

文章编号: 2095-2163(2023)04-0158-05

中图分类号: TP389.1

文献标志码: A

基于 CNN 网络的手写体数字识别系统的实现

杨之杰, 林雪刚, 阮杰

(江苏大学 计算机科学与通信工程学院, 江苏 镇江 212013)

摘要: 手写体数字识别现在仍是图像识别分类的一个热点, 而基于卷积神经网络的深度学习算法具有局部区域连接、权值共享、降采样的结构特点, 使得卷积神经网络在图像处理领域有出色表现。以实现手写体数字高精度识别为目标, 设计并实现一个基于卷积神经网络的高精度手写体数字识别系统。首先, 通过 Pyqt5 平台设计一个人机交互的 GUI 界面, 其次进行手写体数字图像的采集与预处理, 转换成规范的三维向量输入到 CNN 网络卷积层中, 接着进行各个网络层的运算处理, 最后通过 *Softmax* 输出分类结果。仿真实验结果下 MNIST 数据集识别模式下的识别率为 99.9%, 手写输入识别模式下的识别率为 98%。结果表明: 基于 CNN 的神经网络识别准确率高, 实现技术简单, 实用性高。

关键词: 卷积神经网络; GUI 界面系统; Pyqt5; 手写体数字识别

Implementation of handwritten digit recognition system based on CNN network

YANG Zhijie, LIN Xuegang, RUAN Jie

(School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang Jiangsu 212013, China)

[Abstract] Handwritten digit recognition is still a hot spot in image recognition and classification, and the deep learning algorithm based on convolutional neural network has the structural characteristics of local area connection, weight sharing and downsampling, which makes convolutional neural network perform well in the field of image processing. Aiming at realizing high-precision handwritten digit recognition, a high-precision handwritten digit recognition system based on convolutional neural network is designed and implemented. Firstly, use the Pyqt5 platform to build a GUI interface for human-computer interaction; secondly, carry out the collection and preprocessing of handwritten digital images, which are converted into standardized three-dimensional vectors and input into the convolution layer of the CNN network, and then start the operation processing of each network layer; finally, the classification results are output through *Softmax*. The simulation results show that the recognition rate in the MNIST dataset recognition mode is 99.9%, and the recognition rate in the handwriting input recognition mode is 98%. The results show that the neural network based on CNN has high recognition accuracy, simple implementation technology and high practicability.

[Key words] convolutional neural network; GUI interface system; Pyqt5; handwritten digit recognition

0 引言

随着机器学习的发展, 基于神经网络的深度学习已成为热点, 深度学习技术已被广泛应用在文字、图像识别分类研究中。目前在国内外, 针对手写体数字识别技术已经比较成熟, 相较于传统光学字符识别(OCR)图像识别技术, 基于深度学习的卷积神经网络算法可以在复杂场景下快速、准确、有效地获取并识别场景中文字。由于手写体存在形态各异、千差万别、随意性大、书写不规范的情况, 同时还会存在数据采集时的光线、角度不同等问题, 手写数字

识别问题有着很大的挑战性^[1]。

卷积神经网络是一种受到人类的视觉神经系统和早期的时延神经网络(Time-Delay Neural NetWork)的启发而设计提出的多层神经网络。卷积神经网络结合了共享权重、局部感受野、空间或时间上的下采样三种思想, 使得网络具有较少的训练参数、简单的网络结构且适应性强等优点^[2]

1994年, 文献[3]提出LeNet, 定义了卷积神经网络的基本架构是卷积和池化, 在 手写体数字领域识别率达到 99.13%。2012年, 文献[4]提出 AlexNet, 采用双 GPU 网络结构并使用 *ReLU* 作为激

作者简介: 杨之杰(2000-), 男, 本科生, 主要研究方向: 通信系统; 林雪刚(2001-), 男, 本科生, 主要研究方向: 人工智能; 阮杰(2001-), 男, 本科生, 主要研究方向: 机器学习。

