计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(11):261−267 [doi: 10.15888/j.cnki.csa.008795]

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

基于骨架的快速手势识别模型①

赵　阳, 刘汉超, 董兰芳

(中国科学技术大学 计算机科学与技术学院, 合肥 230026)
通信作者: 董兰芳, E-mail: lfdong@ustc.edu.cn

摘　要: 对于手势识别来说, 骨架数据是一种紧凑且对环境条件稳健的数据模态. 最近基于骨架的手势识别研究多使用深度神经网络去提取空间和时间的信息, 然而这些方法可能存在复杂的计算和大量的模型参数的问题. 为了解决这个问题, 我们提出一种轻量高效的手势识别模型. 该模型使用从骨架序列上计算出的两种空间几何特征, 以及自动学习的运动轨迹特征, 然后只使用卷积网络作为骨干网络实现手势分类. 最终我们的模型参数量最少情况下仅为 0.16 M, 计算复杂度最大情况为 0.03 GFLOPs. 我们在公开的两个数据集上评估了我们的方法, 与其他输入为骨架模态的方法相比, 我们的方法取得了相应数据集上最好的结果.

关键词: 动态手势识别; 卷积神经网络; 动作识别; 骨架数据; 特征提取; 深度学习

引用格式: 赵阳,刘汉超,董兰芳.基于骨架的快速手势识别模型.计算机系统应用,2022,31(11):261–267. http://www.c-s-a.org.cn/1003-3254/8795.html

Fast Skeleton-based Hand Gesture Recognition Model

ZHAO Yang, LIU Han-Chao, DONG Lan-Fang

(School of Computer Science and Technology, University of Science and Technology of China, Hefei 230026, China)

Abstract: Skeleton data is compact and robust to environmental conditions for hand gesture recognition. Recent studies of skeleton-based hand gesture recognition often use deep neural networks to extract spatial and temporal information. However, these methods are likely to have problems such as complicated computation and a large number of model parameters. To solve this problem, this study presents a lightweight and efficient hand gesture recognition model. It uses two spatial geometric features calculated from skeleton sequences and automatically learned motion trajectory features to achieve hand gesture classification with convolutional networks alone as its backbone network. The proposed model has a minimum number of parameters as small as 0.16M and a maximum computational complexity of 0.03 GFLOPs. This method is also evaluated on two public datasets, where it outperforms the other methods that use skeleton modality as input.

Key words: dynamic hand gesture recognition; convolutional neural network (CNN); action recognition; skeleton data; feature extraction; deep learning

在人机交互的场景中手势识别是一个基础研究, 因而在计算机视觉领域被广泛研究. 近年来, 随着廉价的动作传感器[1,2] 的普及和高效的手势估计算法[3−5] 的发展, 基于骨架的手势识别吸引了越来越多研究人员的关注. 骨架数据相比于其他视频数据能够提供更为紧凑和对环境条件稳健的信息, 因此使用骨架实现更为准确的手势识别在人机交互[6]、人体行为理解[7] 和监护系统[8] 等方面具有非常有潜力的应用价值.

在实际应用中, 我们希望手势识别方法尽可能的快速轻量, 方便在低功耗的计算设备上部署应用. 因此,

① 基金项目: 国家重点研发计划重点专项 (2020YFB1313602)
收稿时间: 2022-02-24; 修改时间: 2022-03-28; 采用时间: 2022-04-12; csa 在线出版时间: 2022-07-07

如何使手势识别的方法更加轻量高效是在实际应用中最关键的一个问题.

我们将基于骨架的手势识别研究在特征设计方面大体分为两种: 一种是手工设计的描述符, 一种是基于深度学习自动学习的特征.

