

基于深度学习的连续手语语句识别算法

李 晨¹, 黄元元¹, 胡作进²

(1. 南京航空航天大学 计算机科学与技术学院, 江苏 南京 211106;

2. 南京特殊教育师范学院 数学与信息科学学院, 江苏 南京 210038)

摘 要:目前,关于连续手语语句识别的研究相对较少,原因在于难以有效地分割出手语词。该文利用卷积神经网络提取手语词的手型特征,同时利用轨迹归一化算法提取手语词的轨迹特征,并在此基础上完成长短期记忆网络的构建,从而为手语语句识别准备好手语词分类器。对于一个待识别的手语语句,采用基于右手心轨迹信息的分割算法来检测过渡动作。由过渡动作可以将语句分割为多个片段,考虑到某些过渡动作可能是手语词内部的动作,所以将若干个片段拼接成一个复合段,并按照层次遍历的次序对所有复合段运用手语词分类器进行识别。最后,采用跨段搜索的动态规划算法寻找最大后验概率的词汇序列,从而完成手语语句的识别。实验结果表明,该算法可以对47个常用手语词组成的语句做出识别,且具有较高的准确性和实时性。

关键词:连续手语语句识别;过渡动作;卷积神经网络;长短期记忆网络;词间转移概率

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2021)01-0001-06

doi:10.3969/j.issn.1673-629X.2021.01.001

Recognition Algorithm for Continuous Sign Language Sentence Based on Deep Learning

LI Chen¹, HUANG Yuan-yuan¹, HU Zuo-jin²

(1. School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China;

2. School of Math and Information Science, Nanjing Normal University of Special Education, Nanjing 210038, China)

Abstract: At present, the researches on sign language sentences recognition are relatively few, because it is difficult to effectively split out sign language words. For sign language words, a convolutional neural network is used to extract hand-shape feature, and the trajectory normalization algorithm is used to extract trajectory feature. Based on that, a long short-term memory network is completed so as to prepare a sign-words classifier for sentence recognition. For a sign language sentence to be recognized, it can be split into several fragments by transition actions which are detected by the segmentation algorithm based on the right-hand trajectory. Since some transition actions may be actions inside words, we stitch several fragments into a composite segment and all composite segments are classified by the sign-words classifier in the order of hierarchical traverse. Finally, a dynamic programming algorithm with cross-segment search is used to find the word sequence with the greatest posterior probability, which realizes the recognition of sign language sentence. The experiment shows that the proposed algorithm can recognize sentences composed of 47 commonly used sign language words with high accuracy and real-time performance.

Key words: continuous sign language sentence recognition; transition action; convolutional neural network; long short-term memory network; transition probability between words

0 引 言

在当今的人机交互技术中,手势是输入信息的一种媒介。作为特殊的手势类别,手语是聋哑人的重要

交际工具。因此,研究手语识别不仅可以促进人机交互技术的发展,还可以促进聋哑人和健全人之间的交流。

收稿日期:2020-02-11

修回日期:2020-06-12

基金项目:江苏省青年自然科学基金(BK20170768)

作者简介:李 晨(1995-),女,硕士生,研究方向为模式识别、图像处理;黄元元,博士,副教授,研究方向为多媒体技术、图像处理、模式识别;胡作进,博士,教授,研究方向为数据处理、机器学习。

1 相关工作

连续手语语句是由手语词和连接手语词的过渡动作组成^[1]。由于手语动作的连贯性,从手语语句中分割出手语词变得极其困难,因此如何准确地检测手语词边界是连续手语语句识别的最大挑战。

