卷积神经网络识别汉字验证码

范 望,韩俊刚,苟 凡,李 帅
FAN Wang, HAN Jungang, GOU Fan, LI Shuai

西安邮电大学 研究生学院,西安 710121
School of Postgraduate, Xi'an University of Posts and Telecommunications, Xi'an 710121, China

FAN Wang, HAN Jungang, GOU Fan, et al. Chinese character CAPTCHA recognition based on convolution neural network. Computer Engineering and Applications, 2018, 54(3):160-165.

Abstract: CAPTCHAs (Completed Automated Public Turing test to tell Computers and Humans Apart) have already been widely applied in various fields of social life. Automatic recognition of CAPTCHAs consisting of English letters and Arabic numerals has already reached an advanced level. While with general methods identifing the CAPTCHAs consisting of Chinese characters seems too difficult and the accuracy needs to be promoted. This paper mainly proposes a method of automatic identification CAPTCHAs which is based on convolutional neural network to improve the accuracy of characters recognition. In order to improve the generalization performance of the model by which adopting the framework of Keras convolution neural network and designing of multilayer convolution to extract deep-layer image information of which identifing Chinese characters CAPTCHAs and alphanumeric CAPTCHAs respectively. The experimental results indicate that the accuracy of identification has been promoted remarkably. The identification rate of Chinese characters is up to 99.4%. Meanwhile, the maximum of the identification rate of alphanumeric four-character CAPTCHAs is as high as 99.3%. These findings show that the Deep Neural Network possesses an excellent perceptivity against complex structures. It can be seen from the comparative experiments that the framework of Keras convolution neural network has better performance than other frameworks in CAPTCHAs recognition.

Key words: CAPTCHAs(Completed Automated Public Turing test to tell Computers and Humans Apart); Chinese character CAPTCHAs; CNN; Keras framework

摘 要: 验证码今已广泛应用在各个领域,常见的英文字母与数字组合的验证码自动识别准确率已达到较高的水准,而汉字因其字符复杂,用传统方法进行自动识别难度很大。提出一种基于卷积神经网络的验证码自动识别方法来提高字符的识别准确率。采用Keras卷积神经网络框架,设计多层卷积来提取深层次图像信息,分别对汉字验证码和字母数字验证码进行识别,以提高模型的泛化性。实验结果表明用该方法汉字验证码的单字识别率已达到99.4%;传统四字符字母数字验证码的识别率最高达到99.3%。这一结果表明深度神经网络对验证码复杂结构的感知能力很强大,通过对比实验发现Keras框架在验证码识别领域有较好效果。

关键词: 验证码;汉字验证码;CNN;Keras框架
文献标志码: A **中图分类号:** TP391 **doi:** 10.3778/j.issn.1002-8331.1706-0304

1 引言

随着互联网技术的快速发展,网络安全成了人们关注的一个重点内容。验证码主要是用于区分机器自动程序与人类用户的差异性,抵御恶意程序,防止滥用网络资源。验证码自动识别技术可以提高现有验证码安全性,并帮助设计出更安全的验证码,进而有效地确保网络安全。

国内外学者在验证码的识别领域已有了较多的研

作者简介: 范望(1994—),男,硕士研究生,研究领域为基于深度学习的图像识别,医学图像识别,E-mail:978683267@qq.com;韩俊刚(1943—),男,二级教授,研究领域为软件和硬件的形式化验证,图形处理器和新型计算机体系结构;苟凡(1993—),男,硕士研究生,研究领域为基于深度学习的图像识别;李帅(1992—),男,硕士研究生,研究领域为计算机图像识别。

收稿日期: 2017-06-23 **修回日期:** 2017-08-08 **文章编号:** 1002-8331(2018)03-0160-06

究。吕霁综合了一类神经网络方法,并将其应用于验证码识别研究[1];刘欢整体研究了卷积神经网络在可分割与不可分割验证码识别上的应用[2];王璐则对粘连字符验证码的识别进行了研究[3];LvYanping 等人对汉字验证码使用卷积神经网络识别进行深度研究[4];Garg 和 Pollett 提出了使用深层神经网络,开发一个能够打破所有基于字符的验证码的单一神经网络[5];Shen Yunhang 等人对汉字 Touclick CAPTCHAs 提出了一种多尺度角结构模型识别[6]。

