doi:10.6041/j.issn.1000-1298.2020.10.028

基于迁移学习的农作物病虫害检测方法研究与应用

余小东[1,2]　杨孟辑[1,2]　张海清[3]　李　丹[1]　唐毅谦[1]　于　曦[1,2]

(1. 成都大学计算机学院, 成都 610106; 2. 成都大学模式识别与智能信息处理四川省高校重点实验室, 成都 610106;
3. 成都信息工程大学软件工程学院, 成都 610225)

摘要: 为了提高农作物病虫害严重程度(健康、一般、严重)的分类效果,采用迁移学习方式并结合深度学习提出了一种基于残差网络(ResNet 50)的 CDCNNv2 算法。通过对 10 类作物的 3 万多幅病虫害图像进行训练,获得了病虫害严重程度分类模型,其识别准确率可达 91.51%。为了验证 CDCNNv2 模型的鲁棒性,分别与使用迁移学习的 ResNet 50、Xception、VGG16、VGG19、DenseNet 121 模型进行对比试验,结果表明,CDCNNv2 模型比其他模型的平均精度提升了 2.78 ~ 10.93 个百分点,具有更高的分类精度,病虫害严重程度识别的鲁棒性增强。基于该算法所训练的模型,结合 Android 技术开发了一款实时在线农作物病虫害等级识别 APP,通过拍摄农作物叶片病虫害区域图像,能够在 0.1 ~ 0.5 s 之内获取识别结果(物种-病害种类-严重程度)及防治建议。

关键词: 农作物病虫害; 病虫害等级分类; 深度迁移学习; ResNet 50; 移动应用程序

中图分类号: TP391.41; TP181　　**文献标识码**: A　　**文章编号**: 1000-1298(2020)10-0252-07　　**OSID**:

Research and Application of Crop Diseases Detection Method Based on Transfer Learning

YU Xiaodong[1,2]　YANG Mengji[1,2]　ZHANG Haiqing[3]　LI Dan[1]　TANG Yiqian[1]　YU Xi[1,2]

(1. *College of Computer Science*, *Chengdu University*, *Chengdu* 610106, *China*
2. *Key Laboratory of Pattern Recognition and Intelligent Information Processing in Sichuan*, *Chengdu University*, *Chengdu* 610106, *China*
3. *School of Software Engineering*, *Chengdu University of Information Technology*, *Chengdu* 610225, *China*)

Abstract: Classifying the severity of crop diseases is the staple basic element of the plant pathology for making disease prevent and control strategies. In order to achieve better results in the classification of the severity (healthy, general or severe) of crop diseases, a CDCNNv2 algorithm based on residual network (ResNet 50) and deep transfer learning was proposed. By training more than 30,000 crop disease images which were divided into 10 species, a model for the classification of disease severity was obtained, and the recognition accuracy could reach 91.51%. For verifying the robustness of the CDCNNv2 model, comparative experiments were carried out with ResNet 50, Xception, VGG16, VGG19 and DenseNet 121 that used transfer learning. The experimental results showed that the average accuracy of the CDCNNv2 model was increased by 2.78 ~ 10.93 percentage points, which had higher classification accuracy and strengthened the robustness of crop disease severity identification. At the same time, based on the model trained by this algorithm, combined with Android technology, a real-time and online crop diseases severity recognition APP was developed. By photographing the disease areas of the crop leaves, the recognition results (species - disease - severity) and recommendations for prevention and treatment can be obtained between 0.1 s and 0.5 s. In addition, other related supporting functions such as disease encyclopedia made the APP more complete, which can provide effective solutions and ideas for the prevention and treatment of crop diseases.