在国内,张继海等^[2]将手语语句进行首轮粗分割后得到的多个片段送入手语词的隐马尔可夫模型(hidden Markov model, HMM)中,并借助阈值矩阵和动态时间规整算法(dynamic time warping, DTW)确定出可能的候选词及它们的结束帧,再根据比率阈值进一步确定本轮粗分割的最优候选词,并以其结束帧的下一帧为起点,继续进行下一轮的粗分割……最后将得到的多个最优候选词按照先后顺序串联起来,即可获得语句的识别结果。该算法在包含 34 个词汇的手语语句库中取得 77.8% 的识别率,但由于它在确定候选词的结束帧时采用逐帧遍历的方法,因此运行效率较低。杨文文等^[3]采用基于 HMM 的逐层构筑算法,同时辅以手语词帧长的约束和 n 元语法模型,最终在由 21 个词汇组成的 20 个手语语句上取得 12.2% 的错误率。然而该算法中语句的平均识别时间超过 8 秒,显然无法实现手语语句的实时识别。徐鑫鑫等^[4]根据点密度提取手语的关键帧序列,然后利用若干连续关键帧的权值之和对关键帧序列进行分割和识别,从而获得手语语句的识别结果。该算法的运行效率较高,但如果大权值的关键帧出现漏检或者误识,将无法识别出正确的手语词边界。

在国外, Yang 等^[5]利用基于条件随机场(conditional random field, CRF)的阈值模型判断语句中各帧是手语词还是过渡动作,然后利用 CRF 对分割后的手语词进行识别,最终在由 48 个词汇组成的美国手语语句库中取得 87% 的识别率。由于非特定人群手语数据的差异性较大,所以阈值模型在实际应用时手语词边界的检测效果并不理想。Cui 等^[6]通过卷积神经网络(convolutional neural networks, CNN)提取每帧图像的空间特征,再通过叠加的时间卷积层和时间池化层提取各手语片段的空间-时间特征,并将其送入双向的长短期记忆网络(long short-term memory, LSTM)中建模,最后采用连接时序分类(connectionist temporal classification, CTC)算法作为整个架构的目标函数。在 2012 年的德国天气预报手语库中,该算法的错误率为 38.7%。由于手语片段的类别概率分布大多较分散,所以采用波束搜索法进行 CTC 解码时,可能剔除部分片段的正确类别,进而影响手语词边界的准确性。Koller 等^[7]先利用 CNN 计算出每帧图像的隐状态类别概率分布向量,再通过 Viterbi 算法、一阶隐马尔可夫过程及 n 元语法模型求解手语语句的最优

词汇序列,最终在 2012 年的德国天气预报手语库上取得 32% 的错误率。由于该算法在寻找手语词边界时需要对三个超参数进行网格搜索,因此算法的时间损耗较高。

目前,大多数的手语词边界检测算法对非特定人群没有很好的鲁棒性,这在一定程度上影响了手语语句的识别效果。

2 算法步骤

该文利用轨迹归一化算法提取手语词的轨迹特征,同时利用卷积神经网络提取手语词的手型特征,并在此基础上训练基于长短期记忆网络的手语词分类器。对于一个待识别的手语语句,该文采用基于轨迹信息的分割算法检测过渡动作。由过渡动作将语句分割为多个片段后,考虑到过渡动作可能是手语词内部的动作,所以将若干片段拼接成复合段,并对所有复合段运用手语词识别算法进行分类,然后跨段搜索出目标词汇序列,从而完成手语语句的识别。

2.1 手语数据的获取

该文借助 Kinect 获取手语者的手心位置和深度图像,并在此基础上获得手语数据。

2.1.1 手型图像的获取

该文将深度图像和手心位置相结合,从而实现手型图像的快速提取^[8]。图 1 为手型图像的提取效果。

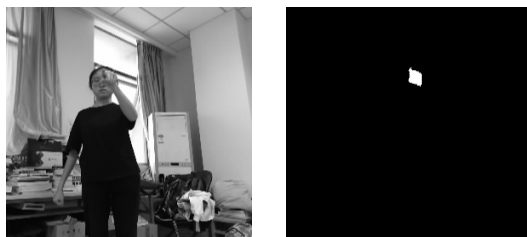


图 1 手型图像的提取效果

由于获取的手型图像比较粗糙,为了更精确地描述手语动作,在手型图像的基础上,引入了轨迹数据。

2.1.2 轨迹数据的获取

将手语持续时间内、经卡尔曼滤波校正后的手心位置按照先后顺序连接起来,即可获得手心的轨迹。为了进一步地去除噪声,该文对左、右手心轨迹分别应用长度为 3 的均值滤波进行平滑。平滑后的左、右手心轨迹构成了轨迹数据。

