Computers and Electronics in Agriculture 189 (2021) 106379

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

MEAN-SSD: A novel real-time detector for apple leaf diseases using improved light-weight convolutional neural networks

Henan Sun [a,1], Haowei Xu [a,1], Bin Liu [a,b,c,*], Dongjian He [b,c,d], Jinrong He [e], Haixi Zhang [a], Nan Geng [a,b,c]

[a] College of Information Engineering, Northwest A&F University, Yangling, China
[b] Key Laboratory of Agricultural Internet of Things, Ministry of Agriculture and Rural Affairs, Northwest A&F University, Yangling, China
[c] Shaanxi Key Laboratory of Agricultural Information Perception and Intelligent Service, Northwest A&F University, Yangling, China
[d] College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling, China
[e] College of Mathematics and Computer Science, Yan'an University, Yan'an, Shaanxi, China

ARTICLE INFO

Keywords:
Apple leaf diseases
Mobile end real-time detection
Light-weight convolutional neural networks

ABSTRACT

Alternaria blotch, Brown spot, Mosaic, Grey spot, and Rust are 5 common apple leaf diseases that severely impact apple production and quality. At present, although many CNN methods have been proposed for apple leaf diseases, there are still lack of apple leaf disease detection models that can be applied on mobile devices, which limits their application in practical production. This paper proposes a light-weight CNN model that can be deployed on mobile devices to detect apple leaf diseases in real time. First, a dataset of apple leaf diseases composed of simple background images and complex background images, which is called AppleDisease5, is constructed via data augmentation technology and data annotation technology. Then a basic module called MEAN block(Mobile End AppleNet block) is proposed to increase the detection speed and reduce model's size by reconstructing the common 3×3 convolution. Meanwhile, the Apple-Inception module is built by introducing GoogLeNet's Inception module and replacing all 3×3 convolution kernels with MEAN block in Inception. Finally, a new apple leaf disease detection model, MEAN-SSD(Mobile End AppleNet based SSD algorithm), is built by using the MEAN block and Apple-Inception module. The experiment results show that MEAN-SSD can achieve the detection performance of 83.12% mAP and a speed of 12.53 FPS, which illustrates that the novel MEAN-SSD model can efficiently and accurately detect 5 common apple leaf diseases on mobile devices.

1. Introduction

Among agricultural products, apple is one of the most important economic crops in China. In 2019, China's cultivation area exceeded 3 million hectares, making it the country with the greatest apple production and consumption in the world. During the growth and development of apples, various leaf diseases often occur and seriously affect the yield and quality of apples due to the influences of the environment and bacteria (Khan et al., 2020).

Traditionally, visual inspection by experienced experts is required to diagnose plant leaf diseases. This method is time-consuming and labor-intensive, which usually results in a high risk of error and significant cost (Sharif et al., 2018; Adeel et al., 2020; Khan et al., 2020). As smart agriculture develops, methods based on machine learning are gradually becoming more commonly utilized to detect plant disease, making the diagnosis process more convenient (Chen et al., 2020). Researchers have proposed several machine-learning-based models for plant disease detection, such as SVM(support vector machine) and random forest (Kaur et al., 2019). However, features extracted in these methods heavily depend on professional experience and knowledge, limiting the generalization ability of machine-learning-based models. In addition, machine-learning-based models cannot achieve satisfactory accuracy, and they are susceptible to artificially selected features.

In recent years, convolutional neural networks (CNNs) have achieved important breakthroughs in the plant disease detection area (Khan et al., 2018; Zhou et al., 2020). Rather than manually selecting features to feed traditional machine-learning-based models, CNNs provide end-to-end pipelines to automatically extract advanced robust features and

* Corresponding author at: College of Information Engineering, Northwest A&F University, Yangling, China.
E-mail address: liubin0929@nwsuaf.edu.cn (B. Liu).
[1] Henan Sun and Haowei Xu are the co-first authors.

https://doi.org/10.1016/j.compag.2021.106379
Received 4 November 2020; Received in revised form 4 August 2021; Accepted 5 August 2021

thus significantly improve the usability of plant leaf detection (Kim et al., 2020; Zhang et al., 2018; Hasan et al., 2019; Fu et al., 2021). However, most of the CNN models for apple leaf disease detection are implemented on high performance servers with GPU acceleration due to the huge amount of weight parameters and high computational complexity, which limits their application in actual industrial production (Agarwal et al., 2020). During recent years, more attention has been focused on online crop leaf disease detection (Deng et al., 2020). However, the detection speed of online deployment with web server technology is greatly limited by the Internet speed, which is not suitable for real-time detection. Thus, deploying CNN models offline to mobile end instead of cloud servers is the future development trend, which requires a light-weight CNN model. However, the CNN models mentioned above are either too large to be deployed on mobile devices or too slow to detect diseases in real time because of the limitation of Internet speed, while the light-weight CNN detection model based on mobile devices proposed in this paper can achieve high accuracy and real-time detection for apple leaf diseases.

In conclusion, this paper proposes a novel real-time detection model for apple leaf diseases using improved light-weight convolutional neural networks. The contributions are summarized as follows:

- An apple leaf disease dataset called AppleDisease5 is established to ensure the generalization performance of the MEAN-SSD model. First, in order to improve the practical ability of the model, images of infected apple leaves with simple and complex environments are collected in the orchard. Furthermore, the AppleDisease5 dataset is enlarged through digital image processing technology for solving problems of CNN overfitting.
- A real-time and light-weight CNN model for the detection of apple leaf diseases, MEAN-SSD, is proposed. In order to reduce the size and computational complexity while maintaining the accuracy, the core module MEAN block is first designed based on the ideas of split convolution and group convolution. Then the Apple-Inception module is proposed by adding a branch of 7×7 convolution and replacing all 3×3 convolutions with MEAN blocks in the basic Inception module, in order to extract the multi-scale features of apple leaf diseases. Finally, the pre-network MEAN of the proposed model is built based on the MEAN block and Apple-Inception module.
- A light-weight CNN detection model is first deployed on mobile devices, which realizes the fast and accurate detection for apple leaf diseases on the mobile end. The proposed model is an end-to-end deep learning model and can automatically extract discriminative features of apple leaf diseases. Based on principles of simple operation, strong real-time performance, low cost and vigorous promotion, the novel mobile-based apple leaf disease detection model is designed and developed.

