

基于改进 CNN 的铝轮毂背腔字符识别

程淑红, 周 斌

(燕山大学 电气工程学院, 河北 秦皇岛 066000)

摘 要: 铝轮毂背腔字符分辨率较低、背景噪声较大, 对其进行识别时不易提取几何特征和纹理特征。为此, 提出一种基于改进卷积神经网络(CNN)的字符识别方法。在原始 CNN 的基础上引入改进的 inception 结构对网络构架进行优化, 以提升计算资源的利用率, 并在保持网络计算资源不变的前提下增加网络的宽度和深度, 降低字符识别时间。实验结果表明, 该方法训练准确率达 99% 以上, 识别准确率达 98.5%, 识别效果优于支持向量机、BP 神经网络等方法。

关键词: 卷积神经网络; inception 结构; 网络构架; 背腔字符; 损失函数优化

中文引用格式: 程淑红, 周斌. 基于改进 CNN 的铝轮毂背腔字符识别[J]. 计算机工程, 2019, 45(5): 182-186.

英文引用格式: CHENG Shuhong, ZHOU Bin. Recognition of characters in aluminum wheel back cavity based on improved convolution neural network[J]. Computer Engineering, 2019, 45(5): 182-186.

Recognition of Characters in Aluminum Wheel Back Cavity Based on Improved Convolution Neural Network

CHENG Shuhong, ZHOU Bin

(School of Electrical Engineering, Yanshan University, Qinhuangdao, Hebei 066000, China)

[Abstract] It is difficult to extract geometric and texture features when recognizing the characters in the aluminium wheel back cavity because of its low resolution and strong background noise. Therefore, a character recognition method based on improved Convolution Neural Network(CNN) is proposed. On the basis of the original CNN, an improved inception structure is introduced to optimize the network architecture to improve the utilization of computing resources, increase the width and depth of the network and reduce the time of character recognition while keeping the network computing resources unchanged. Experimental results show that the training accuracy of this method is over 99% and the recognition accuracy is 98.5%, which is better than that of Support Vector Machine(SVM) and BP neural network.

[Key words] Convolution Neural Network(CNN); inception structure; network architecture; back cavity character; optimization of loss function

DOI: 10.19678/j.issn.1000-3428.0050060

0 概述

目标识别是图像处理中重要的组成部分,也是其主要的结果输出^[1]。在目标识别中,字符识别占据很大比例。随着社会的日益发展,新鲜事物层出不穷,需要应用目标识别技术的领域逐渐扩展,且应用的层次逐渐加深,应用的场景趋于复杂化。因此,简单的识别方法(如模板匹配)已无法满足目标识别的要求,这对识别技术提出了新的挑战。基于此,机器学习应运而生。文献[2]提出一种基于支持向量机(Support Vector Machine, SVM)分类的高效混合

图像去噪方法。在该方法中,输入噪声图像首先被划分成大量重叠的补丁,然后根据尺度不变特征从每个补丁中提取局部特征,用 SVM 分类后进行 2 种去噪,最后再将 2 种去噪进行混合。该方法能够取得较好的去噪效果,但耗时较多且特征提取较麻烦。为解决该问题,研究人员开始致力于研究更便捷、智能的机器学习方法,即人工神经网络和深度学习^[3]。文献[4]提出一种多层人工神经网络 LeNet-5,并利用反向传播算法对其进行训练。在此之后,神经网络得到迅速发展。文献[5]提出一种深度卷积神经网络(Convolution Neural Network, CNN) AlexNet,其