现如今,深度学习技术被广泛应用于各领域,硬件能力与算法的改进更是使其形成良性循环。在图像领域,卷积神经网络采用随机梯度下降算法(Stochastic GradientDescent,SGD)[7]和 GPU 加快训练过程,从而使得训练大量的图像数据更为方便。卷积神经网络在以往被用来研究验证码识别并不少见,但往往是基于 Caffe、Mxnet 或者 Tensorflow 框架,本文所有实验均采用 Keras[8]神经网络框架,旨在探究一种适合识别验证码的并且明显提升准确率的神经网络框架。Keras 实质上是一个高层神经网络库,它是基于 Theano 和 Tensorflow 的深度学习库,是一个极简单和高度模块化的神经网络库。本文设计新的网络结构,在 Keras 网络框架下使用 CNN 分析研究传统字母数字验证码与汉字验证码[9],并取得了很好的效果。

2　数据准备

本文共设置了两个实验来验证在 Keras 框架下使用深层神经网络对验证码识别性能的提升。首先,训练传统字母数字验证码,并将测试结果进行统计;其次,对汉字验证码进行训练与测试。

2.1　传统的字母数字验证码

传统的字母数字验证码(如图 1 所示)的数据来源于各大公司网站,本文采取网络爬虫爬取数据的方式,对各种字母数字验证码进行获取,使数据具有代表性。其中大部分来自于百度,腾讯,网易云音乐,小部分来自于其他网站以及自己模仿生成的一些四字符验证码,数据的多样性保证了系统的稳定性。

图 1　传统的字母数字验证码

最后获取训练数据 50 000 张,测试数据 1 000 张。采取人工方式为每张验证码打上了标记(Label),确保了数据的准确性。

2.2　汉字验证码

为了对比,汉字验证码的实验是以腾讯 qq 安全中心找回密码出现的点击验证为蓝本。点击验证码(如图 2 所示)识别的思想就是首先将大图中的每个字分割下来进行处理,然后放到神经网络里边去做预测;再将小图中的每个字分割下来进行处理做预测,然后将大小图的预测值进行匹配(计算两个预测矩阵的最短欧几里德距离[10]),匹配成功则返回大图坐标,点击即可。

图 2　点击验证码

要做到点击验证码识别的效果,必须保证训练的模型尽可能包含所有汉字的模式,包括汉字的种类、样式以及各种变换。本文训练数据来源于自主生成的验证码(如图 3 所示)。虽然现存汉字大约有 91 251 个,但是汉字验证码都是一些常用汉字,所以本文提取常用的 3 500 个汉字,将单个汉字对应的标记转换为汉字在 txt 文件中的序号。随后生成单字验证码进行训练,这样不仅保证了汉字种类的涵盖,同样保证了模型不会存在数据冗余。

图 3　汉字验证码

汉字的样式不同主要是字体的不同,在选定 3 500 个汉字之后,为了使字体覆盖全面,使用微软字体库,方正字体库的字体相结合并加上扭曲、旋转,噪声生成了 32×32 大小的单字验证码,其中 600 000 张用于训练,20 000 张用于测试。

3　实验过程

本文采用卷积神经网络开源框架 Keras 进行实验研究。服务器配置如下:

14.04-Ubuntu,CP 为 Intel 酷睿 i7-5820K 处理器,主频 3.30 GHz,32 GB 内存,GPU 为 NVIDIA GeForce GTX 970,4 GB 显存。

