

基于深度卷积神经网络的织物瑕疵识别方法

张卷卷^{1*}, 董 骅², 朱 涛³

¹中国移动通信集团浙江有限公司, 浙江 杭州

²中国信息通信研究院, 北京

³杭州电子科技大学自动化学院, 浙江 杭州

Email: *13588881960@139.com, donghua@caict.ac.cn, tzhu_zt@163.com

收稿日期: 2021年1月17日; 录用日期: 2021年2月12日; 发布日期: 2021年2月19日

摘 要

针对传统基于图像特征提取的瑕疵检测方法过于依赖特征提取效果, 且泛化能力较差以及人工质检存在的效率低、易受主观因素影响等问题, 本文提出了一种基于深度卷积神经网络的图像瑕疵识别方法, 基于ResNet50卷积神经网络, 构建了分类模型。并增大了输入网络的图像尺寸; 采用多种图像变换增强数据; 修改损失函数让模型更加关注困难样本。该模型在测试集上的AUC (Area Under Curve)值可以达到0.905, 同时F1分数达到了0.81。此外本文提出了一种基于滑动窗口检测的瑕疵识别方法, 提高对图像中细节的关注, 大幅提升了原模型的分性能。

关键词

卷积神经网络, 深度学习, 滑动窗口, 瑕疵识别

Fabric Defect Recognition Method Based on Deep Convolution Neural Network

Juanjuan Zhang^{1*}, Hua Dong², Tao Zhu³

¹China Mobile Zhejiang Company, Hangzhou Zhejiang

²China Institute of Information and Communication, Beijing

³School of Automation, Hangzhou Dianzi University, Hangzhou Zhejiang

Email: *13588881960@139.com, donghua@caict.ac.cn, tzhu_zt@163.com

*通讯作者。

Received: Jan. 17th, 2021; accepted: Feb. 12th, 2021; published: Feb. 19th, 2021

Abstract

Aiming at the problems of traditional defect detection methods based on image feature extraction, such as too much dependence on the effect of feature extraction, poor generalization ability, low efficiency of artificial quality inspection and vulnerability to subjective factors, an image defect recognition method based on deep convolution neural network is proposed. Based on ResNet50 convolution neural network, the classification model is constructed, and the size of the input network image is increased, various image transformations are used to enhance the data, and the loss function is modified to make the model pay more attention to the difficult samples. The AUC (Area Under Curve) value of the model on the test set can reach 0.905, and the F1 score can reach 0.81. In addition, a defect recognition method based on sliding window detection is proposed in this paper, which can improve the attention of image details and greatly improve the classification performance of the original model.

Keywords

Convolution Neural Network, Deep Learning, Sliding Window, Fabric Quality Recognition

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

瑕疵识别一直都是图像领域一个重要的分支,而布匹良品检验是纺织行业生产和质量管理的重要环节,但一直以来布匹良品和瑕疵品都是依靠人眼检验区分的。人工检验存在速度慢、劳动强度大,易受主观因素影响,缺乏一致性问题,严重降低了纺织生产流程的自动化程度。因此,研究布匹良品自动检验方法对于实际生产有着重要意义。

目前对于布匹良品检验的方法,主要是针对布匹瑕疵检测。而布匹瑕疵图案的检测方法主要为:基于频域特征提取的方法[1] [2];基于空间域的灰度共生矩阵法[3];基于模型算法的自回归模型[4];基于布匹瑕疵纹理结构的算法[5],混合算法[6] [7] [8] [9]等。上述算法仅适用于背景纹理较为简单的布匹图像瑕疵检测,对于复杂背景的图像,特征提取效果较差,且泛化性能较差。

随着深度学习的兴起,深度卷积神经网络(Deep Convolutional Neural Network)因不需要人工设计特征提取算法、模型泛化能力强等优点在计算机视觉中的应用越来越广泛。文献[10]构建了一个具有多层结构的CNN网络并结合softmax来解决色织物缺陷分类的问题;文献[11]分别采用了AlexNet [12]和GoogleNet [13]对织物的花型进行分类;文献[14]验证了CNN网络能够有效的区分钢材表面的缺陷。这些研究表明深度卷积神经网络在图像分类中具有很好的适用性,克服了传统方法的不足,但上述方法的分类准确率仍有较大的提升空间。

本文基于ResNet50 [15]深度卷积网络,分别在数据处理、网络结构和损失函数上进行改进,并构建了基于滑动窗口的分类模型,应用于布匹瑕疵检测,大幅提升了模型对正常布匹和瑕疵布匹的区分能力。