基金项目: 国家自然科学基金(61601400); 河北省博士后择优项目(B2016003027)。

作者简介: 程淑红(1978—), 女, 教授、博士, 主研方向为视觉检测、图像处理、水质监测; 周 斌(通信作者), 硕士研究生。

收稿日期: 2018-01-10 **修回日期:** 2018-03-30 **E-mail:** 3114884526@qq.com

在 LeNet 的基础上调整网络架构并加深网络深度。AlexNet 在 2012 年的 ImageNet 大赛上以明显优势夺得冠军。文献[6]提出一种针对图像分类的卷积神经网络 google Net,并建立 inception 结构,其使用 1×1 、 3×3 和 5×5 的卷积核代替 7×7 的卷积核,实验结果验证了该方法优异的分类性能。

本文基于改进的 inception 结构建立一种卷积神经网络,通过提取卷积特征对铝轮毂背腔字符进行分类识别,以降低图片的低分辨率对识别结果的影响,提高分类识别的精确度。

1 卷积神经网络

CNN 的基本结构主要包括 2 层^[7]:

1) 卷积层。使用设计好的卷积核对图片进行卷积,将卷积值加权并增加偏置,再通过一个激活函数得到卷积层。

2) 池化层,也称下采样层。利用图像的局部相关性原理对图像进行自抽样,从而降低数据处理量并保留有用信息。

除上述 2 层外,CNN 还包括全连接层和 softmax 分类层。这种网络结构能够降低反馈神经网络的复杂性,因此,CNN 得到了越来越多的关注。

1.1 卷积层

卷积层操作即局部特征的提取过程。卷积层中含有多个卷积核,卷积核对上一层特征图进行卷积以提取卷积特征,同一个卷积核的权值共享,不同的卷积核权值不同,提取的特征也不同。卷积层计算公式如式(1)所示。

$$X_j^l = f\left(\sum_{i \in M_j} X_i^{l-1} * k_{ij}^l + b_j^l\right) \quad (1)$$

其中, l 表示层数, i, j 表示特征图标签, M_j 表示输入层的感受野, X 表示特征图, k 表示卷积核, b 表示输出图的偏置。卷积层操作示意图如图 1 所示。

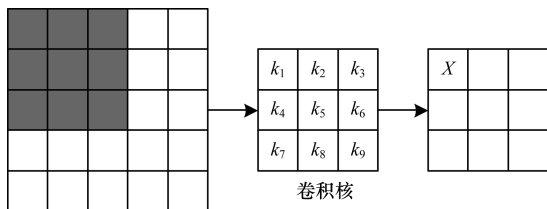


图 1 卷积层操作示意图

1.2 池化层

在 CNN 的卷积过程中,往往会因为卷积层的增加而导致数据处理量暴增,这将大幅增加数据处理的难度和时间。为降低特征图分辨率,减少特征维度,本文在卷积层后增加一个池化层。输入的特征图在经过池化层后个数不变,当池化层尺寸为 n 时,

输出的特征图是输入特征图的 $1/n$ 。池化层计算公式如式(2)所示。

$$X_k^l = f(\beta_k^l \text{down}(X_k^{l-1}) + b_k^l) \quad (2)$$

其中, $\text{down}()$ 表示池化函数, β 表示权重系数。最常见的池化方法有 2 种:平均值池化和最大值池化,本文使用最大值池化。池化层操作示意图如图 2 所示。

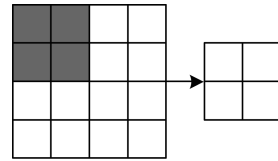


图 2 池化层操作示意图

1.3 反向传播

反向传播是神经网络有监督学习中的一种常用方法,其目的是不断更新卷积核权值,使该值朝着有利于分类的方向变化^[8]。反向传播主要基于梯度下降法向训练误差减小的方向调整。误差平方和损失函数计算公式如式(3)所示。

$$E^N = \frac{1}{2} \sum_{n=1}^N \sum_{k=1}^c (t_k^n - y_k^n)^2 \quad (3)$$

其中, N 表示样本数量, c 表示类别数量, E^N 表示样本总误差, t_k^n 表示与第 n 个样本相对应的第 k 维标签, y_k^n 表示第 n 个样本对应输出中预测的第 k 个输出。