基于手工设计的描述符的方法除了直接使用骨架关节点的坐标[9–12] 外, 还会利用关节点之间的联系设计一些几何特征和运动特征. 文献 [13] 使用关节点间距离、关节点间向量夹角、关节点间向量的模比这 3 种骨架的几何特征来描述手势动作, 然后利用长短期记忆网络 (long short-term memory, LSTM) 模型建立出多层级的手势识别模型. 文献 [14] 提出手在空间中运动的全局特征和局部特征, 然后将这两种特征融合后进行线性判别分析特征降维, 最后利用高斯核的支持向量机 (support vector machine, SVM) 进行动态的手势识别分类. 这里, 手部运动的全局特征为整个手部的平移和旋转运动, 局部特征为手指的平移和旋转运动. 文献 [11,12] 提出使用腕关节和手心关节来描述手部的方向, 使用分组的关节坐标描述手部的形状, 运动信息使用关节点的速度作为特征, 最后使用线性核的 SVM 作为分类器实现手势识别. 文献 [15,16] 提出计算每个关节点对的欧式距离和偏移, 以及关节点运动轨迹的双速度作为输入特征, 使用卷积网络提取特征完成手势分类. 虽然相比自动学习的特征, 手工设计的特征更为紧凑高效, 但是因为这些特征都是基于经验设计的特征, 对于难以充分描述的时域特征还有待充分挖掘.

随着机器学习的发展, 基于深度学习的手势识别方法成为研究人员的主要方向. 此类方法往往使用端对端的方式实现手势识别. 由于图卷积在节点分类、图分类上取得了不错的进展, 有很多文献使用图卷积提取骨架特征实现手势识别. 由于图卷积只能提取空域的特征, 但是对于动态手势识别来说, 时域信息也非常重要. 为了解决这个问题, 文献 [17] 使用了一种称为时空图卷积的模型, 可以同时提取骨架的空间特征和时间特征. 文献 [18] 提出使用图卷积和自注意力机制提取关节点和骨骼在空间和时间的特征. 然而除了图卷积比普通的卷积需要更多的计算量外, 有些关节点在骨架图上距离很远, 但在实际运动时却有很强的相关性, 这往往会限制图卷积学习到的骨架特征.

针对以上存在的问题, 我们提出一种轻量快速的手势识别模型. 它使用从骨架的关节点坐标序列经过简单的计算得到的几何特征, 以及自动学习的关节点运动轨迹特征. 然后只使用卷积网络作为骨干网络提取高层次的特征实现手势识别.

1 动态骨架序列的特征

为了简化手势识别模型的复杂度, 我们提出了 3 种从骨架序列中提取的特征来作为模型的输入. 其中前两种特征是每一帧的空间特征, 可以通过简单的几何计算快速获得; 后一种是帧与帧之间的运动特征, 通过浅层的神经网络自动学习而来.

1.1 关节点集距离特征

受到文献 [15] 的启发, 我们采用关节点集距离 (joint collection distance, *JCD*) 作为一个空间信息特征. 关节点距离特征的主要优点具有位置-视角不变性, 并且相比同类的特征[19] 包含更少的元素.

关节点集距离的计算方式是计算骨架的每对关节点的欧式距离, 这样可以得到一个包含所有关节点对距离的对称矩阵. 然后为了压缩重复的信息, 我们只使用矩阵的下三角元素来构成一个特征向量. 具体来说, 我们假设一个骨架序列有 T 帧, 每帧的骨架有 N 个关节点, 那么在第 k 帧的关节点 n 的坐标可以表示为 $J_n^k=(x_n^k,y_n^k,z_n^k)$. 对于第 k 帧的骨架, 它的关节点集距离特征的公式为:

$$JCD^k=\begin{bmatrix} \left\|J_2^kJ_1^k\right\| & & \\ \vdots & \ddots & \\ \left\|J_N^kJ_1^k\right\| & \cdots & \left\|J_N^kJ_{N-1}^k\right\| \end{bmatrix}$$

其中, $\left\|J_i^kJ_j^k\right\|=\sqrt{\left(x_i^k-x_j^k\right)^2+\left(y_i^k-y_j^k\right)^2+\left(z_i^k-z_j^k\right)^2}$ 表示关节点 J_i^k 与关节点 J_j^k 的欧式距离. 在输入模型的时候, JCD^k 特征会扁平化为长度为 $\binom{N}{2}$ 的向量.

