

# 基于 BP 神经网络的摔倒检测算法



周立鹏<sup>1</sup> 孟利民<sup>1</sup> 周磊<sup>1</sup> 蒋维<sup>2</sup> 董建平<sup>3</sup>

1 浙江工业大学信息工程学院 杭州 310023

2 浙江树人大学信息科技学院 杭州 310015

3 台州市行政学院信息处 浙江 台州 318000

(2111703015@zjut.edu.cn)

**摘要** 摔倒对于老年人来说是一个十分严重的问题,实时检测老年人是否摔倒对于减轻摔倒造成的伤害具有重要意义。为此,文中提出了一种基于 BP 神经网络的摔倒检测算法。该算法采用佩戴于腰部的六轴传感器(MPU6050)来采集人体运动数据,使用简单的统计学方法对数据进行特征提取,并以提取到的特征为 BP 神经网络的输入神经元,用 Levenberg-Marquardt 算法训练神经网络模型,使其能够实现摔倒检测的功能。实验结果表明,该算法可以较好地识别摔倒,其准确率可以达到 99.55%。

**关键词** BP 神经网络;模式识别;摔倒检测;特征提取;可穿戴式设备

**中图分类号** TP183

## Fall Detection Algorithm Based on BP Neural Network

ZHOU Li-peng<sup>1</sup>, MENG Li-min<sup>1</sup>, ZHOU Lei<sup>1</sup>, JIANG Wei<sup>2</sup> and DONG Jian-ping<sup>3</sup>

1 College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China

2 College of Information Science and Technology, Zhejiang Shuren University, Hangzhou 310015, China

3 Information Department, Taizhou Administrative College, Taizhou, Zhejiang 318000, China

**Abstract** Fall is a very serious problem for the elderly. Real-time detection of whether the elderly fall or not is of great significance to reduce the injury caused by falling. Therefore, a fall detection algorithm based on BP neural network is proposed in this paper. The algorithm collects human motion data with a six-axis sensor (MPU6050) worn at the waist, and uses a simple statistical method to extract features from the data. The extracted features are used as input neurons of BP neural network, and Levenberg-Marquardt algorithm is used to train the neural network model, so that it can realize the function of fall detection. Experimental results show that the algorithm can recognize falls well and the accuracy can reach 99.55%.

**Keywords** BP neural network, Pattern recognition, Fall detection, Feature extraction, Wearable equipment

随着人们生活水平以及医疗卫生条件的提高,人均寿命不断延长,人口老龄化已经成为国内以及国际社会普遍存在的问题。与此同时,老年人口迅速增加,给国内的医疗保障系统带来了极大的压力<sup>[1]</sup>。老年人由于身体机能下降,极易发生摔倒,而在当今社会快节奏的生活工作背景下,子女工作繁忙,老年人大多处于无人看护的状态,一旦发生摔倒,通常会长时间没人发现,从而造成十分严重的后果。因此,设计一种摔倒检测算法,实时监控老年人是否摔倒,并在摔倒后及时报警,就可以缩短从摔倒到就医的时间,从而减小摔倒造成的伤害。

目前国内外的摔倒检测算法主要分为 3 类:基于视频图像处理[2-3]的摔倒检测算法、基于外部传感器的摔倒检测算法<sup>[4]</sup>、基于可穿戴传感器的摔倒检测算法<sup>[5-8]</sup>。其中基于视频和基于外部传感器的摔倒检测算法需要在生活环境中布置摄像头或一系列其他传感器(如红外传感器、压力传感器等),它们只能检测传感器范围内的摔倒,一旦人离开该区域则无法

检测摔倒,具有一定的局限性。而基于可穿戴式传感器的摔倒检测算法由于其佩戴方便,可以实时检测摔倒,是目前使用得最多的摔倒检测方法。文献[5]提出了一种基于三轴加速度传感器的老年人摔倒检测算法,通过确定阈值的方法区分摔倒与日常行为,这种方法计算量低、实现简单,但是在实际测试中,由于日常行为中跳、跑等动作的加速度有时会超过阈值,因此误报率较高。文献[6]通过手机内置重力传感器采集人体运动数据,对数据进行特征提取和奇异值分解,通过使用径向基核函数的支持向量机对数据进行分类,但是由于提取的特征不够全面,其对于坐下、躺下、蹲下等行为的误判较高。文献[7]提出了一种基于奇异值分解的一维卷积神经网络(Convolutional Neural Networks, CNN)摔倒检测算法,对加速度数据进行奇异值分解,并将其输入一维卷积神经网络,但是由于对数据进行了降维处理,并且训练迭代次数较小,导致算法准确率不高。文献[8]提出一种基于决策二叉树的固定

本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家自然科学基金项目(61871349);浙江省基础公益项目(LY18F010024, LQ19F010013)

This work was supported by the National Natural Science Foundation of China (61871349) and Basic public welfare projects in Zhejiang (LY18F010024, LQ19F010013).