Key words: crop disease; classification of disease rank; deep transfer learning; ResNet 50; APP

收稿日期: 2020-06-11　修回日期: 2020-08-12

基金项目: 国家自然科学基金项目(61602064)和欧盟 Erasmus + SHYFTE 项目(598649-EPP-1-2018-1-FR-EPPKA2-CBHE-JP)

作者简介: 余小东(1988—),男,讲师,博士,主要从事机器学习研究,E-mail: yuxiaodong@cdu.edu.cn

通信作者: 于曦(1973—),男,教授,博士生导师,主要从事机器学习与产品生命周期评估研究,E-mail: yuxi@cdu.edu.cn

0　引言

农作物病虫害是农业领域面临的主要挑战之一，灾难性的作物病害加剧了粮食供应的短缺[1-2]。根据相关统计和分析，2006—2015 年，全国农作物病虫害的受灾面积为 4.63 亿 hm^2 到 5.075 亿 hm^2。根据全国农业技术推广服务中心的预测，2020 年我国农作物病虫害的受灾面积将累计达到 3 亿 hm^2，这将造成巨大的粮食与经济损失[3]。因此，快速有效地检测农作物病虫害是采取相关防治措施及时止损的第一步。

农作物病虫害的鉴别方式与途径有多种，以前多依靠农业专家或当地植保部门。近年来随着作物病虫害专家知识库的不断完善，建立了各种农业相关网站，通过专家系统或自行检索能够得到一些防治建议。但这些方式具有一定的局限性：其一，农作物的病虫害种类繁多，如马铃薯的相关病害就超过 100 多种[4]，而农业专家是有限的，有时很难找到相关病害领域的专家；其二，植保部门的人员有限，一些地区距离较远、交通不便，无法及时到现场对农作物病虫害进行鉴别。这些因素都容易导致病情延误，无法及时得到有效的防治建议。

随着移动网络、智能手机以及计算机技术的飞速发展，研究者选择计算机视觉技术[5]作为开发移动应用的基础来检测农作物病虫害。在计算机视觉领域，图像识别[6]是典型的应用之一，而卷积神经网络[7]（Convolutional neural network，CNN）可以以像素级提取图像的特征，使 CNN 在图像识别领域有足够好的表现，并深受开发者青睐。

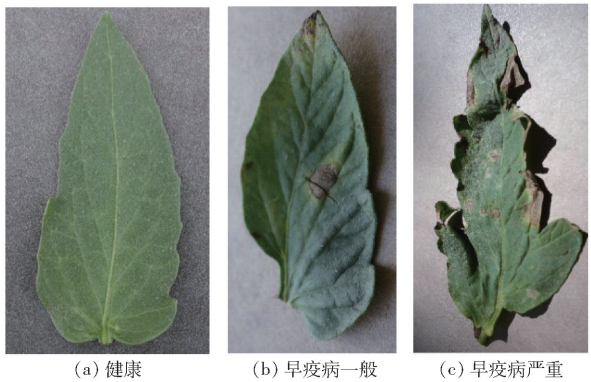
文献[8]利用数字图像处理技术，通过对彩色图像的分析识别香蕉叶斑病的症状，图像识别的平均正确率最高为 94.5%，最低为 13.2%。文献[9]利用支持向量机（Support vector machine，SVM）对甜菜叶片进行检测与分类，对健康、患病叶片的鉴别准确率可达 97%，但是根据疾病类型和患病程度进行分类，准确率在 65% 到 90% 之间。文献[10]使用 Inception v3 网络对木薯 5 种病害进行识别，总体准确率可达 93%。文献[11-12]利用深度学习检测不同作物的多种病虫害，在识别作物-疾病组合方面均能够达到 99% 的准确率。传统的数字图像处理方法应用广泛、处理精度高、灵活性强，但处理后的图像需要在专业人员的指导下进行观察与评估。机器学习方法所训练出的模型虽然能够自动识别作物病害，但是在模型训练之前，需要经历复杂的特征工程，同时需要专业人员的指导。深度学习方法可以将上述问题的工作量最小化，只需要专业人员对原始图像是否存在病变进行标注即可。基于这些简单的标签可以使流程实现从训练到识别的全自动化。一些研究注重实际的应用开发，如文献[13]采用标准区域图技术开发了一款 iOS 应用，以评估病害的严重程度，但是该应用仅适用于甜菜病害严重程度的分类，并且需要用户从预先设置的列表中选择病害类别和计算标准。文献[14]利用两种不同类型的权重对每种病害进行加权，建立了病害模型，但是并未对病害的严重程度进行划分。文献[15]通过计算病害区域的面积来评估病害严重程度，APP 的操作需要用户具有一定的专业知识。

