 算法的性能有较大的提升.

表 1　在 X-Sub 数据子集上的性能对比 (%)

算法	top-1	top-5
ST-GCN	81.5	88.3
ST-AGCN	83.2	97.3
AST-GCN	82.4	97.4
AAST-GCN	83.7	97.4

在 NTU-RGB+D 60 数据集的 X-Sub 和 X-View 数据子集上, 将最终的网络模型 AAST-GCN 与其他人体动作识别模型进行性能对比. 采用 top-1 准确率来评估模型性能. 实验结果如表 2 所示.

表 2　AAST-GCN 算法与其他模型的 top-1 性能对比 (%)

算法	X-Sub	X-View
Lie Group[21]	50.1	82.8
Deep-LSTM[22]	60.7	67.3
TCN[23]	74.3	83.1
ST-LSTM[24]	69.2	77.7
ST-GCN	81.5	88.3
DPRL+GCNN[25]	83.5	89.8
AAST-GCN (Ours)	83.7	90.8

从表 2 中可知, AAST-GCN 算法在 X-Sub 数据子集上 top-1 准确率为 83.7%, 在 X-View 数据子集上 top-1 准确率为 90.8%, 分别比基线 ST-GCN 算法高出 2.2% 和 2.5%, 与其他的人体动作识别算法相比, AAST-GCN 算法的性能均有所提升.

(2) Kinetics-Skeleton 数据集训练及验证结果

将改进的算法使用 SGD 优化训练过程, 迭代训练 50 个 epoch, 初始学习率为 0.1, 在第 20、30 和 40 轮时进行学习率衰减, 每次衰减为原来的 1/10. 训练过程中损失值的变化曲线如图 14 所示.

表 3 为改进的算法与其他人体动作识别方法在 Kinetics-Skeleton 数据集上的识别性能对比. 可以看出, AAST-GCN 算法的性能优于其他算法, 与 ST-GCN 算法相比 top-1 和 top-5 准确率的评价指标结果分别提高 3.1% 和 4%.

(3) 自建跌倒数据集 IBFD 训练及验证结果

将改进的算法使用 SGD 优化训练过程, 迭代训练 50 个 epoch, 初始学习率为 0.1, 在第 20、30 和 40 轮进行学习率衰减, 每次衰减为原来的 1/10. 采用准确率 *Accuracy* 评估算法性能, 行为分类正确为 True (T), 反之为 False (F), 计算公式如式 (15) 所示:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{15}$$

其中, *TP* 表示真正例, *TN* 表示真反例, *FP* 表示是假正例, *FN* 表示假反例.

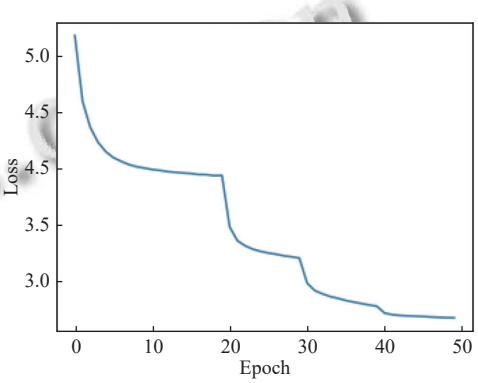

图 14　Kinetics-Skeleton 训练损失变化曲线

表 3　AAST-GCN 与其他模型的性能对比 (%)

算法	top-1	top-5
Feature Enc[26]	14.9	25.8
Deep-LSTM	16.4	35.3
TCN	20.3	40.0
ST-GCN	30.7	52.8
AAST-GCN	33.8	56.8

在自建跌倒数据集 IBFD 上进行实验, 实验结果如表 4 所示. IBFD 数据集在 ST-GCN 算法上的准确率为 87.1%, 在 AAST-GCN 算法上的准确率为 91.8%, 与 ST-GCN 算法相比提高 4.7%.

表 4　自建数据集 IBFD 的实验对比 (%)

算法	准确率
ST-GCN	87.1
AAST-GCN	91.8

将 IBFD 数据集分为正常行为和跌倒行为. 从表 5 中可以看到跌倒行为的精确率为 93.1%, 正常行为的精确率为 91.5%. 由于环境复杂和视频拍摄角度的单一, 存在人体遮挡的情况, 导致一定的误检.

表 5　IBFD 数据集分类测试

行为类别	总视频数	精确率 (%)
跌倒行为	88	93.1
正常行为	402	91.5

5 结论与展望

针对动作识别准确率不高、需要预先定义人体骨架拓扑图等问题, 使用 OpenPose 算法提取视频的人体骨骼关键点信息, 将骨骼点信息放入改进的 ST-GCN 算法中进行动作分类与识别. 在 ST-GCN 算法的基础上引入自适应图卷积模块和注意力机制模块, 提高算法对不同动作类型特征提取的灵活性和动作识别能力. 实验结果表明, 改进的算法适用于实际应用需求, 在自建跌倒数据集上能够实现 93.1% 的跌倒检测精确率, 具有良好的鲁棒性. 但对于部分行为还存在误判, 未来将进一步改善, 同时将改进算法应用于跟随机器人中完成人体跌倒检测.