2. 基于 ResNet50 网络的布匹良品检验算法

2.1. 网络结构

ResNet 的主要优点是可以利用更深层次的网络解决训练误差随网络层数的增加而增大的问题。为了解决该问题，ResNet 对传统的网络结构进行了调整，其关键结构是将基本的网络单元增加了一个恒等的快捷连接。如图 1 所示， $H(x)$ 为理想映射， $F(x)$ 为残差映射， $H(x) = F(x) + x$ 。通过将拟合目标函数 $H(x)$ 转变为拟合残差函数 $F(x)$ ，把输出变为拟合和输入的叠加，使得网络对输出 $H(x)$ 与输入 x 之间的微小波动更加敏感。

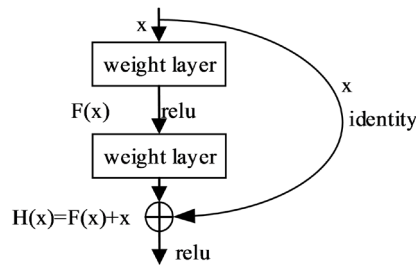


Figure 1. Added residual network unit of shortcut connection

图1. 添加了快捷连接的残差网络单元

综合考虑了网络深度以及计算复杂度的问题，本文选取了ResNet50作为分类网络。ResNet50是残差网络的典型代表之一，与ResNet101和ResNet152相比，有适中的网络深度和计算量，同时能够保持较高的分类准确度。

ResNet50的网络结构图如图2所示，它有两个基本构建块，一个是Identity Block，另一个是Conv Block。Identity Block不改变输入特征图的尺寸和通道数，可以连续串联，其结构如图3所示。Conv Block中存在卷积核大小为 1×1 ，步长为2的卷积层，缩小了特征图的尺寸，起到下采样的作用。此外，快捷连接上的特征图与主支路上的特征图在模块末尾进行融合，经过Conv Block的特征图通道数会翻倍，其结构如图4所示。

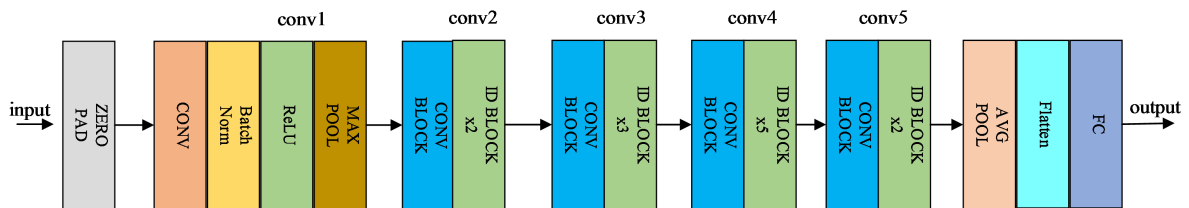


Figure 2. ResNet50 network structure

图2. ResNet50网络结构

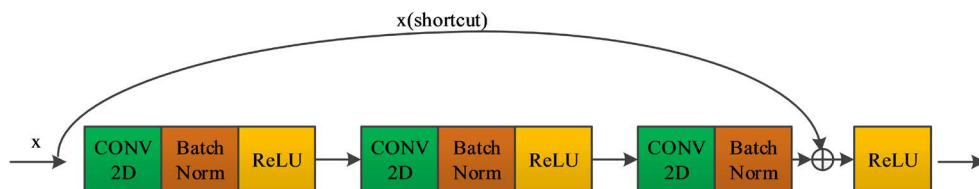


Figure 3. Identity Block structure diagram

图3. Identity Block结构图

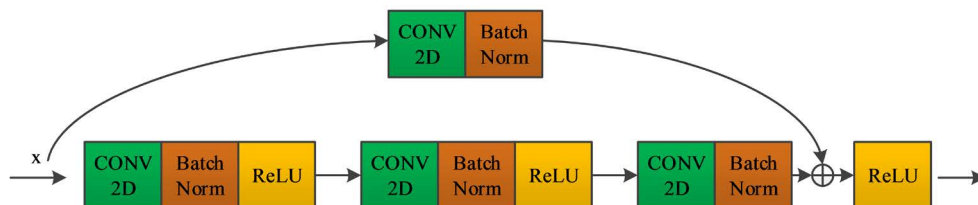


Figure 4. Conv Block structure diagram

图4. Conv Block结构图

ResNet50网络的默认输入为 224×224 像素的图像，图像经过conv1~conv5的处理后得到大小为 7×7 的特征图，再经过全局平均池化层和全连接层，最终通过softmax输出对应每个类别的概率。