通过对以上研究方法与移动应用的总结，本文提出一种基于残差网络[16]（ResNet 50）的 CDCNNv2 算法，并结合深度迁移学习[17]开发农作物病虫害严重程度等级划分系统（物种-病害种类-严重程度），以实现农作物病虫害的实时、全自动检测。

1　试验数据

1.1　数据集描述

本试验采用由“AI Challenger 2018”提供的农作物病虫害公开数据集，共包含 36 261 幅标注图像，其中训练集有 31 721 幅图像，验证集包含 4 540 幅图像。整个数据集共有 10 个物种（苹果、桃子、草莓、玉米、辣椒、番茄、橘子、葡萄、樱桃、马铃薯）、27 种病害。按照物种-病虫害种类-严重程度划分，共有 61 类，以番茄叶片为例，如图 1 所示。



(a) 健康　(b) 早疫病一般　(c) 早疫病严重

图 1　番茄叶片展示

Fig. 1　Example for tomato leaves

1.2　图像预处理

病虫害的识别与拍摄环境有很大关系，例如光照、角度等。图像品质影响识别精度，为了使模型具有更强的泛化能力，采用数据增强技术[18]，对现有图像进行随机光亮度增减、随机旋转与镜面翻转等操作。

当随机增减光亮度时，遵循公式

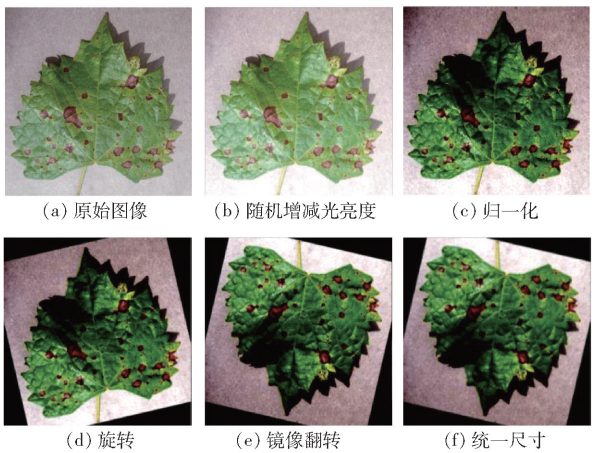
$$\varphi = \gamma\alpha + \beta \quad (\alpha \in (0.9, 1.1), \beta \in (-10, 10))$$

式中 φ——处理后的图像

γ——原始图像

α——对比度 β——光亮度

当图像进行了随机光亮度增减之后,将图像归一化到 -1 ~1 之间,随后设置中心点随机旋转一定角度,再进行镜面翻转,最后将图像统一到 229 像素 × 229 像素。数据增强的结果如图 2 所示。



(a) 原始图像 (b) 随机增减光亮度 (c) 归一化

(d) 旋转 (e) 镜像翻转 (f) 统一尺寸

图 2 数据增强结果

Fig. 2 Results of data augmentation

2 CDCNNv2 模型

CDCNNv2 模型是基于残差网络 ResNet 50 改进而来,按照深度学习的效果而言,识别性能会随着网络层数的加深而提高,但识别性能会在网络加深到某一个层数的时候达到饱和,此时继续加深网络层数,只会使得该网络在训练集上的表现有所减弱,称此现象为网络退化。而 ResNet 50 网络在训练时并非如此,其核心残差模块在网络达到最优时,可以通过恒等映射向后传输信息,理论上网络将会一直处于最优状态,ResNet 50 的具体结构与原理可参照文献[16]。

2.1 深度迁移学习

为了使模型加速收敛并具有更强的泛化能力,选择使用迁移学习技术。选择合适的预训练模型是迁移学习成功的关键,最重要的一点就是预训练模型的数据集与本试验的数据集有一定的关联与相似性。例如,预训练模型的任务是苹果分类,它可能对本试验的橘子分类有帮助。

在本试验中,使用在 ImageNet[19] 上预训练的 ResNet 50 模型的权重文件。ImageNet 是一个庞大的自然图像数据库,有超过 1 500 万幅图像与 2 万多个类别。迁移其权重将会对本试验的模型训练有极大帮助,使用原始的 ResNet 50 网络并没有达到预期效果,所以本试验保留 ResNet 50 的卷积部分,改变其中多层以满足并提高识别的精确度。