2.2 手语词识别算法

该文录制了 47 个常用的手语词。在获得这些手语词样本的轨迹特征和手型特征的基础上,开展手语词分类器的训练。

2.2.1 轨迹特征的提取

为了消除手心轨迹的尺度差异、采样点数差异和起始点差异,提出了一种轨迹归一化算法。

假设有一个持续时间为 n 帧的手语词样本,它的左手心轨迹 $P = \{p_1, p_2, \dots, p_n\}$, 其中 $p_i (1 \leq i \leq n)$ 表示第 i 帧左手心的位置。轨迹 P 的归一化过程如下:

(1) 创建一个长度为 50 的时间序列 Q 来存储归一化后的轨迹。

(2) 计算轨迹 P 的尺度缩放因子 α_s 和采样点数缩放因子 α_n :

$$\alpha_s = 1 / \| (\text{Neck}_x, \text{Neck}_y, \text{Neck}_z) - (\text{SpineMid}_x, \text{SpineMid}_y, \text{SpineMid}_z) \| \quad (1)$$

$$\alpha_n = \frac{50}{n} \quad (2)$$

其中, $(\text{Neck}_x, \text{Neck}_y, \text{Neck}_z)$ 、 $(\text{SpineMid}_x, \text{SpineMid}_y, \text{SpineMid}_z)$ 分别表示脖子和脊柱中心的位置。将轨迹 P 各采样点的手心位置乘上 α_s 可以实现尺度归一化;将轨迹 P 各采样点的序号乘上 α_n 可以指导采样点数归一化操作。

(3) 计算轨迹 P 的第 $i (1 \leq i \leq n)$ 个采样点归一化后的下标 j :

$$j = \lceil n * \alpha_n \rceil \quad (3)$$

其中, $\lceil x \rceil$ 表示对 x 进行四舍五入取整。如果在归一化的轨迹 Q 中 q_j 未被赋值,则将轨迹 P 的起始点 p_1 与原点对齐,并将 p_i 尺度归一化后的值赋给 q_j :

$$q_j = \alpha_s * (p_i - p_1) \quad (4)$$

如果 q_j 已被赋值,则将轨迹 P 的起始点 p_1 与原点对齐,并把尺度归一化后的 p_i 和 q_j 的均值赋给 q_j :

$$q_j = \frac{[\alpha_s * (p_i - p_1) + q_j]}{2} \quad (5)$$

(4) 遍历轨迹 Q , 对所有未赋值的 q_i , 采用线性插值法补充数据:

$$q_i = (q_{i-1} + q_{i+1}) / 2 \quad (6)$$

(5) 返回归一化的轨迹 Q 。

对左、右手心轨迹分别进行归一化后,该文使用归一化的左、右手心轨迹共同描述手语的轨迹特征。

2.2.2 手型特征的提取

对 MobileNetV2^[9] 稍加修改后,搭建出如表 1 所示的卷积神经网络。

这里的 conv2d 表示标准卷积层, avgPool 表示全局池化层。由表 1 可以看出,该网络由 3 个标准卷积层、9 个 bottleneck 模块和 1 个全局池化层组成。该网络的输入为 $224 \times 224 \times 1$ 的手型图像,经过网络各层的作用后,最后输出 61 维的手型类别概率分布向量。

当完成卷积神经网络的训练后,移除网络的最后一个标准卷积层,剩余的网络架构可以用作手型特征提取器。因此,输入一张手型图像,该网络可以提取出 160 维的手型特征;输入一个手语词样本,该网络可以提取出它的手型特征序列,将该序列归一化到 50 个采

样点,即可获得它的手型特征。

表 1 卷积神经网络的参数信息

输入尺寸	类型	扩展因子	输出通道数	重复次数	步长
224×224	conv2d 3×3	—	32	1	2
112×112×32	bottleneck	3	16	1	1
112×112×16	bottleneck	3	32	2	2
56×56×32	bottleneck	3	64	2	2
28×28×64	bottleneck	3	96	2	2
14×14×96	bottleneck	3	128	2	2
7×7×128	conv2d 1×1	—	160	1	1
7×7×160	avgPool 7×7	—	—	1	—
1×1×160	conv2d 1×1	—	61	1	1

