文章编号：2095-2163(2020)10-0027-05 中图分类号：TP391.4 文献标志码：A

基于卷积神经网络的手势识别研究

周亦敏，李锡麟
（上海理工大学 光电信息与计算机工程学院，上海 200093）

摘 要：在手势识别的过程中，手势的多样性和复杂性会对识别的可靠性和准确性带来较大影响。为了能够提高手势识别的识别速度和准确率。本文使用 Google 的开源 Tensorflow 框架构建手势识别模型，介绍了 Tensorflow 的平台特征，并提出了基于 Tensorflow 框架的卷积网络模型。该实验的数据集是结合已有的的数据集和自收集的数据集进行设计的。实验结果表明，该模型具有较高的识别精度，较高的计算效率，较强的鲁棒性等特点，可以轻松调整网络结构，快速找到最优模型，较好地完成手势识别任务。

关键词：卷积神经网络；手势识别；Tensorflow；计算机视觉

Research on CNN-based Gesture Recognition

ZHOU Yimin，LI Xilin
（School of Optical-Electrical & Computer Engineering，University of Shanghai for Science & Technology，Shanghai 200093，China）

【Abstract】In the process of gesture recognition，reliability and accuracy of the diversity and complexity of gesture recognition will bring a greater impact. In order to improve the recognition speed and accuracy of the gesture recognition. In this paper，the use of Google´s newest open - source framework for buildingTensorflow gesture recognition model，introduced the platform features Tensorflow，and proposed a convolution Tensorflow network model based on the framework. The experimental data set is combined with the existing data sets and data sets collected from the design. Experimental results show that the model has higher recognition accuracy，higher calculation efficiency，and stronger robustness. It can easily adjust the network structure，quickly find the optimal model，and complete the gesture recognition task well.

【Key words】Convolutional Ceural network；Gesture Recognition；Tensorflow；Computer Vision

0 引 言

手势是一种重要的沟通方式，手势识别技术开创了人类与机器，设备或计算机交互的新局面。随着科学技术的发展，特别是无人机、智能假体和虚拟现实的发展，通过手势识别技术进行人机交互的需求越来越广泛，手势识别已成为国内外学者关注的焦点。因此，手势识别技术的研究具有重要的意义[1]。

手势识别的步骤一般分为测试分割、特征分析和特征识别。测试分割主要完成手势检查和分割的任务，特征分析主要进行特征检测，特征识别完成特征提取和手势识别[2]。在特征分析和识别过程中，大多数现有的手势识别算法都需要手动提取特征来完成手势识别任务。

近年来，随着人工智能技术的飞速发展，深度学习算法受到了广泛的关注。卷积神经网络（Convolution Neural Network，CNN）[3]因其快速、强大的分类能力而成为图像分类领域的热门算法。卷积神经网络是基于视觉神经细胞模拟的模型。它具有特征学习的能力，不需要手动提取功能，可以训练未处理的图像并自动生成特征提取分类器[4]。Tensorflow 是 Google 开发的开源深度学习框架[5]。它的前端支持许多开发语言，例如 Python、C ++和 Java 等；后端是用 C ++、CUDA 等编写的。在此框架中实现的算法，可以轻松移植到许多异构系统上，受到许多开发人员的青睐。它可以从底层实现队列和线程操作，快速调用硬件资源，提供输入数据、图形节点结构和对象功能，并可以将节点分配给许多设备进行并行操作。本文在 Tensorflow 框架下建立了基于卷积神经网络的手势识别模型，并设计了基础网络。优化网络参数后，分析了网络的识别效果。

1 神经网络的结构设计

1.1 LeNet-G 神经网络的概念

LeNet-5[6]是 CNN 的经典 8 层卷积网络结构，它最初用于手写数字识别。该网络结构除两个卷积层和两个池层外，还有两个完全连接的层以及两个

作者简介：周亦敏（1962-），男，硕士，副教授，主要研究方向：嵌入式系统、计算机系统结构、网络应用与智能设备等；李锡麟（1995-），男，硕士研究生，主要研究方向：计算机视觉、手势识别。

