仪器仪表学报
Chinese Journal of Scientific Instrument
ISSN 0254-3087,CN 11-2179/TH

《仪器仪表学报》网络首发论文

题目：人行为骨架特征识别边缘计算方法研究
作者：游伟，王雪
收稿日期：2020-08-04
网络首发日期：2020-10-15
引用格式：游伟，王雪．人行为骨架特征识别边缘计算方法研究．仪器仪表学报.
https://kns.cnki.net/kcms/detail/11.2179.TH.20201015.1140.020.html

网络首发时间：2020-10-15 13:37:31
网络首发地址：https://kns.cnki.net/kcms/detail/11.2179.TH.20201015.1140.020.html

人行为骨架特征识别边缘计算方法研究

游 伟，王 雪

(清华大学精密仪器系 精密测试技术及仪器国家重点实验室 北京 100084)

摘 要：人行为识别是智能安防监控领域的重要任务之一，传统的行为识别方法将原始图像集中上传至云端服务器，造成网络拥堵、服务器计算压力与计算延迟等问题。本文提出一种采用边缘计算的多时间尺度骨架特征融合行为识别方法。首先，从人行为的原始图像中提取关键点空间位置，构建人行为骨架特征。然后，将骨架特征提取与识别任务部署至多个边缘节点，在各个边缘节点上分别提取不同时间尺度的骨架特征并进行识别，所有边缘节点的识别结果上传至云端服务器并进行融合决策。本方法不仅能够根据准确率需求调整边缘节点数量，实现计算资源动态调度，缓解网络拥堵现象并减轻服务器计算压力，而且显著提高了识别准确率，对智能安防领域的人行为识别具有重要的实际应用价值。

关键词：行为识别；边缘计算；骨架特征；多时间尺度；特征提取

中图分类号：TP391.4 TH702 **文献标识码**：A **国家标准学科分类代码**：520.6040

Study on the edge computing method for skeleton-based human action feature recognition

You Wei, Wang Xue

(State Key Laboratory of Precision Measurement Technology and Instruments, Department of Precision Instrument, Tsinghua University, Beijing 100084, China)

Abstract： Human action recognition is an important task in intelligent security monitoring field. Traditional action recognition method uploads the original image signal to the cloud server in a centralized manner, which brings problems such as network congestion, server computing pressure and computing delay. This paper proposes an edge computing-based multi-time scale skeleton feature fusion method for action recognition. Firstly, the spatial positions of human key points are extracted from the original images of human action and the skeleton features of human action are constructed. Then, the skeleton feature extraction and recognition task are deployed to multiple edge nodes. In the edge nodes the skeleton features on different time scales are extracted and recognized respectively. The results in all the edge nodes are uploaded to the cloud server and fused to make a decision. The proposed method not only achieves the dynamic scheduling of computing resources through adjusting the number of edge nodes according to the accuracy requirements, which relieves network congestion and lightens server computing pressure, but also significantly improves the recognition accuracy, which has important practical application value for the human action recognition in intelligent security field

Keywords： action recognition; edge computing; skeleton feature; multi-time scale; feature extraction

1 引 言

人行为识别在智能安防监控领域具有重要的意义和广泛的应用前景[1-3]，对街道、医院、学校等重要区域的监控视频中人的行为进行检测与分析，能够实现公共安全事件的及时自动报警，保证生产生活环境的安全与稳定。行为识别的主要任务是根据视觉传感器采集的数据推断人的行为动作。目前人行为识别采用的数据源有 RGB 视频、深度图以及骨架数据等。骨架数据是通过姿态估计方法从RGB视频和深度图中提取的关键点空间位置，能够克服 RGB 视频的光照变化、背景干扰和图像噪声等影响，并且具有数据量小的优点，已成为行为识别的重要数据源[1]。

早期的行为识别方法采用人工定义的特征，如李氏群点 [4] 和三维关节直方图 [5]。这些特征利用关键点的几何或统计信息，根据先验知识进行设计，具有明确的含义，但是容易受到骨架噪声、视角变化以及自

收稿日期：2020-08-04 Received Date: 2020-08-04
基金项目：国家重点研发计划项目（2018YFB2003500）资助

身遮挡的影响。近年来，深度学习方法在行为识别领域取得了良好效果，根据分类模型的种类，可分为卷积神经网络方法[6-11]，循环神经网络方法[12-16]、图卷积网络方法[17]以及时间卷积网络（Temporal Convolutional Network，TCN）方法[18]，以上深度学习方法的识别准确率超过了采用人工定义特征的方法。随着识别复杂度的提高，可将目前提高准确率的途径分为两类：（1）改进特征提取与特征表示方法，提高特征表达能力，（2）优化分类模型，提高模型学习能力。

在骨架特征提取方面，卷积神经网络方法将骨架时间序列排列为二维图像，采用各类成熟的图像识别模型对骨架特征进行分类。Ke 等人[7]提出将骨架三维坐标的每个维度编码为一张图片，采用卷积神经网络同时对时间和空间信息进行特征提取。Liu 等人[10]将骨架数据编码为一系列二维图像，采用图像增强方法提高特征表达能力。