通讯作者: 杨之杰 Email: 2643291352@qq.com

收稿日期: 2022-05-07

活函数,使得网络能够获得更加丰富的特征。2014年,文献 [5] 提出 VGG 系列模型(包括 VGG-11/VGG-13/VGG-16/VGG-19),使用很“深”的网络结构并在同年的 ImageNet Challenge 上获得分类任务第二名、定位(Localization)任务第一名。

随着神经网络的发展,网络结构越来越深,但是对于小图像容量的手写体数字识别的实现并不需要大而深的网络结构,否则对计算机处理性能将会造成巨大负担,因此,本文设计一个简单适宜的 CNN 网络结构,实现了一个可以用于高精度识别的手写体数字识别系统。本文研发的手写体数字识别系统主要分为 2 个部分:GUI 交互界面与 CNN 网络模型。其中,GUI 交互界面是通过 Pyqt5 工具包进行搭建,再通过所搭建 CNN 网络模型进行相应的图像训练与测试,实验仿真不同模式下的图像识别场景,得出结论为:基于 CNN 网络的图像识别算法的识别率可以达到 99.9%,具有高精度的识别性能。本文实现的 CNN 手写体数字识别系统不仅可以有效实现手写体数字识别,同时还具有简单易行、识别性能优良的优点。

1 基于 CNN 的手写体数字识别系统

1.1 系统实现概述

本文提出的手写体数字识别系统是基于卷积神经网络模型实现的一种高效简易的图像识别系统。主要实现流程如图 1 所示。

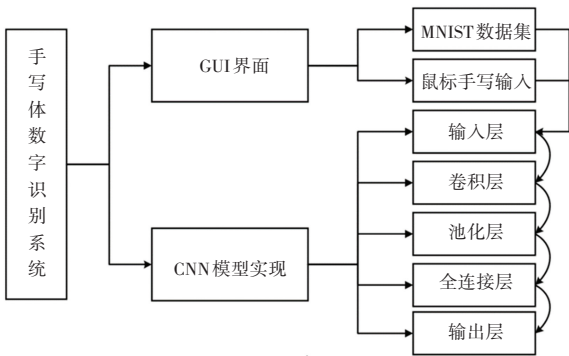


图 1 手写体数字识别系统实现流程图

Fig. 1 The flow chart of handwritten digit recognition system implementation

由图 1 可见,手写体数字识别系统主要分为 2 个主要模块,分别是:基于 Pyqt5 的 GUI 界面和基于 CNN 的深度学习模型。其中,Pyqt5 实现的 GUI 界面用于用户与 PC 端的人机交互,用户可以选择系统识别的模式:MNIST 数据集随机抽取、鼠标手写输入;基于卷积神经网络的识别模型主要通过 5 个网络层:输

入层、卷积层、池化层、全连接层以及输出层,实现对用户所选取的识别模式下的手写体图片识别。

1.2 图像采集与处理

1.2.1 MNIST 数据集识别模式

MNIST 数据集是由手写数字的图片和相应的标签组成,共有 10 类,分别对应数字为 0~9。训练图片一共有 60 000 张,可采用学习方法训练出相应的模型。测试图片一共有 10 000 张,可用于评估训练模型的性能^[6]。

MNIST 数据集预处理过程如图 2 所示。MNIST 数据集抽取模式,要先将 MNIST 数据集下载,手写数字识别系统使用自定义 `load_mnist` 函数进行数据集的下载与本地保存。本文中先将训练集与测试集中的手写体图像进行预处理,下载的数据集图像保存为归一化的像素值(范围是 0.0~1.0),同时将对应的数字标签输入展开为一个 784 的一维数组。这样有助于后续网络模型的输入处理。同时,在 GUI 界面的画板区域内将 MNIST 数据集的抽取图像转换成 Qimage 对象,用于后续的识别图像处理。