1.2 关节点集方向特征

关节点集距离只描述了每对关节点的距离, 而这些距离可以唯一的确定出一个没有方向 (或者旋转) 信息的形状. 但是我们认为骨架的方向信息对手势识别来说也很重要. 比如在前面拍篮球的手势, 和在侧边拍篮球的手势, 这两个动作的唯一的区别就是手部的朝向. 基于以上的观察, 我们又补充了关节点集方向特征作为手势识别模型的输入.

为了描述关节点集的方向, 我们考虑了两种方式

来表示这一信息: 关节点方向锥和关节点方向树.

(1) 关节点方向锥 (joint direction cone, *JDC*)

关节点方向锥是通过从关节点集中选择一个点作为起始点, 然后计算其他点与该点的偏移. 具体来说, 假如我们选择的起始点是手腕节点, 它的编号为 1, 那么第 k 帧的骨架的关节点方向锥特征的计算公式如下:

$$JDC^k = \left(\overrightarrow{J_1^k J_2^k}, \overrightarrow{J_1^k J_3^k}, \cdots, \overrightarrow{J_1^k J_N^k}\right)$$

其中, $\overrightarrow{J_i^k J_j^k} = \left(x_j^k - x_i^k, y_j^k - y_i^k, z_j^k - z_i^k\right)$.

(2) 关节点方向树 (joint direction tree, JDT)

关节点方向树是在骨架的生理结构上定义的. 如图 1 所示, 我们在手部骨架上选择一个关节点作为树的根节点, 然后沿着手部骨架的骨骼逐步计算相邻关节点的偏移.

图 1 关节点方向树

一般来说, 假如骨架是一个无向图, 那么我们可以用类似于拓扑排序的方式得到这个图的关节点方向树特征. 具体算法如算法 1.

算法 1. 关节点方向树特征的算法

输入: 骨架 $S=(V,E)$ 是一个无向图, 第 s 个关节点 $v_s \in V$ 是关节点方向树的根节点

输出: JDT 列表即为关节点方向树特征

```
1. JDT = []
2. S = [s]
3. Visited = [s]
4. while S is not empty do
5.     i = Pop(S)
6.     for all ⟨i, j⟩ ∈ E do
7.         Append(JDT, v_i v_j)
8.         Append(S, j)
9.         Append(Visited, j)
10.    end for
11. end while
```

这样, 可以通过以上算法获得一个唯一的描述关节点方向树的关节点偏移顺序, 进而就可以得到每一帧的 JDT 特征.

1.3 运动轨迹特征

虽然关节点集距离特征和方向特征描述了骨架的空间特征, 但是它不包含全局的运动信息. 当一个手势运动涉及到运动轨迹时, 比如画 V 字型或者 X 字型的手势, 这时只使用空间信息是不够的.

在描述骨架运动时, 许多文献[9,12,15,16,20] 使用速度或者加速度来描述, 并且文献 [15] 提出使用双速度来弥补运动速率不确定的问题. 而速度、加速度以及双速度的计算都是从同一关节点的连续 3 帧上计算得到的. 基于以上的观察, 我们提出一种简单的策略来自动学习运动轨迹特征, 就是使用核大小为 1×3 的 2 维卷积对骨架序列进行特征提取. 其中, 卷积核的 1×3 分别对应这骨架关节点数维度和骨架序列的时间长度. 我们叠加了多个 2 维卷积来更好地学习运动轨迹的语义特征.

2 基于骨架的手势识别

在这一节, 我们介绍一下手势识别的数据预处理流程和手势识别模型的架构.

2.1 骨架序列预处理

给定一段骨架序列, 我们首先使用中值滤波器 (核大小为 3) 来对数据的每一维度降噪, 然后对每个维度使用三次样条插值算法将骨架序列采样为固定长度的序列 (本文使用 32 帧), 并且再次使用中值滤波器降噪.

在送入模型提取特征之前, 我们对骨架序列做了归一化处理. 我们对每一维度减去该维度的均值, 这样既不影响关节点特征的计算, 又可以将骨架序列移动到中点位置起到归一化的效果.