2 改进的卷积神经网络

2.1 改进的 inception 结构

本文引入 inception 卷积层结构对整个 CNN 进行优化。inception 的核心思想是使用局部最优的稀疏结构取代原有的全连接方式,使用多层感知器对局部感知域进行计算,以最大程度地避免冗余,最终加快训练速度。原始 inception 结构如图 3 所示,其采用 1×1 、 3×3 和 5×5 的卷积核以及 3×3 的最大池化来提取上一层的卷积特征,这种结构的每个单元都单独接受上一层的输出为输入, 5×5 的卷积核每次需要处理的数据量为 $5 \times 5 = 25$,因此,其完成整幅图片的卷积操作时计算量非常大。

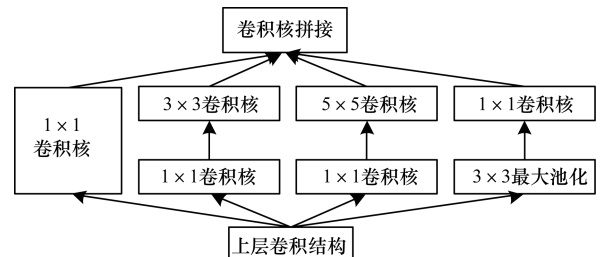


图 3 原始 inception 结构

为解决上述问题,研究人员根据“network in network”的思想将输入的特征量进行拼接操作,利用其非线性特性进行再取样和滤波,并尝试将 $n \times n$ 的卷积核拆分成 $1 \times n$ 和 $n \times 1$ 的卷积核,从而设计出适应于低分辨率图片的更小量级的 inception 网络结构^[9]。将 5×5 的卷积核拆分成 2 个 3×3 的卷积核,其计算量为 $3 \times 3 + 3 \times 3 = 18$,计算消耗降低了 $1 - 18/25 = 28\%$;再将 3×3 的卷积核拆分成 2 个 1×3 和 3×1 的卷积核,其计算量为 $3 + 3 = 6$,计算性能可以再次提升 $1 - 6/9 \approx 33\%$ 。因此,小卷积核结构能够大幅降低数据处理量,减少训练时间。小卷积核 inception 网络结构如图 4 所示。

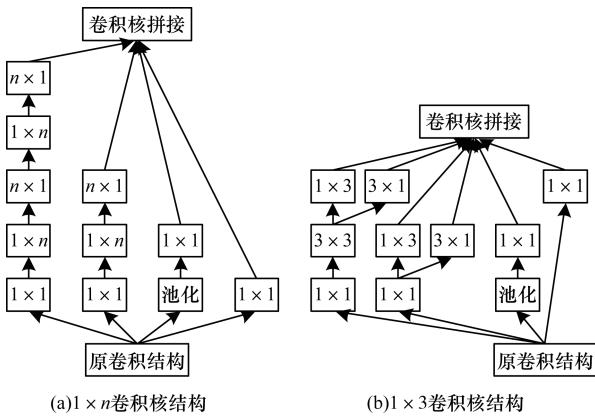


图 4 小卷积核 inception 结构

若特征数据的输入尺寸为 $28 \times 28 \times 256$, 共含 96 个 5×5 卷积核,在卷积核前加入 64 个 1×1 卷积核,则卷积核拆分后的运算复杂度如下:

5×5 卷积核为 $28^2 \times 1^2 \times 256 \times 64 + 28^2 \times 5^2 \times 64 \times 96 \approx 1.3 \times 10^8$ 。

3×3 卷积核为 $28^2 \times 1^2 \times 256 \times 64 + 28^2 \times 3^2 \times 64 \times 96 \times 2 \approx 1.0 \times 10^8$ 。

1×3 和 3×1 卷积核为 $28^2 \times 1^2 \times 256 \times 64 + 28^2 \times 3 \times 64 \times 96 \times 2 \approx 0.7 \times 10^8$ 。

