

基于深度学习的自适应睡枕设计^①

余益臻¹, 任佳¹, 刘瑜¹, 郭力宁²

¹(浙江理工大学机械与自动控制学院, 杭州 310018)

²(深圳市深智杰科技有限公司, 深圳 518102)

通讯作者: 刘瑜, E-mail: liuyu@zstu.edu.cn



摘要: 睡眠质量是影响人体健康的因素之一, 而一个高度合适的枕头能够有效地改善睡眠质量. 大量研究表明, 人在侧躺时所需的枕头高度应大于平躺时所需的枕头高度. 本文提出了一种基于深度学习的自适应睡枕设计方案, 可以识别人体的睡姿, 调节枕头高度, 并给出了硬件平台设计、网络模型搭建和移植. 由枕头内部的压力传感器和气压传感器, 分别采集头部对枕头的压力和枕头气囊内的气压, 生成时间序列数据帧, 再通过一维卷积网络 (1DCNN) 和门控循环单元网络 (GRU) 结合的网络模型对睡姿进行识别分类, 最后根据不同的分类结果调节枕头高度.

关键词: 睡姿识别; 时间序列; 深度学习; 1DCNN; GRU

引用格式: 余益臻, 任佳, 刘瑜, 郭力宁. 基于深度学习的自适应睡枕设计. 计算机系统应用, 2021, 30(7): 117-123. <http://www.c-s-a.org.cn/1003-3254/7972.html>

Design of Adaptive Pillow Based on Deep Learning

YU Yi-Zhen¹, REN Jia¹, LIU Yu¹, GUO Li-Ning²

¹(Faculty of Mechanical Engineering and Automation, Zhejiang Sci-Tech University, Hangzhou 310018, China)

²(Shenzhen Shenzhijie Technology Co. Ltd., Shenzhen 518102, China)

Abstract: Sleep quality, which influences human health, can be greatly enhanced by a pillow with proper height. Substantial studies have revealed that the pillow's height for persons lying on their sides should be greater than that for them lying on their backs. This paper introduces an adaptive pillow based on deep learning, which can recognize human sleeping positions to adjust the pillow's height. The paper also presents hardware platform design and the construction and transplanted of neural network models. First, the pressure sensor and the air pressure sensor embedded in the pillow respectively collect the pressure of the head on the pillow and the air pressure in the pillow airbag to generate a time-series data frame. Then, the 1DCNN-GRU network model identifies and classifies the sleeping positions. Finally, the pillow's height is adjusted according to classifications.

Key words: sleeping position recognition; time series; deep learning; 1DCNN; GRU

睡眠约占人生命总时长的三分之一, 随着生活节奏的加快, 越来越多的人饱受睡眠问题的困扰, 睡眠质量不好是危害健康的一大隐患^[1]. 越来越多的流行病学证据支持睡眠时间短与肥胖和糖尿病风险之间的联系. 与行为或睡眠障碍有关的慢性睡眠丧失, 可能是体重增加、胰岛素抵抗和 2 型糖尿病的一个新的危险因素^[2].

有关研究表明, 人侧躺时睡枕的高度应大于平躺时的高度^[3], 以减轻侧躺时肩部承受的压力. 因此, 需要设计一种能根据睡姿来调整高度的枕头来改善睡眠质量. 于是, 如何有效地识别人体平躺和侧躺这两种睡姿就成了解决该问题的关键.

目前, 对于人体睡姿识别的研究还比较少, 已知的

① 基金项目: 浙江省公益技术研究项目 (LGG20F030007)

Foundation item: Public Welfare Technology Research Project of Zhejiang Province (LGG20F030007)

收稿时间: 2020-10-22; 修改时间: 2020-11-18; 采用时间: 2020-12-01; csa 在线出版时间: 2021-06-30

睡姿识别方法有基于睡姿图像技术,即摄像头采集人体睡眠时的图像,通过数字图像处理和机器学习算法对该图像进行分类^[4],识别出此时人体的睡姿.也有通过置于床垫中的阵列式压力传感器,采集人体压力分布,来识别当前睡姿^[5].然而,以上两种方案存在着一些不足和需要改进的地方.首先,通过摄像头采集睡姿图像的方法,涉及了用户隐私,以及当人体被物体遮盖时(如脸被头发遮挡,身体被子遮盖),识别错误的问题.其次,采集人体压力分布的方法,置于床垫中的阵列式压力传感器面积大,传感器损耗大,且精度要求高,整个系统成本较高.