2.2 手势识别模型的架构

我们的手势识别模型的架构如图 2 所示. 相仿文献 [15], 我们使用超参数 filters 来改变模型的参数量, 进而控制模型的计算复杂度. 在实验时我们使用 16、32、64 这 3 个级别展开对比.

对于预处理后的骨架序列, 假设其大小为 $T \times N \times D$, 其中 T 代表序列长度 (时间维度), N 代表骨架关节点数 (点数维度), D 代表关节点信息 (信息维度, 对于 3 维骨架来说 $D=3$). 我们分别计算每帧的关节点集距离特征和关节点集方向特征, 它们是长度为 $\binom{N}{2}$ 和 $D \cdot \binom{N}{1}$ 的特征向量. 对于运动轨迹特征, 那么经过 2 维

卷积后得到的输出大小为 $T\times(N\times\text{filters}/2)$, 即每帧的运动特征长度为 $N\times\text{filters}/2$. 然后这 3 个特征都会经过 3 个 1 维卷积来自动学习更高层的语义特征.

图 2 中, 标记“CNN (1, 2×filters), /2”表示使用的卷积为 1 维卷积 (2 维卷积使用“CNN2”表示), 卷积核大小为 1, 输出通道的数量为 2×filters, 并且使用 Batch-Norm 和 LeakyReLU 激活函数. “/2”表示会对卷积的输出使用核大小为 2 的最大池化层以及在点数维度上的概率为 0.1 的 Dropout. 标记“⊕”表示连接操作, 这里是将提取的 3 种高层语义特征在信息维度上连起来. 标记“Adaptive Max Pooling”表示可适应最大池化操作, 它可以固定最大化池操作的输出维度, 这里我们将输出的时间维度池化掉, 以保证在最后预测分类结果时使用相同的全连接层. 标记“FC (128)”代表输出通道数为 128 的全连接层.

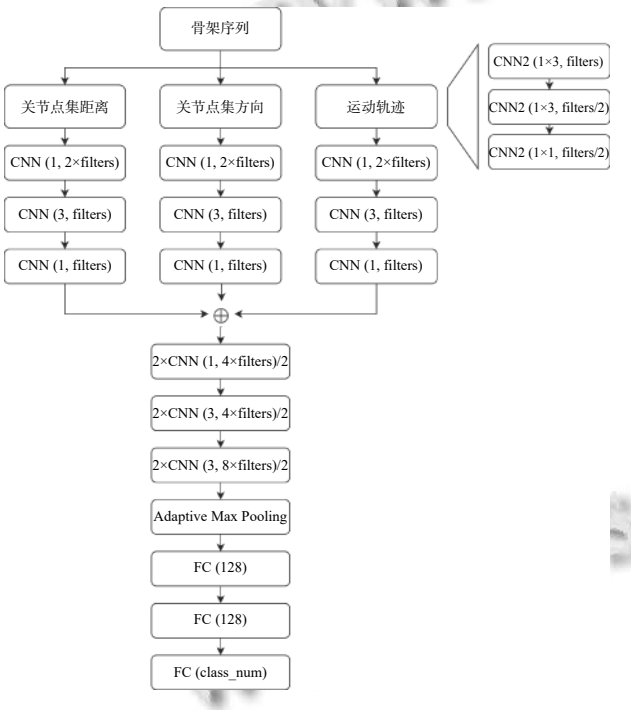

图 2 模型结构

由于我们设计的 3 个特征能够较好地代表骨架运动, 所以不需要使用复杂的循环神经网络、图卷积或者自注意力机制等网络端对端地从关节点坐标上学习抽象特征, 而是只使用普通的卷积网络来做网络骨干即可取得较好的手势分类结果. 在设计卷积时, 我们使用的卷积都只在时间维度上提取特征, 而不在点数维度上提取特征. 这是因为不同的数据集在设计骨架关节点的顺序时, 序号连续的关节点不一定在空间和运动上有很强的相关关系.