收稿日期：2020-04-24

输入和输出层[7-8]。但是,网络结构越复杂,所需的训练时间就越长,这严重影响了网络的通用性和实时性,不能满足手势识别的实时性要求[9]。本文基于 LeNet-5 结构,设计了卷积神经网络 LeNet-G,用于手势识别。网络可以在相对较短的时间内提取手势进行分类的特征。LeNet-G 网络的两层卷积层和两层池化层交替连接提取要素,以连接完整的连接层。在输出层中使用具有强非线性分类能力的 Softmax 分类器,在网络中使用 Softplus 激活函数[10]。本文神经网络结构如图 1 所示。

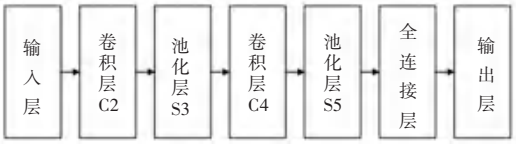

图 1 LeNet-G 神经网络结构

Fig. 1 LeNet-G neural network structure

1.2 卷积层

卷积层是用于提取图像特征的最重要的网络层。卷积是指通过权重共享和局部连接的方法,对具有学习能力卷积核的输入图像或上层输出特征图像进行运算。在激活功能的作用下,获取新特征图像的过程。每个特征图像代表图像的一个特征,在网络的每一层上都有多个特征图。在卷积层 C2 中,可以获得边缘、线条等低级特征;在卷积层 C4 中可以获得手掌等特征。

考虑到手势识别的复杂性,为了完全提取手势的特征,对卷积层进行了如下改进:

(1)由于卷积核的大小会影响手势分类的准确性,不同的手具势有较小的特征差异,因此较小的卷积核更适用于提取局部特征信息。3 * 3 卷积核对不同类别的手势识别能力更强,并且可以减少卷积层中的参数,提高模型的性能。因此在本文中,将 5 * 5 卷积核替换为 3 * 3 卷积核。

(2)采用 Softplus 函数[11]可以提高网络的收敛速度,并且缩短预训练时间。因此采用 Softplus 函数代替原来的 sigmoid 激活函数。Softplus 和 Sigmoid 函数可以表示为:

$$f_{softplus} = \log(1 + e^{x}), \tag{1}$$

$$f_{Sigmoid} = \frac{1}{1 + e^{x}}. \tag{2}$$

1.3 池化层

池化层是缩小维度,并从特征图像中提取特征的过程,也称为下采样层。通过减小特征图像的尺寸来获得池化层中的每个特征图像。池化对输出的上层进行下采样的过程,会减少参数矩阵的尺寸并保留有用的信息,从而减少最后全连层中的参数数量。一般在池化层的处理方式有最大池化采样和平均池化采样,本文采用最大池化采样[12],减小过拟合,同时提高模型的容错性。

1.4 全连接层

LeNet-G 网络通过两次卷积和下采样特征提取操作,来连接完整的连接层,以减小网络规模并增强非线性映射的能力。全连接层的每个神经元都与所有上层神经元相连。全连接层可以表示为:

$$x_j^l = f\left(\sum_{i=1}^{n} x_j^{l-1} \cdot w_{ij}^l + b_j^l\right). \tag{3}$$

其中,x 是 l 层的第 j 个特征图,n 是前一层神经元的数量,w 与上层神经元的权重,f 是激活函数,b 是偏置。全连接层中的神经元数设置为 600,并完全连接到池化层 S5。

1.5 Softmax 输出层

从卷积层和池化层提取特征后,可以获得完整的特征集。这些功能需要分类器进行分类。由于手势特征较为复杂,因此选择具有较强非线性分类能力的 Softmax 作为分类器[13]。Softmax 的输出层回归包含 8 个神经元,这 8 个神经元代表 8 种分类手势。