针对上述问题,本文提出了一种基于深度学习的自适应睡枕设计方案.通过枕头内部的压力传感器和气压传感器,分别采集头部对枕头的压力和枕头气囊内的气压,生成时间序列数据帧,并利用深度学习方法,自动学习深度特征,解决睡姿识别问题.

自适应睡枕的设计主要分为两个部分:硬件平台设计和睡姿识别算法的设计.

1 硬件平台设计

系统硬件平台的总体设计如图1所示,主控芯片选用意法半导体(ST)公司的STM32F407ZGT6,该MCU采用ARM Cortex-M4架构,主频高达168 MHz,带有FPU,具有强大的浮点运算能力.MCU主要负责处理传感器采集到的数据,根据识别的结果去控制执行器,并每隔一段时间数据写入外部Flash,还通过蓝牙与手机APP通信.传感器有压力传感器和气压传感器,分别使用HX711模块的压力传感器,XGZP6847气压计模块,用于采集头部对枕头的压力和置于枕头内部气囊的气压.执行器为气泵和气阀,分别选择5V微型充气泵和5V微型电磁气阀,用于控制气囊的充放气,达到调整枕头至对应高度的目的.主控芯片通过低功耗HC-42蓝牙5.0BLE模块与手机进行通信,手机APP可实现对枕头高度的设定、固件程序的升级,以及提取储存在Flash中的数据并生成txt文件.

2 睡姿识别算法设计

2.1 数据预处理

通常,自然界中的大部分信号都存在噪声,而直接采集所得的压力和气压原始数据包含了各种噪声,会对后续的工作产生影响,因此,需要对压力和气压原始

数据进行预处理.首先,将数据进行均值滤波,去除尖锐噪声.再将数据进行标准化,将数据限定在一定的范围内,消除奇异样本数据的影响,并且有效提升模型的收敛速度.

$$x^* = \frac{x - \bar{x}}{\sigma} \tag{1}$$

其中, \bar{x} , σ 是原始数据 x 的均值和方差.

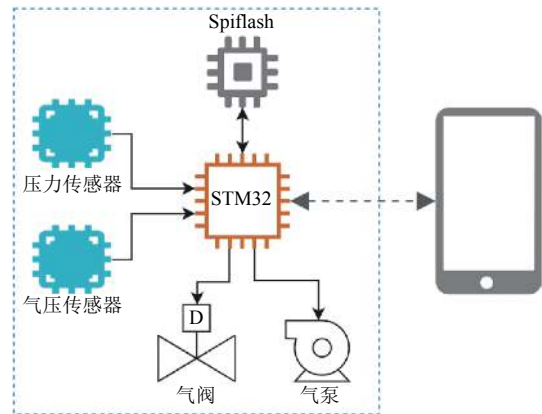


图1 硬件平台示意图

然后将每8s的数据生成一帧时序数据,每两帧数据间的时间重叠为4s.因数据采样频率为10 Hz,每次采样得到1个压力值和1个气压值,所以每一帧时序数据包含了80个压力数据和80个气压数据,如图2所示.

$$x_i = [p_h^i, p_g^i] \tag{2}$$

$$X = [x_1, x_2, \dots, x_{80}]^T \tag{3}$$

其中, X 为一帧时序数据, x_i 为 i 时刻的数据, $x_i \in \mathbb{R}^2$, p_h^i 为 i 时刻的压力值, p_g^i 为 i 时刻的气压值.

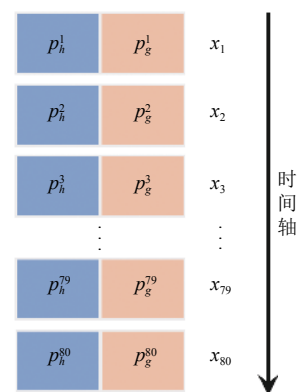


图2 一帧数据示意图

最后将一帧时序数据作为神经网络的输入, 该数据为一维双通道时序数据, 须选择合适网络模型对其进行特征提取和预测分类.