3.1　实验一

进行深度学习的 CNN 网络结构如下,本文将用此结构对传统的字母数字验证码(如图 1 所示)进行训练,训练的网络结构如图 4 所示。

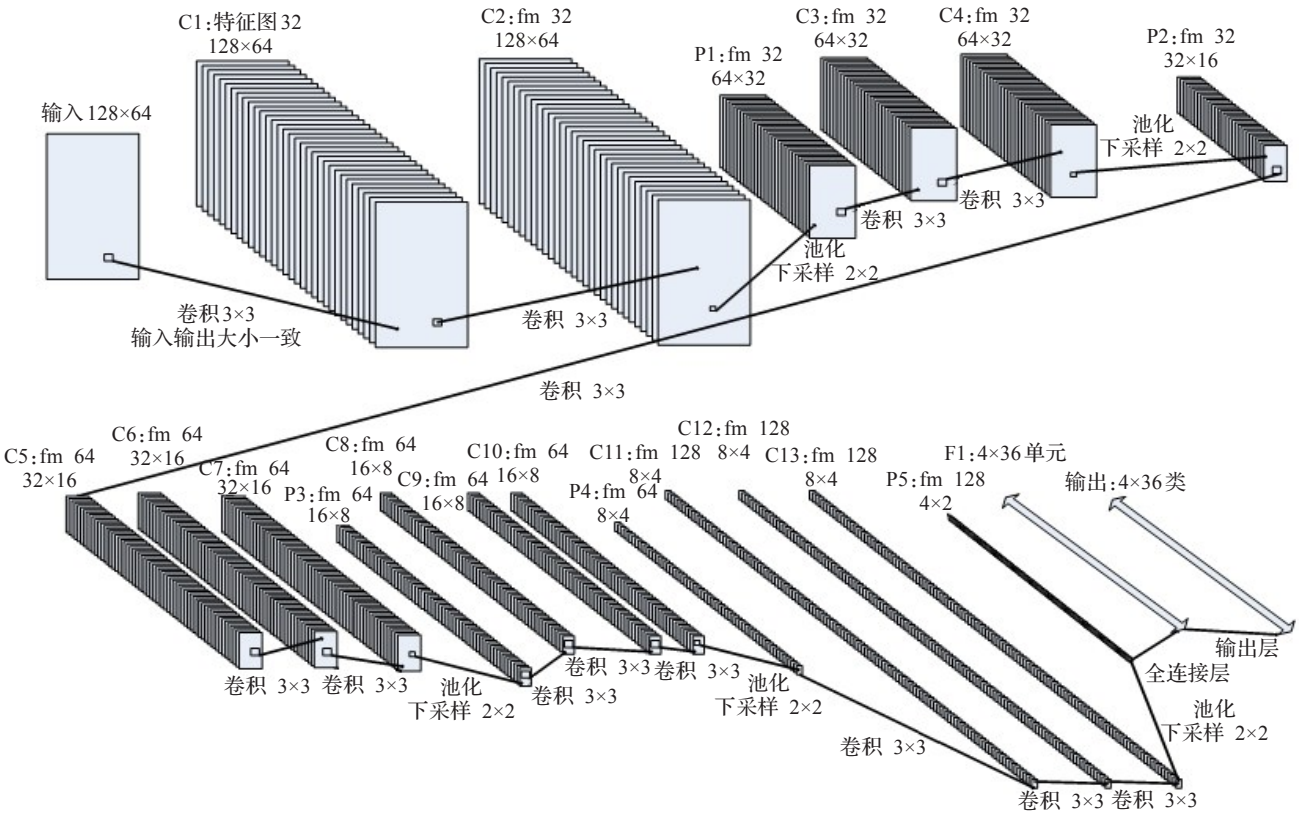

图4　训练字母数字验证码的网络结构

(1)对图片矩阵进行归一化等基本处理后,将大小为128×64的验证码图片送入网络。

(2)实验的CNN包含13个卷积层,5个池化层和一个全连接层。

① C1层卷积核大小为3×3,因选择卷积方式为输入输出尺寸一致,所以C1卷积层包含32个大小为128×64的特征图。每个卷积层之后接特征映射层,采用ReLU[11]激活函数,实现特征映射的位移不变性。