在提高模型学习能力方面，Shahroudy 等人[12]针对循环神经网络模型，提出对人体不同部位进行分组识别的长短时记忆网络模型，根据人的运动特点将关键点的位置分组，对每组关键点分别利用长短时记忆网络模型进行特征提取与融合。Liu 等人[13]在长短时记忆网络的时间轴基础上拓展出空间轴，同时捕捉骨架序列的时间和空间信息，有效提高了识别准确率和鲁棒性。Yan 等人[17]将骨架看作时空图，采用图卷积网络模型进行识别，能够充分利用关键点的时间和空间连接关系。Kim 和 Reiter[18]首次将时间卷积网络用于行为识别，设计了具有 10 个时间卷积层的行为识别模型，其识别准确率超过了循环神经网络方法，同时显著减少了模型的参数量。

目前针对骨架特征提取与模型构建的改进方法在一定程度上提高了行为识别的准确率，但是这些方法均采用集中运算的方式，需要将图像数据上传至云端服务器集中处理，随着行为识别在安防领域的应用，目前的方法面临新的问题。随着视觉传感器数量的增加，需要上传至云端服务器的数据量大幅增加，占用大量带宽，造成网络拥堵；识别任务全部由云端服务器完成，计算资源无法动态调度，占用大量的服务器计算资源；此外，为提高分类模型的学习能力，需要增加分类模型的参数量，然而，模型参数量的增加会导致分类模型占用的计算和存储资源过多，进一步增加服务器计算压力。

为解决上述问题，本文提出一种骨架特征行为识别的边缘计算方法。边缘计算是为了解决云计算面临的网络拥堵与云中心计算压力等问题而出现的新型计算方式，通过在靠近采集设备的一端部署具有计算、存储和通信能力的边缘计算节点，实现数据的就近处理，能够有效缓解网络拥堵，并且由边缘节点分担云中心的计算任务，有效降低云中心计算压力[19]。本文将行为识别任务拆分为可在多个边缘节点执行的子任务，从而将骨架数据的特征提取与识别任务部署在边缘计算节点。为实现任务的拆分，本文借鉴了图像分割与识别领域的多尺度特征提取[20]思想，设计了适用于边缘计算环境的多时间尺度骨架特征提取方法，采用多个边缘节点执行不同尺度的特征提取与识别任务，从而将集中式云计算方法转变为采用多边缘节点的边缘计算方法。本文方法的优点如下：（1）采用面向边缘计算的多时间尺度骨架特征融合行为识别方法，利用多个边缘节点提取包含不同信息的特征，降低了服务器的计算压力。（2）多边缘节点识别结果的融合能够显著提高识别准确率，并且可根据准确率要求动态调整边缘节点数量，实现计算资源动态调度。（3）采用适用于边缘计算环境的低参数量模型，并且通过将特征提取与识别任务部署到边缘节点，显著降低了网络传输量。

2 行为识别边缘计算流程

本文提出的行为识别方法如图 1 所示，该方法的系统架构分为视觉传感器、边缘计算节点（以下简称边缘节点）与云端服务器 3 部分。每个部分的具体功能如下：

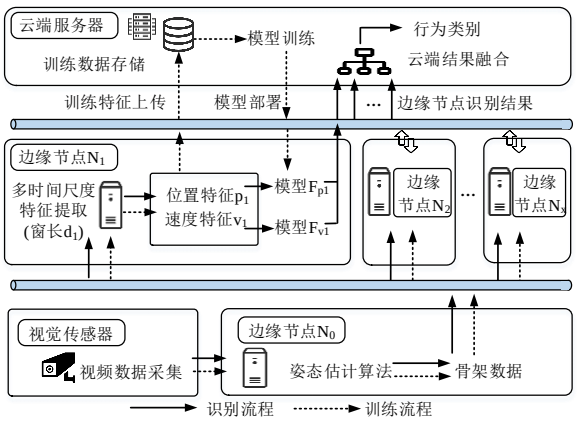

图 1 人行为识别边缘计算流程

Fig. 1 The Edge computing process of human action recognition

（1）网络视觉传感器：执行视频数据采集任务，并将视频数据传输至与其相连的边缘节点 N_0。

（2）边缘节点：N_0 节点执行姿态估计算法，将骨架数据传输至边缘节点 N_1~N_m。边缘节点提取不同时间尺度的位置特征 p 和速度特征 v。

本文的行为识别方法分为训练和识别两个流程。在训练流程中，边缘节点 N_1~N_m 分别将各自提取的骨架特征上传至云端服务器，并接收服务器下发的模型 F_p 和 F_v。在识别流程中，边缘节点将特征输入分类模型并输出识别结果，将识别结果上传至云端服务器。识别流程的特征不上传至云端服务器，从而减小网络传输量，缓解网络拥堵现象。边缘节点的计算任务彼此独立，增加节点能够获取更多尺度的特征，提升识别准确率。在计算资源受限情况下，可调整边缘节点的数量，实现计算资源动态调度。

（3）云端服务器：在训练流程中，云端服务器接收边缘节点 N_1~N_m 上传的特征并构建与各个节点对应的训练数据集，采用该数据集训练分类模型。然后，将训练完成的模型下发至各个边缘节点。在识别流程中，云端服务器接收边缘节点 N_1~N_m 上传的识别结果并将结果进行融合。识别流程的运算任务由多个边缘节点承担，充分利用了边缘节点的计算能力，降低服务器计算压力。