Softmax 适用于进行多分类的情况,可以将多个神经元的输出,映射到(0,1)的概率区间内进行多分类。其通用原型是用于两分类的 Logistic 分类器。数据集可以分成训练集和测试集两部分,数据集中第 i 个数据的 x_i 对应唯一的已标定 y_i,将所有数据集归为 n 类。模型的训练集由 m 个已标记的样本构成:$\{(x_1,y_1),\ldots,(x_m,y_m)\}$。对于模型中 softmax 前一层输入值 x,对每一个类别 j 可以估算出概率 $p(y=j \mid x)$。用一个 k 维的向量来表示 k 个估计的概率值。假设函数 $h_\theta(x)$ 形式如下:

$$h_\theta(x_i) = \begin{bmatrix} p(y_i=1 \mid x_i;\theta) \\ p(y_i=2 \mid x_i;\theta) \\ \vdots \\ p(y_i=k \mid x_i;\theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^{k} e^{\theta_j^T x_i}} \begin{bmatrix} e^{\theta_1^T x_i} \\ e^{\theta_2^T x_i} \\ \vdots \\ e^{\theta_k^T x_i} \end{bmatrix}. \tag{4}$$

其中,θ_k 是模型的参数。对应于 x 的预测值 k,$\sum_{j=1}^{k} e^{\theta_m^T x_i}$ 可以对概率分布进行归一化,确保所有概率之和为 1。代价函数为:

$$J(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=0}^{1} 1\{y_i=j\} \log p(y_i=j \mid x_i,\theta)\right]. \tag{5}$$

其中，$l\{y_i = j\}$ 的值为 1。

网络训练的主要问题是如何降低损失函数。对于此问题通常可以使用梯度下降法解决。

2　基于 Tensorflow 的 CNN 手势识别模型构建

2.1　Tensorflow 的特点

Tensorflow 是一种用数据流图进行计算的开源软件库，TensorFlow 会将所有计算转换为有向图上的节点。每个操作是一个节点，节点之间的连接称为边，每个边代表计算之间的依存关系[14-15]。有向图描述了计算数据的过程，并负责维护和更新状态。用户可以在有向图的分支上执行条件控制或循环操作；可以使用 C ++、Java、Python 等开发语言，来设计用于数据计算的有向图。有向图中的每个节点都具有任意数量的输入和输出；每个节点都描述一个可以视为操作实例的操作；在有向图边缘上流动的数据将成为张量。每个张量都是类型化的多维数据，可以预先定义或从有向图的结构中推断得出。一个节点可以具有零个或多个张量，可以轻松地构建和修改神经网络。本文实验框架基于 Python。Python 库中有许多辅助函数来构造数据图，这些辅助函数通过“导入”的形式加载连接数据库。该过程自动将定义的计算转换为有向图上的节点。

Tensorflow 程序通常分为两个阶段，第一阶段定义有向图中的所有计算，第二阶段执行计算，如图 2 所示。

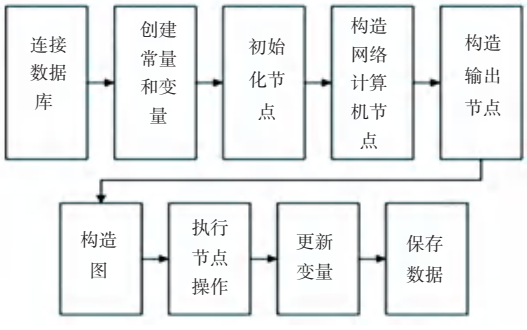

图 2　Tensorflow 程序

Fig. 2　Tensorflow program

第一阶段，在执行计算操作之前先构造一个有向图，然后创建常量和变量。有向图由单个节点和边组成。第二阶段，即通过启动有向图来创建会话对象，并通过执行节点操作以更新变量，最后保存数据。