2.2 具体方法

在模型改进中,使用 Fine-tuning 策略,其主要思想是调整预训练模型的其中一层或多层以达到适应目标任务的目的。本试验保留 ResNet 50 模型的卷积层权重,是因为卷积层提取的是图像的通用特征(Generic features),这些特征对本试验的任务有极大帮助,其他层的具体改动步骤如下:

(1)在图像输入模型之前,在图像四周以 2 × 2 的形式填充“0”值,以便更好地提取图像边缘信息,控制特征图尺寸。

(2)将预训练模型的卷积层权重迁移到 CDCNNv2 模型的卷积层,让权重随着训练同步更新。

(3)在卷积层之后添加 1 个平均池化层,池化窗口为 2 × 2,计算图像特征矩阵 2 × 2 区域的平均值,有利于保存图像更多的细节信息。

(4)随后使用 Flatten 函数将多维矩阵的输入转换为一维矩阵,以加速计算。

(5)Flatten 层之后,存在 2 个全连接层,之间有 1 个 Batch Normalization 函数,能够加快训练同时提高精度。第 1 个全连接层输出维度为 1 024,采用 ReLU 作为激活函数。为了能够适应本项目需求,最后 1 个全连接层输出维度为 61,使用 Softmax 作为激活函数。

简明的网络结构图以及与 ResNet 50 基础模型的对比,如图 3 所示。

3 试验与结果分析

3.1 试验平台

试验平台使用的服务器操作系统为 CentOS 7,处理器为 Intel Xeon E5 -2650 v4 2.20 GHz,运行内存 192 GB,配备 NVIDIA Tesla P100 GPU。使用 Python 3.6.5 与 Keras(2.2.4)人工神经网络库,图像预处理采用 OpenCV 4.1.0 视觉库。同时使用 CUDA 10.1 与 cuDNN 10.1 深度神经网络加速库。

3.2 试验参数设置

每次训练选取的样本数量为 64,共 495 次,进行 80 次迭代,使用 Adam 优化器,初始学习率为 0.001,动量因子为 0.1,当 2 个 epoch 过后,模型性能没有提升,之后学习率将降低,最低的学习率为 0.000 000 1。

3.3 结果分析

为了验证 CDCNNv2 模型的有效性,在相同参数设置下将本模型与同样采用迁移学习技术的 ResNet 50、Xception[20]、VGG16[21]、VGG19[22]、DenseNet 121[23] 原始模型进行比较,试验结果见表 1。