3 边缘节点多时间尺度骨架特征提取

本节介绍边缘节点多时间尺度特征提取算法，各个节点的任务如图 2 所示。视觉传感器采集的视频数据传输到 N_0 节点后，由 N_0 节点利用 OpenPose 等姿态估计工具箱[21]执行姿态估计算法，根据图像序列计算关键点位置，并将骨架序列中每帧的关键点空间位置数据排成特征向量，形成特征向量的时间序列。N_1~N_m 节点采用不同时间窗长度的平均滤波器进行滤波，滤波器在时间窗长度内对骨架数据进行平滑。时间窗长度增加，滤波器对数据的平滑程度加强，能够过滤细节，处理后的特征序列包含全局信息；时间窗长度减小，滤波器对轨迹的平滑程度减弱，可保留关节运动的局部信息。通过调节窗长度，可使滤波后的特征向量序列具有不同的时间尺度，多尺度的特征同时包含了行为的全局和局部信息。以下介绍 N_1~N_m 节点的特征提取算法流程。

人行为的骨架特征可表示为关键点坐标的时间序列。长度为 T 帧的骨架时间序列，第 t 帧的第 j 个关键点表示为 $\boldsymbol{c}_{t,j}=[x_{t,j},y_{t,j},z_{t,j}]^{\mathrm{T}}$，其中 $t\in(1,2,...,T)$，$j\in(1,2,...,N)$，N 表示每帧中关键点的个数。第 t 帧的特征向量为：

$$\tilde{\boldsymbol{p}}_t=[\boldsymbol{c}_{t,1}{}^{\mathrm{T}},...,\boldsymbol{c}_{t,N}{}^{\mathrm{T}}]^{\mathrm{T}} \tag{1}$$

取时间窗长度 d 为奇数，则滤波后的特征向量 $\boldsymbol{p}_{t,d}$ 为时间窗内该时刻特征向量前后共 d 个特征向量的均值：

$$\boldsymbol{p}_{t,d}=\frac{1}{d}\sum_{k=0}^{d-1}\tilde{\boldsymbol{p}}_{t-(d-1)/2+k} \tag{2}$$

将滤波后相邻时刻的特征向量做差分，得到各个关节

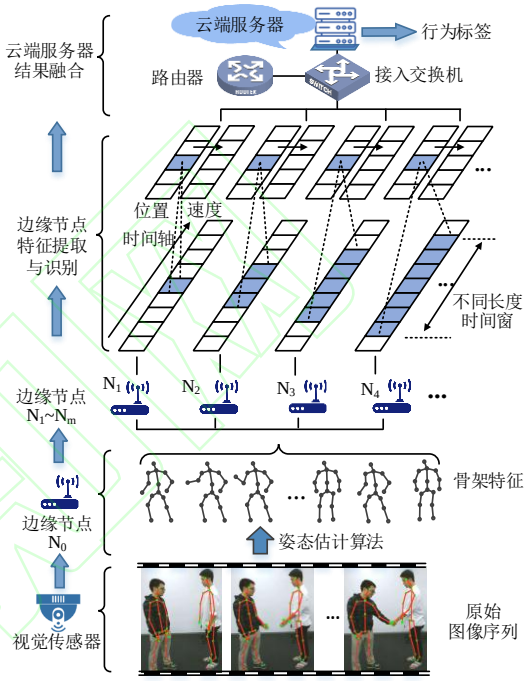

图 2 边缘节点多时间尺度骨架特征提取示意图

Fig. 2 Schematic diagram of the multi-time scale skeleton feature extraction on the edge nodes

点速度的时间序列：

$$\boldsymbol{v}_{t,d}=\boldsymbol{p}_{t,d}-\boldsymbol{p}_{t-1,d} \tag{3}$$

利用上述步骤提取的位置特征与速度特征构建多时间尺度特征，时间窗长度 d 对应的位置特征为：

$$\boldsymbol{P}_d=[\boldsymbol{p}_{1,d},...,\boldsymbol{p}_{T,d}]^{\mathrm{T}}\in\mathbb{R}^{T\times 3N} \tag{4}$$

速度特征为：

$$\boldsymbol{V}_d=[\boldsymbol{v}_{1,d},...\boldsymbol{v}_{T,d}]^{\mathrm{T}}\in\mathbb{R}^{T\times 3N} \tag{5}$$

设时间窗长度集合为 $\boldsymbol{D}=\{d_k\mid k\in[1,m]\}$，共 m 种不同取值，每个边缘节点执行其中一个尺度的滤波任务，获取一组位置和速度特征。m 个节点总共提取 m 组特征。对每一个行为的时间序列样本，多个边缘节点提取到的多尺度特征集合为：

$$\boldsymbol{K}=\{\boldsymbol{P}_d,\boldsymbol{V}_d\mid d\in\boldsymbol{D}\} \tag{6}$$

每组特征是特征向量按时间顺序排列而成的矩阵，矩

阵的每行代表各个时刻滤波后关节点的位置或速度。由于图像采集设备与人的相对位置和角度不确定，由姿态估计方法得到的骨架的视角与空间位置存在较大

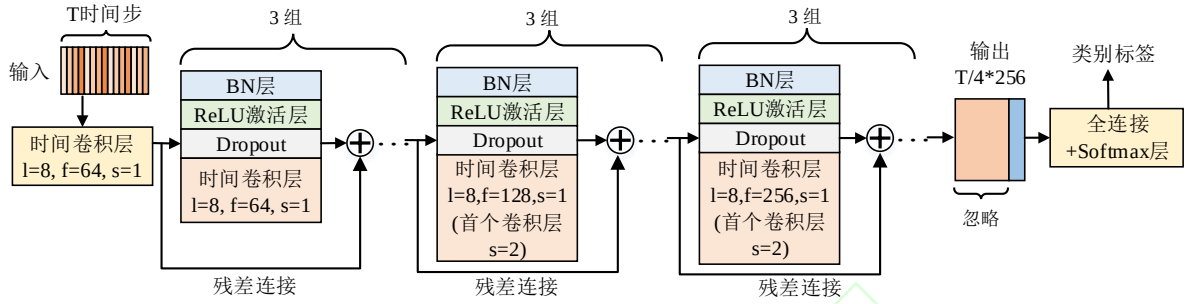

图 3 部署于边缘节点时间卷积网络模型

Fig. 3 The TCN model deployed on the edge nodes

波动，为解决视角变动问题，本文采用文献[10]的方法，在进行多时间尺度特征提取前根据关节点的原始空间位置计算躯干与肩膀连线的空间方向，根据该方向确定目标坐标系，然后将骨架旋转至该坐标系中，并通过将位置和速度特征归一化到[–1,1]区间，解决骨架的空间位置变动造成的样本分布不一致问题。