2.2.3 基于长短期记忆网络的手语词分类器

手型特征和轨迹特征共同组成手语词的特征。考虑到手语词的特征是时间序列数据,而长短期记忆网络(LSTM)善于学习时序数据中的关联信息,于是搭建出如表 2 所示的长短期记忆网络。

表 2 长短期记忆网络的参数信息

输入尺寸	类型	神经元数量
50×166	双向 LSTM	166
50×332	Flatten	16 600
16 600	fc1	1 600
1 600	fc2	300
300	fc3	47

该网络包含一个双向 LSTM 层、一个 Flatten 层以及三个全连接层。网络的输入为手语词的特征,即一个长度为 50 的时间序列,序列中的每个元素为 166 维的向量,经过网络各层的作用后,最终输出 47 维的词汇类别概率分布向量。其中双向 LSTM 层用于捕捉每个采样点的上下文信息;而 Flatten 层是把 50 个采样点的隐状态拼接起来,进而获取整个序列的上下文信息。至于三个全连接层的功能则有点不同,前两个全连接层的作用是实现特征的学习和降维,最后一个全连接层则主要负责分类计算。

2.3 连续手语语句识别算法

对于一个待识别的手语语句,该文先采用分割算法检测过渡动作,然后采用基于过渡动作的手语语句识别算法获取语句的识别结果。

2.3.1 手语语句的分割

鉴于过渡动作的速度相对较快,且方向的偏转角度较小,因此提出了一种基于右手(主导手)轨迹信息的手语语句分割算法,它的详细步骤如下:

(1) 初步确定过渡动作。

在图 2 中, p_{i-1} 、 p_i 和 p_{i+1} 为三个相邻采样点上右

手心的位置。第 i 个采样点的右手心速度 v_i 可以定义为 p_i 和 p_{i+1} 之间的距离,即:

$$v_i = \|p_{i+1} - p_i\| \quad (7)$$

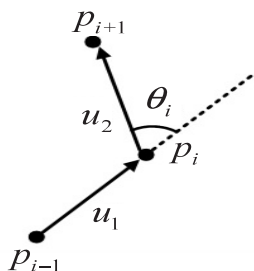


图 2 三个时间上相邻的采样点

图 2 中的 θ_i 表示第 i 个采样点上的方向角,它刻画了右手心在时刻 i 的方向偏转情况,即:

$$\theta_i = \arccos\left(\frac{u_1 \cdot u_2}{\|u_1\| \|u_2\|}\right) \quad (8)$$

其中, $u_1 = p_i - p_{i-1}$, $u_2 = p_{i+1} - p_i$ 。针对右手心的速度,设定阈值 $\rho_v = (2 * \text{avg}(v))/3$; 针对右手心的方向角,设定阈值 $\rho_\theta = 20$ 。其中 $\text{avg}(v)$ 表示所有右手心采样点的速度的均值。当 $v_i \geq \rho_v$ 且 $\theta_i \leq \rho_\theta$ 时,采样点 i 是过渡帧。因为过渡动作不止一帧,所以该文将距离三帧以内的过渡帧合并到同一个过渡动作中。由该方法确定出的第一个过渡动作位于起始手势和第一个手语词之间,而最后一个过渡动作位于最后一个手语词和终止手势之间,它们均不属于过渡动作,因为它们不是相邻手语词之间的连接动作,该文先剔除最后一个过渡动作,至于第一个过渡动作则暂且保留。

(2) 剔除错误的过渡动作。

非特定人群在比划具有语义的关键手势时会降低动作的速度,在轨迹上的表现就是这些手势对应的点密度较大。通过对手语语句样本的观察,发现所有手语词的关键手势的右手心点密度均 ≥ 5 。而过渡动作位于前一手语词的尾个关键手势和后一手语词的首个关键手势之间,所以该文根据右手心的点密度进一步剔除错误的过渡动作。