通信作者:孟利民(mlm@zjut.edu.cn)

阈值的摔倒检测算法,该方法可以区分快跑与摔倒,但是由于其算法中采用的特征为手动提取,只能使用一些比较明显的特征来分类快跑与摔倒,导致其识别准确率不高。综上所述,基于可穿戴传感器的摔倒检测算法仍然存在准确率不高、容易误报等问题,而且在实验过程中采集的活动类型不够全面更会加剧误报和误判等问题。

为了提高摔倒检测算法的准确率,减少误报,本文提出了一种基于 BP 神经网络的摔倒检测算法。BP 神经网络是目前应用得最广泛的一种多层前馈神经网络模型,主要应用于函数逼近<sup>[9]</sup>、模式识别、数据压缩<sup>[10]</sup>等方面,本文需要对人体摔倒行为和日常行为进行区分,这正是 BP 神经网络在模式识别领域的应用。本文算法中可穿戴设备佩戴于腰部<sup>[11]</sup>,通过六轴传感器 MPU6050 采集人体运动的加速度和角速度数据,并通过蓝牙 MCU 芯片 DA14580 将数据上传到手机 APP,将采集到的加速度、角速度数据进行特征提取,将特征提取后的数据的 70% 用于训练 BP 神经网络模型,15% 用于验证,15% 用于测试,训练完成后得到的 BP 神经网络模型就可以实现摔倒识别功能。

## 1 数据采集

本文用于数据采集的可穿戴设备的系统框图如图 1 所示,将设备佩戴于腰间,通过六轴传感器 MPU6050 采集人体三轴加速度与三轴角速度数据,并通过蓝牙芯片 DA14580 将数据通过蓝牙上传到手机 APP。图 2 为设备实物图,图 3 为设备佩戴位置示意图。本文的坐标轴定义如下:人体正常站立时,向前为 X 轴正方向,向左为 Y 轴正方向,向上为 Z 轴正方向。

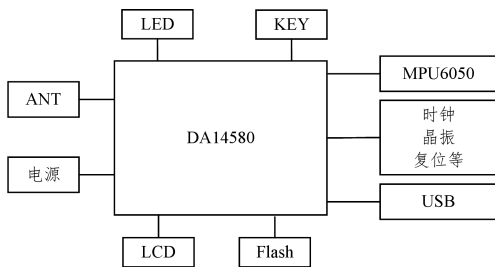


图 1 可穿戴设备系统框图

Fig. 1 System block diagram of wearable equipment

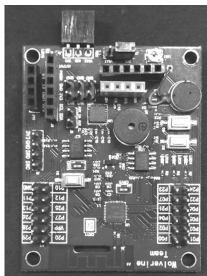


图 2 设备实物图

Fig. 2 Physical drawing of equipment

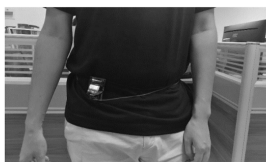


图 3 设备佩戴位置示意图

Fig. 3 Diagram of device wearing position

为了建立摔倒与日常行为运动数据集,本文邀请了 10 名志愿者,要求他们在实验室进行 10 项非摔倒行为和 7 项摔倒行为的实验,并采集了实验过程中的加速度和角速度数据。实验进行的活动如表 1 所列。在采集时,设备每 0.02 s 发送一次加速度和角速度数据,本次实验中每条数据的长度为 2 s,每条数据中有 100 个加速度和角速度数据。本次实验一共采集了 1360 条数据,其中 800 条为非摔倒数据,560 条为摔倒数据。

表 1 实验行为表

非摔倒行为	摔倒行为
静止	向前直接摔倒
走路	向前跪地摔倒
跑步	向后坐地摔倒
下楼梯	向后直接摔倒
上楼梯	向右摔倒
坐下	向左摔倒
坐的状态下起立	靠墙摔倒
躺下	
躺的状态下起立	
跳跃	