最终我们的设计的手势识别模型非常轻量化. 和其他轻量化模型相比, 比如 MobileNetV2[21], 其参数量为 2.1 M 到 6.9 M, 而我们的模型的参数量在最小情况下仅为 0.16 M, 而最大情况下也只有 2.0 M. 虽然我们的模型的参数量非常少, 但是与其他手势识别方法相比性能也毫不逊色, 具体的比较见第 3 节的实验分析部分.

3 实验分析

这一节, 我们分别在公开的数据集 SHREC’17[12] 和 JHMDB[22] 做实验. 我们首先在模型复杂度和两种关节点集方向特征上做实验对比, 最后与其他文献的结果做比较.

3.1 数据集

(1) SHREC’17

SHREC 2017 Track 数据集[12] 是一个公开的动态手势数据集. 该数据集中的手势是从手部运动或者手部形状中定义出来的, 分为粗略的手势和精细的手势. 该数据集一共包含 2 800 个视频段, 分为 14 类手势. 然后每类手势又以两种方式表演: 使用一根手指或者整只手. 该数据集包含深度图像和骨架两种模态的数据, 其中骨架数据的关节点坐标又分为深度图像空间的和三维世界空间. 我们只使用三维世界空间的骨架数据做实验. 该数据集提供了官方的划分, 其中 1 960 段数据用作训练集 (70%), 840 段数据用作测试集 (30%). 在测试时, 为了减少随机效果, 我们随机地选择 3 个不同的随机种子, 最后记录这些结果的平均值和标准差.

(2) JHMDB

关节点标注的人体动作数据集 (joint-annotated human motion data base, JHMDB)[22] 是一个人体动作和姿态的数据集. 该数据集的视频段采自互联网或者电影, 一共包含 21 种动作. 每个视频段为 15 到 40 帧, 并且人工地用人体姿态标注软件为每帧标注了 2D 关节点、缩放、视角等信息. 该数据集提供了官方的 3 种划分, 在测试时我们统计在这 3 个划分上取得结果的平均值和标准差.

3.2 实现细节

因为我们的模型非常轻量, 在训练时我们将整个训练集整合为一个批次送给模型训练, 使用一个 GTX 2080Ti 显卡. 我们选择 Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$)[23]

作为优化器, 使用 ReduceLROnPlateau 作为学习率调整策略将学习率从 10^{-3} 下降到 10^{-5}, 使用交叉熵作为损失函数. 模型代码使用 PyTorch 1.2.0 实现. 训练时没有对数据使用任何增强策略, 也没有使用任何集成策略或者预训练参数来提高性能.

3.3 实验结果

(1) 比较两种关节点集方向特征和模型复杂度

如第 2.2 节所述, 我们使用 filters 的大小来控制模型的复杂度 (我们使用 16、32 和 64 来做对比), 然后在数据集 SHREC'17 和 JHMDB 对比了两种关节点集方向特征的差别, 它们的结果如图 3 和图 4 所示.