2.2. 图片数据预处理

本文使用的布匹数据集来自天池大数据竞赛——雪浪AI制造挑战赛，该数据集的图片可分为正常布匹和含瑕疵布匹两大类，其中瑕疵类别中包含了毛边、擦洞、边扎洞、缺经、缺纬等47种缺陷数据。训练集中正常布匹数量为1316张，含瑕疵的布匹数量为706张，测试集中包含正常与瑕疵布匹图片共662张。为了方便在训练中监督模型的收敛效果，本文从训练集中随机抽取10%的图片作为验证集。图片的分辨率统一为 2560×1920 ，对于包含瑕疵的图片，数据集提供了记录瑕疵位置的标注文件，如图5所示，图5(a)和图5(b)是正常的布匹，图5(c)含毛边，图5(d)含擦洞，均为瑕疵图像。

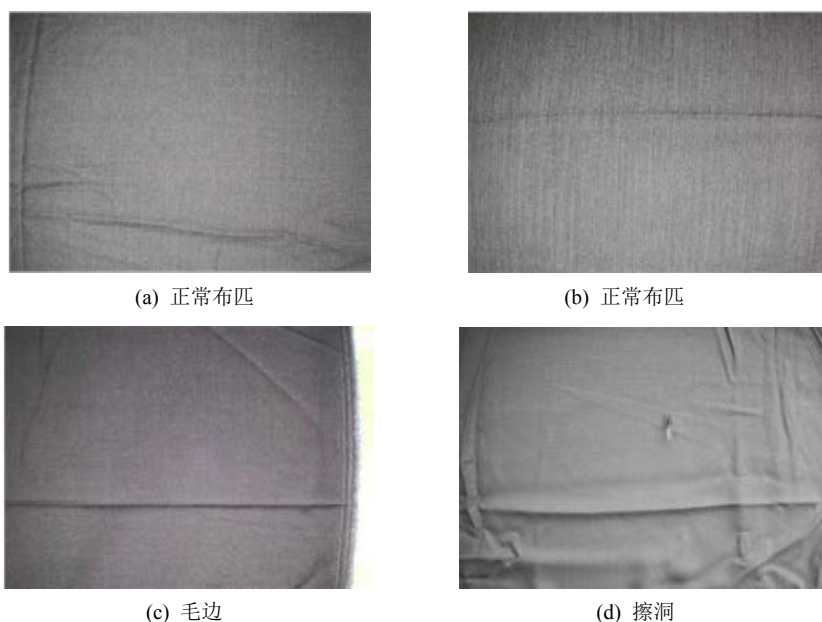


Figure 5. Cloth image data set

图5. 布匹图像数据集

2.2.1. 基于像素区域关系重采样的图片缩放

网络输入图片的像素大小直接影响网络的参数计算量以及特征的学习效果，本文所用的数据集的图片像素均为 2560×1920 。同VGGNet和GoogleNet等网络一样，ResNet50网络的默认图片输入尺寸为 224×224 ，若将图片按原图大小直接输入网络，则会造成网络计算复杂度过大，难以训练。所以需要原数据集的图片进行缩放再放入网络训练。

常见的图片缩放方法为双线性插值法，通过实验发现，采用该方法对原图进行缩放，生成的图片会有波纹状干扰，如图6(a)所示。为了解决此问题，本文采用了基于像素区域关系重采样的图片缩放法，其计算式如下：

$$\begin{aligned} X_{src} &= X_{dst} * \left(\frac{Width_{src}}{Width_{dst}} \right) \\ Y_{src} &= Y_{dst} * \left(\frac{Height_{src}}{Height_{dst}} \right) \end{aligned} \quad (1)$$

其中 X_{src} ， Y_{src} 是原图上的位置信息， X_{dst} ， Y_{dst} 是目标图像的位置信息。生成的图片如图6(b)所示，可见相对于双线性插值法，基于像素区域关系重采样的图片缩放法有效地减少了波纹状干扰。

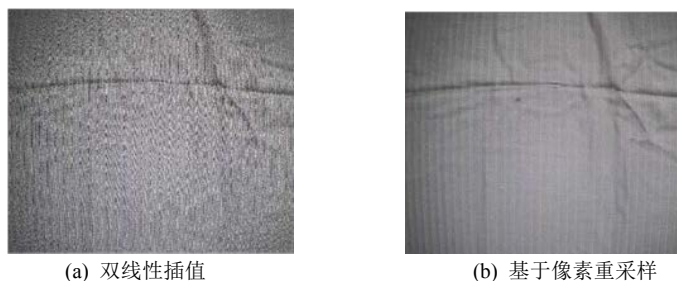


Figure 6. Comparison chart of different zoom methods
图6. 不同缩放方式的对比图

2.2.2. 数据增强

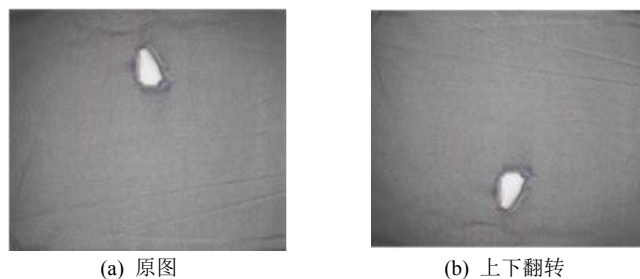
数据增强的作用是扩大有效的训练样本，防止因训练数据太少导致模型学习不充分，训练所得结果无法达到预期[16]。