The rest of the paper is organized as follows: Section 2 introduces and summarizes the related work. Section 3 describes the detailed information of the apple leaf disease dataset, AppleDisease5. Section 4 describes the proposed novel detection model for apple leaf disease in detail. Section 5 provides the evaluation and analyses of the experiment performance. Finally, Section 6 summarizes this paper.

2. Related work

As artificial intelligence rapidly develops, breakthroughs in CNNs have occurred in computer vision. To solve problems of crop disease prevention in large-scale agriculture production, such as Mosaic disease and Brown spot disease, the CNN algorithm has been used to precisely detect plant leaf diseases precisely.

In Fu et al. (2021), Fu provides a method for the detection of kiwifruit. Based on YOLOv3-tiny, the deep YOLOv3-tiny model is developed by adding two convolutional kernels of 3×3 and 1×1, with the highest average precision of 0.9005 and smallest data weight of 27 MB. In Xie et al. (2020), with the Inception-v1 module, SE-blocks and Inception-ResNet-v2 module introduced, a deep-CNN-based Faster DR-IACNN model, which achieves the better feature extraction ability, is proposed. The experiment results show that the accuracy of the Faster DR-IACNN detection model reaches 81.1%, with the detection speed up to 15.01 FPS. In Jadhav et al. (2020), Jadhav provides an identification method based on AlexNet and GoogLeNet for soybean disease identification, which achieves the accuracies of 98.75% and 96.25% respectively. In Adeel et al. (2019), Khan proposes an automatic system for segmentation and recognition of grape leaf diseases, which can reach an average segmentation accuracy rate of 90% and classification accuracy above 92%. However, these models are too large to be deployed on the mobile end because of the heavy use of large convolution and Inception module, while the proposed MEAN-SSD model chooses to split and group the convolution process to reduce the size while maintaining the accuracy so that it can fit well onto mobile devices.

In recent years, mobile-based CNN models are gradually applied into the area of plant disease detection. In Picon et al. (2019), Artzai proposes three different CNN architectures that contain contextual non-image meta-data and encapsulates the deployment model into a Docker container, which has a secure and authenticated REST service to allow it to connect with a mobile APP. In Singh et al. (2018), Kaushik deploys the CNN model for plant disease identification on the cloud server and develops a mobile APP to send the detection outcome to users. Although the models mentioned above achieve the purpose of the mobile end detection, the detection speed is greatly limited by the Internet speed, while the model proposed in this paper achieves a fast detection speed without the effect of the Internet speed, due to its light-weight structure and off-line deployment.

3. Construction of the apple leaf disease dataset

The process of apple leaf disease detection is shown in Fig. 1. First, the original images are obtained from apple orchards and the laboratory. Second, the diseased images are increased through data augmentation, and further processed through expert annotation. Finally, the dataset is partitioned into a training dataset for training the MEAN-SSD model, a validating dataset for adjusting the parameters and model evaluation, and a testing dataset for evaluating the generalization ability of the model.

3.1. Apple leaf disease image collection

Since no suitable dataset for apple leaf disease detection is available, a new apple leaf disease dataset, which is called AppleDisease5, is constructed, with 5 common apple leaf diseases selected as research objects. In order to enhance the diversity of the dataset, apple leaf disease images are collected from the BaiShui experimental apple station, LuoChuan experimental apple station and QingCheng experimental apple station in Northwest A&F University. The images are acquired under different weather conditions, such as sunny days, cloudy days and rainy days, given that different collection conditions can further enhance the diversity of the dataset. A BM-500GE/BB-500GE digital color camera has been used to capture apple leaf images with a resolution of $2{,}456 \times 2{,}058$ pixels.

Fig. 2 shows 5 representative images of the apple leaf dataset. In Fig. 2, the differences among the 5 types of apple diseases can be clearly observed. In the early stage of Alternaria blotch, brown spots appear on the surface of leaves, and then gradually expand to reddish-brown. The edge of the spot is purple-brown, and there is a dark spot or concentric ring pattern in the center of the lesion. The Brown spot is round, yellowing all around, dark brown in the middle, in which small black spots arranged in the shape of concentric whorls, and the area surrounding the lesions is green. The Mosaic spots are various types of bright yellow spots, and the symptoms vary greatly. Gray spot disease generates a lesion similar to round brown and clear edges at the early stage of leaf

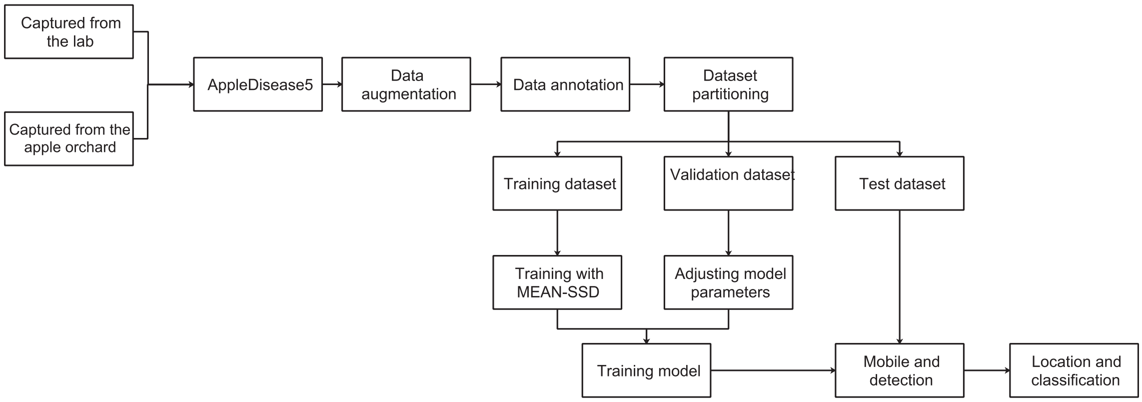
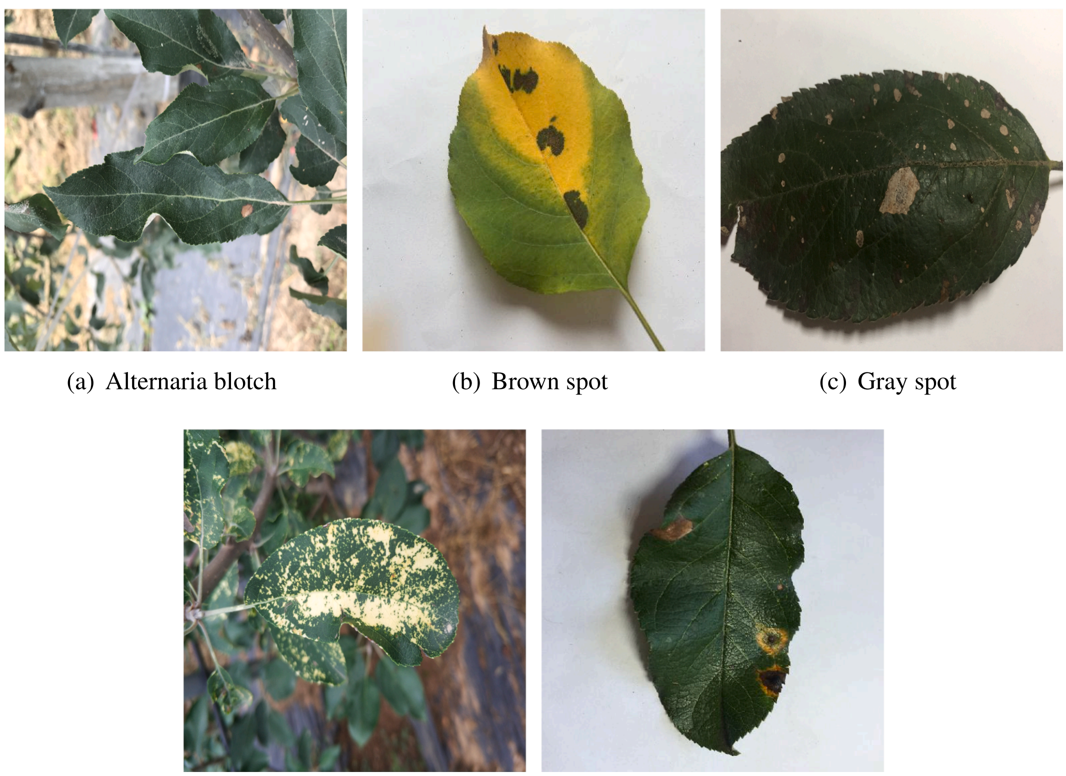