## 2 特征提取

本文算法需要对传感器得到的三轴加速度数据 ( $a_x, a_y, a_z$  分别为 X 轴、Y 轴、Z 轴方向的加速度) 和三轴角速度数据 ( $\omega_x, \omega_y, \omega_z$  分别为绕 X 轴、Y 轴、Z 轴旋转的角速度) 进行预处理,并进行特征提取。首先对三轴加速度和角速度数据求和得到和加速度,公式如下:

$$\begin{cases} a = \sqrt{a_x^2 + a_y^2 + a_z^2} \\ \omega = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2} \end{cases} \quad (1)$$

其中,  $a_x, a_y, a_z$  分别为 X 轴、Y 轴、Z 轴方向的加速度,  $a$  为和加速度,  $\omega_x, \omega_y, \omega_z$  分别为绕 X 轴、Y 轴、Z 轴旋转的角速度,  $\omega$  为和角速度。

将加速度进行连续的积分可得到速度数据,对得到的速度信号积分可得到位移数据,由于采集到的数据是离散的,本文算法将积分更换为求和,公式如下:

$$\begin{cases} v_n = \sqrt{(\sum_{i=1}^n a_{xi} \Delta t)^2 + (\sum_{i=1}^n a_{yi} \Delta t)^2 + (\sum_{i=1}^n a_{zi} \Delta t)^2} \\ s_n = \sqrt{(\sum_{i=1}^n a_{xi} \Delta t^2)^2 + (\sum_{i=1}^n a_{yi} \Delta t^2)^2 + (\sum_{i=1}^n a_{zi} \Delta t^2)^2} \end{cases} \quad (2)$$

其中,  $v_n$  为第  $n$  个点的速度,  $a_{xi}, a_{yi}, a_{zi}$  分别为第  $i$  个点的 X 轴、Y 轴、Z 轴方向的加速度,  $s_n$  表示第  $n$  个点的位移,  $\Delta t$  为时间间隔,因为本文中每条数据的长度为 2 s,一共 100 个点,所以取  $\Delta t = 0.02$  s。

之后,将和角速度数据  $\omega$ 、和加速度数据  $a$ 、速度数据  $v$  以及位移数据  $s$  等 4 组数据分别求最大值、最小值、范围、方差、均方差、平均值,如此,每组数据得到 24 个特征值。

## 3 BP 神经网络模型的建立与训练

本文算法所用的 BP 神经网络模型为典型的三层 BP 神经网络,由输入层、隐含层、输出层组成,层与层之间采用全连接的方式。本文算法中,输入层由 24 个神经元组成,即为前一步中提取的 24 个特征值,隐含层由 10 个神经元组成,隐含层神经元个数由经典算法<sup>[12]</sup>决定,公式如下:

$$m = \sqrt{n+l} + \alpha \quad (3)$$

其中,  $m$  为隐含层神经元个数,  $n$  为输入层神经元个数,  $l$  为输出层神经元的个数,  $\alpha$  为 1 到 10 之间的常数, 实际应用中,  $\alpha$  的值根据实际效果决定, 本次实验中发现当隐含层神经元个数取 10 时神经网络迭代次数较少, 因此隐含层由 10 个神经元组成。

由于需要识别摔倒与非摔倒数据, 对数据进行二分类, 因此输出层由两个神经元组成。本文算法构建的 BP 神经网络的结构如图 4 所示。

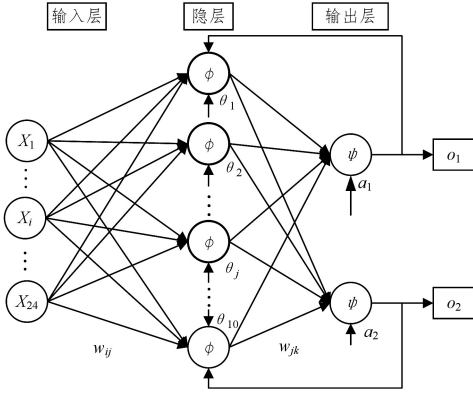


图 4 BP 神经网络的结构图

Fig. 4 Structural chart of BP neural network

图 4 中,  $x_i$  表示输入层第  $i$  个神经元的输入,  $i=1, \dots, 24$ ;  $w_{ij}$  表示输入层第  $i$  个神经元到隐含层第  $j$  个神经元之间的权值,  $j=1, \dots, 10$ ;  $\theta_j$  表示隐含层第  $j$  个神经元的偏置;  $\phi$  表示隐含层的激活函数;  $w_{jk}$  表示隐含层第  $j$  个神经元到输出层第  $k$  个神经元之间的权值;  $a_k$  表示输出层第  $k$  个神经元的偏置,  $k=1, 2$ ;  $\psi$  表示输出层激活函数。  $o_k$  表示输出层第  $k$  个神经元的输出。