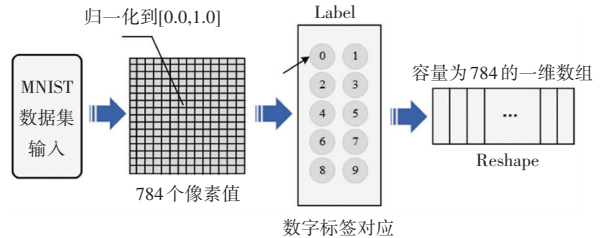


图 2 MNIST 数据集预处理过程

Fig. 2 Preprocessing process of MNIST dataset

1.2.2 手写输入识别模式

鼠标手写模式是基于 Pyqt5 的 GUI 开发框架实现的^[7],在 GUI 界面上设置一个空白画板,背景色设置 RGB(0,0,0),也就是纯黑色;设置画笔颜色为(255,255,255),也就是纯白色。利用 Pyqt5 类库中的绘图工具进行手写数字的交互事件实现,本文实现的鼠标手写图像是只有黑白色的图像,以方便 2 种图像输入模式后续使用相同的图像识别处理过程。用户在 GUI 界面的画板上写下相应的数字,点击“识别”后,本程序就可以将画板的内容获取并转换成 Qimage 对象,再通过 Reshape 将其变换成符合网络输入规范的三维向量,用于后续 CNN 网络模型识别处理。手写模式的图像采集过程如图 3 所示。

1.2.3 CNN 网络实现

识别处理是针对上述的 2 种输入识别模式所采集的手写体数字图像进行 CNN 网络模型输入的规范化图像对象处理,保证统一化的网络输入层的参

数输入。在获取了 Qimage 对象后,需要将其转换成 Python 中的 PIL image 对象,将图像大小修改成 1 通道的 28×28 像素大小,同时批量大小为 1,再将其转换成灰度值归一化的数组,规范成网络输入类型。手写体数字图像的预处理过程如图 4 所示。

相比较传统神经网络,CNN 网络新增了卷积层 (Convolution 层)和池化层 (Pooling 层),因此一般的 CNN 模型就是由卷积层、池化层和全连接层构成的。Convolution 层实现结构如图 5 所示。本文通过一个简单合适的 CNN 模型进行手写体数字识别实现,网络结构为“Convolution-ReLU-Pooling-Affine-ReLU-Affine-Softmax”。CNN 网络模型实现架构如图 6 所示。由图 6 可看到,对其中各组成部分拟展开阐释分述如下。

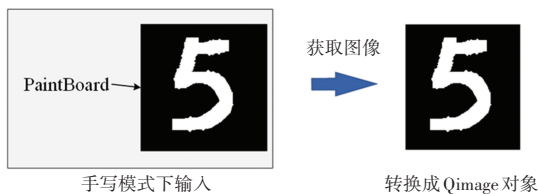


图 3 手写模式的图像采集过程

Fig. 3 Image acquisition process in handwriting mode

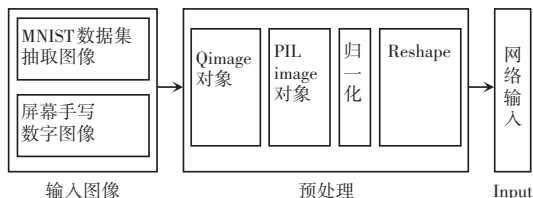


图 4 手写体数字图像的预处理过程

Fig. 4 Preprocessing of handwritten digital images

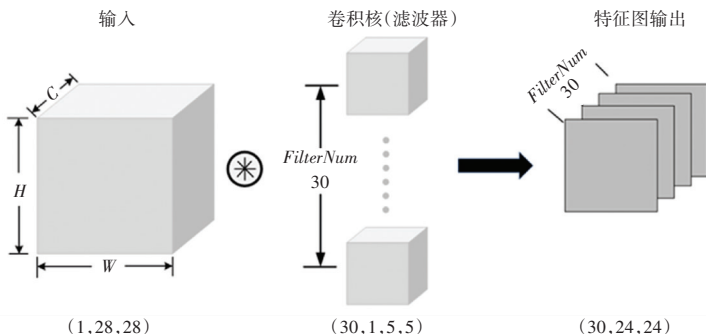


图 5 Convolution 层实现结构

Fig. 5 Convolution layer implementation structure

(1) 卷积层。在通过前一步的图像采集与预处理后,输入到卷积层 (Convolution 层),该层的参数是通过反向传播算法优化得到的,随后再将输入元素进行卷积运算输入到激活函数中,得到该层的输出特征图,卷积运算见式(1):