2.2　CNN 模型结构

基于 Tensorflow 的 CNN 手势识别模型的框架结构如图 3 所示。

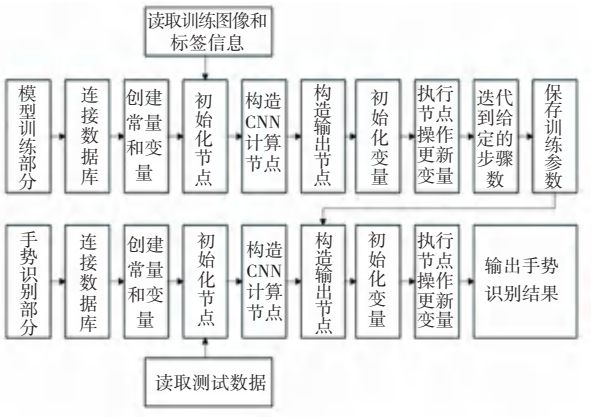

图 3　CNN 模型的框架结构图

Fig. 3　CNN model frame structure diagram

该模型分为二部分：训练和识别。这二部分的节点结构和操作基本相同，其主要区别在于权重变量的初始分配。模型给出了网络训练部分的初始权重，并通过训练样本对卷积网络结构每一层的权重进行了调整，以减少实际输出误差。识别部分的权重直接使用从训练中获得的权重，并用测试集进行测试，输出为手势识别的结果。

2.3　数据集

为了保证数据的有效性，实验数据采用国际公认的数据集和自行收集的数据集。国际公认的手势集来自“美国手语”手势集。该手势集有 24 个手势，每个手势都有 2 500 个自然色样本图像。本次实验在手语数据集中随机选择了 8 个手势，每个手势为 1 250 张图像，总共获取了 10 000 个有效样本。

自我收集的数据集由 10 名学生作为参与者收集。手势样本是从不同的角度和不同的背景环境中拍摄的。例如：左右手，采用室外自然光背景和室内自然光背景，增强数据的有效性。总共获取了 10 000 个有效样本。

本文的有效样本图像是基于肤色的二值化模型，在一定的肤色空间内，肤色具有一定的聚类特性。通过选择适当的阈值以检测和分割肤色，可以将手势图像与背景图像分离，然后根据高斯概率模型将手势图像转换为像素值为 0-255 的灰度图像。图片像素大小缩放为 32 * 32。

人手肤色在 YCrCb[16] 颜色范围内具有出色的聚类特征。将获得的样本图像从 RGB 空间转换为 YCrCb 颜色空间。亮度信息 Y 对颜色聚类没有影响。为了降低计算难度，仅针对 Cr 和 Cb 计算高斯概率。使用高斯卷积检查包含手势的图像，以进行卷积运算以获得概率值，并将概率值与阈值进行比较以提取手势特征。高斯概率模型可以表示为：

$$P(Cr,Cb) = \exp[-0.5(x-A)^T C^{-1}(x-A)]. \quad (6)$$

其中，$x-[Cr, Cb]^T$，A 是通过对许多人的皮肤颜色进行采样而得出的平均皮肤颜色。通过上式可以计算出样本图像与肤色之间相似度的概率矩阵。收集了许多黄种人的肤色值，并求出了平均值：$A=[140.231, 199.157]^T$。

样本的三维图像是在高斯概率模型下计算得到，如图 4 所示。

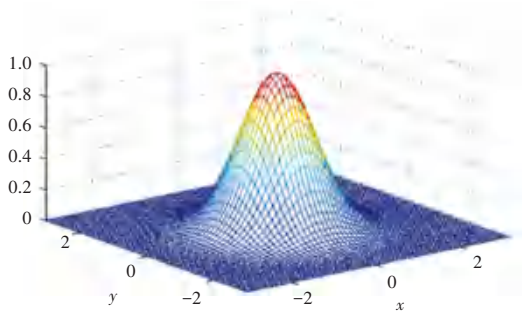

图 4 三维图像

Fig. 4 3D image

图 4 中，x、y 是图像在灰色空间中的坐标，而 z 轴是相应的概率值。在 z 轴上设置阈值，提取高于阈值的手势图像，并对其进行灰度处理和 Canny 边界检测[17]，以获取标准图像。最后，图像尺寸设为 32 * 32，如图 5 所示。