Fig. 1. Flow chart of the mobile end real-time detection for apple leaf diseases.



(a) Alternaria blotch (b) Brown spot (c) Gray spot (d) Mosaic (e) Rust

Fig. 2. Images of five common types of apple leaf disease.

disease, and then becomes gray. In the early stage of Rust disease, small oily orange-red spots appear on the front of the leaves, and then slowly expand to form round orange-yellow spots with red on the edges of the spots. The differences among these disease spots exert a large influence on the detection of various apple leaf diseases.

3.2. Image augmentation

On the basis of the common characteristics analysis of apple leaf diseases, to prevent the overfitting problem of network training caused by insufficient data samples, data augmentation is implemented to simulate the interference in practical scenes, which plays a critical role in the stage of model's training. Through 12 operations including image rotation, horizontal and vertical mirroring, sharpening, brightening, and contrast adjustment, a dataset with 5 categories including a total of 2,230 apple leaf disease images is increased to 26,767 images. Specifically, image intensity interference simulates the impact of weather factors during shooting, which includes brightness, contrast, and sharpness interference. Additionally, the relative positions between the camera and infected apple leaves are simulated by rotation, horizontal symmetry and vertical symmetry. To obtain a standard format for the subsequent network model training, the dataset has been transformed into VOC2007 format.

In this experiment, the number of original images is expanded to 26,767 through digital image processing technology. The whole dataset is randomly divided into training, validating and testing sets with the ratio of 3:1:1. The detailed information of the dataset is described in Table 1. Fig. 3 presents an apple leaf disease image generated through data augmentation technology.

Table 1
Apple leaf disease dataset.

Disease	Total (original)	Total (augmented)	Training	Validation	Test
Alternaria blotch	394	4720	2832	944	944
Brown spot	461	5532	3324	1104	1104
Gray spot	448	5380	3230	1075	1075
Mosaic	430	5165	3155	1005	1005
Rust	497	5970	3560	1205	1205
Total	2230	26767	16101	5333	5333

3.3. Image annotation

Image annotation is essential when building the AppleDisease5 dataset. An annotation tool has been developed to label the locations and types of spots on diseased leaves under the guidance of experienced experts. Following the annotation step, each annotated image generates XML files which contain information such as the coordinate values of each spot's bounding boxes and the classes of the diseases. Taking a Rust-infected leaf as an example, as Fig. 4(a) shows, the infected spots surrounded by boxes are labelled in the annotated image. Fig. 4(b) is a piece of code of the generated XML file, with the disease classification of Rust labelled, and the position of the disease spots are located by the lower right and upper left coordinates of the boxes. Due to the limitations of manual annotation, random errors inevitably appear in the annotation process. For reducing the impact of the following experiments' errors, the image annotations have been repeatedly verified.

4. Mobile end detection model for apple leaf diseases

In this section, a novel model, MEAN-SSD, is described in detail. The proposed MEAN-SSD model consists of two parts: (1) a pre-network to extract the apple leaf disease features. (2) a Single Shot MultiBox Detector(SSD) to locate the disease spots. The following section starts with the core modules of MEAN-SSD, which are the MEAN block and Apple-Inception module. Then the structure of the pre-network and the whole model are described. The overall framework is shown as Fig. 5.

4.1. The MEAN block

In order to reduce the size and computational complexity of the model while maintaining the accuracy, the core MEAN block is proposed as the model's basic module to replace all 3×3 convolution kernels, which is shown in Fig. 6. As described in Hua et al. (2019), a large amount of information redundancy exists among convolution kernels in the same layer of the CNN. Thus, the CNN can be compressed by reducing the information redundancy. Inspired by MobileNet(Howard et al., 2017) and EffNet(Freeman et al., 2018), the computation process of common convolution is split to achieve the purpose of reducing computation. At the same time, group convolution is added into the MEAN block to further reduce the number of parameters and the information redundancy.