本文首先采用的数据增强方法是对图像进行随机上下翻转和镜像翻转以及90°旋转生成新的图像。布匹瑕疵不但具有不同的形状，而且瑕疵的角度具有一定随机性。通过原图的随机翻转及旋转可以一定程度上消除样本多样性不足的问题，增强效果如图7所示。其中图7(a)为原图，图7(b)~(d)分别为上下翻转，镜像翻转以及随机旋转90°的效果图。

此外，还对图片进行了水平和垂直方向上的平移以及随机改变图片亮度并添加高斯噪声来进一步增强训练集，图8(a)是改变了图片亮度及添加了高斯噪声的效果图，图8(b)和图8(c)分别是在水平方向平移和垂直方向平移之后的效果图。

2.3. 基于滑动窗口的检测方法

除了将原图直接缩放到固定尺寸进行训练和测试外，为了更加关注图像中的细节，借鉴了滑动窗口检测[17]的思想构建了新的分类模型，具体做法如下：



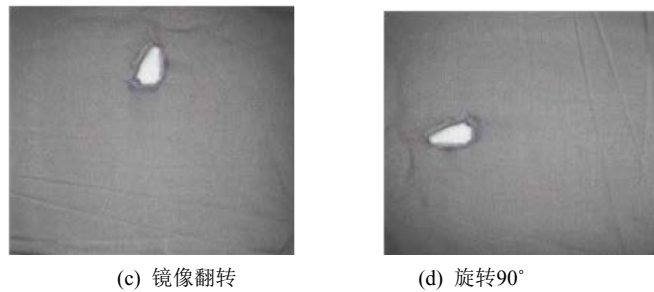


Figure 7. Data enhancement effect chart (1)
图7. 数据增强效果图(1)

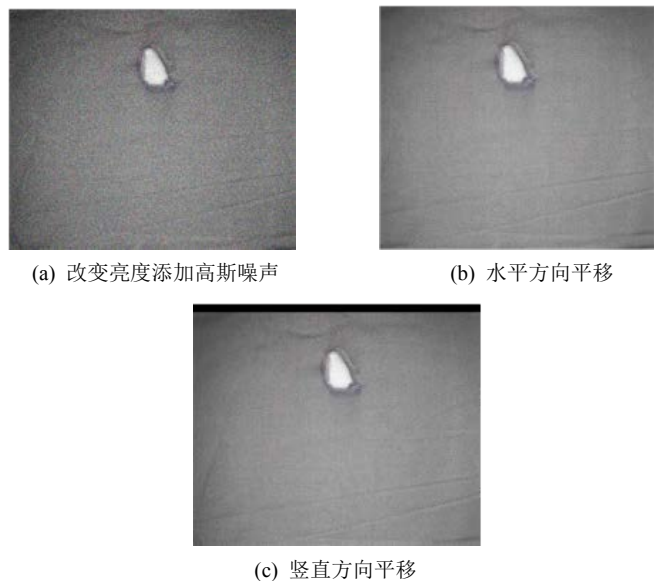


Figure 8. Data enhancement effect chart (2)
图8. 数据增强效果图(2)

① 对于训练集中的正常布匹图片，采用 512×512 的滑动窗口，步长为256进行重采样，获得小图。对于包含瑕疵的图片，若采用同样的方式进行采样，由于大多数缺陷在原图中的面积占比较小，采样得到的大部分图片都不包含瑕疵且存在部分图片仅包含瑕疵边缘。本文计算了重采样得到的小图中瑕疵部分面积与原图瑕疵面积的比值 θ 。在本文的实验中：当 θ 大于0.1时，将该小图归为含瑕疵样本，其余图片均归于正常样本。

$$\text{Class} = \begin{cases} \text{normal} & \theta < 0.1 \\ \text{defect} & \theta \geq 0.1 \end{cases} \quad (2)$$

为了保持合理的正负样本比例，本文从正常样本中随机抽取5%的样本量(与原数据集正负样本比例保持接近)作为最终训练集中的正常布匹样本，得到的数据仍使用ResNet50网络进行训练。