$$X_j^l = \text{ReLU}\left(\sum_i X_i^{l-1} * k_{ij}^l + b\right) \quad (1)$$

其中, X_j^{l-1} 表示卷积层特征图被 j 个卷积所覆盖的元素; k_{ij}^l 表示该层卷积核中的元素; b 表示偏置; ReLU 表示激活函数。

卷积层的作用相当于图像处理过程中的“滤波器”运算,以此获取图像的特征,本文设计的卷积层的卷积核(滤波器)的数量为 30,大小为 5×5,步幅为 1,填充为 0。

(2) ReLU 层。ReLU 层可以理解为激活函数层,当输入为正的时候,导数不为零,从而进行基于梯度的学习,对于图像数据输入而言,ReLU 函数很适合进行模型中非线性映射学习,减少了参数之间的相互依存关系,避免了过拟合现象的发生^[8]。

ReLU 函数的数学公式为:

$$y = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (2)$$

(3) 池化层 (Pooling 层)。可以缩小图像尺寸,减少运算量,通常在卷积层之后会得到维度很大的特征,将特征切分为多个区域,取其最大值,得到新的、维度较小的特征,并在偏置处理后通过激活函数输出,如式(3)所示:

$$X_j^l = \text{ReLU}(\mu_j^{l-1} + b) \quad (3)$$

其中, μ_j^{l-1} 是卷积层图像块降采样后得到的输出; X_j^l 是该层输出,也是下一层的输入元素; ReLU 表示激活函数。

本文实现的池化层,主要通过数据展开、求最大值、Reshape 成规范大小三个步骤来实现。其中,数据展开是为了简化本网络中的向量运算,将四维向量展开成二维向量,便于数据处理;接着,求出每一行的最大值,作为输出的元素;最后,将先前处理后的元素 Reshape 成一个 (30, 1, 12, 12) 的四维向量,输入到下一层。

(4) Affine 层。可以理解为全连接层, 在网络层中的正向传播中进行矩阵乘积运算, 主要是进行神经网络中的加权运算与偏置运算, 在本文系统设计中运算为: $np.dot(X, W) + B$ 。

(5) Softmax 层。作为最后的输出层, 主要用来将前一层 (全连接层) 的输出作为输入值进行正规

化, 调整到 0~1 之间后再输出, 用于实现最终的图像识别分类。此处需用到的数学公式为:

$$Softmax(y)_i = \frac{e^{y_i}}{\sum_{j=1}^n e^{y_j}} \quad (4)$$

其中, y_i 是前一层的输出目标值。

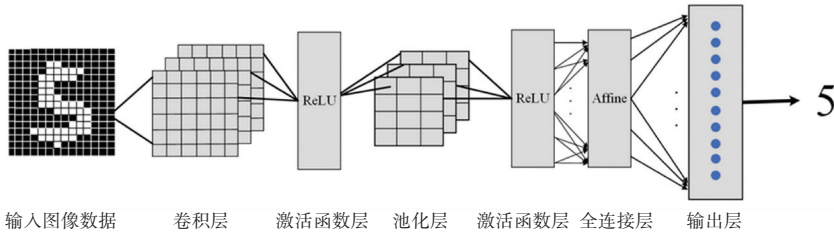


图 6 CNN 网络模型实现架构

Fig. 6 Implementation architecture of CNN network model

1.2.4 界面显示

本文实现的 PC 端的手写体数字识别系统界面是通过 PyQt 工具包搭建的, PyQt 是一个用于创建 GUI 应用程序的跨平台工具包, 可将 Python 与 Qt 库融为一体。也就是说, PyQt 允许使用 Python 语言调用 Qt 库中的 API。本文使用 Python3.0 版本在 VScode 编辑器进行手写体数字识别 GUI 应用开发。