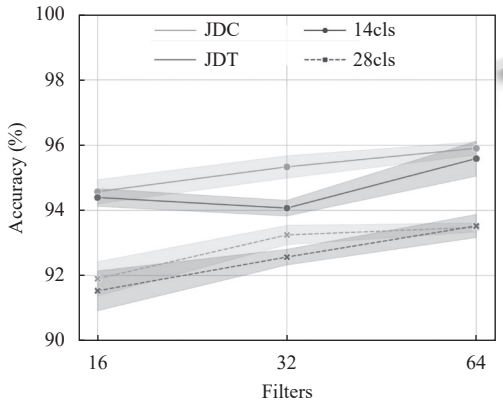

图 3 本文方法在数据集 SHREC'17 的结果

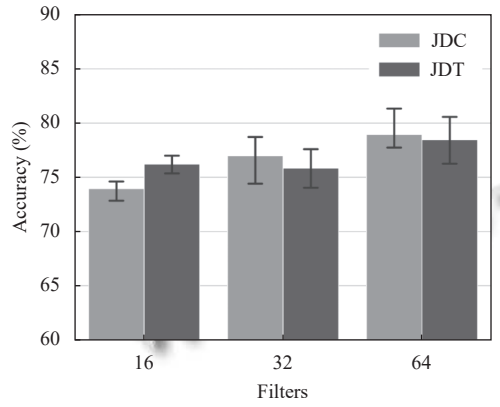

图 4 本文方法在数据集 JHMDB 的结果

在数据集 SHREC'17 上, 对于 14 种手势的分类, 使用 JDC 定义的模型在不同的复杂度下取得结果分别为 94.57% (±0.3%), 95.33% (±0.3%), 95.90% (±0.2%); 使用 JDT 在不同的模型复杂度下取得结果分别为 94.40% (±0.2%), 94.07% (±0.2%), 95.59% (±0.4%). 而在要求更为精细的 28 种手势分类下, JDC 取得的结果为 91.91% (±0.5%), 93.25% (±0.2%), 93.48% (±0.1%); JDT 取得的结果为 91.54% (±0.5%), 92.57% (±0.2%), 93.52% (±0.3%).

在数据集 JHMDB 上, 使用 JDC取得的结果分别为 73.96% (±0.8%), 76.98% (±1.9%), 78.94% (±1.7%); 使用 JDT 取得的结果分别为 76.21% (±0.7%), 75.84% (±1.5%), 78.45% (±1.8%).

从结果我们可以看得出来, 这两种表示关节点集方向特征的方式具有相似的成效, 特别是对数据集 JHMDB 而言. 我们分析, 数据集 JHMDB 的关节点为 2 维坐标, 其运动轨迹没有像 3 维坐标那么有相似性, 所以该数据集的分类性能主要依赖空间特征而非运动特征. 但是总体来说, JDC 略好于 JDT.

(2) 和其他文献的结果对比

我们将这两个数据集的实验结果和其他文献对比, 结果展示在表 1 和表 2 中. 在数据集 SHREC'17 上, 和使用骨架的方法相比, 我们的方法在参数量仅为 0.16 M 时已与之前的最佳结果相近, 我们的方法最佳结果在 14 种手势上提高了 1.3% 的准确率, 在 28 种手势上提高了 1.6% 的准确率. 和使用其他模态的方法相比, 我们的方法在 14 种手势上与最佳方法接近, 而在 28 种手势上略低于最佳的方法. 但是由于我们的模型在提取特征的方式上非常简单直接, 相比于使用点云的最佳方法, 我们的模型的计算速度会快很多. 事实上, 使用点云的方法的计算量为 16.1 GFLOPs, 而我们最佳结果的方法计算量仅为 0.03 GFLOPs. 在数据集 JHMDB 上, 我们的方法在参数量为 0.53 M 时与当前最佳方法的结果接近, 而我们最佳结果提高了 1.7% 的准确率.

表 1 数据集 SHREC'17 上的结果

方法	模态	参数量 (M)	14种手势的准确率 (%)	28种手势的准确率 (%)
Key frames[12]	深度	7.92	82.9	71.9
PointLSTM[24]	点云	1.2	**95.9**	**94.7**
SoCJ+HoHD+HoWR[11]	骨架	—	88.2	81.9
Res-TCN[25]	骨架	—	91.1	87.3
STA-Res-TCN[25]	骨架	5–6	93.6	90.7
ST-GCN[26]	骨架	—	92.7	87.7
DG-STA[18]	骨架	—	94.4	90.7
DD-Net[15]	骨架	1.82	94.6	91.9
本文 (filters=16, JDC)	骨架	0.16	94.6 (±0.3)	91.9 (±0.5)
本文 (filters=32, JDC)	骨架	0.54	95.3 (±0.3)	93.3 (±0.2)
本文 (filters=64, JDC)	骨架	2.00	**95.9 (±0.2)**	93.5 (±0.1)
本文 (filters=64, JDT)	骨架	2.00	95.6 (±0.4)	93.5 (±0.3)

最后, 我们将我们的方法与其他方法进行收敛速度的对比. 我们对比的方法为同样轻量化的手势识别模型 DD-Net[15], 该模型为数据集 SHREC'17 和 JHMDB 的当前最佳方法. 我们对比的代码为该模型公开的 PyTorch 实现版本. 我们在数据集 SHREC'17 的 28 种手势上使用相同的训练方式进行的对比, 图 5 为每个训练轮次模型在测试集上的准确率变化图, 可以观察到我们的方法具有更快的收敛速度.