② 对于①中得到的训练集，本文仍采用2.2中的数据增强方式对训练集进行数据增强。

③ 对于测试集，采用同样的滑窗重采样方式得到小图，对小图分别预测该图为含瑕疵布匹类别的概率，再将排名前三的概率求均值作为该图为含瑕疵布匹类别的概率。

基于滑动窗口的检测方法示意图如图9所示，其中网络输入是由滑动窗口重采样得到的子图，通过网络可以预测每张子图所属的类别。

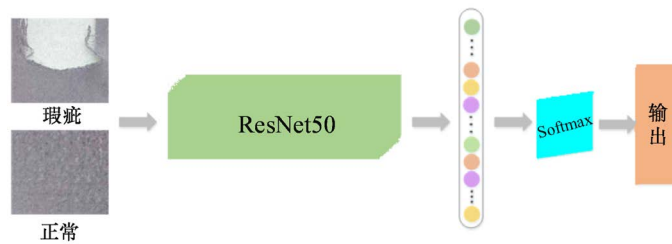


Figure 9. Schematic diagram of sliding window detection
图9. 滑窗检测示意图

2.4. 损失函数的优化

交叉熵[18]是分类任务中常用的损失函数，对于二分类模型它的损失函数如下：

$$L = -y \log y' - (1 - y') = \begin{cases} -\log y' & y = 1 \\ -\log(1 - y') & y = 0 \end{cases} \quad (3)$$

其中 y' 是模型对于实例为正类的预测概率， $y' \in (0,1)$ 。使用交叉熵损失函数，对于正样本而言，输出的预测概率越大，则损失越小。对于负样本而言，输出的概率越小则损失越小。当训练样本中存在较多简单的样本，则训练过程会在这些简单样本的迭代上消耗较多时间，且对于那些较难分类的样本，很难进行优化。

本文使用的布匹数据集的瑕疵类别的样本中，存在着一些如破洞、扎洞等较为明显的瑕疵，但同样也存在着如含油脂、织稀等较难区分的样本，为了改善上述的情况，本文借鉴了 Focal Loss [19]的思想，重新定义了损失函数：

$$L_i = \begin{cases} -(1 - y') \log y' & y = 1 \\ -y'^{\gamma} \log(1 - y') & y = 0 \end{cases} \quad (4)$$

$\gamma > 0$ ，与标准的交叉熵损失函数相比，当预测概率较大时，新的损失函数的损失值将会减小，而当预测概率较小时，损失值将会增大，使得训练过程更加关注那些较难区分的样本。

2.5. 训练与测试

数据集的训练与测试的流程图如图 10 所示。

按照图 10(a)的过程对训练集进行训练，每训练完一个 epoch，对验证集进行测试，记录在测试集上的准确率并保存模型。训练完后，选取在验证集上表现最好的模型按照图 10(b)的步骤对图片进行测试，记录每张图对应于含瑕疵布匹类别的概率，最终计算 AUC 以及 F1 分数等评价指标。

3. 实验结果及分析

3.1. 实验设置

按 2.2 中的处理方法将数据集分别处理成 224×224 、 800×600 两种不同输入尺寸。此部分数据集均包含训练集 1822 张、验证集 200 张和测试集 662 张。再通过 2.3 所述的滑动窗口重采样的方法得到的 512×512 大小的数据集。共有训练集 6484 张，验证集 922 张，采用带动量因子(momentum)的小批量(mini-batch)随机梯度下降法(stochastic gradient descent, SGD) [20]并加载 ImageNet [21]预训练模型对网络进行训练。对于不同输入尺寸大小的图片，网络初始学习率、每个训练批次图片数量以及训练轮次的设置如表 1 所示。

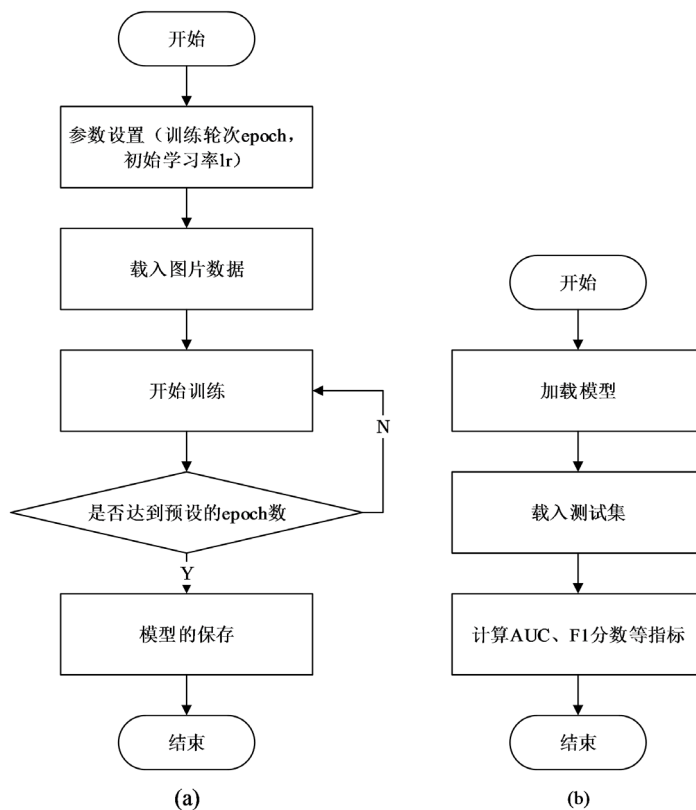


Figure 10. Flow chart of training and testing process
图10. 训练及测试过程的流程图

Table 1. Training parameter settings
表1. 训练参数设置

输入尺寸	初始学习率(lr)	每个批次图片数量(batch size)	训练轮次(epoch)
224 × 224	0.001	32	30
800 × 600	0.001	16	30
512 × 512 (滑窗)	0.001	16	30