2.2 卷积神经网络

卷积神经网络 (CNN) 是一种处理具有类似网格结构的数据的神经网络, 与传统人工神经网络结构不同, CNN 包含了卷积层和池化层. 卷积层之间采用稀疏连接和权值共享的连接方式, 可大幅降低参数数量^[6], 加快了学习速率. 池化层降低了输出至下一层的数据维度, 简化了网络的复杂度, 同时也在一定程度上减少了过拟合的可能.

与二维、三维卷积神经网络相比, 对于固定长度片段的数据, 且该数据的一些特征在片段中的位置不具有高度相关性时, 一维卷积神经网络 (1DCNN) 可以有效地将这些特征提取出来. 本文中每个通道的数据为一维数据, 1DCNN 的卷积核朝一个方向移动, 如图 3 箭头所示. 一维卷积运算公式如下:

$$s(n) = (f * g)[n] = \sum_{m=0}^{N-1} f(m)g(n-m) \quad (4)$$

其中, $s(n)$ 为卷积后的序列, f 为待卷积的离散信号, g 为卷积核, N 是信号 f 的长度. 为了去除冗余信息, 简化网络复杂度, 防止数据过拟合, 需在一维卷积层后添加池化层.

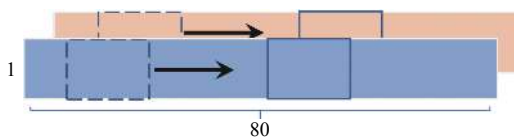


图 3 1DCNN 卷积操作

2.3 循环神经网络

循环神经网络 (RNN) 是专门用于处理序列的神经网络, 长短期记忆网络 (LSTM) 是一种特殊的 RNN, 是为了解决 RNN 梯度消失和梯度爆炸而提出的. 在传统的 RNN 中, 训练算法使用的是 BPTT, 当时间比较长时, 需要回传的残差会指数下降, 导致网络权重更新缓慢, 无法体现出 RNN 的长期记忆的效果, 需要一个存储记忆的单元, 因此, LSTM 模型被提出.

LSTM 单元结构如图 4 所示, $t-1$ 表示上一时刻, t 表示当前时刻. LSTM 的核心在于, 每个 LSTM 单元的长期记忆中的信息状态 (cell state). LSTM 通过设置 3 个门来控制长期记忆中的信息, 这 3 个门分别是: 遗

忘门 (forget gate)、输入门 (input gate) 和输出门 (output gate)^[7]. 门由一个 Sigmoid 层和一个点乘操作组成, 可以让信息选择性保留.

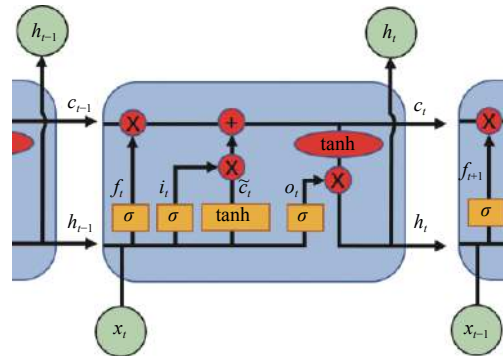


图 4 LSTM 单元结构

遗忘门可以决定清除或保留长期记忆中的信息.

$$f_t = \sigma(W_f \cdot h_{t-1} + U_f \cdot x_t + b_f) \quad (5)$$

在 Sigmoid 层输入上一时刻的隐含状态 h_{t-1} 和当前时刻的输入 x_t , 由循环权重 W_f 、输入权重 U_f 和偏置 b_f , 可以计算得到一个 0 与 1 之间的数 f_t , 来决定上一时刻需要保留的信息量.

输入门决定了当前时刻被存进长期记忆中的信息.

$$i_t = \sigma(W_i \cdot h_{t-1} + U_i \cdot x_t + b_i) \quad (6)$$

$$\tilde{C}_t = \tanh(W_C \cdot h_{t-1} + U_C \cdot x_t + b_C) \quad (7)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (8)$$

在 tanh 层输入上一时刻的隐含状态 h_{t-1} 和当前时刻的输入 x_t , tanh 激活函数可以将输出值限制在 -1 和 +1 之间, 输出一个新的向量 \tilde{C}_t , 表示当前时刻待存储的信息. 同样地, h_{t-1} 和 x_t 经过一个 Sigmoid 层, 计算得到 i_t 来决定 \tilde{C}_t 中需被存储的信息量. C_t 表示经过遗忘门和输入门添加了一部分新信息后, 当前时刻长期记忆中的信息.