因为铝轮毂背腔字符分辨率较低,且小卷积核 inception 结构对数据的处理能力优于原始 inception 结构,所以本文选用图 4 所示的 inception 结构设计 CNN,如图 5 所示。

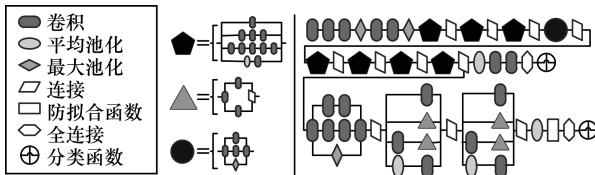


图 5 基于小卷积核 inception 结构的 CNN

改进后的 inception 结构具有良好的特征抽取能力,可以将字符图像的特征信息抽取为数字特征,并实现多分类,最终识别出不同的字符。本文 CNN 网络结构参数如表 1 所示。

表 1 网络结构参数

类型	卷积核尺寸/步长	输入尺寸	注释
卷积	$3 \times 3/2$	$299 \times 299 \times 3$	—
卷积	$3 \times 3/1$	$149 \times 149 \times 32$	—
卷积	$3 \times 3/1$	$147 \times 147 \times 32$	—
池化	$3 \times 3/2$	$147 \times 147 \times 64$	—
卷积	$3 \times 3/1$	$73 \times 73 \times 64$	—
卷积	$3 \times 3/2$	$71 \times 71 \times 80$	—
卷积	$3 \times 3/1$	$35 \times 35 \times 192$	—
inception 模块	—	$35 \times 35 \times 288$	3 组
inception 模块	—	$17 \times 17 \times 768$	5 组
inception 模块	—	$8 \times 8 \times 128$	3 组
池化	$8 \times 8/1$	$8 \times 8 \times 2048$	—
线性	—	$1 \times 1 \times 2048$	概率
Softmax	—	$1 \times 1 \times 1000$	分类输出

2.2 损失函数的最优选择

目前,在损失函数优化方面,常用方法有批量梯度下降法 (Batch Gradient Descent, BGD)^[10]、随机梯度下降法 (Stochastic Gradient Descent, SGD)^[11] 等。这些方法一般将线性回归函数假设如下:

$$h_{\theta} = \sum_{i=0}^n \theta_i x_i$$

$$\theta_i = \theta_i - \alpha \frac{\partial}{\partial \theta_i} J(\theta) \quad (4)$$

其中, x_i 表示样本特征 x 的第 i 个元素, θ_i 表示权重参数, α 表示步长,用于控制 θ 每次向使损失函数 $J(\theta)$ 变小方向迭代时的变化幅度。对于 BGD 方法而言,其对应的损失函数如式(5)所示。

$$J_{\text{train}}(\theta) = \frac{1}{2m} \sum_{i=1}^m [h_{\theta}(x_i) - y_i]^2 \quad (5)$$

其中, $h_{\theta}(x_i)$ 为假设函数, y_i 为样本输出 y 的第 i 个元素, m 为样本数目。

对 θ_j 求导可得:

$$\frac{\partial J(\theta)}{\partial \theta_j} = -\frac{1}{m} \sum_{i=1}^m (y_i - h_{\theta}(x_i)) x_i^j \quad (6)$$

$$\theta_j' = \theta_j + \frac{1}{m} \sum_{i=1}^m (y_i - h_{\theta}(x_i)) x_i^j \quad (7)$$

SGD 方法对应的损失函数如式(8)所示。

$$J_{\theta} = \frac{1}{m} \sum_{i=1}^m \frac{1}{2} (y_i - h_{\theta}(x_i))^2 = \frac{1}{m} \sum_{i=1}^m \text{cost}(\theta, (x_i, y_i)) \quad (8)$$