表 2 数据集 JHMDB 上的结果

方法	参数量 (M)	骨架模态的准确率 (%)
Chained Net[27]	17.5	56.8
EHPI[28]	1.22	65.5
PoTion[29]	4.87	67.9
DD-Net[15]	1.82	77.2
本文 (filters=16, JDC)	0.16	74.0 (±0.8)
本文 (filters=32, JDC)	0.53	77.0 (±1.9)
本文 (filters=64, JDC)	1.95	**78.9** (±1.7)
本文 (filters=64, JDT)	1.95	78.5 (±1.8)

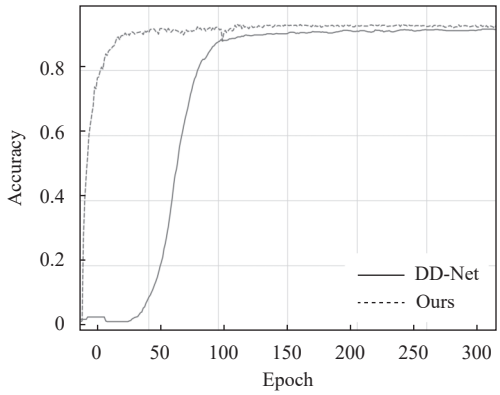

图 5 本文方法在收敛速度上的对比

4 结论与展望

本文提出了一种轻量的基于骨架的手势识别模型. 我们为这个模型提供了 3 种输入特征, 它们分别为关节点集距离特征、关节点集方向特征和运动轨迹特征. 本文为关节点集方向特征提出两种基本等效的描述方式, 它们分别为关节点方向锥 (JDC) 和关节点方向树 (JDT). 我们将本文提出的方法在公开数据集 SHREC'17 和数据集 JHMDB 上进行了评估, 与输入为骨架模态的方法相比, 我们的方法上都取得了最好的识别结果. 在未来, 我们计划扩展我们模型的输入模态支持 RGB 和深度图像数据, 进一步提高模型的准确率.

参考文献

1 Cao Z, Hidalgo G, Simon T, *et al*. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(1): 172–186. [doi: 10.1109/TPAMI.2019.2929257]

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

3 Zhang F, Bazarevsky V, Vakunov A, *et al*. MediaPipe hands: On-device real-time hand tracking. arXiv: 2006.10214, 2020.

4 Panteleris P, Oikonomidis I, Argyros A. Using a single RGB frame for real time 3D hand pose estimation in the wild. Proceedings of 2018 IEEE Winter Conference on Applications of Computer Vision. Lake Tahoe: IEEE, 2018. 436–445.

5 Ge LH, Liang H, Yuan JS, *et al*. Real-time 3D hand pose estimation with 3D convolutional neural networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(4): 956–970. [doi: 10.1109/TPAMI.2018.2827052]

6 Ren Z, Meng JJ, Yuan JS, *et al*. Robust hand gesture recognition with Kinect sensor. Proceedings of the 19th ACM International Conference on Multimedia. Scottsdale: ACM, 2011. 759–760.

7 Wei SE, Tang NC, Lin YY, *et al*. Skeleton-augmented human action understanding by learning with progressively refined data. Proceedings of the 1st ACM International Workshop on Human Centered Event Understanding from Multimedia. Orlando: ACM, 2014. 7–10.

8 Vijayakumari SG, Nandhinee G, Sneha KR. Hand gesture controlled aerial surveillance drone. International Journal of Pure and Applied Mathematics, 2018, 119(15): 897–901.