② C2将激活后的C1作为输入,再做卷积;C2包含32个大小为128×64的特征图,其卷积核大小为3×3。

③ P1层是将激活后的C2作为输入的池化层,P1包含32个大小为64×32的特征图,其局部感受区域为2×2。

④ C3将P1作为输入,该卷积层包含32个大小为64×32的特征图,其卷积核大小为3×3。

⑤ C4将激活后的C3作为输入,该卷积层包含32个大小为64×32的特征图,其卷积核大小为3×3。

⑥ P2层是将激活后的C4作为输入的池化层,P2包含32个大小为32×16的特征图,其局部感受区域为2×2。

⑦ C5将P2作为输入,该卷积层包含64个大小为32×16的特征图,其卷积核大小为3×3。

⑧ C6将激活后的C5作为输入,该卷积层包含64个大小为32×16的特征图,其卷积核大小为3×3。

⑨ C7将激活后的C6作为输入,该卷积层包含64个大小为32×16的特征图,其卷积核大小为3×3。

⑩ P3层是将激活后的C7作为输入的池化层,P3包含64个大小为16×8的特征图,其局部感受区域为2×2。

⑪ C8将P3作为输入,该卷积层包含64个大小为16×8的特征图,其卷积核大小为3×3。

⑫ C9将激活后的C8作为输入,该卷积层包含64个大小为16×8的特征图,其卷积核大小为3×3。

⑬ C10将激活后的C9作为输入,该卷积层包含64个大小为16×8的特征图,其卷积核大小为3×3。

⑭ P4层是将激活后的C10作为输入的池化层,P4包含64个大小为8×4的特征图,其局部感受区域为2×2。

⑮ C11将P4作为输入,该卷积层包含128个大小为8×4的特征图,其卷积核大小为3×3。

⑯ C12将激活后的C11作为输入,该卷积层包含128个大小为8×4的特征图,其卷积核大小为3×3。

⑰ C13将激活后的C12作为输入,该卷积层包含128个大小为8×4的特征图,其卷积核大小为3×3。

⑱ P5层是将激活后的C13作为输入的池化层,P5包含128个大小为4×2的特征图,其局部感受区域为2×2。

⑲ F1全连接层将P5作为输入,包含4×36=144个连接单元。

⑳ 输出层采用softmax[12]分类,每一个字符分为36类(该验证码用到了26个字母和10个数字,其中字母大小写不作区分),4字符同时输出;softmax函数如下:

$$\sigma(Z)_j=\frac{e^{Z_j}}{\sum_{k=1}^{K}e^{Z_k}},\ j=1,2,\cdots,k \tag{1}$$

其中,K 为类别数,图5表示一个多输入单输出神经元。

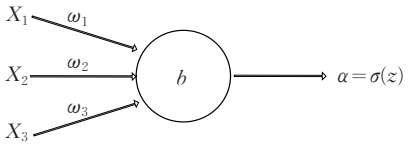

图5　多输入单输出神经元

图5中 $z=\sum_j \omega_j x_j+b$ 为全连接层输出，X_1，X_2，X_3 表示输入，ω 表示权值，b 表示偏置。每一类将会有一个属于其本身的 z，softmax 会计算出该类的 $\sigma(z)$，$\sigma(z)$ 表示该类存在的概率。比如在本实验中，每个字符通过神经网络计算会有36个分类，则会有36个 $\sigma(z)$ 与之对应，并且和为1。随后将取 $\sigma(z)=\max(\sigma(z))$，对应的损失函数如下：

$$C=-\frac{1}{n}\sum_x y\ln\alpha+(1-y)\ln(1-a) \tag{2}$$

其中 y 表示目标值，$\alpha=\sigma(z)$ 表示实际值，$z=\sum_j \omega_j x_j+b$，x 表示样本，n 为训练样本总数。

对上式求导得：

$$\begin{cases}\dfrac{\partial C}{\partial \omega_j}=\dfrac{1}{n}\sum_x X_j\left(\sigma(z)-y\right)\\[2ex] \dfrac{\partial C}{\partial b}=\dfrac{1}{n}\sum_x\left(\sigma(z)-y\right)\end{cases} \tag{3}$$

随后对权值进行优化。

3.2　实验二

进行深度学习的CNN网络结构如下，本文将用此结构对汉字验证码(如图3所示)进行训练：

(1)对图片矩阵进行归一化等基本处理后，将大小为32×32的验证码图片送入网络。

(2)实验的CNN包含10个卷积层，4个池化层和一个全连接层(如图6所示)。

① C1层卷积核大小为3×3，因选择卷积方式为输入输出尺寸一致，所以C1卷积层包含32个大小为32×32的特征图。每个卷积层之后接特征映射层，采用ReLU激活函数，实现特征映射的位移不变性。