Based on the ideas above, the MEAN block starts with a 3×3 group convolution. Different from the common 3×3 group convolution, the 3×3 convolution is divided into a 1×3 convolution and a 3×1 convolution. Instead of using a stride of 2 to achieve the purpose of

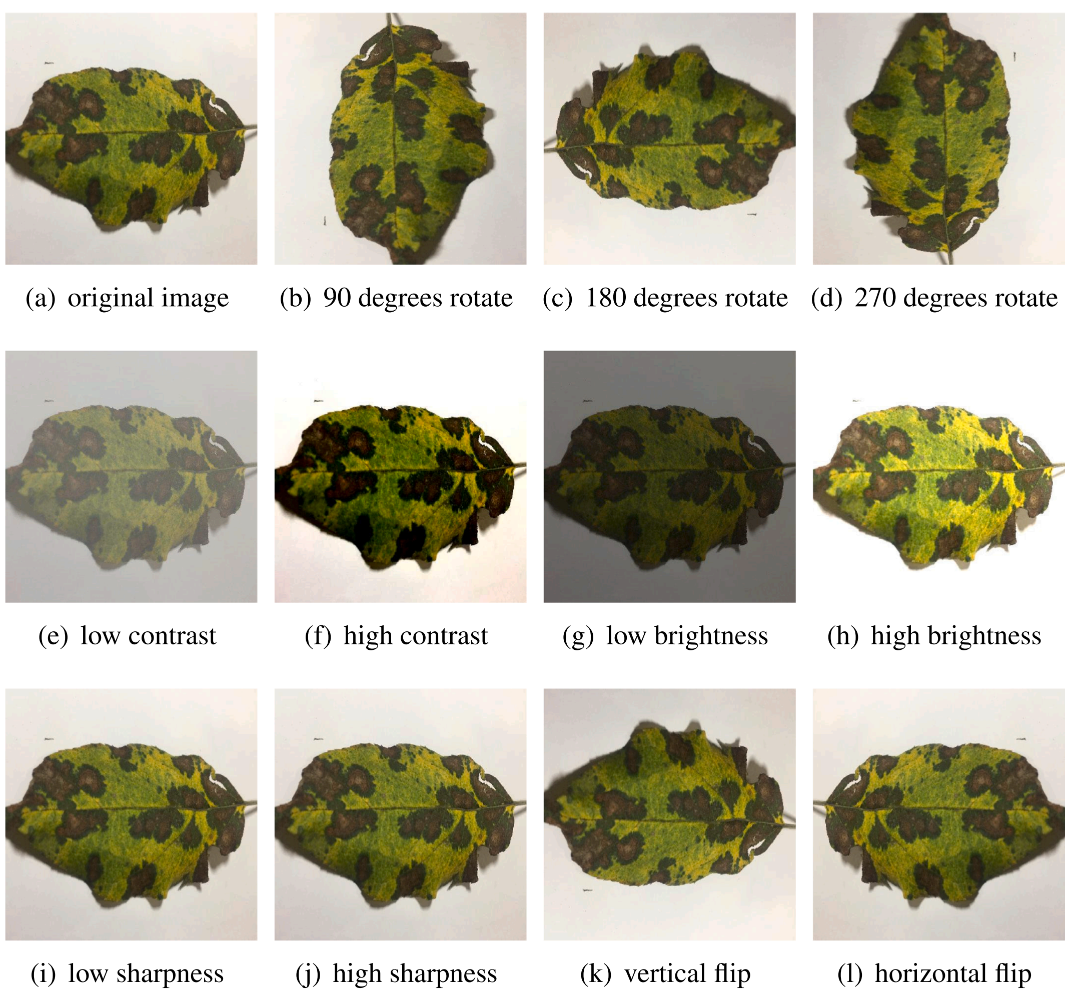
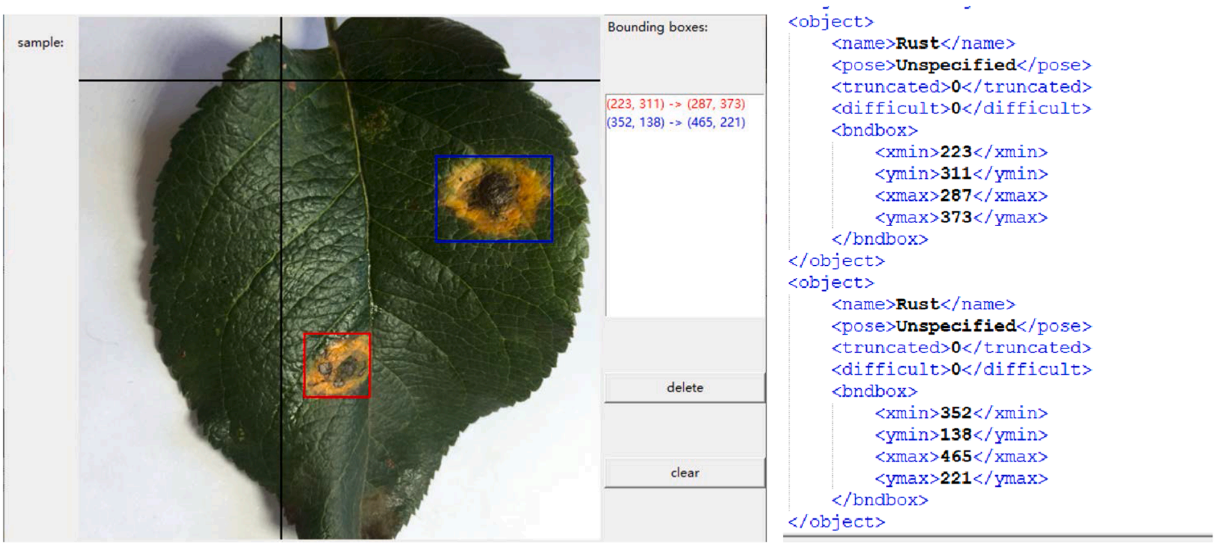


Fig. 3. Data augmentation of an apple leaf disease image.



(a) (b)

Fig. 4. The example of Rust-infected leaf annotation (a) The annotated image; (b) The XML document.

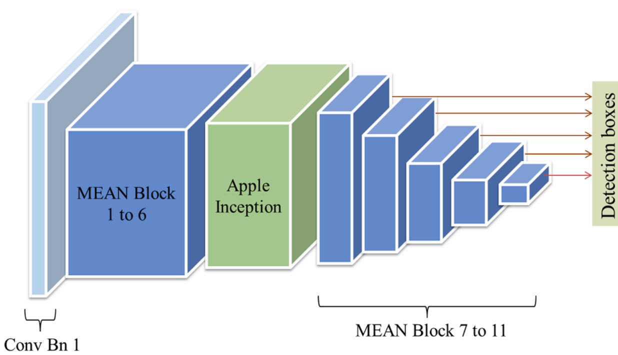


Fig. 5. The MEAN-SSD model.

downsampling, a pooling layer is added between the 1 × 3 convolution and the 3 × 1 convolution to reduce the computation, inspired by EffNet (Freeman et al., 2018) who claimed a better efficiency is achieved by adding a pooling layer into two convolution layers instead of using the stride of 2. In the experiment, it's found that this method not only maintains the model's accuracy, but also reduces the model's size and computation consumption, thus conserving the inference time. Similar to MobileNet(Howard et al., 2017), a 1 × 1 pointwise convolution is added after the 3 × 3 grouping convolution to fuse the information among different channels in the same convolution kernel and control the channel number of the output feature maps. A batch normalization layer is added to both the 3 × 3 group convolution and the 1 × 1 pointwise convolution to solve the gradient dispersion problem. To further refine the model's prediction accuracy, the residual connection module of ResNet(He et al., 2016) is introduced. Moreover, a pointwise convolution is added on the shortcut to control the number of channels, and also an average pooling layer is added after the pointwise convolution if the feature map needs to be downsampled, as shown in Fig. 6.