BP 神经网络的训练过程包括信号的正向传播与误差的反向传播两个方面, 计算神经网络的输出时是从输入到输出方向进行的, 而更新神经网络中各神经元的权值与偏置时是从输出到输入方向进行的。训练开始前, 初始的权值和偏置都是随机生成的。

3.1 信号的前向传播

隐含层第  $j$  个神经元的输入  $net_j^h$  为:

$$net_j^h = \sum_{i=1}^{24} w_{ij} x_i + \theta_j \tag{4}$$

隐含层第  $j$  个神经元的输出  $out_j^h$  为:

$$out_j^h = \phi(net_j^h) \tag{5}$$

输出层第  $k$  个神经元的输入  $net_o_k$  为:

$$net_o_k = \sum_{j=1}^{10} w_{jk} \cdot out_j^h + a_k \tag{6}$$

输出层第  $k$  个神经元的输出  $out_o_k$  为:

$$out_o_k = \psi(net_o_k) \tag{7}$$

在本文算法中, 隐含层的激活函数为 Sigmoid 函数, 公式如下:

$$\phi(x) = \frac{1}{1 + e^{-x}} \tag{8}$$

输出层的激活函数为 Softmax 函数, 公式如下:

$$\psi(z)_k = \frac{e^{z_k}}{\sum_{n=1}^N e^{z_n}} \tag{9}$$

其中,  $k=1, \dots, N$ , 本文中  $N=2$ 。

3.2 误差的反向传播

误差的反向传播, 就是由输出层开始逐层计算各层神经

元的输出误差, 使用训练算法来调节各层的权值和偏置, 使修改后的神经网络的输出可以接近期望值。使用不同的训练算法会影响训练的效果和速度。本文采用的训练算法为 Levenberg-Marquardt 算法<sup>[13]</sup>, 该算法将高斯-牛顿法与 BP 神经网络传统的梯度下降法相结合, 提高了网络的收敛速度, 减少了网络训练的迭代次数, 而且训练出来的模型精度更高。

首先定义误差函数, 对于总体  $P$  个样本, 误差函数定义为  $E(w)$ 。

$$E(w) = \frac{1}{2} \sum_{p=1}^P (\mathbf{Y}_p - \mathbf{Y}_p')^2 = \frac{1}{2} \sum_{p=1}^P e_p^2(w) \tag{10}$$

其中,  $\mathbf{Y}_p$  表示第  $p$  个样本的期望输出向量,  $\mathbf{Y}_p'$  表示第  $p$  个样本的实际输出向量,  $w$  表示网络的权值和偏置组成的向量,  $e_p(w)$  表示误差。

设  $w^k$  表示第  $k$  次迭代权值和偏置所组成的向量,  $w^{k+1}$  表示第  $k+1$  次迭代权值和偏置所组成的向量, 且  $w^{k+1} = w^k + \Delta w$ 。在 L-M 算法中, 增量矩阵  $\Delta w$  的计算方法为:

$$\Delta w = [J^T(w)J(w) + \mu I]^{-1} J^T(w)e(w) \tag{11}$$

其中,  $\mu$  为比例系数, 当取很大的值时, 上式接近梯度下降法, 当取 0 时上式为高斯-牛顿法;  $I$  为单位矩阵;  $J(w)$  为雅可比矩阵, 表示如下:

$$J(w) = \begin{bmatrix} \frac{\partial e_1(w)}{\partial w_1} & \frac{\partial e_1(w)}{\partial w_2} & \dots & \frac{\partial e_1(w)}{\partial w_n} \\ \frac{\partial e_2(w)}{\partial w_1} & \frac{\partial e_2(w)}{\partial w_2} & \dots & \frac{\partial e_2(w)}{\partial w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_P(w)}{\partial w_1} & \frac{\partial e_P(w)}{\partial w_2} & \dots & \frac{\partial e_P(w)}{\partial w_n} \end{bmatrix} \tag{12}$$