对于表 1 中前两组数据训练所得模型，测试集分别缩放至同样的尺寸再进行测试。对于滑动窗口检测模型，测试集同训练集一样，每张 2560 × 1920 像素大小的原图由 512 × 512 大小的窗口用 256 步长采样得到小图。对小图分别进行测试，求他们为瑕疵类别的概率，再将概率大小前三的图片的预测概率求均值作为原图为瑕疵类别的概率。

本文所有实验均是在 keras 深度学习框架下进行的，电脑的操作系统为 ubuntu16.04，CPU 为英特尔 I7 处理器，32 G 运行内存，GPU 为 TITAN X (12 GB)，使用 GPU 加速计算。

3.2. 结果分析

3.2.1. 评价指标

1) AUC (Area Under the Curve)

ROC (Receiver Operating Characteristic)曲线是用于评价分类模型性能的常用标准之一。它的横坐标为伪正类率 FPR (False Positive Rate)，即预测为正但实际为负的样本占有所有负例样本的比例，纵坐标为

真正类率 TPR, (True Positive Rate), 即与预测为正且实际为正的样本占有所有正样本的比例, FPR 和 TPR 的计算公式如下:

$$\begin{aligned} \text{FPR} &= \text{FP}/(\text{TN} + \text{FP}) \\ \text{TPR} &= \text{TP}/(\text{TP} + \text{FN}) \end{aligned} \quad (5)$$

其中 TP 为被模型预测为正类的正类样本数; TN 为被模型预测为负类的负样本数; FP 为被模型预测为负类的正类样本数; FN 为被模型预测为正类的负样本数。

对于二分类模型, 通过分类得到每个实例属于正类的概率 P , 设定阈值 θ , 当 $P > \theta$ 时, 该实例为正类, 否则为负类, 对应可以计算出一组(FPR, TPR), 从小到大调整阈值, 将得到的点连接即得到 ROC 曲线。AUC 为 ROC 曲线下的面积, 单独看 ROC 曲线难以评判两个分类器的好坏, AUC 的值越大, 表明模型的泛化性能更好, 可以通过比较 AUC 数值的大小来衡量分类模型的性能好坏。

2) F1-Score

F1 分数(F1 Score), 同样是衡量分类模型精度的重要指标, 它同时兼顾了分类模型的精准率(Precision)和召回率(Recall), 可以看做是二者的加权平均, 它的计算公式如下:

$$\text{F1} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6)$$

其中精准率表示被判定为正类的样本中实际为正类的比重, 召回率表示被判定为正类且实际为正类的样本占总的正样本的比重, 二者的计算公式如下:

$$\begin{aligned} \text{Precision} &= \text{TP}/(\text{TP} + \text{FP}) \\ \text{Recall} &= \text{TP}/(\text{TP} + \text{FN}) \end{aligned} \quad (7)$$

3.2.2. 不同网络结构及输入图片尺寸的比较

对于缩放至 224×224 , 800×600 两种不同尺寸的数据集, 按照表 1 中的参数设置分别用 GoogleNet、VGG16 和 ResNet50 网络进行训练、测试和比较。图 11 是三个网络在不同尺寸数据集上测试结果的 ROC