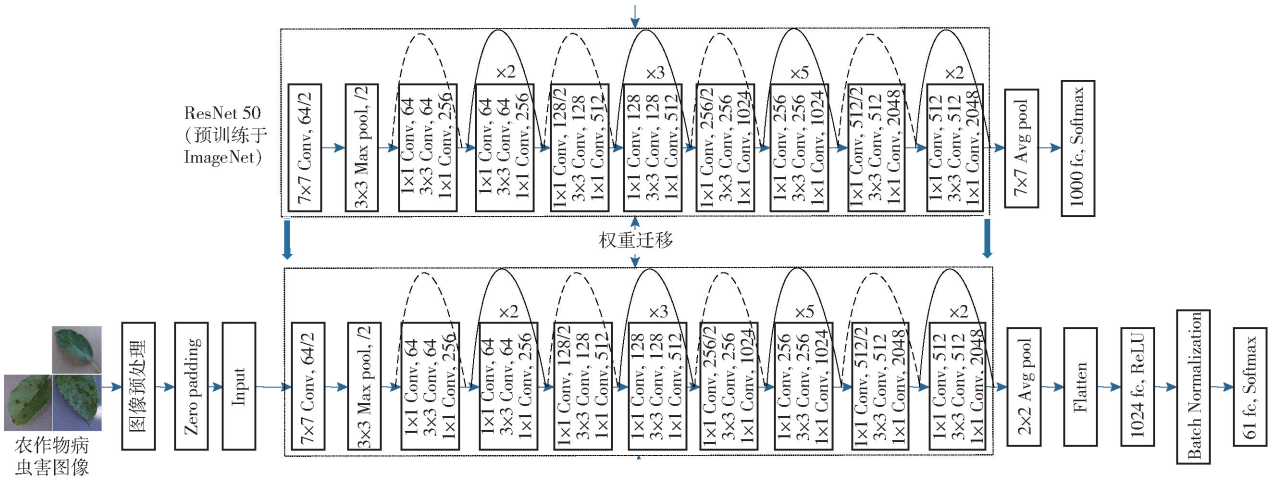


图 3　CDCNNv2 网络框架图

Fig. 3　Concise model structure of CDCNNv2

表 1　对比试验结果

Tab. 1　Results of comparative experiments

模型	试验次数	平均精度/%	平均收敛时间/h
ResNet 50	10	87. 52	9. 78
Xception	10	88. 48	7. 02
VGG16	10	88. 73	12. 20
VGG19	10	85. 54	18. 20
DenseNet 121	10	80. 58	16. 30
CDCNNv2	10	91. 51	8. 48

从表 1 中可以看出，在相同的试验条件下，尽管 CDCNNv2 模型收敛时间不是最优，但是在可接受的范围内；CDCNNv2 模型的平均精度为 91. 51%，比 ResNet 50 模型的平均精度高 3. 99 个百分点，相较于其他模型也有 2. 78 ~ 10. 93 个百分点的提升。

同样，从图 4 中可以证明，CDCNNv2 模型在大约第 25 次迭代就已经收敛，在所有模型中收敛速度处于前列，仅低于 Xception 模型。

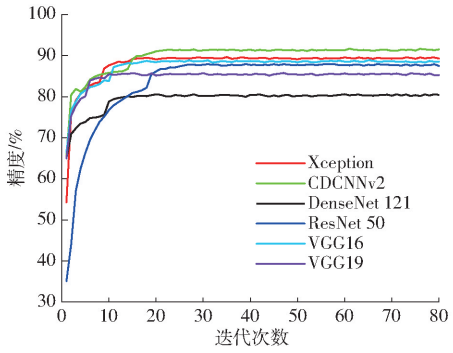


图 4　各模型精度收敛曲线

Fig. 4　Accuracy convergence curves of each model

从图 5 中可以看出，CDCNNv2 模型的损失值在第 25 次迭代达到了稳定点，下降到了 0. 20 左右，为最低。从损失值的收敛情况来看，CDCNNv2 模型的训练效果理想。

综合以上结果可看出，CDCNNv2 模型能够很好地完成农作物病虫害等级分类的任务，具有鲁棒性与高精度，这使得此模型在农作物病虫害领域能够成为一个非常有用的咨询与预警工具。

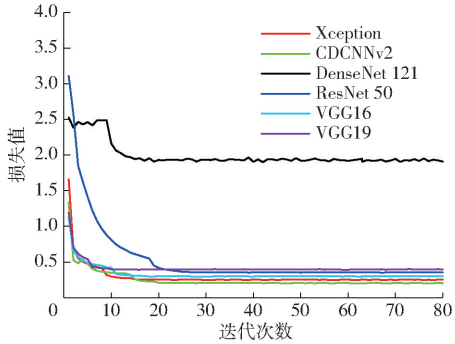


图 5　各模型损失值比较曲线

Fig. 5　Comparison curves of each model loss value

4　模型部署与 APP 设计

4.1　部署工具

本文模型的部署用到了 2 个重要的工具：TensorFlow Serving[24] 和 Docker 容器[25]。TensorFlow Serving 是一种高性能、灵活的机器学习服务系统，用于部署机器学习或者深度学习模型，可以同时服务于多个模型，或者同一模型的不同版本，并且公开 gRPC 和 HTTP 接口 API。Docker 容器的使用能够避免与繁琐的环境配置打交道，可以将多种服务以镜像的形式挂载其中，本项目将 TensorFlow Serving 挂载于 Docker 中，方便了模型部署。如果需要用到 TensorFlow Serving 挂载模型，Docker 是与之配合使用的最佳选择之一。