图 5 手势图像

Fig. 5 Gesture image

3 实验结果与分析

3.1 实验环境

本文实验在 Windows10 操作系统中运行 Tensorflow，以测试 LeNet-G 网络。计算机 CPU 为 i7-8700K，内存为 32G，用于 CUDA 加速的 GPU 为 NVIDIA GTX2080。数据集分为训练集和测试集。训练集随机选择每个手势图像的 2 000 个图像，总共 16 000 个样本；测试集每个手势图像的 500 个图像，总共 4 000 个样本。

3.2 不同学习率对 LeNet-G 识别效果的影响

学习率是根据当前收敛程度和梯度更新情况的重要参数。在模型开始训练时，用较大的学习率可以跳出局部最优解。随着模型训练迭代次数增加，逐渐减小学习率，逼近全局最优解。

设置学习速率的方法有二种：自适应调整和手动设置。当权重值具有较大的波动时，降低学习率；而当权重值趋于不变时，提高学习率。神经网络参数是动态变化的，最好的方法是为每个权重值设置不同的学习率。可以利用上一层数值更新的变化来建立两个权重更新之间的关系，从而加快学习过程。本文采用自适应梯度下降算法来更新参数，然后比较全局学习率的初始值的选择。如果初始值太大，则会错过极值点，并且系统将不会收敛，并且错误率很高。如果初始值太小，系统将收敛太慢，甚至会陷入局部最优状态。因此，根据经验初始学习率分别设置为 0.05、0.01、0. 005、0.001、0.0005，并且在批量处理 100 个样本的情况下进行了训练。识别效果见表 1。

表 1 不同学习率下的手势识别效果

Tab. 1 Gesture recognition effect under different learning rates

学习率	准确率/%	训练时间/s
0.05	97.364	2 508
0.01	98.516	2 629
0.005	99.176	2 753
0.001	99.208	2 873
0.000 5	98.236	2 996

根据表 1 的值，当初始学习率设置为 0.05、0.0.1 和0. 000 5时，系统的准确率远低于 0.001 和 0.005。这表明神经网络陷入局部最优的情况，错误率增加。将识别结果与初始学习率分别为 0.001 和 0.005 二种情况进行比较，可以看出 2 个值的错误率没有较大差异，但是训练时间却有很大差异。综合分析表明，初始学习率应设置在 0.005 左右。

3.3 卷积核数目对 LeNet-G 识别效果的影响

卷积层通过手势样本与卷积核之间的卷积运算来提取特定的手势特征。每个卷积核代表一个特殊功能。更多的卷积核可以更准确，更真实地识别样本图像。

为了验证卷积核数目对 LeNet-G 网络的影响，设置了两个卷积层中的卷积核数目，分别为 8-16、16-16、16-32、32-32。然后训练手势数据集并记录识别效果。识别效果见表 2。由表 2 可知，随着卷积核数的增加，错误率逐渐降低。但是，卷积核的数量越多，计算量增大，训练时间增加，并对计算机性能的要求增加。均衡考虑计算机性能、训练时间和识别效果，设置卷积核为 16-32 结构就可以满足网

络的要求。

表 2　不同卷积核下的识别效果

Tab. 2　Recognition effects under different convolution kernels　%

卷积核结构	准确率	
	训练	测试
8-16	97.762	95.517
16-16	98.215	96.056
16-32	99.086	97.699
32-32	99.166	97.826

3.4　神经元数量对 LeNet-G 识别效果的影响

手势样本图像用作输入数据，LeNet-G 网络用于训练样本。全连接层中初始神经元的数量分别设置为 600 个、800 个和1 000 个，进行5 000次训练。根据全连接层中神经元的数量，测试样本训练后神经网络模型的准确率见表 3。