假设由步骤(1)获得过渡动作序列 $T = \{t_1, t_2, \dots, t_m\}$, 其中 m 表示过渡动作数量。初始化 $i = 1$, 接下来采用迭代算法剔除错误的过渡动作:

①若 $i \geq m$, 考虑到 t_1 不是过渡动作, 所以将 t_1 从序列 T 中剔除, 并得到最终的过渡动作序列, 否则进入步骤②;

②若 t_i 的终止帧到 t_{i+1} 的起始帧之间的区间不存在在右手心点密度 ≥ 5 的采样点, 则剔除 t_{i+1} , 并更新序列 T 和数量 m , 然后重复该步骤; 否则保留 t_{i+1} , 并令 $i = i + 1$, 跳转至步骤①继续判断后续的过渡动作。

2.3.2 基于过渡动作的手语语句识别算法

假设对一个手语语句运用上述分割算法检测到

$T-1$ 个过渡动作, 由这些过渡动作可以将手语语句分割为 T 个片段。因为检测出的词间过渡动作包含下一手语词的部分信息, 所以对于任意相邻过渡动作间的手语片段而言, 为了尽可能地保留手语词的特征, 该文将前个过渡动作的中位点帧设为起始帧, 同时为了尽可能地剔除手语词的上下文信息, 将后个过渡动作前右手心点密度大于 3 的帧设为终止帧。

考虑到检测出的过渡动作可能是手语词内部的动作, 所以该文将若干个片段拼接在一起形成复合段。因为语句样本中的词汇平均大约包含 1.7 个手语片段, 所以根据片段数 T 粗略预估语句中的手语词数量 N 。

$$N = \lceil T/1.7 \rceil \quad (9)$$

其中, $\lceil x \rceil$ 表示对 x 进行四舍五入取整。由于语句样本中的手语词至多包含 5 个片段, 为了避免过度的片段拼接给后续识别带来干扰, 由式(10)预估手语词的最大片段数 β 。

$$\beta = \min(5, \lceil T - (N - 1) \times 1.3 \rceil) \quad (10)$$

为了进行手语语句的识别, 该文需要在线创建类别标签矩阵 C 和分类概率矩阵 S , 并将它们的元素初始化为 0。对于以片段 t 的起始帧开始、以片段 t' 的终止帧结束的复合段, $C_{t,t',q}$ 保存该复合段的候选手语词的类别标签, $S_{t,t',q}$ 则保存该复合段是手语词 $C_{t,t',q}$ 的概率, 其中 $1 \leq t \leq T$, $t \leq t' \leq \min(t + \beta - 1, T)$, $1 \leq q \leq 5$ 。后续的手语语句识别过程如下:

(1) 复合段的分类。

首先初始化当前层各复合段的起始帧为片段 1 的起始帧, 并令 $t = 1$, 接下来开始复合段的分类工作。

①依次截取以片段 t 的开始帧为起点, 以片段 t' ($t \leq t' \leq \min(t + \beta - 1, T)$) 的结束帧为终点的复合段, 并对这些复合段运用手语词识别算法进行分类。如果这些复合段存在概率值 ≥ 0.2 的类别, 则把对应的类别和概率分别存入类别矩阵 C 和概率矩阵 S 中;

②令 $t = t + 1$, 跳转至步骤①, 继续对下一层的复合段进行分类。

(2) 目标词汇序列的跨段搜索。

定义 $\delta(t, t', q)$ 表示以片段 t 的起始帧开始、以片段 t' 的终止帧结束的复合段是手语词 $C_{t,t',q}$ 的累积概率, 其中 $1 \leq t \leq T$, $t \leq t' \leq \min(t + \beta - 1, T)$, $1 \leq q \leq 5$ 。令 $\langle t, t', q \rangle$ 表示 $\delta(t, t', q)$ 的参数元组, 所有合法的 $\langle t, t', q \rangle$ 组成了元组的集合 L 。为了方便回溯出搜索路径, 令 $\varphi(t, t', q)$ 保存 $\delta(t, t', q)$ 前个节点参数元组。同时为了避免识别出过长的词汇序列, 定义 $\eta(t, t', q)$ 表示搜索到 $\delta(t, t', q)$ 为止, 路径上已有词汇的数量。