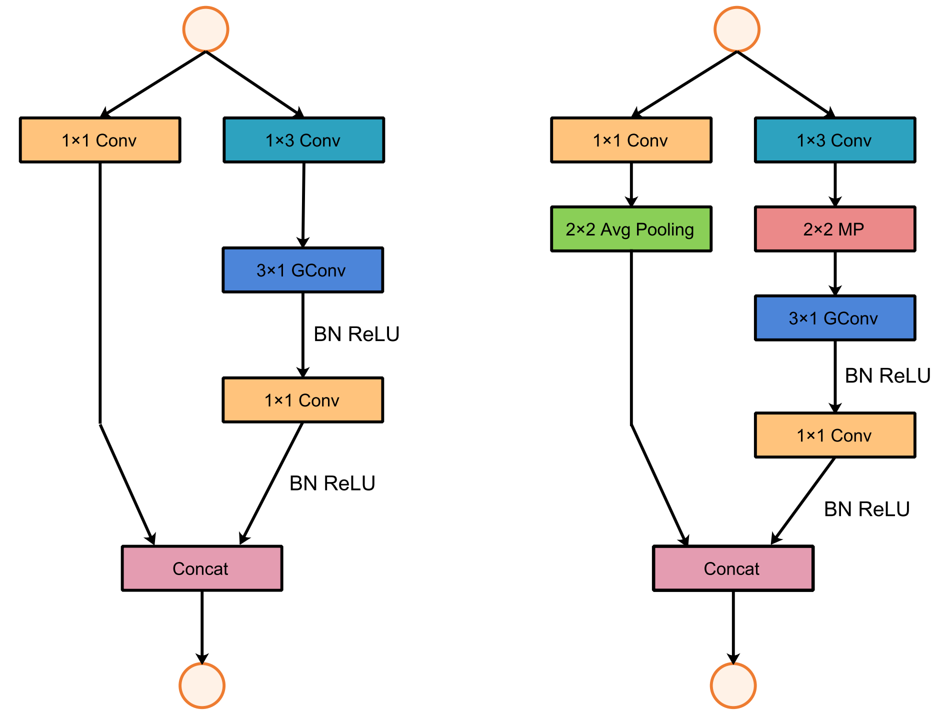
4.2. The Apple-Inception module

Apple leaf disease spots tend to have various sizes. Alternaria blotch spots, Grey spots and Rust spots tend to have a size of approximately 45 × 47 pixels, while Brown spots tend to have a larger size of 91 × 57 pixels. Mosaic spots, however, usually have a tiny size of approximately 5 × 6 pixels. As a result, GoogLeNet(Szegedy et al., 2015)'s Inception module is used in the MEAN to solve the multi-scale feature extraction problems of apple leaf diseases. Different from the basic Inception module, an additional branch consisting of a 1 × 1 convolution and a 7 × 7 convolution is added to better identify tiny spots like Mosaic, while the original 3 × 3 convolution and 5 × 5 convolution are used to identify spots such as Alternaria blotch. The idea of the VGG model (Simonyan et al., 2014) is accepted, and all the convolutions of 5 × 5 and 7 × 7 in Inception are replaced with the small but effective MEAN block module, so as to reduce the amount of model calculation. Through experiments, it's found that although the Inception module consumes many computing resources, its multi-scale feature extraction capability is well fit to the needs of apple leaf disease detection. Fig. 7 shows the Apple-Inception module based on the basic Inception module.

4.3. Design of the pre-network MEAN

Early CNN models, such as AlexNet(Krizhevsky et al., 2012) and GoogLeNet(Szegedy et al., 2015), face difficulty in realizing real-time detection on the mobile end with limited computing resources, due to their large amount of parameters, while some light-weight networks, such as ShuffleNet(Zhang et al., 2018), MobileNet(Howard et al., 2017), EffNet(Freeman et al., 2018), etc., have been successfully deployed to the mobile end. Thus, the model's structure refers to the construction of the light-weight networks above. As MobileNet(Howard et al., 2017) has been proven most suitable for the detection of the apple leaf diseases through the experiments, MobileNet(Howard et al., 2017) is chosen as the basic network architecture of the MEAN model.

Table 2 lists the detailed parameters of the MEAN. The previous several 3 × 3 convolutional layers are usually used to learn low-level features, such as colors and edges, while the deeper convolutional layers are used to extract complex and complete features. Therefore, the first several depthwise separable convolution layers of MobileNet are retained, except that each depthwise separable convolution layer is replaced with a MEAN block to reduce the number of parameters and computations of the model. On the eighth layer, the improved Apple-Inception module is placed to extract the multi-scale features to improve the model's accuracy. Five more MEAN blocks are placed after the Apple-Inception module to further extract the deeper-level features. Finally, the MEAN model is connected with a fully connected layer and a softmax function layer to classify the categories of apple leaf diseases.



(a) The MEAN block without downsampling (b) The MEAN block with downsampling

Fig. 6. The MEAN block (”MP” means max pooling; ”GConv” means group convolution; ”BN” means batch normalization).