表 3　全连接层中不同数目的神经元下的网络识别效果

Tab. 3　Recognition effect under different number of neurons

神经元的个数	准确率/%
600	99.847
800	96.674
1 000	96.991

从表 3 可以看出，当通过梯度下降法改变整个连接层中神经元的数量时，损失函数的准确度将发生变化。可以得出结论，增加全连接层中的神经元可以在一定程度上提高准确度。

4　结束语

通过实验表明，基于卷积神经网络的手势识别模型具有编程简单、使用灵活等优点。它可以有效地提高建模、编程和分析的效率。本文中采用基于 Tensorflow 平台的卷积神经网络手势识别框架，并设计了一个 7 层 LeNet-G 卷积神经网络用于手势识别。通过改变初始学习速率、全连接层中卷积核和神经元数，得出结论：当初始学习率为 0.005 时，两个卷积层中的卷积核数设置为 16-32，全连接层中的神经元数为 600，卷积神经网络可以获得更好的识别效果。由于所收集的手势样本是多个背景的手势，因此所设计的网络训练的模型具有很强的鲁棒性。同时，由于实验是在 GPU 模式下进行的，因此运算速度相对较快，可以快速找到运行效率最高的网络结构训练模型，并且在处理速度上具有很大的优势。可以看出，基于卷积神经网络的手势识别模型的设计具有很大的开发和应用潜力。

参考文献

[1] LI Yuan, WANG Xinggang, LIU Wenyu, et al. Deep attention network for joint hand gesture localization and recognition using static RGB-D images[J]. Information Sciences, 2018, 441.

[2] 陈甜甜，姚璜，左明章，等. 基于深度信息的动态手势识别综述[J]. 计算机科学，2018，45(12)：42-51，76.

[3] LECUN Y, BOSER B, DENKER J S, et al. Backpropagation applied to handwritten zip code recognition [J]. Neural computation, 1989, 1(4): 541-551.

[4] 常亮，邓小明，周明全，等. 图像理解中的卷积神经网络[J]. 自动化学报，2016，42(9)：1300-1312.

[5] GÉRON A. Hands-On Machine Learning with Scikit-Learn, Keras, and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems[M]. O'Reilly Media, 2019.

[6] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[7] WANG W, NEUMANN U. Depth-awarecnn for rgb-d segmentation[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018. 135-150.

[8] 朱越，李振伟，杨晓利，等. 基于视觉的静态手势识别系统[J]. 计算机技术与发展，2019，29(2)：69-72.

[9] LV X, XU Y, WANG M. Real-Time Hand Gesture Recognition for Robot Hand Interface [M]//Life System Modeling and Simulation. Springer, Berlin, Heidelberg, 2014. 209-214.

[10] CHEN L, ZHOU M, SU W, et al. Softmax regression based deep sparse autoencoder network for facial emotion recognition in human-robot interaction[J]. Information Sciences, 2018, 428: 49-61.

[11] ZHAO H, LIU F, LI L, et al. A novelsoftplus linear unit for deep convolutional neural networks[J]. Applied Intelligence, 2018, 48(7): 1707-1720.

[12] WU H, GU X. Max-pooling dropout for regularization of convolutional neural networks[C]//International Conference on Neural Information Processing. Springer, Cham, 2015. 46-54.

[13] VAMPLEW P, DAZELEY R, FOALE C. Softmax exploration strategies for multiobjective reinforcement learning [J]. Neurocomputing, 2017, 263: 74-86.

[14] PHAN-XUAN H, LE-TIEN T, NGUYEN-TAN S. FPGA platform applied for facial expression recognition system using convolutional neural networks[J]. Procedia Computer Science, 2019, 151: 651-658.

[15] LIU S, YU M, LI M, et al. The research of virtual face based on Deep Convolutional Generative Adversarial Networks using Tensor Flow[J]. Physica A: Statistical Mechanics and its Applications, 2019, 521: 667-680.

[16] BIANCO S, GASPARINI F, SCHETTINI R. Computational strategies for skin detection [C]//International Workshop on Computational Color Imaging. Springer, Berlin, Heidelberg, 2013. 199-211.

[17] YANG A, JIANG W, CHEN L. An Adaptive Edge Detection Algorithm Based on Improved Canny [M]//Advanced Computational Methods in Life System Modeling and Simulation. Springer, Singapore, 2017. 566-575.