本方法利用了平均滤波器能够对关节轨迹进行平滑和模糊的特性，其目的并不是滤除噪声。采用不同时间窗处理后的特征所包含的信息彼此互补，并无优劣之分。其他滤波方式，如中值滤波，其主要作用为抑制椒盐噪声，滤波后的位置存在跳变，对本文的特征提取并无优势。高斯滤波在对轨迹进行平滑与模糊方面与平均滤波无本质区别，因此本文采用最简单的平均滤波方法进行多时间尺度特征提取。

4 多边缘节点时间卷积网络融合识别

本节介绍边缘节点上运行的时间卷积网络模型以及边缘节点与云端服务器的融合识别机制。

4.1 边缘节点时间卷积网络模型

边缘节点 N_1~N_m 执行边缘端行为识别任务。每个边缘节点上部署针对位置特征的模型 F_p 和针对速度特征的模型 F_v，两种特征均采用相同结构的时间卷积网络模型。该模型的结构基于文献[18]，在该结构基础上为适应多时间尺度特征识别，采用空洞卷积扩大了网络的感受野，提高模型捕获多尺度上下文信息的能力。如图 3 所示，时间卷积网络模型由输入层、带有空洞卷积和残差连接的时间卷积层以及全连接输出层 3 部分组成。

输入层：设输入特征为 $\boldsymbol{X}_0 \in \mathbb{R}^{T\times N}$。$T$ 代表序列长度，N 代表每个时刻的特征向量维数。输入特征首先通过一个一维卷积层。一维卷积的运算可表示为：

$$\tilde{\boldsymbol{X}}_l = \mathrm{f}(\boldsymbol{X}_{l-1} * \boldsymbol{W}_l + \boldsymbol{b}_l) \tag{7}$$

其中 $\boldsymbol{X}_{l-1}$ 是前一层的输出。$\boldsymbol{W}_l$ 是第 l 层的卷积核，$\boldsymbol{W}_l = \{\boldsymbol{W}_l^{(i)}\}_{i=1}^{F_l}$，$\boldsymbol{W}_l^{(i)} \in \mathbb{R}^{d\times F_{l-1}}$，$\boldsymbol{b}_l \in \mathbb{R}^{F_l}$ 为该层的偏置，“$*$”代表卷积运算。f 表示激活函数，在本文中激活函数全部采用线性整流函数（Rectified Linear Unit, ReLU）。

空洞卷积模块：图 4 展示了一个带有空洞卷积的卷积核的运算过程。卷积核 $\boldsymbol{W}_l^{(i)}$ 表示第 l 层卷积的第 i 个卷积核，在特征图上沿时间轴做一维运动。卷积核相邻行中注入了时间长度为 d（空洞系数）的空洞，使卷积核跳过某些时间步，捕获更长时间的上下文信息。改变空洞系数，能够改变卷积核提取特征所覆盖的时间范围，从而捕获多时间尺度的特征。一维空洞卷积可表达为：

$$F_d(t) = \sum_{j=1}^{k} \boldsymbol{W}^{(i),j} \cdot \boldsymbol{X}^{t-d\cdot(j-1)} \tag{8}$$

输出特征图的第 i 列由卷积核 $\boldsymbol{W}^{(i)}$（为简化表达省略下标 l）生成，$\boldsymbol{W}^{(i),j}$ 表示 $\boldsymbol{W}^{(i)}$ 的第 j 行。$\boldsymbol{X}$ 的上标表示时间步。

空洞卷积模块共 3 组，每组包含 3 个卷积层。卷积层前依次为批归一化层（Batch Normalization，BN），ReLU 激活层，Dropout 层。卷积核的长度、数量以及卷积核的移动步长分别用 l，f，s 表示（图 3）。每个子模块都采用残差连接。每组的首个子模块采用步长为 2 的卷积，并且卷积核数量增加 1 倍，从而使特征图的时间步长缩减为原来的 1/2，特征维数变为上一组卷积的 2 倍，造成残差连接中相加两个特征图维度不一致。为解决此问题，将上个子模块输出的特征图经过卷积核长度为 1，步长为 2 的一维卷积调整为相同维

度后再相加。

每组空洞卷积的 3 个子模块的空洞系数分别为 1, 2, 4。每层感受野如表 1 所示，“模型层”列的 conv1-1 表示第 1 组卷积的第 1 个卷积层，“参数设置”列的参数分别为空洞系数、卷积核移动步长和感受野。