LSTM 单元的输出, 即当前时刻的隐含状态 h_t , 取决于 C_t 和输出门.

$$o_t = \sigma(W_o \cdot h_{t-1} + U_o \cdot x_t + b_o) \quad (9)$$

$$h_t = o_t * \tanh(C_t) \quad (10)$$

将 C_t 输入至一个 tanh 层, 得到一个待输出量. 同样地, 将在 Sigmoid 层输入 h_{t-1} 和 x_t , 计算得到 o_t 来决定输出, 最后得到该 LSTM 单元的输出 h_t .

门控循环单元网络 (GRU) 是一种特殊的 RNN,

是 LSTM 的一个变种,如图 5 所示,与 LSTM 主要不同之处在于,GRU 单个门控单元同时控制遗忘因子和更新状态单元的决定. GRU 将 LSTM 的遗忘门、输入门和输出门进行了改动,变为更新门 (update gate) 和重置门 (reset gate). 在很多情况下,GRU 实际表现效果与 LSTM 接近,而 GRU 模型比 LSTM 模型更简单,能够有效减少模型消耗的资源^[8].

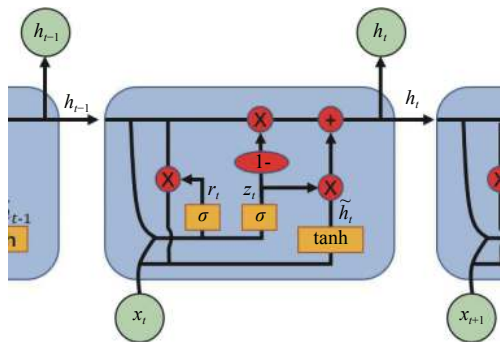


图 5 GRU 结构

重置门和更新门的计算公式如下:

$$r_t = \sigma(W_r \cdot h_{t-1} + U_r \cdot x_t + b_r) \quad (11)$$

$$z_t = \sigma(W_z \cdot h_{t-1} + U_z \cdot x_t + b_z) \quad (12)$$

更新门用于控制上一时刻的状态信息被保留至当前状态中的程度,更新门的值越大表示上一时刻的状态信息保留越多.重置门用于控制忽略前一刻的状态信息的程度,重置门的值越大表示忽略越少.

$$\tilde{h}_t = \tanh(r_t \cdot W \cdot h_{t-1} + U \cdot x_t) \quad (13)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (14)$$

与 LSTM 中的一些功能类似,重置门控制的上一时刻的信息量 h_{t-1} 需要被保留.更新门控制 h_{t-1} 中需要丢弃的信息和 \tilde{h}_t 中多少信息会被保留,最后得到该 GRU 单元的输出 h_t .

2.4 本文采用的网络模型

本文采用的 1DCNN-GRU 网络模型,由一维卷积神经网络 (1DCNN) 与门控循环单元网络 (GRU) 组成.先由 1DCNN 网络对经过预处理的时序数据进行一维卷积操作,提取数据在空间结构上的特征,再将提取出来的特征传入 GRU 网络.经过 1DCNN 后,提取得到的特征仍具有时序特性,所以 GRU 网络可以有效地对这些特征进行处理,并对处理结果进行

有效地预测和分类.通常,分类模型的最后一层由一个 Softmax 函数激活的全连接层构成,Softmax 函数定义如下:

$$f_{\text{Softmax}}(x)_i = \frac{\exp(x_i)}{\sum_{j=1}^n \exp(x_j)} \quad (15)$$

Softmax 函数将 n 个范围在 $(-\infty, +\infty)$ 的数,映射为 n 个 $(0, 1)$ 之间的概率.同时,使用 Softmax 函数可以防止数值出现上溢和下溢.经过该全连接层后,输出模型预测的分类.

3 实验

3.1 数据采集

本文将整个睡眠状态细分为 6 种,分别为: 1) 平躺保持不动; 2) 平躺虽有变化但仍为平躺; 3) 平躺变侧躺; 4) 侧躺保持不动; 5) 侧躺虽有变化但仍为侧躺; 6) 侧躺变平躺.其中,平躺时枕头高度应处在低位,侧躺时应升至高位.