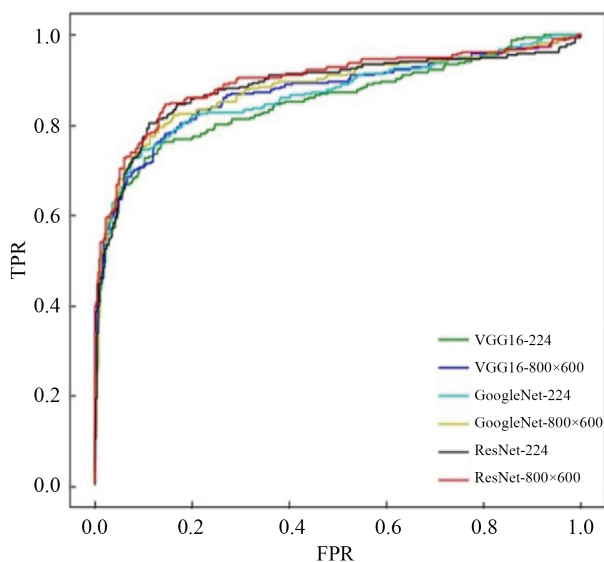


Figure 11. The model corresponds to the ROC curve of different size inputs

图 11. 模型对应于不同尺寸输入的 ROC 曲线

曲线, 表 2 记录了图 10 中每条曲线的 AUC 值, 即曲线下方所围成的面积, 表 3 记录了不同模型对应的 F1 分数。从表 2 和表 3 可以看出, ResNet50 训练所得的模型在测试集上的 AUC 值以及 F1 分数明显优于 GoogleNet 和 VGG16, 通过对比 224×224 与 800×600 两种不同图片输入尺寸所得的模型的测试结果, 可以发现, 增大输入图片的尺寸, 减少原图在缩放过程中的特征丢失, 有利于网络分类性能的提升。

Table 2. AUC test results of different networks for different input sizes

表2. 不同网络对于不同输入尺寸的AUC测试结果

网络	输入尺寸	
	224×224	800×600
VGG16	0.853	0.873
GoogleNet	0.867	0.879
ResNet50	0.885	0.898

Table 3. F1 scores of different networks for different input sizes

表3. 不同网络对于不同输入尺寸的F1分数

网络	输入尺寸	
	224×224	800×600
VGG16	0.763	0.775
GoogleNet	0.776	0.787
ResNet50	0.791	0.805

3.2.3. loss 优化的检验

上述实验均采用交叉熵作为损失函数, 此部分实验将损失函数替换成了2.4中重新定义的损失函数, 图片输入尺寸大小统一设置为 800×600 , 分别对三个网络重新进行训练和测试, 测试结果的AUC值及F1分数如表4和表5所示。

改进的损失函数使得训练中更加关注那些较难学习的样本, 从表3可以看出, 使用改进的损失函数后, 三个网络模型在AUC和F1分数上的表现均有上升, 证明增大困难样本在训练过程中的损失, 能够提高模型分类性能。

Table 4. Comparison of AUC of model test obtained from different loss function training

表4. 不同损失函数训练所得模型测试的AUC比较

Loss	VGG16	GoogleNet	ResNet50
交叉熵	0.873	0.879	0.898
改进的损失函数	0.881	0.885	0.905

Table 5. Comparison of F1 scores of models trained with different loss functions

表5. 不同损失函数训练所得模型的F1分数比较

Loss	VGG16	GoogleNet	ResNet50
交叉熵	0.775	0.787	0.805
改进的损失函数	0.781	0.79	0.81

3.2.4. 滑窗检测方法的测试结果

基于上述实验结果, 本文将滑窗检测方法的采样块大小设置为 512×512 像素, 网络为 ResNet50, 训练过程采用 2.4 中重新定义的损失函数, 实验结果如表 6 所示:

Table 6. Test results of sliding window model

表6. 滑动窗口模型的测试结果

	AUC	F1 Score
滑窗检测	0.935	0.841

滑窗检测方法的测试结果 AUC 高达 0.935 同时 F1 分数可以达到 0.841。两个评价指标均是所有模型当中最高的。可见利用原图缩放后所得图片进行训练, 会因为原图的部分特征丢失, 影响模型对图像特征的学习效果, 而基于滑窗检测的模型, 更能关注图像中的细节, 所以拥有更好的分类性能。

3.2.5. 模型分类速率的比较

网络的参数量、深度以及输入图片尺寸的大小等因素均会影响模型最终的分类速率, 本文对不同网络结构、不同输入尺寸以及基于滑窗检测的模型分别进行了分类速率评估, 表 7 是模型在测试集上处理图片的平均时间。结果显示模型的分类速率随着输入图片的尺寸增大而下降, 此外增加滑窗操作后, 模型的分类速率也会有一定程度上的下降。

Table 7. The average time for different models to process a picture (unit: second)

表7. 不同模型处理一张图片的平均时间(单位: 秒)

VGG16		GoogleNet		ResNet		滑窗检测
224 × 224	800 × 600	224 × 224	800 × 600	224 × 224	800 × 600	512 × 512
0.008	0.039	0.015	0.03	0.017	0.045	1.08

4. 结论

本文提出一种基于深度卷积神经网络的图像瑕疵识别方法, 并应用于布匹良品检验环境中。本文基于 ResNet 网络, 在网络输入、数据增强以及损失函数上进行了优化和改进。此外, 提出的基于滑窗的瑕疵检测方法, 更加关注图像中的细节, 训练所得模型在测试集上的 AUC 达到了 0.935 的同时 F1 分数可以达到 0.841, 相较于原 ResNet 训练所得模型有较大提升。尽管增加滑窗的操作降低了模型的分类速率, 但在分类准确率上有较大幅度的提升, 布匹质检的环境对模型的实时性要求并不高, 相较于速度而言, 模型对缺陷的识别能力更为重要。基于滑窗检测的模型是本文提出的性能最好的模型。