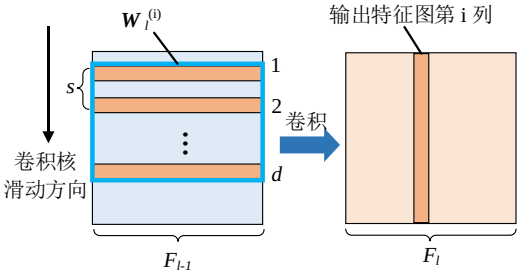

图 4 时间空洞卷积运算过程

Fig. 4 The calculation process in dilated temporal convolution

表 1 TCN 模型各层参数及感受野

Table 1 The parameters and receptive field of various layers in the TCN model

模型层	参数设置	模型层	参数设置
input	1, 1, 8	conv2-2	2, 1, 92
conv1-1	1, 1, 15	conv2-3	4, 1, 148
conv1-2	2, 1, 29	conv3-1	1, 2, 162
conv1-3	4, 1, 57	conv3-2	2, 1, 218
conv2-1	1, 2, 64	conv3-3	4, 1, 330

输出层：空洞卷积模块的输出特征图时间步长为 $T/4$。由于采用空洞卷积，每个时间步的感受野都能覆盖整个时间序列，因此只保留最后一个时间步的特征，舍弃其他时间步的冗余信息。在此之后采用全连接层和 Softmax 层，全连接层的神经元数量与待分类行为种类数相同。模型的输出值为输入样本属于各类行为的概率。

4.2 云端服务器融合识别

按第 3 节的步骤得到多时间尺度的特征集合 K，将该集合中的每一组特征及其标签作为训练数据，采用第 2 节所示流程训练对应的时间卷积网络模型 F_p 与 F_v。训练完成后，云端服务器将模型下发至边缘节点。边缘节点将每组特征分别输入模型进行识别。所有节点将识别结果上传至云端服务器进行融合，最终的识别结果为：

$$y_{fusion}=\arg\max_{i\in\{1,2,\ldots,C\}}\left[\sum_{d\in \boldsymbol{D}}[F_p(\boldsymbol{P}_d)+F_v(\boldsymbol{V}_d)]\right]^{(i)} \tag{9}$$

其中 y 是行为标签，C 是待识别的行为种类数，i 表示向量的第 i 个元素。采用输出结果取平均值的方式进行结果融合。

5 实验结果与分析

为验证多时间尺度特征融合的有效性以及边缘计算方法的优势，本文在南洋理工大学的行为识别 60 类样本数据集（NTU60）[12]和 120 类样本数据集（NTU120）[22]上进行实验，与其他基于骨架的行为识别方法进行对比，并对边缘计算方法的模型参数量和网络传输量进行分析。

5.1 数据集与实验系统设置

（1）实验数据集

NTU60 数据集。该数据集包含由 Kinect V2 相机采集的 56880 个样本，涉及 40 位被测对象的 60 类行为，数据集部分行为如图 5 所示。数据集包含大量的骨架噪声、较大的类内差异以及视角变动，增加了识别难度。该数据集提供两种验证标准：跨对象（Cross-Subject）验证标准和跨视角（Cross-View）验证标准。Cross-Subject 验证标准采用被测对象划分训练集和测试集，训练集为 20 人的 40320 个样本，测试集为其余 20 人的 16560 个样本。Cross-View 验证标准根据拍摄视角划分，训练集为 2 号和 3 号相机采集的 37920 个样本，测试集为 1 号相机采集的 18960 个样本。该数据集为考虑行为识别的实际场景，将单人行为和两人交互行为混杂在一起，将同一时刻两个被测对象的特征拼接为一个特征向量，如果仅有一个被测对象，则将特征重复排列两次。

(a)打电话 (b)打字 (c)掉落物品 (d)握手

(a) Make Phone call (b) Typing (c) Drop item (d) Shake hand

(e)摘帽 (f)摔倒 (g)拳打/掌击 (h)穿衣

(e)Take off a hat (f) Fall down (g)Punch/Slap (h) Put on clothes

图 5 NTU60 数据集样本

Fig. 5 Samples of NTU60 dataset

NTU120 数据集。该数据集包含 114480 个样本，是目前提供骨架数据样本数最多的行为识别数据集之一。在 NTU60 数据集的基础上添加了 60 类行为，包含多达 120 类行为，不同拍摄高度和距离下的拍摄视角增加为 155 个，被测对象增加为 106 人。行为内容

覆盖日常生活、健康、以及两人交互行为，视角变化多样，能够反映行为识别方法在实际应用场景中的应用效果。数据集提供数据集提供跨对象（Cross-Subject）验证标准和跨设置（Cross-Setup）验证标准两种验证标准。Cross-Subject 验证标准采用 53 人的行为样本作为训练集，其余 53 人作为测试集。Cross-Setup 验证标准对所有视角进行编号，偶数编号的视角为训练集，奇数编号的视角为测试集。

（2）实验系统参数设置

边缘节点数量可根据计算资源与准确率需求调节。本实验采用 N_1~N_5 共 5 个边缘节点进行特征提取与识别，各个节点的平均滤波器时间窗长度见表 2。提取后的特征在时间轴上长度不同，统一采用三次样条插值将序列长度调整为 128。时间卷积网络模型权重的初始化采用 He-Normal 方法，采用交叉熵损失函数，dropout 设置为 0.4，batch size 设置为 128。

模型采用随机梯度下降法进行训练，初始学习率 0.01，loss 下降进入平台期后学习率降为 0.001。训练 100 轮后停止，取 5 次训练的平均值作为实验结果。采用以 TensorFlow 为后端的 Keras 深度学习库实现模型训练和特征识别。