9 Ghorbel E, Boutteau R, Boonaert J, *et al*. Kinematic spline curves: A temporal invariant descriptor for fast action recognition. Image and Vision Computing, 2018, 77: 60–71. [doi: 10.1016/j.imavis.2018.06.004]

10 Wang HS, Wang L. Beyond joints: Learning representations from primitive geometries for skeleton-based action recognition and detection. IEEE Transactions on Image Processing, 2018, 27(9): 4382–4394. [doi: 10.1109/TIP.2018.2837386]

11 de Smedt Q, Wannous H, Vandeborre JP. Skeleton-based dynamic hand gesture recognition. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops. Las Vegas: IEEE, 2016.

1206–1214.
12 de Smedt Q, Wannous H, Vandeborre JP, *et al*. SHREC'17 track: 3D hand gesture recognition using a depth and skeletal dataset. Proceedings of the 10th Eurographics Workshop on 3D Object Retrieval. Lyon: HAL, 2017. 1–6.
13 郭贺圆. 基于 Kinect 的手势动作识别技术研究 [硕士学位论文]. 长春: 长春理工大学, 2020.
14 缪永伟, 李佳颖, 孙树森. 融合手势全局运动和手指局部运动的动态手势识别. 计算机辅助设计与图形学学报, 2020, 32(9): 1492–1501.
15 Yang F, Sakti S, Wu Y, *et al*. Make skeleton-based action recognition model smaller, faster and better. arXiv: 1907.09658, 2020.
16 Emporio M, Caputo A, Giachetti A. STRONGER: Simple trajectory-based online gesture recognizer. In: Frosini P, Giorgi D, Melzi S, *et al*., eds. Proceedings of Eurographics Italian Chapter Conference—Smart Tools and Apps for Graphics. The Euro-graphics Association, 2021.
17 王焱章. 基于时空图卷积网络的手语翻译 [硕士学位论文]. 南京: 南京邮电大学, 2020.
18 Chen YX, Zhao L, Peng X, *et al*. Construct dynamic graphs for hand gesture recognition via spatial-temporal attention. Proceedings of the 30th British Machine Vision Conference. Cardiff: BMVA Press, 2019. 103.
19 Li CK, Wang PC, Wang S, *et al*. Skeleton-based action recognition using LSTM and CNN. Proceedings of 2017 IEEE International Conference on Multimedia & Expo Workshops. Hong Kong: IEEE, 2017. 585–590.
20 Wang YX, Shi YB, Wei GL. A novel local feature descriptor based on energy information for human activity recognition. Neurocomputing, 2017, 228: 19–28. [doi: 10.1016/j.neucom.2016.07.058]
21 Sandler M, Howard A, Zhu ML, *et al*. MobileNetV2: Inverted residuals and linear bottlenecks. Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 4510–4520.
22 Jhuang H, Gall J, Zuffi S, *et al*. Towards understanding action recognition. Proceedings of 2013 IEEE International Conference on Computer Vision. Sydney: IEEE, 2013. 3192–3199.
23 Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv: 1412.6980, 2014.
24 Min YC, Zhang YX, Chai XJ, *et al*. An efficient PointLSTM for point clouds based gesture recognition. Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 5760–5769.
25 Hou JX, Wang GJ, Chen XH, *et al*. Spatial-temporal attention Res-TCN for skeleton-based dynamic hand gesture recognition. Proceedings of ECCV 2018 Workshops on Computer Vision. Munich: Springer, 2018. 273–286.
26 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 and 30th Innovative Applications of Artificial Intelligence Conference and 8th AAAI Symposium on Educational Advances in Artificial Intelligence. New Orleans: AAAI, 2018. 912.
27 Zolfaghari M, Oliveira GL, Sedaghat N, *et al*. Chained multi-stream networks exploiting pose, motion, and appearance for action classification and detection. Proceedings of 2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 2904–2913.
28 Ludl D, Gulde T, Curio C. Simple yet efficient real-time pose-based action recognition. Proceedings of 2019 IEEE Intelligent Transportation Systems Conference. Auckland: IEE, 2019. 581–588.
29 Choutas V, Weinzaepfel P, Revaud J, *et al*. PoTion: Pose motion representation for action recognition. Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 7024–7033.

(校对责编: 孙君艳)