目标词汇序列的跨段搜索算法如下:

①初始化。

$$\delta(1, t', q) = \begin{cases} 0, & \text{若 } C_{1,t',q} = 0 \\ S_{1,t',q}, & \text{其他} \end{cases} \quad (11)$$

$$\varphi(1, t', q) = \text{NULL} \quad (12)$$

$$\eta(1, t', q) = \begin{cases} 0, & \text{若 } C_{1,t',q} = 0 \\ 1, & \text{其他} \end{cases} \quad (13)$$

②递归。

$$\delta(t, t', q) = \begin{cases} 0, & \text{若 } C_{t,t',q} = 0 \\ \max_{\langle \tilde{t}, t-1, q' \rangle \in L} \{ \delta(\tilde{t}, t-1, q') \times S_{t,t',q} \}, & \text{其他} \end{cases} \quad (14)$$

$$\varphi(t, t', q) = \begin{cases} \text{NULL}, & \text{若 } C_{t,t',q} = 0 \\ \arg \max_{\langle \tilde{t}, t-1, q' \rangle \in L} \{ \delta(\tilde{t}, t-1, q') \}, & \text{其他} \end{cases} \quad (15)$$

$$\eta(t, t', q) = \begin{cases} 0, & \text{若 } \varphi(t, t', q) = \text{NULL} \\ \eta(\varphi(t, t', q)) + 1, & \text{其他} \end{cases} \quad (16)$$

其中, L 表示满足 $P(C_{t,t',q} | C_{\tilde{t},t-1,q'}) \neq 0$ 且 $\eta(\tilde{t}, t-1, q') < N$ 的 $\langle \tilde{t}, t-1, q' \rangle$ 的集合, 这里的 $P(C_{t,t',q} | C_{\tilde{t},t-1,q'})$ 表示词汇 $C_{\tilde{t},t-1,q'}$ 到词汇 $C_{t,t',q}$ 的转移概率, 它可以通过统计语料库中 $C_{\tilde{t},t-1,q'}$ 的下一个词汇是 $C_{t,t',q}$ 的概率来确定取值。

③终止。

$$P^* = \max_{\langle t, T, q \rangle \in L} \delta(t, T, q) \quad (17)$$

$$\langle t_1^*, T, q_1^* \rangle = \arg \max_{\langle t, T, q \rangle \in L} \delta(t, T, q) \quad (18)$$

④路径回溯。

令 $t_0^* = T + 1$, 递归 $\langle t_{i+1}^*, t_i^* - 1, q_{i+1}^* \rangle = \varphi(t_i^*, t_{i-1}^* - 1, q_i^*)$, 其中 i 的值从 1 开始递增, 直到 $\varphi(t_i^*, t_{i-1}^* - 1, q_i^*) = \text{NULL}$ 停止递归。

目标词汇序列的倒数第 j 个词汇为类别矩阵 C 中索引为 $\langle t_j^*, t_{j-1}^* - 1, q_j^* \rangle$ 的那个元素, 其中 $j = 1, 2, \dots, i$ 。将识别出的词汇按照时间先后串联起来, 即可得到语句识别结果。

3 实验结果与分析

为了验证手语词识别算法的有效性, 邀请 6 名手语者参与 47 类手语词的样本录制。此外, 以这 47 个词组成的 30 条手语语句作为样本进行语句识别实验。共有 6 名手语者参与语句样本的录制, 其中 2 名是熟练手语者, 2 名是次熟练手语者, 还有 2 名是不熟练手语者。需注意, 参与词汇样本采集的手语者和参与语句样本采集的手语者不重叠。

3.1 卷积神经网络的训练

针对录制的 47 类手语词的样本, 该文使用关键动作提取算法提取关键手型^[10], 然后采用 K 均值算法对关键手型进行聚类^[11], 其中 K 设为 60, 由此可以获得