本文参与实验的人员共有 10 人,包括 8 名大学生和 2 名教师.图 6-图 11 为 2 号实验员的其中一组数据,该组数据记录了该实验员的 6 种睡眠状态,每种睡眠状态包含了连续 16 s 的头部对枕头的压力值和枕头气囊内的气压值.该数据为一维双通道时序数据,片段长度固定且该数据的一些特征在片段中的位置不具有高度相关性.

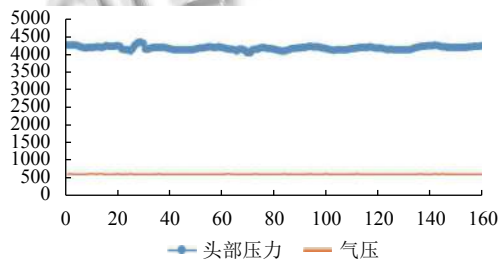


图 6 状态 1 数据曲线图

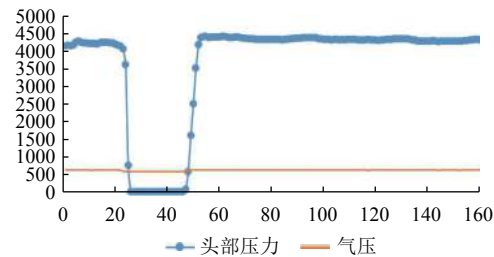


图 7 状态 2 数据曲线图

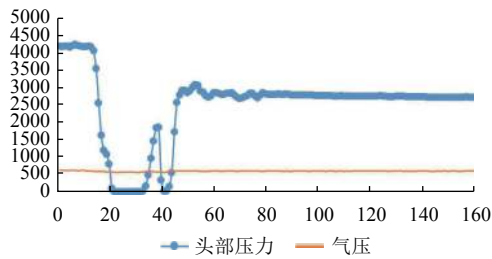


图8 状态3数据曲线图

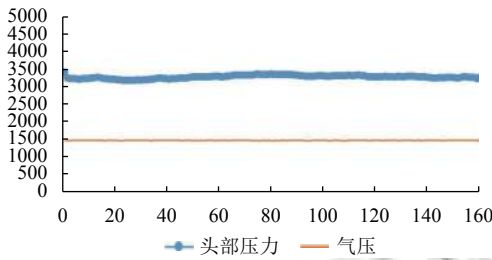


图9 状态4数据曲线图

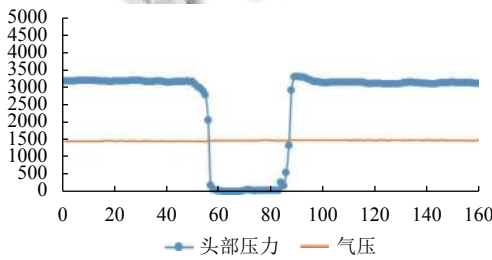


图10 状态5数据曲线图

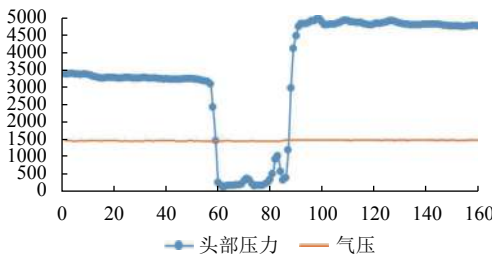


图11 状态6数据曲线图

3.2 模型参数设置

本文网络模型搭建和训练在PC机上进行,处理器为Intel Core i5-7300H,内存为8 GB,显卡为NVIDIA GTX1050 (2 GB),使用TensorFlow和Keras深度学习框架。

网络模型的深度会很大程度地影响特征的学习。对于层数较少的浅层网络,该网络学习得到的特征表征能力有限,随着层数的增加,网络可以学习到表征能力

更强的深度特征,可以更好地拟合复杂的特征输入。然而,网络参数量会随着网络的加深而增加,消耗更多的计算和存储资源,而且网络的加深可能会导致模型过拟合,因此需要对模型结构进行综合考虑。

神经网络主要分为输入层、隐藏层和输出层。网络的第一层为输入层,在该层传入预处理完的一帧时序数据。网络的输出层由1个Softmax全连接层构成,该层的维数等于需分类类别的个数,本文一共需要识别6种状态,所以输出层的维数是6。