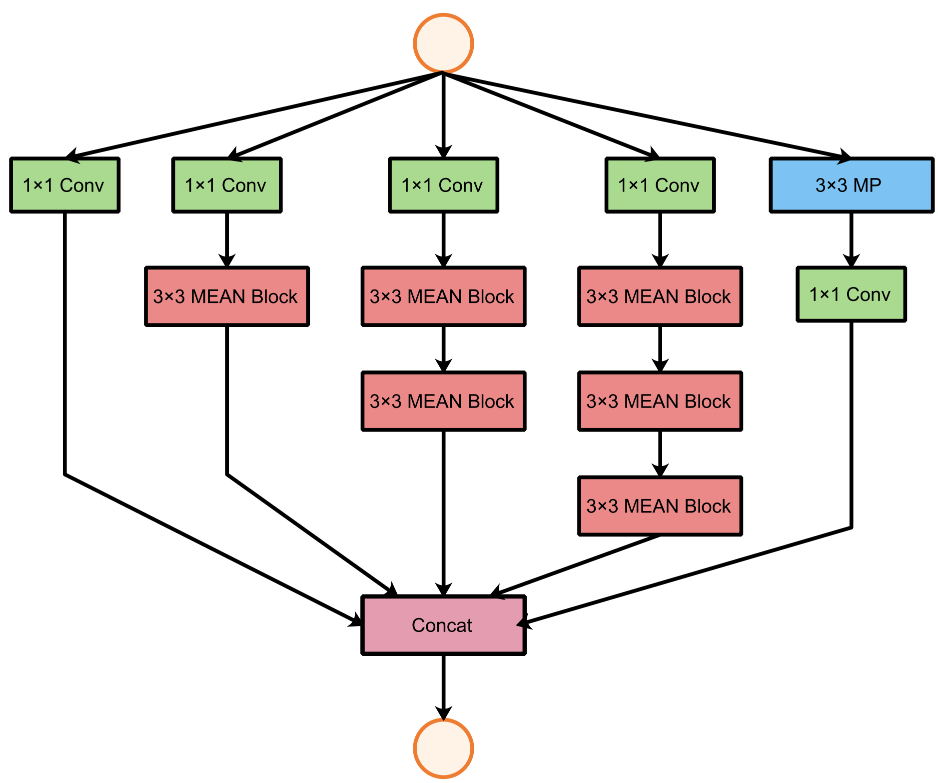


Fig. 7. The Apple-Inception Module.

4.4. *Design of the MEAN-SSD model*

SSD, proposed in Liu et al. (2016), is one of the classic one-stage detection models that can both predict and locate the object directly, without using region proposals. Inspired by Liu et al. (2016), the new mobile-based detection model is proposed by replacing the pre-network

Table 2
Related parameters of the MEAN model ("Conv_BN" means convolution layer with a batch normalization layer; "FC_With_Softmax" means fully connected layer with the activation function of softmax).

Name	Kernel Size/Stride	Output Size
Conv_BN_1	$3 \times 3/2$	$256 \times 256 \times 8$
MEAN_block_1	$3 \times 3/1$	$256 \times 256 \times 16$
MEAN_block_2	$3 \times 3/2$	$128 \times 128 \times 32$
MEAN_block_3	$3 \times 3/1$	$128 \times 128 \times 64$
MEAN_block_4	$3 \times 3/2$	$64 \times 64 \times 128$
MEAN_block_5	$3 \times 3/1$	$64 \times 64 \times 256$
MEAN_block_6	$3 \times 3/2$	$32 \times 32 \times 512$
Apple_Inception	—	$32 \times 32 \times 512$
MEAN_block_7	$3 \times 3/1$	$32 \times 32 \times 512$
MEAN_block_8	$3 \times 3/2$	$16 \times 16 \times 256$
MEAN_block_9	$3 \times 3/1$	$16 \times 16 \times 128$
MEAN_block_10	$3 \times 3/2$	$8 \times 8 \times 64$
MEAN_block_11	$3 \times 3/1$	$8 \times 8 \times 64$
FC_With_Softmax	—	5

in the original SSD with the proposed pre-network of MEAN. As discussed in 4.2, apple leaf disease spots tend to have various sizes, such as the 5×6 size of Mosaic spots and the 91×57 size of Brown spot. As a result, the idea of feature pyramid is introduced into the MEAN-SSD model. The fully connected layer and softmax layer are removed and the MEAN_block_7 to MEAN_block_11 feature maps in the MEAN, with the sizes changed from 32×32 to 8×8, are sent to the SSD's detector to predict the type and generate the bounding box. The feature maps with larger sizes are used to detect tiny spots like Mosaic, while those with smaller sizes are used to detect large spots like Brown spot. The overall framework of the MEAN-SSD model is shown in Fig. 5.

5. Experiments and analyses

The experiments' setup is described in this section. Then, details on the experiments are provided, followed by the experimental results and analyses.

5.1. The setup of experiments

The experiments are conducted on an Ubuntu 16.04.2 server and a mobile phone device. The high performance Ubuntu server contains two Tesla P100 processors with 16 GB memory size, which is used for the model's training. Also, the Keras and TensorFlow frameworks are utilized to deploy the MEAN model on the Ubuntu server, due to the convenience of their Python interfaces.

The detailed information of the Ubuntu server is shown in Table 3.

In terms of the training strategy, the RMSProp algorithm is implemented to minimize the cost function during the training process. The settings of related parameters are shown in Table 4.

The configuration parameters of the mobile phone device, which is used for the detection process of the model, are listed in Table 5.

Table 3
The high performance Ubuntu server.

Configuration	Value
Central processing unit	Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20 GHz (X2)
Graphics processor unit	GP100GL [Tesla P100 PCIe 16 GB]
Operation system	Ubuntu 16.04.2 LTS (64-bit)
Memory	503 GB System memory
Hard Disk	2 TB
Language	Python 3.6
Deep learning framework	PaddlePaddle 2.6.0, TensorFlow 2.0.0

Table 4
Training strategy.

Parameter	Value
Optimizer	RMSProp
Batch size	32
Basic learning rate	10^{-3}
Decay rate	0.5
Top-N strategy	Top-1

Table 5
Mobile end environment.

Configuration	Value
Phone	HUAWEI Mate30 Pro 5G
CPU	HUAWEI Kirin 990 5G
Camera resolution	7296×5472
Operating system	EMUI 10.0

5.2. Accuracy comparison of pre-networks

In order to detect the location and category of lesions from the disease images, some necessary feature information is extracted from the diseased image. The pre-network part is responsible for extracting features from the image.

This section discusses the detailed information of the training and validating processes of several classical CNNs, for example VGG16, MobileNet, ResNet series and GoogLeNet. The recognition accuracy of MEAN is compared with the CNNs above on the AppleDisease5 dataset.

The comparison of the pre-networks' accuracy is shown in Fig. 8 and Table 6, where accuracy means the percentage of the correctly classified images in all images. According to Table 6 and Fig. 8, the recognition results of MobileNet and GoogLeNet are satisfactory, which inspires us to make the most of their strengths to build MEAN. Finally, the whole structure of MEAN is constructed based on the combination of MobileNet and the Inception module, which achieves a satisfactory accuracy (97.07%) and the fewest parameters(2,152,744) in terms of recognition performance.