表 2 边缘节点对应的时间窗长度

Table 2 The time window lengths corresponding to the edge nodes

边缘节点	N_1	N_2	N_3	N_4	N_5
时间窗	d_1	d_2	d_3	d_4	d_5
窗长度	1	3	5	7	9

（3）实验系统软硬件设置

实验系统由高清网络视觉传感器、边缘节点与云端服务器组成。视觉传感器采用具有以太网接口并支持实时流传输协议（Real Time Streaming Protocol，RTSP）的高清网络 CCD，分辨率 1920×1080，高清码率 2Mbps。采用 5 台 MINI 主机作为边缘节点，采用 1 台机架式服务器作为云端服务器。为模拟边缘节点资源受限的情况，采用 VMWare 虚拟机实现对边缘节点计算、内存及存储资源进行限制。边缘节点与云端服务器的参数对比如表 3 所示。视觉传感器、边缘节点与云端服务器均采用千兆网线与接入交换机连接构成局域网。局域网 IP 地址由路由器统一分配。视觉传感器、边缘节点以及云端服务器可通过 IP 地址访问。

原始图像到骨架数据的姿态估计算法并不在本文讨论范围之内，并且目前已有成熟的骨架数据集，因此本文将已有的数据集保存至边缘节点，采用边缘节点上传数据的方式模拟实际场景下的数据采集与姿态估计过程。

表 3 边缘节点与云端服务器硬件参数

Table 3 The hardware parameters of the edge node and cloud server

平台	硬件	硬件资源		
		计算资源	内存资源	存储资源
边缘节点	MINI 主机	CPU : i5-8400 GPU: RTX2060	8GB	128GB
云端服务器	机架式服务器	CPU: Xeon-4116 GPU: Tesla V100	128GB	1TB

5.2 边缘节点多时间尺度特征融合效果验证

为验证边缘节点多时间尺度特征融合的有效性并与其他行为识别方法对比，将 NTU60 数据集存储在边缘节点 N_0 中，采用由 N_0 节点上传数据的方式模拟视觉传感器的数据采集与姿态估计过程。表 4 展示了每个边缘节点在不同时间尺度下的识别结果。Pos 与 Vel 分别代表位置特征与速度特征，P+V 表示位置与速度特征的识别结果进行融合。

以 CS 验证方式的位置特征为例，不同边缘节点的识别准确率在 77.7%至 78.3%之间，平均值为 77.9%，融合后的结果为 80.1%，比各个节点的平均准确率提升约 2.2%。两种验证标准下，所有边缘节点的位置和速度特征识别结果融合后均超过任何单个节点采用单一特征的识别准确率，分别达到 82.8%与 89.7%。根据以上结果可知，采用多时间尺度特征可将该数据集的识别准确率稳定提升 2%以上。

表 4 NTU60 数据集多时间尺度特征融合实验结果（准确率%）

Table 4 Experiment results of multi-time scale feature fusion on NTU60 dataset (accuracy %)

边缘节点	Cross-Subject 验证标准（%）			Cross-View 验证标准（%）		
	Pos	Vel	P+V	Pos	Vel	P+V
N_1	78.3	73.0	81.1	87.6	83.2	87.9
N_2	78.1	73.9	80.7	87.2	84.3	87.9
N_3	77.9	74.1	81.0	86.9	84.9	87.5
N_4	77.7	73.9	80.8	87.1	83.3	87.4
N_5	77.7	73.6	80.7	87.0	83.4	87.1
融合	80.1	76.2	**82.8**	88.5	86.3	**89.7**

5.3 识别准确率比较

本文的多时间尺度骨架特征融合方法与其他方法在 NTU60 数据集上的实验结果对比如表 5 所示。表格对比了采用原始位置特征的单个边缘节点（Single Node）以及采用多边缘节点特征融合（Multiple Nodes）的结果。参比方法包括采用循环神经网络的时空

ST-LSTM 方法[13]和采用卷积神经网络的 JTM 方法[6]。

由表 5 可知，本文方法的准确率超过了循环神经网络方法和卷积神经网络方法，本文采用的识别模型是在时间卷积网络基础上通过采用空洞卷积等模型改进方法得到，在采用原始位置特征和单个时间卷积网络的情况下，Cross-Subject 和 Cross-View 的识别准确率分别为 78.3%与 87.6%。在此基础上，采用多时间尺度骨架特征融合方法，可将准确率提高到 82.8%和 89.7%，验证了多边缘节点的融合能够提高识别准确率。

循环神经网络在对特征进行识别时，需要将每个时刻的特征依次输入模型，模型对下一时刻特征的处理依赖于上一时刻的运算结果。而卷积神经网络可对所有时刻的特征进行并行处理，能够充分利用计算设备的运算能力，在运算实时性方面具有优势。采用卷积神经网络模型的方法取得了良好的效果，但是为了提高模型学习能力增加了对计算和存储资源的要求。因此，本文方法在准确率和模型参数量方面都具有明显优势，模型参数量的比较见 5.5 节。

表 5 NTU60 数据集实验结果及比较

Table 5 Experiment results and comparison on NTU60 dataset

方法	Cross-Subject 验证标准（%）	Cross-View 验证标准（%）
ST-LSTM[13]	69.2	77.7
JTM[6]	76.3	81.1
Single Node	78.3	87.6
Multiple Nodes	**82.8**	**89.7**