② C2将激活后的C1作为输入，再做卷积；C2包含32个大小为32×32的特征图，其卷积核大小为3×3。

③ P1层是将激活后的C2作为输入的池化层，P1包含32个大小为16×16的特征图，其局部感受区域为2×2。

④ C3将P1作为输入，该卷积层包含64个大小为16×16的特征图，其卷积核大小为3×3。

⑤ C4将激活后的C3作为输入，该卷积层包含64个大小为16×16的特征图，其卷积核大小为3×3。

⑥ P2层是将激活后的C4作为输入的池化层，P2包含64个大小为8×8的特征图，其局部感受区域为2×2。

⑦ C5将P2作为输入，该卷积层包含128个大小为

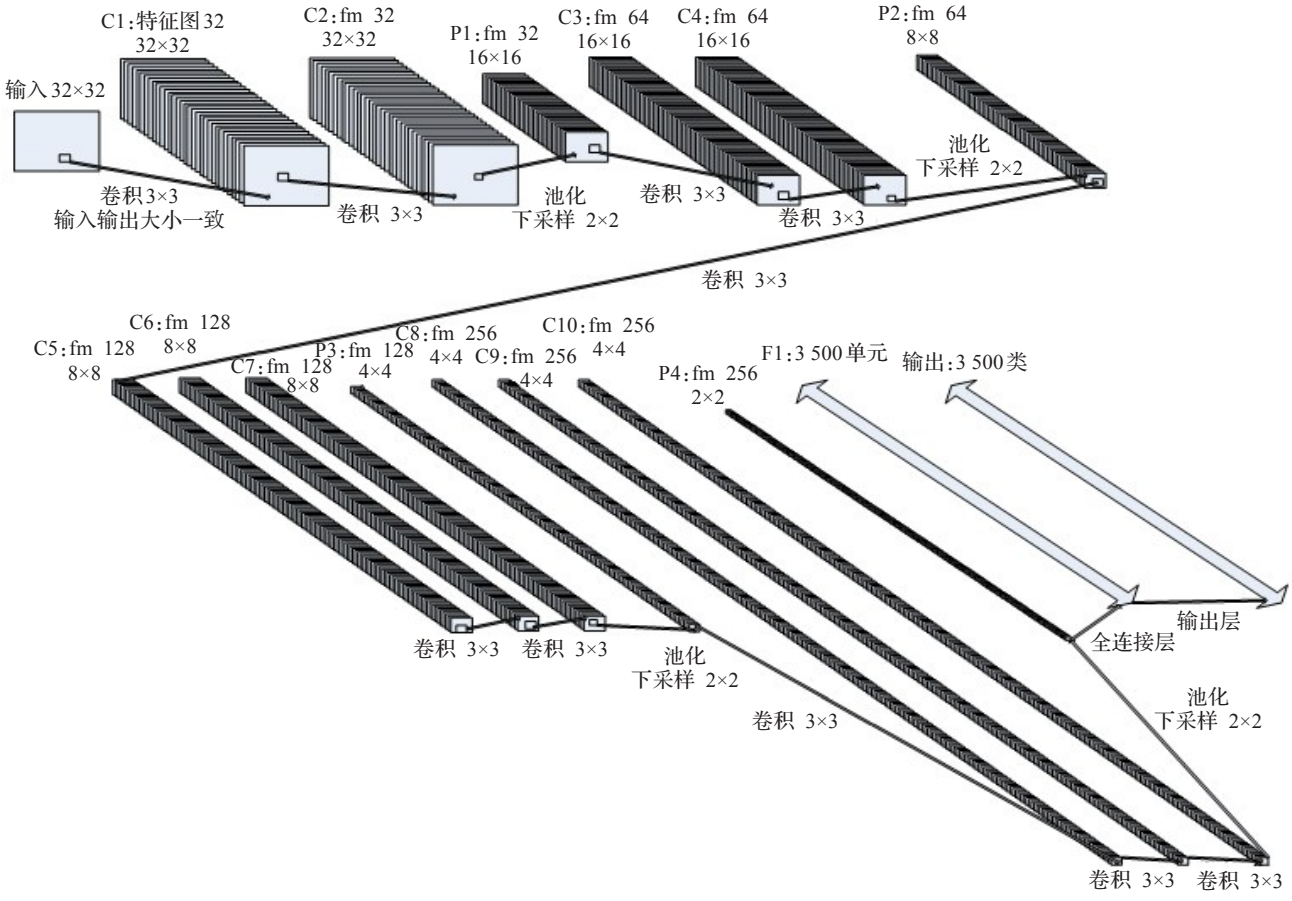

图6　训练汉字验证码的网络结构

8×8的特征图,其卷积核大小为3×3。

⑧ C6将激活后的C5作为输入,该卷积层包含128个大小为8×8的特征图,其卷积核大小为3×3。

⑨ C7将激活后的C6作为输入,该卷积层包含128个大小为8×8的特征图,其卷积核大小为3×3。

⑩ P3层是将激活后的C7作为输入的池化层,P3包含128个大小为4×4的特征图,其局部感受区域为2×2。

⑪ C8将P3作为输入,该卷积层包含256个大小为4×4的特征图,其卷积核大小为3×3。

⑫ C9将激活后的C8作为输入,该卷积层包含256个大小为4×4的特征图,其卷积核大小为3×3。

⑬ C10将激活后的C9作为输入,该卷积层包含256个大小为4×4的特征图,其卷积核大小为3×3。

⑭ P4层是将激活后的C10作为输入的池化层,P4包含256个大小为2×2的特征图,其局部感受区域为2×2。

⑮ F1全连接层将P4作为输入,包含3 500个连接单元。

⑯ 输出层采用softmax分类,每一个字符分为3 500类(该验证码用到了3 500个汉字)。

为了更好地提取验证码特征,本文设计了上述CNN网络结构,该网络结构对此有着显著的优势。首先,每个卷积层中使用较小的卷积核(3×3)来捕获字符复杂的笔画特征。其次,使用更多的层来捕获两类验证码深层结构信息。

对于字符数字四字符验证码,在输出时直接获取softmax预测概率最大的一类;考虑到汉字的复杂性,在做点击验证识别时,将大图与小图逐个进行预测,然后将大小图的softmax预测矩阵进行匹配,即计算两个预测矩阵的欧几里德距离,距离最短的两个预测矩阵所对应的图片即为匹配成功的图片。

4　实验结果

针对实验一参数配置:

优化器采用Adam[13],Adam优化器是由Kingma和Lei Ba在Adam:A method for stochastic optimization这篇文章中提出来的,它相对于其他优化方法对存储器要求不高,并且非常适合数据和参数较大的问题。超参数设置将采用Adam的默认值,选用的块大小(batch size)为224,采用GPU模式加速训练过程,最大训练迭代次数为100。训练结果如表1所示。

表1　实验一训练结果

迭代次数	测试精度/%
25	75.27
50	92.16
75	98.22
100	99.25

Garg和Pollett[5]提出的深层神经网络与本文网络结构得到的结果对比如表2所示。

表2　实验一不同方法结果对比

实验名称	实验结果/%
深层网络	99.00
本文算法	99.25

针对实验二参数配置:

优化器仍然采用Adam,超参数设置部分改变。本文将学习率(learning rate)lr设置为0.000 1,其余的采用Adam的默认值,选用的块大小(batch size)为384,采用GPU模式加速训练过程,最大训练迭代次数为25。训练结果如表3。

表3　实验二训练结果

迭代次数	测试精度/%
5	81.20
10	93.24
20	97.71
25	99.47

LvYanpin[4]等人提出的基于卷积神经网络的方法与本文的卷积网络结构得到的实验结果对比如表4所示。

表4　实验二不同方法结果对比

实验名称	实验结果/%
基于卷积神经网络	97.72
本文方法	99.47

实验二以点击验证为蓝本,对于汉字的分割以及处理采用HSV[14]与边缘检测[15]结合的方法。边缘检测是基于提取图像中不连续部分的特征,根据闭合的边缘确定区域的一种方法。John Canny在A Computational Approach to Edge Detection这篇文章中详细对边缘检测这种计算方法进行说明,通过这个思路提取的汉字效果较好,但对于一些验证码背景会出现一些不连续部分,比如像一些山的棱角,这样边缘检测会提取出部分背景,造成误差。本文采用HSV方法就会避开这个问题,图片通常是以RGB格式存储,难以分离出颜色,可以将其转换到HSV空间进行颜色分离。HSV事实上就是描述一种比RGB更加详细准确的颜色之间的联系,同时也简化了一些复杂的计算。其中H指hue(色相),是色彩的最基本属性,通常就是指颜色名称;S指saturation(饱和度),是表示色彩的纯度,值越高色彩越纯;V指value(色调)。本文使用HSV分割的思路是使用OPENCV将图片从RGB空间转化到HSV空间;由于点击验证验证码中的汉字颜色有限,所以很容易给出待检测区域的H、S、V值,用形态学的膨胀和腐蚀算子,可以使检测出的区域形成一个整体;找出物体的轮廓,再找出检测区域的中心,依据这个中心对检测区域画框,截取。为了提高分割效果,本文借鉴了张国权[14]等提出的在HSV空