其中, $\text{cost}(\theta, (x_i, y_i))$ 表示每个样本的损失函数。

BGD 方法在更新每一个参数时都使用到所有的样本,其优点是可以找到全局最优解,缺点是数据量庞大,易导致训练速度减慢。SGD 方法通过部分样本进行迭代更新,但是其噪音较多,使得该方法每次迭代的优化方向不确定。因此,本文在上述 2 种方法的基础上,实现一种小批量梯度下降法^[12]。该方法随机选取一定数量的样本进行更新,其设计思想是初始化参数值并迭代更新这些参数使目标函数不断变小,即在训练过程中,使用验证集周期性地测试效果,如果效果较好,将其保存起来,如果多次迭代后分类效果几乎没有提升,则停止迭代。

3 实验结果与分析

3.1 实验样本采集

本文实验样本取自环形光源照射下的铝轮毂背腔图片,通过对原始图片进行预处理、区域定位以及字符分割后,得到需要识别的样本字符,如图 6 所示。



图 6 实验样本图片

3.2 字符识别

本文采集 11 050 幅字符图片作为训练样本,5 300 幅字符图片作为测试样本。网络训练次数为 5 000,最终训练准确率达 100%,验证准确率达 97%,测试准确率达 98.5%,交叉熵为 0.005。图 7 所示为本文方法的训练准确率与训练损失率的对比关系。

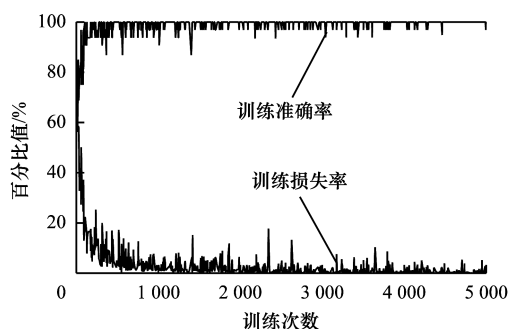


图 7 训练过程中的 2 种参数变化情况

由图 7 可以看出,本文神经网络训练效果较好,在训练次数约为 1 000 时,训练准确率和损失率基本达到较好的水平,且该训练结果基本保持稳定直至训练完成。

本文用置信度来表示字符的分类识别结果,置信度越大,说明待检测字符越接近分类目标。在本次实验中,每类字符的 2 组分类识别置信度统计结果如图 8 所示。由图 8 可以看出,每组字符的分类置信度最大值选项都为待检测字符的真值,虽然某些字符干扰项与真值相差不大,但仍然能够将两者区分开。因此,在进行字符识别时选取置信度最大的分类选项,能够提高字符识别的准确率。

0 <score=0.990 99> 6<score=0.008 85> 8<score=0.000 11> 9<score=0.000 03>	1 <score=0.992 37> 7<score=0.004 14> 4<score=0.001 72> 9<score=0.000 76>
2 <score=0.999 89> 1<score=0.000 07> 7<score=0.000 03> 5<score=0.000 01>	3 <score=0.345 48> 2<score=0.334 20> 5<score=0.291 76> 6<score=0.021 43>
4 <score=0.999 64> 7<score=0.000 25> 1<score=0.000 10> 2<score=0.000 01>	5 <score=0.847 81> 2<score=0.079 58> 0<score=0.044 97> 6<score=0.011 27>
6 <score=0.996 95> 0<score=0.001 93> 8<score=0.000 53> 9<score=0.000 20>	7 <score=0.996 86> 4<score=0.001 57> 3<score=0.000 90> 2<score=0.000 24>
8 <score=0.561 26> 6<score=0.433 64> 3<score=0.001 89> 9<score=0.001 71>	9 <score=0.808 73> 6<score=0.127 99> 0<score=0.060 63> 8<score=0.001 04>