4.2　部署方法

模型部署的环境包含操作系统 Ubuntu 18、Docker 18、TensorFlow Serving。模型与它们之间的简略包含关系如图 6 所示，Docker 容器运行在 Ubuntu 操作系统中，TensorFlow Serving 以镜像的形

式挂载于 Docker 容器内,模型则挂载于 TensorFlow Serving 中。

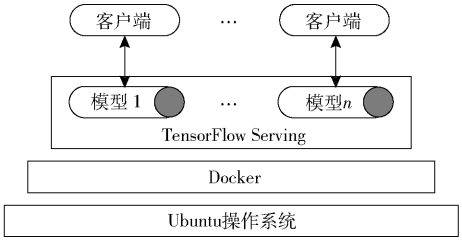


图 6 部署工具之间的关系简图

Fig. 6 Diagram of relationship between deployment tools

模型的具体部署过程较为繁杂,受限于篇幅,本文只指出部署顺序并给出部分重要代码。

首先指定服务的版本号与模型的保存位置,再定义模型的输入与输出格式,核心代码如下

```
model_filePath = './my_model_resnet_v2.h5'
model = load_model(model_filePath)
export_path_base = "serving"
export_version = '2';
export_path = os.path.join(export_path_base,export_version)
builder = tf.compat.v1.saved_model.builder.SavedMdelBuilder(export_path)
tensor_info_input = tf.compat.v1.saved_model.utils.build_tensor_info(model.input)
tensor_info_output = tf.compat.v1.saved_model.utils.build_tensor_info(model.output)
```

随后,将输入与输出封装,以方便地从客户端接收相应格式的参数,简略代码如下

```
prediction_signature =
(
tf.compat.v1.saved_model.signature_def_utils.build_signature_def
(inputs = {'images':tensor_info_inpu},
outputs{'result':tensor_info_output},
method_name = tf.saved_model.PREDICT_METHOD_NAME)
)
builder.add_meta_graph_and_variables
(K.get_session(),
[tf.saved_model.SERVING],
signature_def_map = {'predict_images': prediction_signature,})
builder.save()
```

下一步启动在线服务,在这一步中使用 Docker 容器,具体操作步骤如下

```
Docker run -p port1 -p port2
-mount type = blind, source = THE SOURCE TO MOUNT, target = THE DESTINATION TO MOUNT
-t tensorflow/serving
--model_config_file =/model/models.config
```

'port 1'为主机端口,可以根据需要修改,而'port 2'为固定端口,不能修改。使用配置文件并通过 Docker 挂载可以实现多个模型的部署,只需要在配置文件中表明各个模型的路径与名称。

4.3 Android 端数据发送与接收

Android 端的数据收发并不复杂,由以下几个步骤组成:

(1)将接收到的图像进行 64 位编码。

(2)创建一个新的 Request Body 实例,包含信息格式、编码类型与字符编码。

(3)请求服务器连接,使用. URL 获取服务器 IP 地址,. post 将封装信息发送给服务器。

(4)在定义接收的数据格式之后,通过 Response 实例接收数据信息。

具体流程如图 7 所示。

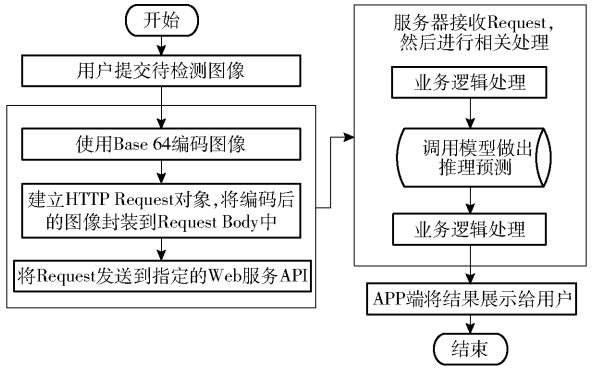


图 7 APP 调用服务器简明流程

Fig. 7 Simplify process of APP calling server

4.4 APP 功能开发与简介

在 4.3 节中,模型的部署与调用已经实现,围绕于此设计了如图 8 所示的配套功能。

各功能模块如下:

(1)用户管理:允许用户权限内的操作,如改变昵称、密码、邮箱、电话等。

(2)检测:允许用户拍照、上传和裁剪图像,在几秒内获取诊断结果,并获得防治建议与药物推荐。

(3)百科全书:允许用户搜索查询农作物相关信息。

(4)社区:允许用户在社区中交流农作物相关信息,同时开发专家系统,以便向专家寻求帮助。

(5)商城:后期将为第三方卖家提供平台售卖农作物相关产品,检测模块中的药物推荐可以跳转到相关商家的销售页面。

检测模块是该 APP 的核心功能,使用苹果-黑

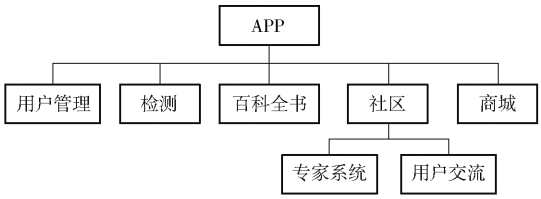


图 8　APP 功能示意图

Fig. 8　Application function diagram

星病-严重的图像进行测试，具体结果如图 9 所示。从图 9 中可以看到，返回的结果包含了病虫害的严重程度，病虫害的描述，检测的可靠性，防治建议与推荐的药物等。



图 9　检测功能界面

Fig. 9　Interface of detection results

4.5　APP 准确性测试

为了验证 APP 的识别准确性，共进行了 200 次验证测试，每验证一次将结果记录在日志文件中，如图 10 所示。可以看出每次测试的结果获取时间都在 0.5 s 以内。随后表 2 中列出了测试的具体结果（苹果黑星病包含了一般与严重两种程度，其他病害类似）。从表 2 中可以看出，本文 APP 具有一定的鲁棒性和实用性。

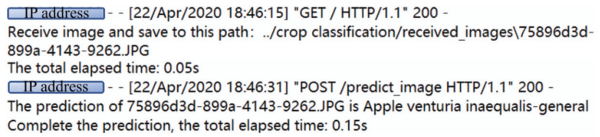


图 10　部分日志文件

Fig. 10　One part of log file

表 2　APP 测试结果

Tab. 2　Test results of APP

种类	数量	正确	错误	精度/%
苹果健康	20	20	0	100
苹果黑星病	20	17	3	85
苹果灰斑病	20	17	3	85
番茄叶霉病	20	17	3	85
番茄早疫病	20	19	1	95
玉米锈病	20	16	4	80
葡萄健康	20	20	0	100
马铃薯早疫病	20	17	3	85
橘子健康	20	20	0	100
草莓叶枯病	20	18	2	90
总计	200	181	19	90.5

5　结论

（1）针对农作物病虫害严重程度分类模型分类精度不够高的问题，结合深度迁移学习提出了一种改进型 ResNet 50 模型——CDCNNv2，在精度上较 ResNet 50 原始模型提升了 3.99 个百分点。

（2）在同样的试验条件下，与 Xception、VGG16、VGG19、DenseNet 121 原始模型进行了对比试验，本文提出的 CDCNNv2 模型的精度提高了 2.78 ~ 10.93 个百分点，同时其损失值最低。因此，该模型具有更好的鲁棒性与可靠性，其综合性能更优。

（3）基于 CDCNNv2 模型开发了一款实时在线农作物病虫害等级识别 APP，该 APP 除了提供识别结果外，还实现了防治建议、药物推荐等一系列配套功能。经过 200 次的实际测试，每次测试结果返回耗时在 0.5 s 以内，测试最终结果表明 APP 具有鲁棒性与实用性，可以为农业病虫害检测提供思路与方案。

参　考　文　献

[1] STRANGE R N, SCOTT P R. Plant disease: a threat to global food security[J]. Annual Review of Phytopathology, 2005, 43: 83 - 116.

[2] TAI A P K, MARTIN M V, HEALD C L. Threat to future global food security from climate change and ozone air pollution[J]. Nature Climate Change, 2014, 4(9): 817 - 821.

[3] 姜玉英，刘万才，黄冲，等. 2020 年全国农作物重大病虫害发生趋势预报[J]. 中国植保导刊, 2020, 40(2): 37 - 39, 53.