60 类关键手型的样本。由于手语动作中还存在关键手型之间的过渡手型, 所以还需为过渡手型类选取样本。鉴于手型样本数有限, 该文采用平移、旋转及缩放变换来扩充样本集。最终每一类手型均有 240 个样本作为训练集, 60 个样本作为测试集。

在交叉熵损失函数^[12]的基础上, 使用随机梯度下降法优化卷积神经网络。设置初始学习率为 0.001, 最大迭代次数为 800。学习率的变化公式如下:

$$\text{lr}_i = \frac{\text{lr}_0}{1 + i * \text{decay}} \quad (19)$$

其中, i 为迭代次数, $\text{decay} = 1.0 \times 10^{-2}$ 。为了防止网络出现过拟合, 该文对除最后一个标准卷积层外的其他卷积层都进行了批量归一化 (batch normalization, BN) 处理。其中训练集的 batchSize 设为 50, 测试集则无需划分 batch, 把所有的测试样本一次性送入网络进行识别。采用 Keras 框架训练该网络, 最终训练出的网络模型在测试集上的精度为 94.58%。

3.2 长短期记忆网络的训练

该文在每个手语词样本的基础上造了 2 个样本, 它们分别保留了原始样本前 14/15 和后 14/15 的采样点。最终每一类手语词的样本总数增加至 162, 该文随机选取其中的 129 个样本用作训练集, 其余的 33 个样本则用于测试。

采用交叉熵损失函数测量长短期记忆网络的分类误差。设置学习率的初始值为 0.001, 最大迭代次数为 500。学习率的变化公式如下:

$$\text{lr}_i = \text{lr}_0 * \text{gamma} \wedge (\text{floor}(i / \text{stepsize})) \quad (20)$$

其中, i 表示迭代次数, $\text{gamma} = 0.1$, $\text{stepsize} = 200$ 。为了防止网络出现过拟合, 该文对双向 LSTM 层和 fc2 都进行了 BN 处理, 其中训练集的 batchSize 设为 30, 测试集无需划分 batch。在 GPU 上训练该模型, 最终训练出的模型在测试集上的精度达 98.55%。

考虑到语句中的手语词和孤立手语词的差异较大, 所以需要人工标注语句中的词汇, 并将其送入长短期记忆网络中训练^[13-14]。该文对熟练、次熟练及不熟练的 3 名手语者的语句样本中的手语词进行标注。每条语句有这 3 名手语者的 27 个样本, 其中的 21 个样本用于网络的再训练, 剩余的 6 个样本用于再测试。由于语句中手语词的样本数有限, 该文采用窗口规整方法^[15]造样本。最终每类手语词用于再训练和再测试的样本数分别为 84、24。对长短期记忆网络再训练 500 次后, 网络在测试集上的精度为 95.32%。

3.3 连续手语语句的识别

为了验证手语语句识别算法的有效性, 与文献[4]以及文献[13]中的算法进行对比。对熟练程度不一的 6 名手语者的语句样本进行识别。其中手语者一

和手语者二能够熟练地表达手语,手语者三和手语者四能够较熟练地表达手语,手语者五和手语者六则无法熟练地表达手语。且手语者一、手语者三和手语者五的部分语句样本参与了长短期记忆网络的训练。运用各算法对上述手语者的语句样本进行分类后,得到的识别准确率和平均识别时间如表 3 所示。

表 3 算法效果对比

	文中算法	文献[4] 的算法	文献[13] 的算法
手语者一	91.83%	94.22%	70.54%
手语者二	89.06%	94.53%	67.49%
手语者三	90.54%	85.27%	63.50%
手语者四	87.58%	84.18%	61.22%
手语者五	84.92%	69.35%	60.39%
手语者六	80.37%	67.70%	56.24%
平均识别时间/s	1.321	1.224	0.825

(1) 文献[4]是基于加权关键帧实现手语语句的识别。该算法的执行效率较高,但是它依赖于大权值的关键帧。对于非熟练的手语者,可能由于动作不够规范导致大权值关键帧的错识概率增高,从而极大地影响语句的识别效果,因此该算法的稳定性较差。