参考文献

1 Wang XY, Ellul J, Azzopardi G. Elderly fall detection systems: A literature survey. Frontiers in Robotics and AI, 2020, 7: 71. [doi: 10.3389/frobt.2020.00071]

2 孙颖, 张吟龙, 王鑫, 等. 面向医疗护理的视觉监控医院患者跌倒检测. 中国医学物理学杂志, 2022, 39(4): 436–441. [doi: 10.3969/j.issn.1005-202X.2022.04.008]

3 Alarifi A, Alwadain A. Killer heuristic optimized convolution neural network-based fall detection with wearable IoT sensor devices. Measurement, 2021, 167: 108258. [doi: 10.1016/j.measurement.2020.108258]

4 Khan SS, Hoey J. Review of fall detection techniques: A data availability perspective. Medical Engineering & Physics, 2017, 39: 12–22.

5 Nooruddin S, Islam MM, Sharna FA, *et al*. Sensor-based fall detection systems: A review. Journal of Ambient Intelligence and Humanized Computing, 2022, 13(5): 2735–2751. [doi: 10.1007/s12652-021-03248-z]

6 Montanini L, del Campo A, Perla D, *et al*. A footwear-based methodology for fall detection. IEEE Sensors Journal, 2018, 18(3): 1233–1242. [doi: 10.1109/JSEN.2017.2778742]

7 Lai CF, Chang SY, Cho HC, *et al*. Detection of cognitive injured body region using multiple triaxial accelerometers for elderly falling. IEEE Sensors Journal, 2011, 11(3): 763–770. [doi: 10.1109/JSEN.2010.2062501]

8 Sheikh SY, Jilani MT. A ubiquitous wheelchair fall detection system using low-cost embedded inertial sensors and unsupervised one-class SVM. Journal of Ambient Intelligence and Humanized Computing, 2023, 14(1): 147–162. [doi: 10.1007/s12652-021-03279-6]

9 Mokhtari G, Aminikhanghahi S, Zhang Q, *et al*. Fall detection in smart home environments using UWB sensors and unsupervised change detection. Journal of Reliable Intelligent Environments, 2018, 4(3): 131–139. [doi: 10.1007/s40860-018-0065-2]

10 Shao Y, Wang XY, Song WJ, *et al*. Feasibility of using floor vibration to detect human falls. International Journal of Environmental Research and Public Health, 2021, 18(1): 200.

11 Zhuang XD, Huang J, Potamianos G, *et al*. Acoustic fall detection using Gaussian mixture models and GMM supervectors. Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. Taipei: IEEE, 2009. 69–72.

12 Carlier A, Peyramaure P, Favre K, *et al*. Fall detector adapted to nursing home needs through an optical-flow based CNN. Proceedings of the 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC). Montreal: IEEE, 2020. 5741–5744.

13 Fan YX, Levine MD, Wen GJ, *et al*. A deep neural network for real-time detection of falling humans in naturally occurring scenes. Neurocomputing, 2017, 260: 43–58. [doi: 10.1016/j.neucom.2017.02.082]

14 马敬奇, 雷欢, 陈敏翼. 基于 AlphaPose 优化模型的老人跌倒行为检测算法. 计算机应用, 2022, 42(1): 294–301.

15 Zhou J, Cui GQ, Hu SD, *et al*. Graph neural networks: A review of methods and applications. AI Open, 2020, 1: 57–81. [doi: 10.1016/j.aiopen.2021.01.001]

16 Bruna J, Zaremba W, Szlam A, *et al*. Spectral networks and locally connected networks on graphs. arXiv:1312.6203, 2013.

17 Yan SJ, Xiong YJ, Lin DH. Spatial temporal graph convolutional networks for skeleton-based action recognition. Proceedings of the 32nd AAAI Conference on Artificial Intelligence. New Orleans: AAAI Press, 2018. 7444–7452.

18 Cao Z, Simon T, Wei SE, *et al*. Realtime multi-person 2D pose estimation using part affinity fields. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu: IEEE, 2017. 1302–1310.

19 王鸿, 陈明举, 熊兴中, 等. 基于 OpenPose 与 AT-STGCN 的电力作业人员行为识别技术. 四川轻化工大学学报 (自然科学版), 2023, 36(4): 61–70. [doi: 10.11863/j.suse.2023.04.08]

20 位俊超, 陈春雨. 基于 SAT-GCN 的花样滑冰选手动作检测算法研究. 应用科技, 2023, 50(1): 7–13.

21 Vemulapalli R, Arrate F, Chellappa R. Human action recognition by representing 3D skeletons as points in a lie group. Proceedings of the 2014 IEEE Conference on

Computer Vision and Pattern Recognition. Columbus: IEEE, 2014. 588–595.

22 Shahroudy A, Liu J, Ng TT, *et al*. NTU RGB+D: A large scale dataset for 3D human activity analysis. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 1010–1019.

23 Kim TS, Reiter A. Interpretable 3D human action analysis with temporal convolutional networks. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Honolulu: IEEE, 2017: 1623–1631.

24 Liu J, Shahroudy A, Xu D, *et al*. Spatio-temporal LSTM with trust gates for 3D human action recognition. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 816–833.

25 Tang YS, Tian Yi, Lu JW, *et al*. Deep progressive reinforcement learning for skeleton-based action recognition. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 5323–5332.

26 Fang Z, Zhang XW, Cao TY, *et al*. Spatial-temporal slowfast graph convolutional network for skeleton-based action recognition. IET Computer Vision, 2022, 16(3): 205–217. [doi: 10.1049/cvi2.12080]

(校对责编: 王欣欣)