[4] HOOKER W J. Compendium of potato diseases [M] // International Potato Center. St. Paul, Minnesota: American Phytopathological Society, 1980.

[5] FORSYTH D A, PONCE J. Computer vision: a modern approach[M] // Lake St, Upper Saddle River, New Jersey: Prentice

Hall, 2002: 597 - 604.
[6] 张琦,张荣梅,陈彬. 基于深度学习的图像识别技术研究综述[J]. 河北省科学院学报, 2019, 36(3): 28 - 36.
ZHANG Qi, ZHANG Rongmei, CHEN Bin. Research review of image recognition technology based on deep learning[J]. Journal of the Hebei Academy of Sciences, 2019, 36(3): 28 - 36. (in Chinese)
[7] SWIETOJANSKI P, GHOSHAL A, RENALS S. Convolutional neural networks for distant speech recognition[J]. IEEE Signal Processing Letters, 2014, 21(9):1120 - 1124.
[8] CAMARGO A, SMITH J S. An image-processing based algorithm to automatically identify plant disease visual symptoms[J]. Biosystems Engineering, 2009, 102(1): 9 - 21.
[9] RUMPF T, MAHLEIN A K, STEINER U, et al. Early detection and classification of plant diseases with support vector machines based on hyperspectral reflectance[J]. Computers and Electronics in Agriculture, 2010, 74(1): 91 - 99.
[10] RAMCHARAN A, BARANOWSKI K, MCCLOSKEY P, et al. Deep learning for image-based cassava disease detection[J]. Frontiers in Plant Science, 2017, 8: 1852.
[11] FERENTINOS K. Deep learning models for plant disease detection and diagnosis[J]. Computers and Electronics in Agriculture, 2018, 145: 311 - 318.
[12] MOHANTY S P, HUGHES D P, SALATHE M. Using deep learning for image-based plant disease detection[J]. Frontiers in Plant Science, 2016, 7: 1419.
[13] PETHYBRIDGE S J, NELSON S C. Estimate, a new iPad application for assessment of plant disease severity using photographic standard area diagrams[J]. Plant Disease, 2018, 102(2): 276 - 281.
[14] GOODRIDGE W, BERNARD M, JORDAN R, et al. Intelligent diagnosis of diseases in plants using a hybrid multi-criteria decision making technique[J]. Computers and Electronics in Agriculture, 2017, 133: 80 - 87.
[15] PETHYBRIDGE S J, NELSON S C. Leaf doctor: a new portable application for quantifying plant disease severity[J]. Plant Disease, 2015, 99(10): 1310 - 1316.
[16] HE K, ZHANG X, REN S, et al. Identity mappings in deep residual networks[C]// European Conference on Computer Vision, Amsterdam, The Netherlands, 2016: 630 - 645.
[17] TAN C, SUN F, KONG T, et al. A survey on deep transfer learning[C]// International Conference on Artificial Neural Networks and Machine Learning (ICANN), Rhodes, Greece, 2018: 270 - 279.
[18] TSUCHIYA H, FUKUI S, IWAHORI Y, et al. A method of data augmentation for classifying road damage considering influence on classification accuracy[J]. Procedia Computer Science, 2019, 159: 1449 - 1458.
[19] KRIZHEVSKY A, SUTSKEVER I, HINTON G. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84 - 90.
[20] CHOLLET F. Xception: deep learning with depthwise separable convolutions[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, USA, 2017: 1800 - 1807.
[21] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[C]// International Conference on Learning Representations, San Diego, California, USA, 2015.
[22] CARVALHO T, REZENDE E R S D, ALVES M T P, et al. Exposing computer generated images by eye's region classification via transfer learning of VGG19 CNN[C]// 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 2017: 866 - 870.
[23] HUANG G, LIU Z, MAATEN L V D, et al. Densely connected convolutional networks[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, USA, 2017: 2261 - 2269.
[24] OLSTON C, FIEDEL N, GOROVOY K, et al. TensorFlow-Serving: flexible, high-performance ML serving[J]. arXiv Preprint arXiv:1712.06139, 2017.
[25] MERKEL D. Docker: Lightweight Linux containers for consistent development and deployment[J]. Linux Journal, 2014,239: 2.