比例系数  $\mu$  为一个试探性参数, 给定一个初始值  $\mu_0$ , 当计算出的阈值和偏置可以使误差减小时  $\mu$  降低, 反之  $\mu$  增加。

本文使用式(11)所示的权值和偏置修正算法, 通过采集的训练样本数据对 BP 神经网络中各神经元的权值和偏置进行调整, 训练完成后得到神经网络模型, 使用验证样本数据对神经网络模型进行验证, 错误小于一定值则完成神经网络模型的构建, 否则进行新一轮的训练, 直到输出最终完成的神经网络模型, 用测试样本数据通过前向传播的方式, 对最终模型进行性能评价。图 5 为 BP 神经网络的训练流程图。

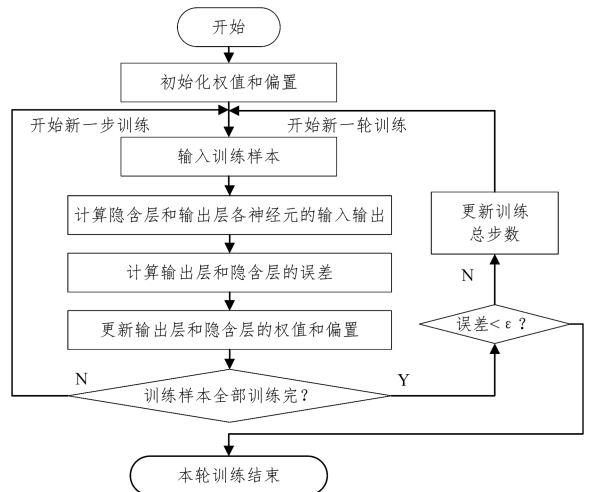


图 5 BP 神经网络的训练流程图

Fig. 5 Training flow chart of BP Neural Network

本文算法的总体流程如下:

(1)对传感器采集到的加速度与角速度数据进行预处理,并进行特征提取;

(2)根据提取的特征以及神经网络的功能建立神经网络模型,确定输入层、隐含层、输出层的神经元个数;

(3)初始化神经网络的权值和偏置,确定最大迭代次数、初始比例系数、比例系数调整系数、最小误差梯度;

(4)根据输入、输出样本集,通过 BP 神经网络的前向算法以及式(10)计算误差;

(5)使用式(11)和式(12)更新神经网络的权值和偏置;

(6)重新计算误差函数值,根据误差变化调整比例系数 $\mu$ ;

(7)判断误差函数值是否小于最小误差梯度,并且判断迭代次数是否到达最大迭代次数,如果满足上述条件之一,则跳出迭代,完成神经网络的训练,否则跳回步骤(5)。

#### 4 算法仿真与性能分析

本文算法的仿真通过 MATLAB R2016b 实现。所使用的计算机配置如表 2 所列,神经网络相关参数如表 3 所列。

表 2 仿真时计算机的配置

Table 2 Computer configuration for simulation

操作系统	Windows 7 64 位
CPU	英特尔酷睿 i5-3337U@1.8 GHz
内存	8G DDR3 1600 MHz
显卡	AMD Radeon HD 8670M+Intel HD Graphics 4000

表 3 网络模型相关参数

Table 3 Relevant parameters of network model

参数	值
最大迭代次数	1000
初始比例系数 $\mu$	0.001
比例系数增加系数	10
比例系数降低系数	0.1
隐层神经元数	10
最小误差梯度 $\epsilon$	$10^{-7}$

本次实验对采集到的 1360 条数据进行特征提取后对神经网络进行训练,得到结果如表 4 所列,表中误报条数指的是将日常行为错判为摔倒的数据条数,漏报条数指的是将摔倒错判为日常行为的数据条数,准确率为正确分类的数据条数占总数据条数的比例。

表 4 训练完成的神经网络模型的摔倒识别结果

Table 4 Fall recognition result of trained neural network model

	总数据 条数/条	摔倒数据 条数/条	非摔倒数据 条数/条	误报 条数/条	漏报 条数/条	准确率/%
训练样本	950	380	570	0	1	99.89
验证样本	205	100	105	0	1	99.51
测试样本	205	80	125	0	4	98.05
总体样本	1360	560	800	0	6	99.55