5.3. The identification performance of MEAN on AppleDisease5

To demonstrate whether the MEAN model could precisely identify various apple leaf diseases, the selected CNN model (MEAN trained with the end-to-end method) is tested on the 5,000 images of 5 diseases. Different environmental conditions are simulated via the digital image processing technology utilized in Section 3.2. The confusion matrix of identification results is shown in Table 7.

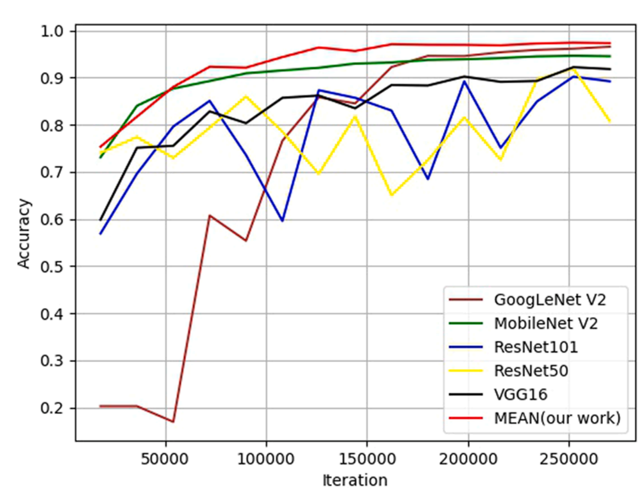


Fig. 8. Accuracy curve of the pre-network models.

It can be seen from the Table 7 that the recognition for Brown spot is the best, with the highest F1-Score of 0.983. The F1-Scores of Rust, Alternaria blotch and Gray spot reach 0.982, 0.980 and 0.970 respectively. The recognition for Mosaic disease is slightly lower, with the F1-Score of 0.966. It is speculated that the spot of Mosaic disease is too small, and it's slightly difficult for MEAN to extract features of such small objects. The overall F1-Score of MEAN is 0.976.

The results show that the MEAN proposed in this paper is capable of identifying most kinds of apple leaf diseases accurately.

5.4. The detection performance on the mobile end

In order to compare the performance of different detection algorithms, SSD and YOLO v3, two classical one-stage algorithms, and Faster R-CNN, a two-stage typical detection algorithm, are selected to detect apple leaf diseases on AppleDisease5 with different backbone networks on the mobile end. The experiment results are shown in Table 8, and Fig. 9 shows one example of the detected apple leaf diseases on the mobile phone.

In the object detection task, the model should be comprehensively evaluated with respect to the two perspectives of location and classification. The mean Average Precision, namely mAP, is the most commonly used index in detection problems, which is selected in the following experiments. Also, the definition of the IoU is as follows:

$$IoU(A,B) = \frac{area(A \cap B)}{area(A \cup B)}$$

In the equation above, A represents the ground-truth box, while B represents the bounding box. The min and max ratios of generated prior boxes are set to 0.2 and 0.9, respectively, and the threshold is set to 0.5.

In the experiment, MEAN-SSD is compared with GoogLeNet (Jadhav et al., 2020), ResNet101 (Xie et al., 2020), MobileNet (Howard et al., 2017), Faster R-CNN (Wang and Qi, 2019) and YOLO v3 (Venkataramanan et al., 2019). As is seen from Table 8, the SSD algorithm using GoogLeNet as the pre-network obtains the highest mAP of 87.90%. The multi-scale feature extraction capability of the Inception module, is well fit to the needs of apple leaf disease detection, due to the various sizes of disease spots, but as a result GoogLeNet consumes excessive computational resources and space, as shown in Table 8. The proposed MEAN-SSD achieves a slightly lower accuracy of 83.12% mAP, but the model size reaches 5.02 MB, which is nearly 5 × smaller than that of GoogLeNet.

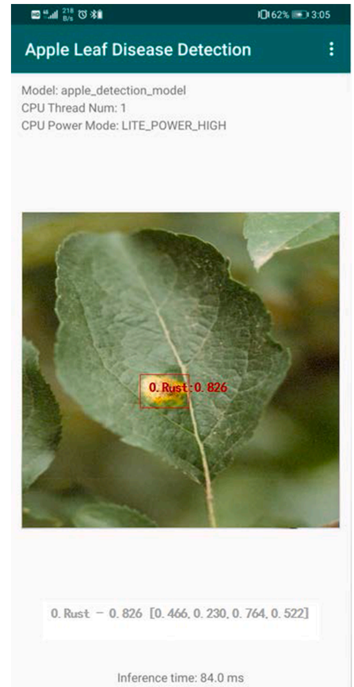


Fig. 9. Apple leaf disease detection results on the mobile phone.

Another important index for evaluating detection algorithms is detection speed, which is essential in real-time detection. Frames per second, namely FPS, is used to measure the speed of object detection. As is commonly known, more FPS enables faster detection speed. In the SSD algorithm, the model using MobileNet as the pre-network achieves the fastest detection speed, up to 15.384 FPS, with the input size of 512 ×

Table 6
Recognition Performance.

Models	GoogLeNetV2	MobileNetV2	ResNet50	ResNet101	VGG16	MEAN(our work)
Accuracy(%)	**97.22**	94.90	90.67	91.76	92.21	97.07
Parameters	11,264,111	3,228,864	26,636,712	23,561,152	138,357,544	**2,152,744**

Table 7
The confusion matrix of identification results.

		Predicted class						Precision	Recall	F1-Score
		Alternaria blotch	Brown spot	Gray spot	Mosaic	Rust	Total			
Actual class	Alternaria blotch	972	11	8	9	0	1000	98.9%	97.2%	0.980
	Brown spot	1	987	7	5	0	1000	97.9%	98.7%	0.983
	Gray spot	1	3	967	29	0	1000	97.4%	96.7%	0.970
	Mosaic	7	0	8	978	7	1000	95.5%	97.8%	0.966
	Rust	2	7	3	3	985	1000	98.0%	98.5%	0.982
	Total	983	1008	993	1024	1005	5000	97.5%	97.8%	0.976

Table 8
Detection results of various CNN models.