网络的隐藏层共有6层,如图12所示,由一维卷积层、池化层和GRU层组成。其中,隐藏层的前3层为一维卷积层,用于提取输入数据特征,文献[9,10]中将层数设置为3时,已经有很好的效果。模型的超参数由多次实验确定。在前两层中,每一层定义64个卷积核用于提取特征,将卷积核大小设置为10,在第3层中定义32个卷积核,将卷积核大小设置为5。第4层是窗口大小为5的最大池化层,第5层为GRU层,将门控单元个数设置为64。优化器选择Adam优化器^[11],损失函数使用交叉熵损失函数。模型迭代50次时,误差损失函数达到收敛状态,所以将训练次数设置为50。训练集和测试集比例为8:2。

3.3 实验结果

本文采用了准确率(Accuracy)、精度(Precision)、召回率(Recall)和F1值(F1-score)对该网络模型进行评价。准确率是预测正确的样本占所有样本的比例。精度表示预测为正样本中,真正是正样本的比例。召回率表示正样本中,有多少是预测正确的正样本。F1值是精度和召回率的加权平均值。

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad (16)$$

$$Precision = \frac{TP}{TP + FP} \quad (17)$$

$$Recall = \frac{TP}{TP + FN} \quad (18)$$

$$F1 = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (19)$$

其中,TP表示将正样本预测为正样本,FN表示将正样本预测为负样本,FP表示将负样本预测为正样本,TN表示将负样本预测为负样本。

对模型进行训练和验证后,得到混淆矩阵、召回率、精度和F1值,如表1所示。

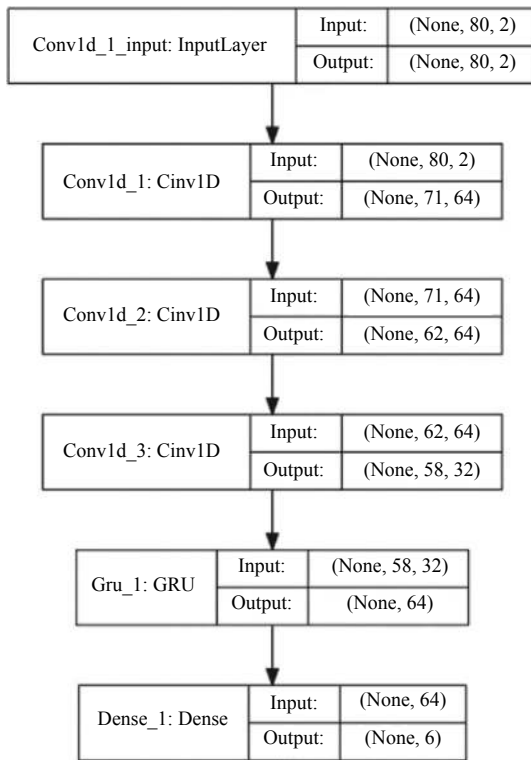


图 12 网络模型结构

表 1 1DCNN-GRU 模型分类结果

真实	预测					
	1	2	3	4	5	6
1	469	7	18	0	0	0
2	82	412	19	0	0	0
3	9	61	443	0	0	0
4	0	0	0	477	51	26
5	0	0	0	67	417	37
6	0	0	0	16	12	577
Recall (%)	94.9	80.3	86.4	86.1	80.0	95.4
Precision (%)	83.8	85.8	92.3	85.2	86.9	90.2
F1-score	0.89	0.83	0.89	0.86	0.83	0.93

作为对比,使用如下的网络模型,进行相同实验,得到不同网络模型的参数数量和准确率,如表 2 所示。

表 2 模型对比结果

模型	Param #	Accuracy (%)
BP	742,506	81.5
1DCNN	57,990	78.2
1DCNN-LSTM	77,862	87.6
1DCNN-GRU	71,654	87.3

3.4 结果分析

结果表明,相比于其他 3 种网络模型,1DCNN-GRU 和 1DCNN-LSTM 模型的准确率较高,1DCNN-GRU 模型参数量更少,更有利于在单片机上的应用。