通过 Qt 界面生成器 (Qt designer) [9], 可以将所需要的控件进行可视化的拖拽放置, 大大提高了界面设计效率。手写体数字识别系统设计界面如图 7 所示。由图 7 可看到, 本文的系统界面主要是 2 个部分。第一部分是手写体数字识别模式选择, 模式选择的布局就是通过 QLabel、QComboBox 以及 QPushButton 控件组成的。其中, QComboBox 下拉框组件实现“MNIST 随机抽取”与“鼠标手写输入”模式选择, 3 个 QPushButton 按钮用于实现 MNIST 图片抽取、清除数据、识别事件。

第二个部分是显示区域, 主要是实现所识别手写体数字图像的显示与识别结果、识别率的显示。通过一个 QLabel 组件用于显示当前所识别的手写体数字图像, 后续的图像采集和预处理过程都是通过 QImage 对象变换实现的。



图 7 手写体数字识别系统设计界面

Fig. 7 Design interface of handwritten digit recognition system

2 仿真结果与分析

基于上述的 GUI 界面与网络模型的实现, 本小节主要针对实验仿真结果进行分析, 并且得出有效的实验结论, 证明本文所实现的手写体数字识别系统有着优良性能。

实验仿真测试分为 2 部分。第一部分是按照“MNIST 随机抽取”识别模式进行仿真测试, 第二部分是按照“鼠标手写输入”识别模式进行仿真测试。2 种模式下分别进行 10 组手写体数字图像识别仿真测试, 每组选取 10 张手写体图像进行识别。

在第一种识别模式下, 通过 MNIST 数据集连续抽取 10 张图片进行识别, 分别进行 10 组图像随机抽取实验, 以每组的识别率均值作为本组实验数据; 在第二种模式下, 通过本系统的 GUI 界面连续手写输入 0~9, 共 10 个数字进行识别, 重复实验 10 组, 以每组的识别率均值作为本组实验数据。MNIST 随机抽取识别率如图 8 所示, 手写输入的识别率如图 9 所示。

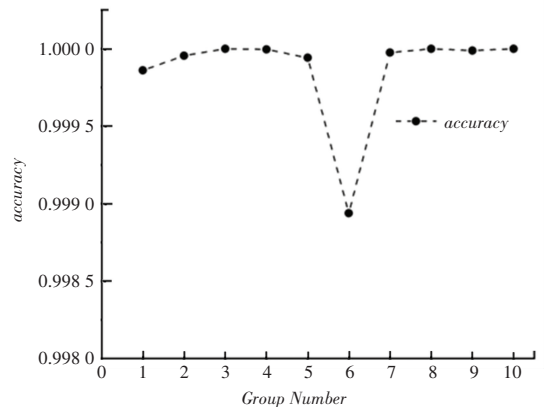


图 8 MNIST 随机抽取识别率

Fig. 8 Recognition rate of MNIST random extraction

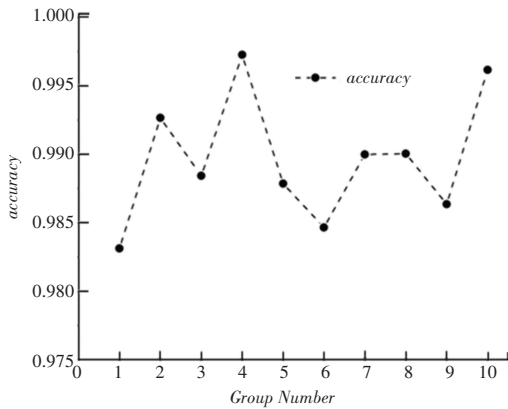


图9 手写输入的认可率

Fig. 9 Recognition rate of handwritten input

实验结果显示, MNIST 随机抽取的 10 组实验图像识别率都在 99.9% 以上, 识别准确率很高; 相比之下, 鼠标手写输入的仿真结果中, 识别率分布在 98.75% ~ 99.75% 区间, 相较于 MNIST 数据集测试的识别率而言, 有所降低。