在 NTU120 数据集上的对比实验结果如表 6 所示。参与对比的方法的准确率来源于文献[22]，由于此数据集增加了行为类别、人数以及拍摄视角，区分同类与异类行为的难度增加，识别准确率相比 NTU60 数据集大幅下降。Clip+CNN+MTLN 方法[7]采用卷积神经网络模型，其识别准确率与采用循环神经网络的 ST-LSTM 方法大致相同。本文方法在采用单个节点时，Cross-Subject 与 Cross-Setup 验证方式下识别准确率为 70.6%与 71.9%。采用多边缘节点的多时间尺度特征后，识别准确率分别提升 2.6%与 2.4%。本文方法的准确率明显超过了其他两种具有代表性的循环神经网络和卷积神经网络方法，证明了本文方法在识别准确率上与传统的集中式运算方法相比具有优势。

表 6 NTU120 数据集实验结果及比较

Table 6 Experiment results and comparison on NTU120 dataset

方法	Cross-Subject 验证标准（%）	Cross-Setup 验证标准（%）
ST-LSTM[13]	55.7	57.9
Clip+CNN+MTLN[7]	58.4	57.9
Single Node	70.6	71.9
Multiple Nodes	**73.2**	**74.3**

5.4 计算资源动态调度分析

现有的行为识别方法大多仅针对算法本身，并且采用集中运算模式，模型所占用运算资源与识别准确率固定，在识别任务的资源调度方面缺少灵活性。本文共采用 5 个边缘节点提取骨架的多时间尺度特征，每个边缘节点运行 2 个识别模型，分别用于识别位置特征与速度特征。本文方法可从两方面进行计算资源的动态调度。首先，由于边缘节点 N_1~N_m 彼此独立，每个节点均可单独完成识别任务，因此可根据准确率要求动态调整节点的数量；此外，每个节点的 2 个识别模型彼此独立，根据模型种类，可分为仅运行位置特征识别模型（Pos）、仅运行速度特征识别模型（Vel）以及同时运行以上两种模型（P+V）共 3 种模式。

在计算资源受限的情况下，本文方法可选择启用或关闭特定的节点，每个节点可选择启用的模型种类，从而显著减小运算资源的消耗。而在识别准确率要求提高时，可充分调动已有资源进行运算。采用不同节点与不同模型种类的识别准确率如图 6 所示，横轴为边缘节点的数量，在 Pos、Vel 和 P+V 模式下，从采用节点 N_1 开始依次增加节点个数，直至采用全部 5 个节点。由图中曲线可知，随着边缘节点数量的增加，以上 3 种模式下识别准确率均获得稳定提升。

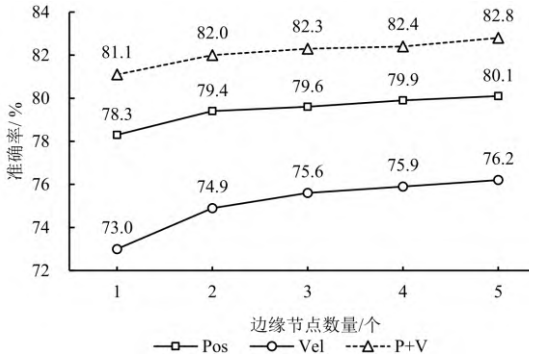

图 6 边缘节点数量对识别准确率的影响（NTU60 数据集，CS 验证标准）

Fig. 6 The influence of the number of edge nodes on the recognition accuracy (NTU60 dataset, Cross-Subject verification criteria)

根据模型数量可估算，当准确率要求最低时，仅需启用 1 个节点的 1 个模型，准确率要求最高时，可启用所有节点的 5×2=10 个模型。最低准确率要求下

的计算量约为最高准确率的 1/10。可根据识别准确率的要求调整边缘节点的数量，从而实现边缘节点计算资源的动态调度。

5.5 模型参数量及网络传输量

在边缘计算环境下，边缘节点的计算资源受限。模型参数量增加将导致模型无法在边缘节点运行。因此，降低模型参数量对行为识别的广泛应用具有重要意义。表 7 列出了在 NTU60 数据集上与本文准确率相近的方法，本文与 ESV 和 JTM 方法均采用了多个模型进行融合，因此模型总参数量为单个模型参数量与模型数量的乘积。参比方法的模型大小根据原文给出的网络结构以单精度浮点数计算得到。本文采用的单个时间卷积网络模型参数量仅 7.3MB，模型总参数量在参比方法中最少，能够显著降低将模型部署到边缘节点时所需的存储空间，在单个模型参数和模型总参数量上都具有优势。

表 7 模型参数数量与准确率比较

Table 7 Comparison of the number of model parameters and accuracy

方法	单个模型参数量 (MB)	模型数量	模型总参数量 (MB)	准确率 (%)
ESV[10]	>200	10	>2000	80.0
JTM[6]	≈233	3	≈700	76.3
Clip+CNN+MTLN[7]	≈90	--	≈90	79.6
本文方法	7.3	10	73	82.8

本文方法将特征提取与识别任务部署至边缘节点，仅需要上传识别结果，而传统方法需要将原始视频数据上传至云端服务器。因此，本文的方法能够降低网络传输量。以分辨率为 1280×720，时间长度为 3s，帧率 30fps 的视频为例，其码率典型值为 2Mbps，识别该段视频的行为所需上传的数据量约为 2Mbps×3s=6Mb。采用本文边缘计算方式后，N_1~N_5 每个边缘节点需上传的识别结果总大小为 10×60×32b＝18.75kb（5 个节点共运行 10 个模型，以识别 60 类行为为例，边缘节点识别结果用 32 位单精度浮点数表示），显著降低了识别所需的网络传输量。

6 结论

本文提出一种人行为骨架特征识别的边缘计算方法，该方法能够将行为识别任务拆分为可在边缘节点上运行的若干子任务。本文在多个边缘节点上分别提取不同时间尺度的骨架特征，并采用时间卷积网络对每组特征进行识别并将结果进行融合。实验结果表明，该方法不仅提高了行为识别的准确率，而且能够根据准确率要求实现计算资源的动态调度，有效缓解传统方法集中上传图像数据造成的网络拥堵现象，减轻服务器计算压力，对安防领域行为识别具有重要的实际应用价值。

参考文献

[1] AGGARWAL J K, XIA L. Human activity recognition from 3D data: A review[J]. Pattern Recognition Letters, 2014, 48: 70-80.