从理论上讲,传入网络的数据为一维双通道时序

数据,1DCNN 可以学习数据在整体结构上的特征,提取数据的局部特征,组合抽象成高层特征.并且卷积层之间采用稀疏连接和权值共享的连接方式,能大幅降低参数数量,加快了学习速率.池化层降低了输出至下一层的数据维度,进一步简化了网络的复杂度.由于经过 1DCNN 后,提取到的特征仍具有时序特性,所以 LSTM 和 GRU 网络可以有效地处理这些特征.而且,GRU 在 LSTM 的基础上优化了门结构,保持了与 LSTM 相近的表现效果的同时,减少了模型的参数量,缩短了模型训练的时间。

4 移植

STM32CubeMX 是一个图形化工具,可以对 STM32 微处理器进行配置,并生成相应可执行的初始化 C 语言代码. STM32CubeAI 是 STM32CubeMX 的一个工具包,可以将预先训练好的人工神经网络转换成可以在单片机上运行的优化代码。

将训练好的模型文件.h5 文件导入 STM32CubeAI,并对该模型进行分析和验证,得出模型所需的资源和部分验证结果的混淆矩阵,如图 13 和表 3 所示,最后生成单片机代码工程文件,并在单片机上运行。

```

acc=100.00%, rmse=0.000000, mac=0.000000

6 classes (10 samples)
-----
C0      0  -  -  -  -  -
C1      -  1  -  -  -  -
C2      -  -  1  -  -  -
C3      -  -  -  0  -  -
C4      -  -  -  -  8  -
C5      -  -  -  -  -  0
    
```

图 13 10 组验证数据混淆矩阵

表 3 网络模型所占资源

input	input_0 [160 items, 640 B, ai_float, FLOAT32, (80, 1, 2)]
inputs (total)	640 B
output	dense_1_nl [6 items, 24 B, ai_float, FLOAT32, (6,)]
outputs (total)	24 B
params #	71 654 items (279.90 KiB)
mac	3441 242
weights (ro)	286 616 B (279.90 KiB)
activations	
(rw)	18 688 B (18.25 KiB)
ram (total)	19 352 B (18.90 KiB)

5 总结

本文针对睡姿识别任务,使用了 1DCNN-GRU 网

络模型,在PC机上进行训练和验证.接着,对比了BP、1DCNN和1DCNN-LSTM网络模型,选择将1DCNN-GRU网络模型移植至单片机,并在单片机上运行,能较好地识别睡姿.由于该实验缺乏相关的公开数据集,数据集只包含了少量实验人员的数据,且数据量少,加之考虑到模型的移植,网络模型不能太复杂,限制了网络的深度,影响了该网络模型的准确率,下一步的研究重点是继续改进和优化模型,进一步提高其准确率.

参考文献

- 1 Woodward M. Sleep in older people. *Reviews in Clinical Gerontology*, 2012, 22(2): 130–149.
- 2 Spiegel K, Knutson K, Leproult R, *et al.* Sleep loss: A novel risk factor for insulin resistance and Type 2 diabetes. *Journal of Applied Physiology*, 2005, 99(5): 2008–2019.
- 3 苏友新, 林善流, 刘献祥. 用枕及睡姿习惯与颈椎病关系的探讨. *中医正骨*, 1998, 10(3): 68–70.
- 4 支一飞. 基于图像技术的可自动调节高度睡枕系统研究 [硕士学位论文]. 西安: 西安电子科技大学, 2014: 1–49.
- 5 任志斌, 李洋, 郭士杰, 等. 基于模糊粗糙集的睡姿压力图像识别. *计算机工程与应用*, 2018, 54(3): 172–177.
- 6 Szegedy C, Vanhoucke V, Ioffe S, *et al.* Rethinking the inception architecture for computer vision. *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA. 2016. 2818–2826.
- 7 Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, 9(8): 1735–1780.
- 8 Chung J, Gulcehre C, Cho KH, *et al.* Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv: 1412.3555*, 2014.
- 9 Plötz T, Hammerla NY, Olivier P, *et al.* Feature learning for activity recognition in ubiquitous computing. *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*. Barcelona, Spain. 2011. 1729–1734.
- 10 沈延斌, 陈岭, 郭浩东, 等. 基于深度学习的放置方式和位置无关运动识别. *浙江大学学报(工学版)*, 2016, 50(6): 1146–1148.
- 11 Kingma D, Ba J. Adam: A method for stochastic optimization. *arXiv: 1412.6980*, 2014.