(2) 文献[13]采用连接时序分类算法实现手语语句的识别。虽然该算法的运行效率高,但它的识别精度较低,这是因为它需要将手语语句划分成多个等长的片段,而大多数片段的类别概率分布比较分散,所以利用波束搜索法进行解码时,手语片段的真实标签可能被剔除,从而极大地影响了语句的识别效果。

(3) 相比较来说,文中算法面向非特定人群的稳定性较高,能够实现手语语句的实时识别。

4 结束语

针对当前手语语句识别算法中存在的问题,提出了一种基于深度学习的手语语句识别算法。它充分利用了卷积神经网络的特征提取能力和长短期记忆网络的时序建模能力,并借助分割算法检测出的过渡动作,将手语语句的识别转化为复合段的分类和目标词汇序列的跨段搜索,降低了手语语句识别的复杂性。实验证明,该算法具有良好的稳定性及实时性。

参考文献:

- [1] 陈福财. 基于 Kinect 的连续中国手语识别[D]. 济南: 山东大学, 2016.
- [2] ZHANG J, ZHOU W, LI H. A threshold-based HMM-DTW approach for continuous sign language recognition[C]//Proceedings of international conference on internet multimedia

computing and service. Xiamen: ACM, 2014: 237-240.

- [3] YANG W W, TAO J X, YE Z F. Continuous sign language recognition using level building based on fast hidden Markov model[J]. Pattern Recognition Letters, 2016, 78(15): 28-35.
- [4] XU X X, HUANG Y Y, HU Z J. Research on continuous sign language sentence recognition algorithm based on weighted key frame[J]. Journal of Advanced Computational Intelligence and Intelligent Informatics, 2018, 22(4): 483-490.
- [5] YANG H D, SCLAROFF S, LEE S W. Sign language spotting with a threshold model based on conditional random fields[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2009, 31(7): 1264-1277.
- [6] CUI R, LIU H, ZHANG C. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization[C]//IEEE conference on computer vision and pattern recognition (CVPR). Honolulu: IEEE, 2017: 1610-1618.
- [7] KOLLER O, ZARGARAN S, NEY H, et al. Deep sign: enabling robust statistical continuous sign language recognition via hybrid CNN-HMMs[J]. International Journal of Computer Vision, 2018, 126(12): 1311-1325.
- [8] 石曼曼. 面向非特定人群的动态手语语句识别系统研究与实现[D]. 南京: 南京航空航天大学, 2017.
- [9] SANDLER M, HOWARD A, ZHU M L, et al. MobileNetV2: inverted residuals and linear bottlenecks[C]//IEEE conference on computer vision and pattern recognition (CVPR). Salt Lake City: IEEE, 2018: 4510-4520.
- [10] 徐鑫鑫, 黄元元, 胡作进. 连续复杂手语中关键动作的提取算法[J]. 计算机科学, 2018, 45(11A): 189-193.
- [11] HARTIGAN J A, WONG M A. A K-Means clustering algorithm[J]. Journal of the Royal Statistical Society. Series C (Applied Statistics), 1979, 28(1): 100-108.
- [12] GOODFELLOW I, BENGIO Y, COURVILLE A. Deep learning[M]. Cambridge: MIT Press, 2016.
- [13] CAMGOZ N C, HADFIELD S, KOLLER O, et al. SubU-Nets: end-to-end hand shape and continuous sign language recognition[C]//IEEE international conference on computer vision (ICCV). Venice: IEEE, 2017: 3075-3084.
- [14] HUANG J, ZHOU W G, ZHANG Q L, et al. Video-based sign language recognition without temporal segmentation[C]//AAAI conference on artificial intelligence. New Orleans: AAAI, 2018: 2-7.
- [15] GUENNEC A L, MALINOWSKI S, TAVENARD R. Data augmentation for time series classification using convolutional neural networks[C]//ECML/PKDD international workshop on advanced analytics and learning on temporal data. Riva Del Garda: HAL, 2016: 11-19.