[2] 戴鹏, 王雪, 谈宇奇, 等. 面向行人检测的异构视觉传感网络自适应标定[J]. 仪器仪表学报, 2016, 37(3): 683-689.
DAI P, WANG X, TAN Y Q, et al. Self-adaptive calibration of hybrid visual sensor networks for pedestrian detection[J]. Chinese Journal of Scientific Instrument, 2016, 37(3): 683-689.

[3] 谈宇奇, 王雪, 林奎成. 基于视觉压缩感知的传感网络行人目标辨识方法[J]. 仪器仪表学报, 2014, 35(11): 2433-2439.
TAN Y Q, WANG X, LIN K CH. Visual compressive sensing-based pedestrian identification in sensor networks. Chinese Journal of Scientific Instrument, 2014, 35(11): 2433-2439.

[4] VEMULAPALLI R, ARRATE F, CHELLAPPA R. Human action recognition by representing 3D skeletons as points in a lie group[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2014: 588-595.

[5] XIA L, CHEN C C, AGGARWAL J K. View invariant human action recognition using histograms of 3D joints[C]. IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2012: 20-27.

[6] WANG P, LI Z, HOU Y, et al. Action recognition based on joint trajectory maps using convolutional neural networks[C]. Proceedings of the 24th ACM international conference on Multimedia. 2016: 102-106.

[7] KE Q, BENNAMOUN M, AN S, et al. A new representation of skeleton sequences for 3D action recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2017: 3288-3297.

[8] LIU J, SHAHROUDY A, WANG G, et al. Skeleton-based online action prediction using scale selection network[J]. IEEE Transactions on Pattern Analysis and Machine

Intelligence, 2019, 42(6): 1453-1467.

[9] LIU M, YUAN J. Recognizing human actions as the evolution of pose estimation maps[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2018: 1159-1168.

[10] LIU M, LIU H, CHEN C. Enhanced skeleton visualization for view invariant human action recognition[J]. Pattern Recognition, 2017, 68: 346-362.

[11] HUANG Z, WAN C, PROBST T, et al. Deep learning on lie groups for skeleton-based action recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2017: 6099-6108.

[12] SHAHROUDY A, LIU J, NG T T, et al. NTU RGB+D: A large scale dataset for 3D human activity analysis[C]. IEEE conference on computer vision and pattern recognition. 2016: 1010-1019.

[13] LIU J, SHAHROUDY A, XU D, et al. Spatio-temporal LSTM with trust gates for 3D human action recognition[C]. European Conference on Computer Vision. 2016: 816-833.

[14] LIU J, WANG G, HU P, et al. Global context-aware attention LSTM networks for 3D action recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1647-1656.

[15] DU Y, WANG W, WANG L. Hierarchical recurrent neural network for skeleton based action recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1110-1118.

[16] WANG H, WANG L. Learning content and style: Joint action recognition and person identification from human skeletons[J]. Pattern Recognition, 2018, 81: 23-35.

[17] YAN S, XIONG Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition[C]. Thirty-second AAAI Conference on Artificial Intelligence. 2018.

[18] KIM T S, REITER A. Interpretable 3D human action analysis with temporal convolutional networks[C]. IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017: 1623-1631.

[19] WANG X, HAN Y, LEUNG V C M, et al. Convergence of edge computing and deep learning: A comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2020, 22(2): 869-904.

[20] PONT-TUSET J, ARBELAEZ P, BARRON J T, et al. Multiscale combinatorial grouping for image segmentation and object proposal generation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 39(1): 128-140.

[21] CAO Z, SIMON T, WEI S E, et al. Realtime multi-person 2D pose estimation using part affinity fields[C]. IEEE Conference on Computer Vision and Pattern Recognition. 2017: 7291-7299.

[22] LIU J, SHAHROUDY A, PEREZ M L, et al. NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019. doi: 10.1109/TPAMI.2019.2916873.

作者简介

游伟，2016年于天津大学获得学士学位，现为清华大学精密仪器系博士研究生，主要研究方向为边缘计算与行为识别。
E-mail: youw16@mails.tsinghua.edu.cn

You Wei received B.Sc. degree in 2016 from Tianjin University. Now, his is a Ph.D candidate in Department of Precision Instrument, Tsinghua University. His current research interest includes edge computing and action recognition.

王雪（通讯作者），1994 年于华中理工大学获博士学位，现为清华大学精密仪器系教授，博士生导师。主要研究方向为精密测试与传感技术和无线传感网络。
E-mail: wangxue@mail.tsinghua.edu.cn

Wang Xue (corresponding author) received his Ph.D. degree from Huazhong University of Science and Technology in 1994. He is a professor and doctoral supervisor in Department of Precision Instrument, Tsinghua University. His research interests are precision measurement and sensor technology, and wireless sensor networks.