图 8 字符分类识别结果

此外,本文还对具有不同倾斜角度的字符以及存在部分遮挡的字符进行识别,结果如图 9、图 10 所示。

0 <score=0.995 15> 6<score=0.009 91> 5<score=0.009 12> 9<score=0.007 85>	1 <score=0.979 89> 4<score=0.015 37> 7<score=0.002 56> 2<score=0.000 97>
0 <score=0.984 63> 6<score=0.008 12> 9<score=0.003 74> 3<score=0.002 47>	1 <score=0.995 56> 3<score=0.002 92> 7<score=0.000 56> 2<score=0.000 32>
2 <score=0.941 25> 3<score=0.039 21> 6<score=0.009 73> 5<score=0.006 41>	3 <score=0.977 29> 6<score=0.016 12> 9<score=0.005 16> 5<score=0.000 65>
2 <score=0.995 31> 5<score=0.004 13> 7<score=0.000 32> 3<score=0.000 14>	3 <score=0.999 14> 8<score=0.000 30> 6<score=0.000 23> 2<score=0.000 18>
4 <score=0.982 94> 1<score=0.014 11> 7<score=0.001 45> 3<score=0.001 37>	5 <score=0.997 41> 2<score=0.000 70> 3<score=0.000 58> 6<score=0.000 33>
4 <score=0.864 49> 7<score=0.112 65> 2<score=0.010 81> 1<score=0.009 35>	5 <score=0.686 50> 2<score=0.284 38> 9<score=0.022 50> 8<score=0.003 23>
6 <score=0.978 91> 0<score=0.009 47> 3<score=0.008 00> 9<score=0.001 28>	7 <score=0.983 56> 5<score=0.008 84> 2<score=0.002 91> 4<score=0.002 51>
9 <score=0.674 20> 6<score=0.282 53> 0<score=0.034 78> 8<score=0.005 03>	7 <score=0.896 62> 1<score=0.042 02> 2<score=0.038 28> 4<score=0.010 40>
8 <score=0.999 54> 3<score=0.000 30> 6<score=0.000 09> 9<score=0.000 09>	9 <score=0.940 96> 6<score=0.052 41> 0<score=0.002 48> 5<score=0.002 10>
8 <score=0.994 26> 6<score=0.003 43> 9<score=0.002 04> 0<score=0.000 16>	6 <score=0.479 93> 9<score=0.327 87> 0<score=0.190 70> 8<score=0.000 88>

图 9 倾斜字符识别结果

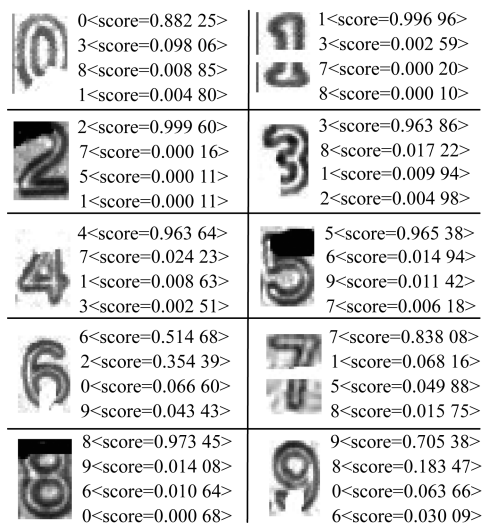


图 10 被遮挡字符识别结果

在图 9 中,包括有一定倾斜角度的图片和倒置的图片。由图 9 可以看出,具有一定倾斜角度字符的识别置信度都大于 0.9,识别效果非常好,而倒置图片的置信度存在小于 0.9 的值,尽管如此,但是大部分字符的识别置信度与其他字符相比相差较大,因此,大部分字符仍能够被正确地识别出。其中,倒置字符识别准确率较低的是“6”和“9”,因为标准印刷体字符的“6”和“9”在倒置之后与另一个数字相同,所以在倒置识别中会出现混淆的情况。但是,本文方法在进行字符分割时会考虑这种情况,即在分割时会矫正偏差较大的字符,因此,在正常的字符识别中,不会出现“6”和“9”混淆的情形。