由表 4 可知,用 L-M 算法训练完成的神经网络模型对摔倒的识别率较高,且不存在误报摔倒的情况。

本文在准确率和时间复杂度的相关方面对本文算法的性能做了进一步分析,结果如表 5 所列。作为对比,我们构建了其他几种神经网络,包括一维卷积神经网络<sup>[7]</sup>、一个长短期记忆(Long Short-Term Memory, LSTM)循环神经网络<sup>[14]</sup>、与传统的梯度下降法作为训练算法的 BP 神经网络。CNN 使用文献[7]中所用参数;LSTM 采用文献[14]中的参数;传统 BP 神经网络参数与本文算法参数相同,并且初始学习速率为 0.001,其训练方法采用的是附加动量的自适应学习速率梯度下降法。其中,CNN 与 LSTM 算法的输入使用的是未经特征提取的原始数据时间序列,长度为 34 s(表 1 中每种行为持续 2 s),而传统 BP 神经网络的输入使用的是特征提取后的特征向量。表 5 中误报率表示误报条数占非摔倒数据条数的比例,漏报率表示漏报条数占摔倒数据条数的比例。

表 5 4 种基于神经网络的摔倒检测算法的比较

Table 5 Comparison of four fall detection algorithms based on neural network

算法	总准确率/%	误报率/%	漏报率/%	模型训练时间/s	训练迭代次数	分类所用时间/ms
1D CNN	94.49	1.87	10.71	/	/	38.2
LSTM	95.59	2.50	7.14	/	/	16.4
传统 BP 神经网络	97.4	1.87	3.57	3	160	13.2
本文算法	99.55	0.0	1.01	2	25	14.1

表 5 中,由于 CNN 与 LSTM 的迭代次数为人为确定的(本文中为 1000),因此讨论其训练时间没有太大意义。由表 5 可知,BP 神经网络相对于 CNN 与 LSTM 的分类用时较短,

由于 BP 神经网络需要对原始数据进行预处理,加上数据预处理的 2 ms 左右,对比 LSTM BP 神经网络并不存在分类用时上的优势,但是相对于 CNN 其分类用时就缩短了,而且 BP 神经网络在精度方面优于 CNN 与 LSTM。对比本文算法与传统 BP 神经网络算法,本文通过使用 L-M 算法,使 BP 神经网络的训练迭代效率大幅提高,迭代次数减少,虽然每次迭代所用的时间延长,但在总体训练用时上要略优于梯度下降法,而且通过 L-M 算法训练出来的神经网络模型的精度更高。综上所述,本文算法兼顾了准确率与时间复杂度,优于其他几种算法。

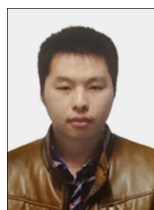
**结束语** 本文提出了一种基于 BP 神经网络的摔倒检测算法,该算法通过六轴传感器(MPU6050)采集人体加速度与角速度数据,对采集的数据进行特征提取,并用提取的特征值通过 L-M 算法训练 BP 神经网络,使 BP 神经网络可以准确分类人体日常行为与摔倒行为。实验结果表明,本文算法与其他几种神经网络算法相比,能更好地对摔倒行为进行识别,并且可以区分摔倒与跑步、跳跃等干扰较大的行为,摔倒检测的准确率达到 99.55%。而且,本文算法的计算量也未高于其他算法,具有良好的应用前景。

本文算法也具有一定的局限性:考虑到实验具有一定危险性,本文数据收集的志愿者都为青壮年,想要更好地识别老年人的摔倒,需要采集老年人的运动数据,使模型更加完善。另外,为了更好地在可穿戴设备上应用该算法,应当对当前的模型进行优化,精简特征向量,进一步减小计算量。