间中颜色距离的定义,并根据定义的颜色距离,用Sobel梯度算子的变形对彩色图像在颜色分量上求出分割边界,这种方法得到的分割汉字更清晰。

采用边缘检测与HSV结合的办法,有效地使分割的汉字清晰,使验证码自动识别率提高,通过这种方法得到的数据如图7所示。点击验证码测试的结果如表5所示。

图7　处理后数据图

表5　点击验证码测试结果

测试内容	值
分割处理后数据/张	500
识别准确率/%	87
实际处理成功数据/张	437
实际数据识别准确率/%	92.8

5　总结

本文设计基于Keras神经网络框架的CNN,有效地提高了汉字验证码以及常见字符验证码的识别率;即使对训练数据加入扭曲、旋转、背景噪声,预测时依然表现出很强的鲁棒性。鉴于点击验证识别有待提高,后期将会继续加大训练数据,囊括各路汉字验证码进行训练,旨在建立真正意义上的汉字验证码高效自动识别。最终的目标是不再区分字符种类,实现所有验证码高效自动识别系统。

参考文献:

[1] 吕霁.基于神经网络的验证码识别技术研究[D].福建泉州:华侨大学,2015.

[2] 刘欢,邵蔚元,郭跃飞.卷积神经网络在验证码识别上的应用与研究[J].计算机工程与应用,2016,52(18):1-7.

[3] 王璐,张荣,尹东,等.粘连字符的图片验证码识别[J].计算机工程与应用,2011,47(28):150-153.

[4] Lv Yingping, Cai Feipeng, Lin Dazhen, et al.Chinese character CAPTCHA recognition based on convolution-neural network[C]//Proceedings of the IEEE Congress on Evolutionary Computation(CEC).Vancouver,BC,Canada:IEEE,2016:4854-4859.

[5] Garg G,Pollett C.Neural network CAPTCHA crackers[C]//Proceedings of the Future Technologies Conference(FTC).San Francisco,CA,USA:IEEE,2016:853-861.

[6] Shen Yunhang, Ji Rongrong, Cao Donglin, et al.Hacking Chinese touclick CAPTCHA by multiscale corner structure model with fast pattern matching[C]//Proceedings of the ACM International Conference on Multimedia(MM'14).New York,NY,USA:ACM,2014:853-856.

[7] Bottou L.Large-scale machine learning with stochastic gradient descent[C]//Proceedings of COMPSTAT' 2010.Heidelberg,Germany:Physica-Verlag HD,2010:177-186.

[8] Ketkar N.Deep learning with python[M].California:Apress,2017:95-109.

[9] Algwil A, Ciresan D, Liu Beibei, et al.A security analysis of automated chinese turing tests[C]//Proceedings of the 32nd Annual Conference on Computer Security Applications.New York,NY,USA:ACM,2016:520-532.

[10] 周志华.机器学习[M].北京:清华大学出版社,2016.

[11] Hara K,Saito D,Shouno H.Analysis of function of rectified linear unit used in deep learning[C]//Proceedings of the International Joint Conference on Neural Networks(IJCNN).Killarney,Ireland:IEEE,2015:1-8.

[12] Zang Fei,Zhang Jiangshe.Softmax discriminant classifier[C]//Proceedings of the 2011 Third International Conference on Multimedia Information Networking and Security(MINES).Shanghai,China:IEEE,2011:16-19.

[13] Kingma D,Ba J.Adam:A method for stochastic optimization[EB/OL].(2017-01-30)[2017-05-10].http://arxiv.org/abs/1412.6980.

[14] 张国权,李战明,李向伟,等.HSV空间中彩色图像分割研究[J].计算机工程与应用,2010,46(26):179-181.

[15] Canny J.A computational approach to edge detection[J].IEEE Transactions on Pattern Analysis and Machine Intelligence,1986,8(6):679-698.