由图 10 可以看出,运用本文方法时,被遮挡字符的识别置信度虽然没有正常字符高,但是其中每个字符仍然能够被准确地识别出来。

3.3 对比分析

为进一步分析各方法的分类性能,使用 SVM^[13]、BP 神经网络^[14]、LeNet-5 卷积神经网络^[15]以及本文方法进行字符识别对比实验,结果如表 2 所示。其中,训练时间表示每张图片的计算时间。

表 2 不同方法字符识别性能对比

方法	训练时间/s	识别时间/s	识别准确率/%
SVM	0.01	0.023	90.65
BP 神经网络	0.04	0.011	91.20
LeNet-5	0.11	0.080	95.70
本文方法	1.80	0.800	98.50

从表 2 可以看出,本文方法虽然时间消耗较高,但是其识别准确率具有较大优势,即更加适合于铝轮毂背腔字符的识别。

4 结束语

本文使用改进的 inception 结构构建一种卷积神经网络,通过提取卷积特征解决铝轮毂背腔字符分

辨率低、背景噪声大等问题,最终完成该类字符的分类识别。实验结果验证了该网络结构在铝轮毂背腔字符识别领域的可行性与有效性。但是,在本文分类实验中还存在个别字符分类置信度较低的情况,解决该问题将是下一步的研究方向。

参考文献

- [1] 单连平, 窦强. 基于深度学习的海战场图像目标识别[J]. 指挥控制与仿真, 2019(1): 1-5.
- [2] ROUTHAY S, RAY A K, MISHRA C, et al. Efficient hybrid image denoising scheme based on SVM classification[J]. Optik, 2018, 157: 503-511.
- [3] 余永维, 殷国富, 殷鹰, 等. 基于深度学习网络的射线图像缺陷识别方法[J]. 仪器仪表学报, 2014, 35(9): 2012-2018.
- [4] 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.
- [5] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[C]//Proceedings of the 25th International Conference on Neural Information Processing Systems. New York, USA: ACM Press, 2012: 1097-1105.
- [6] SZEGEDY C, LIU Wei, JIA Yangqing, et al. Going deeper with convolutions[C]//Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2015: 1-9.
- [7] HUBEL D H, WIESEL T N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex[J]. The Journal of Physiology, 1962, 160(1): 106-154.
- [8] 肖志鹏, 王小华, 杨冰, 等. 基于卷积神经网络的绘画图像分类研究[J]. 中国计量大学学报, 2017, 28(2): 227-232.
- [9] SZEGEDY C, VANHOUCKE V, IOFFE S, et al. Rethinking the inception architecture for computer vision[C]//Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2016: 2818-2826.
- [10] WILSON D R, MARTINEZ T R. The general inefficiency of batch training for gradient descent learning[J]. Neural Networks, 2003, 16(10): 1429-1451.
- [11] ZINKEVICH M, WEIMER M, SMOLA A J, et al. Parallelized stochastic gradient descent[EB/OL]. [2017-12-25]. <http://martin.zinkevich.org/publications/nips2010.pdf>.
- [12] KONECNY J, LIU Jie, RICHTARIK P, et al. Mini-batch semi-stochastic gradient descent in the proximal setting[J]. IEEE Journal of Selected Topics in Signal Processing, 2016, 10(2): 242-255.
- [13] 刘立峰, 武奇生, 姚博彬. 基于高斯尺度空间和 SVM 的桥梁裂缝检测研究[J]. 工业仪表与自动化装置, 2019(1): 13-16, 114.
- [14] 胡翩翩, 曾碧卿. 基于 HS-BP 神经网络的认知无线电频谱预测技术[J]. 计算机工程, 2017, 43(7): 146-150, 155.
- [15] 邹冲, 蔡敦波, 赵娜, 等. 基于 SVM-LeNet 模型融合的行人检测算法[J]. 计算机工程, 2017, 43(5): 170-173.

编辑 吴云芳