#### 参考文献

[1] MA L P, LI N, YANG W, et al. The challenge of aging on the

- healthcare delivery in China [J]. Chinese Hospitals, 2019, 23(4):1-3.
- [2] YU M, YU Y, RHUMA A, et al. An Online One Class Support Vector Machine-Based Person-Specific Fall Detection System for Monitoring an Elderly Individual in a Room Environment [J]. IEEE Journal of Biomedical and Health Informatics, 2013, 17(6):1002-1014.
- [3] ROUGIER C, MEUNIER J, ST-ARNAUD A, et al. Robust Video Surveillance for Fall Detection Based on Human Shape Deformation [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2011, 21(5):611-622.
- [4] LI Y, HO K C, POPESCU M. A Microphone Array System for Automatic Fall Detection [J]. IEEE Transactions on Bio-medical Engineering, 2012, 59(5):1291-1301.
- [5] TANG Y S, XIE N, HE J Q. Design and Implementation of Senile Fall Detection Algorithm Based on Triaxial Accelerometer [J]. Microcomputer Applications, 2019, 35(2):42-44.
- [6] HOU M, WANG H, XIAO Z, et al. An SVM fall recognition algorithm based on a gravity acceleration sensor [J]. Systems Science & Control Engineering, 2018, 6(3):208-214.
- [7] CHO H, YOON S M. Applying singular value decomposition on accelerometer data for 1D convolutional neural network based fall detection [J]. Electronics Letters, 2019, 55(6):320-322.
- [8] LIU P, LU T C, LV Y Y, et al. MEMS Tri-Axial Accelerometer Based Fall Detection [J]. Chinese Journal of Sensors and Actuators, 2014, 27(4):570-574.
- [9] LI J Y. Bp Neural Network Optimized by PSO and its Application in Function Approximation [J]. Advanced Materials Research, 2014:945-949.
- [10] LIU P, ZHANG W. An Fault Diagnosis Intelligent Algorithm Based on Improved BP Neural Network [J]. International Journal of Pattern Recognition and Artificial Intelligence, 2018(4).
- [11] LAI C F, CHANG S Y, CHAO H C, et al. Detection of Cognitive Injured Body Region Using Multiple Triaxial Accelerometers for Elderly Falling [J]. IEEE Sensors Journal, 2011, 11(3):763-770.
- [12] XIA K W, LI C B, SHEN J Y. An Optimization Algorithm on the Number of Hidden Layer Nodes in Feed-forward Neural Network [J]. Computer Science, 2005(10):143-145.
- [13] MADSEN K, NIELSEN H B, TINGLEFF O. Methods for non-linear least squares problems (2nd Edition) [M]. Informatics and Mathematical Modellings, Technical University of Denmark, 2004.
- [14] HU S J, QIN J B, GUO W. A Fall Detection Algorithm with Automatic Feature Extraction [J]. Chinese Journal of Sensors and Actuators, 2018, 31(12):66-71.



**ZHOU Li-peng**, born in 1994, postgraduate. His main research interests include wireless communication signal processing and system design, machine learning, etc.



**MENG Li-min**, born in 1963, Ph.D, professor, Ph.D supervisor. Her main research interests include wireless communication and network, intelligent information system, network management, multimedia digital communication and network, etc.

(上接第 241 页)

- [J]. IEEE Journal of Solid-state Circuits, 1988, 23(2):358-367.
- [8] JÄHNE B, SCHARR H, KÖRKE S. Principles of filter design [C] // Handbook of Computer Vision and Applications. Academic Press, 1999.
- [9] ACHANTA R, ESTRADA F, WILS P, et al. Salient region detection and segmentation [C] // International Conference on Computer Vision Systems. Springer, 2008:66-75.
- [10] ITTI L, KOCH C, NIEBUR E. A model of saliency-based visual attention for rapid scene analysis [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20(11):1254-1259.
- [11] CHENG M M, ZHANG G X, MITRA N J, et al. Global Contrast Based Salient Region Detection [C] // 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2011.
- [12] ZHAI Y, SHAH M. Visual attention detection in video sequences using spatiotemporal cues [C] // Proceedings of the 14th ACM International Conference on Multimedia. Santa Barbara, CA, USA, ACM, 2006:23-27.
- [13] GUO C, MA Q, ZHANG L. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform [C] // 2008 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2008:1-8.
- [14] HOU X, ZHANG L. Saliency detection: A spectral residual approach [C] // CVPR'07. IEEE, 2007:1-8.
- [15] HAREL J, KOCH C, PERONA P. Graph-based visual saliency [C] // Advances in Neural Information Processing Systems. 2007:545-552.
- [16] ACHANTA R, SÜSSTRUNK S. Saliency detection using maximum symmetric surround [C] // 2010 IEEE International Conference on Image Processing. IEEE, 2010:2653-2656.
- [17] MURRAY N, VANRELL M, OTAZU X, et al. Saliency estimation using a non-parametric low-level vision model [C] // CVPR 2011. IEEE, 2011:433-440.



**LIU Jun-qi**, born in 1995, postgraduate. His main research interests include object detection and artificial intelligence.



**LI Zhi**, born in 1973, Ph.D, professor, Ph.D supervisor. His main research interests include space system application and so on.