对于手写输入模式下的手写体数字识别, 在仿真实验中可以观察到不同手写数字的识别率会出现较大差异, 如图 10 所示。由图 10 可看到, 数字中“0”、“1”、“2”、“3”、“5”、“6”、“7”、“8”的识别率都较稳定, 在 99% 左右, 而数字“4”的识别率在 97.5% 左右, 数字“9”的识别率在 94% 左右, 识别精确度相较于其他数字较低。

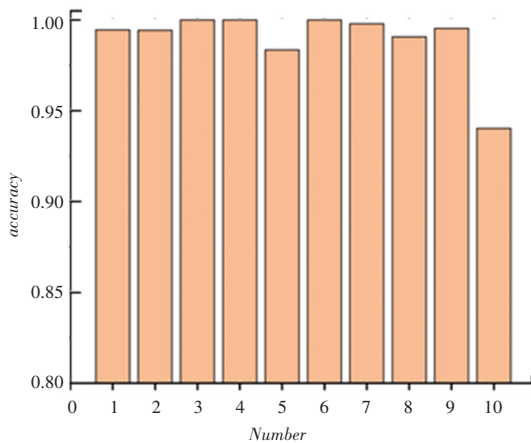


图10 数字0~9的识别率

Fig. 10 Recognition rate of numbers 0 ~ 9

以数字“9”为例, 数字“9”的手写体与“1”、“7”数字的形体十分相似, 对识别造成了干扰。误判结果如图 11 所示, 即使识别率达到了 99.94% 但是却误判成数字“1”, 也就是本文设计实现得到的 CNN 深度学习网络对容易混淆的数字手写体的识别会出现误判现象^[10]。即使是人为判断, 因个体差异也会出现对数字的错误辨别, 对于机器识别出现特殊字

体的误判也在合理范围内。



图11 数字“9”的误判

Fig. 11 Misjudgment of the number “9”

3 结束语

从 2 种识别模式下的仿真结果可以看出, MNIST 数据集抽取的图像, 系统识别率接近 99.9%, 说明本文实现的 CNN 网络的识别性能优良, 简单易行, 具有很好的实用性, 基本达到了高精度目标识别的要求; 而鼠标手写输入下的图像, 系统识别率却会在 98% ~ 99% 之间波动, 同时也会根据用户书写规范程度有很大关系, 这说明了本文的 CNN 网络模型出现了过拟合现象, 只针对 MNIST 数据集中的数据有很好的识别性能, 而对于个体差异的书写体却不能做到精准识别。因此需要对网络模型进行再深层的优化, 提高可信度。

参考文献

- [1] LIU Chenglin, NAKASHIMA K, SAKO H, et al. Handwritten digit recognition: benchmarking of state-of-the-art techniques [J]. Pattern Recognition, 2003, 36(10): 2271-2285.
- [2] 肖驰. 四种机器学习算法在 MNIST 数据集上的对比研究 [J]. 智能计算机与应用, 2020, 10(12): 185-188.
- [3] BOTTOU L, CORTES C, DENKER J S, et al. Comparison of classifier methods: A case study in handwritten digit recognition [C] // International Conference on Pattern Recognition. Jerusalem, Israel: IEEE Computer Society, 1994: 77-82.
- [4] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [C] // International Conference on Neural Information Processing Systems. Doha, Qatar: Curran Associates Inc., 2012: 1097-1105.
- [5] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition [J]. CoRR abs/1409.1556, 2014.
- [6] 郑继燕. 基于 CNN 的手写数字识别与试卷管理系统设计 [D]. 北京: 北京邮电大学, 2020.
- [7] 肖文鹏. 用 PyQt 进行 Python 下的 GUI 开发 [J]. 中文信息: 程序春秋, 2002(07): 73-75.
- [8] 曲景影, 孙显, 高鑫. 基于 CNN 模型的高分辨率遥感图像目标识别 [J]. 国外电子测量技术, 2016, 35(08): 45-50.
- [9] 桑晓丹, 郭锐. 基于 PyQt5 的数字图像处理实验平台设计 [J]. 电子技术与软件工程, 2021(18): 129-130.
- [10] 刘辰雨. 基于卷积神经网络的手写数字识别研究与设计 [D]. 成都: 成都理工大学, 2018.