布匹的瑕疵类别多种多样, 后期的工作将在利用更细致的样本库, 针对不同的瑕疵进行精准检测和识别。

基金项目

本项目获国家自然科学基金(No. 61971168)和浙江省自然科学基金(No. LY18F030009)支持。

参考文献

- [1] Kumar, A. (2008) Computer-Vision-Based Fabric Defect Detection: A Survey. *IEEE Transactions on Industrial Electronics*, **55**, 348-363.
- [2] 李精忠, 张津铭. 一种基于傅里叶变换的光滑边界面状要素 Morphing 方法[J]. 武汉大学学报·信息科学版, 2017,

- 42(8): 1104-1109.
- [3] 吴亚新. 基于灰度共生矩阵的纺织品瑕疵识别算法[J]. 电脑知识与技术, 2017, 13(28): 204-205.
- [4] 张靖, 周明全, 张雨禾, 等. 基于马尔科夫随机场的散乱点云全局特征提取[J]. 自动化学报, 2016, 42(7): 1090-1099.
- [5] Hamdi, A.A., Sayed, M.S., Fouad, M.M., *et al.* (2018) Unsupervised Patterned Fabric Defect Detection Using Texture Filtering and K-Means Clustering. *IEEE International Conference on Innovative Trends in Computer Engineering (ITCE)*, Aswan, 2-4 February 20, 130-144.
- [6] 李春雷, 高广帅, 刘洲峰, 等. 应用方向梯度直方图和低秩分解的织物疵点检测算法[J]. 纺织学报, 2017, 38(3): 149-154.
- [7] 薛婷婷, 刘秀平, 张凯兵, 等. 基于学习的 Gabor 滤波器多样式布匹瑕疵检测[J]. 西安工程大学学报, 2017(6): 775-780.
- [8] Jing, J., Zhang, H., Wang, J., *et al.* (2013) Fabric Defect Detection Using Gabor Filters and Defect Classification Based on LBP and Tamura Method. *Journal of the Textile Institute*, **104**, 18-27.
- [9] Ngan, H.Y.T., Pang, G.K.H. and Yung, N.H.C. (2010) Performance Evaluation for Motif-Based Patterned Texture Defect Detection. *IEEE Transactions on Automation Science & Engineering*, **7**, 58-72.
<https://doi.org/10.1109/TASE.2008.2005418>
- [10] 景军锋, 范晓婷, 李鹏飞, 等. 应用深度卷积神经网络的色织物缺陷检测[J]. 纺织学报, 2017, 38(2): 68-74.
- [11] 张宏伟, 张凌婕, 李鹏飞. 基于深度卷积神经网络的织物花型分类[J]. 纺织高校基础科学学报, 2017, 30(2): 261-265.
- [12] Krzhevaky, A., Sutskever, I. and Hinton, G.E. (2012) ImageNet Classification with Deep Convolutional Neural Networks. *International Conference on Neural Information Processing Systems*, Doha, 12-15 November 2012, 1097-1105.
- [13] Szegedy, C., Liu, W., Jia, Y., *et al.* (2015) Going Deeper with Convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, 7-12 June 2015, 1-9.
- [14] Masci, J., Meier, U., Ciresan, D., *et al.* (2012) Steel Defect Classification with Max-Pooling Convolutional Neural Networks. *The 2012 IEEE International Joint Conference on Neural Networks (IJCNN)*, Brisbane, 10-15 June 2012, 1-6.
- [15] He, K., Zhang, X., Ren, S., *et al.* (2001) Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Kauai, 8-14 December 2001, 770-778.
- [16] Dyk, D.A.V. and Meng, X.L. (2001) The Art of Data Augmentation. *Journal of Computational & Graphical Statistics*, **10**, 1-50.
- [17] Datar, M., Gionis, A., Indyk, P., *et al.* (2002) Maintaining Stream Statistics over Sliding Windows. *Siam Journal on Computing*, **31**, 1794-1813.
- [18] Shore, J. and Johnson, R. (1980) Axiomatic Derivation of the Principle of Maximum Entropy and the Principle of Minimum Cross-Entropy. *IEEE Transactions on Information Theory*, **26**, 26-37.
- [19] Lin, T.Y., Goyal, P., Girshick, R., *et al.* (2017) Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **39**, 2999-3007.
- [20] Theodoridis, S. (2015) Stochastic Gradient Descent. *Machine Learning*, 161-231.
- [21] Russakovsky, O., Deng, J., Su, H., *et al.* (2015) ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, **115**, 211-252.