Method	SSD				Faster R-CNN	YOLO v3
Backbone	GoogLeNet	ResNet101	MobileNet	MEAN(our work)	MEAN	MEAN
mAP(%)	**87.90**	74.82	81.59	83.12	85.42	81.46
Speed(FPS)	7.54	5.10	**15.38**	12.53	11.62	13.67
Size(MB)	24.40	7.84	9.31	**5.02**	7.49	7.54

512. The reason is that the depthwise separable convolutions trade off a reasonable amount of accuracy to reduce size and latency, which increases the speed of feature extraction. A lower detection speed of 12.53 FPS but a higher mAP versus those of the detection model with MobileNet are realized by the proposed pre-network MEAN.

Additionally, the light-weight model is required, so it is still necessary to choose size to evaluate the model. The MEAN-SSD model achieves the best performance in terms of the model size. While maintaining the approximate detection speed of MobileNet, the model size is reduced by nearly one-half. This is because the computation process of common convolution is split to achieve the purpose of reducing computation. At the same time, group convolution is added into the MEAN block to further reduce the number of parameters and the information redundancy among the convolution kernels of the same layer.

Overall, the MEAN-SSD model can detect most of the apple leaf diseases on the mobile end efficiently and accurately, achieving a good trade-off among mAP, speed and size.

5.5. Ablation study on the model's performance

As is shown in Table 9, two ablation experiments are carried out to investigate the performance of the MEAN block and the Apple-Inception module, and another two ablation experiments are carried out to investigate the effects of simple and complex backgrounds of images on the model's performance.

First, the effect of split convolution mentioned in Section 4.1 is analysed, with all split convolutions of the MEAN block replaced with the common 3 × 3 convolutions in MEAN-SSD. It can be observed that the number of parameters rises sharply, while the mAP doesn't change much compared to the baseline model(the third row in Table 9), verifying the effect of split convolution that is to reduce the number of parameters and the information redundancy while maintaining the accuracy. And then, the impact of the Apple-Inception module mentioned in Section 4.2 has been tested. Common convolutions are used to replace the Apple-Inception module, decreasing mAP from 83.12% (baseline) to 80.32%. It demonstrates that the Apple-Inception module works well to extract multi-scale features from the disease spots, thus enhancing the detection accuracy of MEAN-SSD.

Regarding the differences in the detection accuracy between the original dataset and the complex dataset, it can be seen from Table 9 that the mAP of the model trained with the original dataset reaches 86.45%. After adding images of complex background into the original dataset, which becomes the complex dataset, the mAP decreases to 83.12%. The statistical results demonstrate that the complex backgrounds contain too much noise that has no relevance to the diseases. However, in order to improve the generalization ability and apply it to actual production requirements, the complex dataset is chosen to train the MEAN-SSD model in the experiments.

5.6. Design for Android platform

5.6.1. Paddle Lite

Paddle Lite is a mobile-end inference engine designed to deploy deep learning models on mobile devices. It consists of two components: the opt tool and the Paddle Lite interpreter for APP development. The opt tool is first to convert the trained model into the Paddle Lite model, which is nb model. Then, the Paddle Lite model is downloaded to the Android Studio platform. Finally, the Paddle Lite interpreter is configured to deploy the nb model into the smart phone and run the detection model with the resources of mobile devices.

5.6.2. APP development

As shown in Fig. 10, the apple leaf disease detection APP is composed of 4 modules: image capturing module, image preprocessing module,

Table 9
Results of ablation experiment.

Apple-Inception	Split Convolution	Original Dataset	Complex Dataset	mAP(%)	Parameters	Size(MB)
✓	×	✓	×	82.86	2,855,048	8.62
×	✓	✓	×	80.32	1,917,589	4.87
✓	✓	✓	×	86.45	2,152,744	5.02
✓	✓	×	✓	83.12	2,152,744	5.02

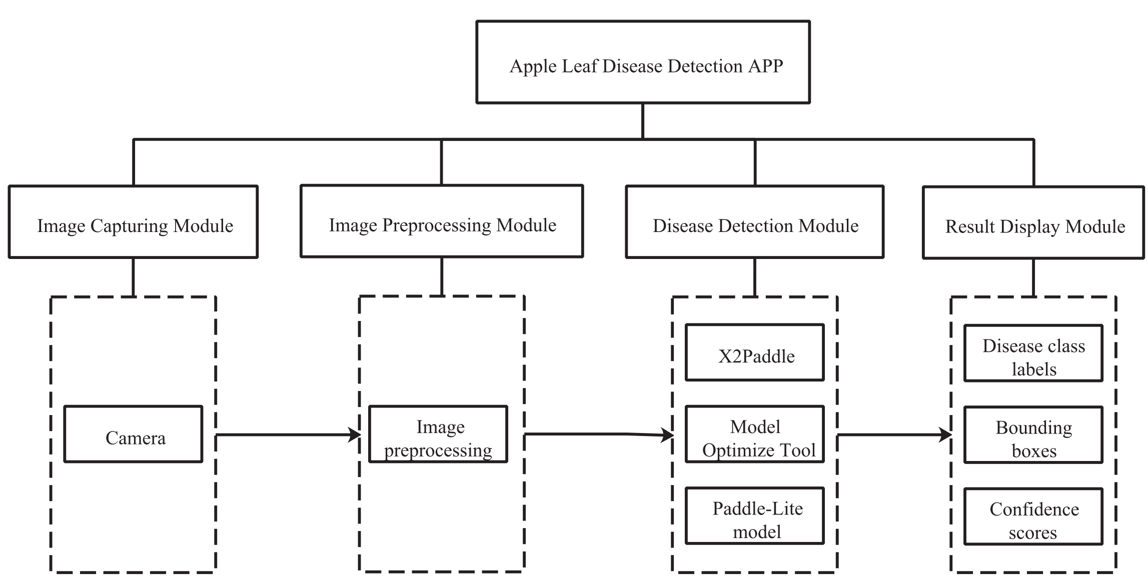


Fig. 10. The overall structure of the apple leaf disease detection APP.

disease detection module and result display module. First, the image capturing module captures an apple leaf disease image via the phone camera. Second, the input apple leaf disease image is cropped to the required shape of 256 × 256 resolution by the image preprocessing module, which can speed up image processing. Third, the disease detection module runs the Paddle Lite model configured by the Paddle Lite interpreter to detect the apple leaf diseases. Finally, the result display module shows disease class labels, confidence scores and bounding boxes of apple leaf diseases in the image. The APP development environment is based on the Windows 10 operating system and Java language, which is configured by Android Studio 4.1.0, JDK 11.0.9, and JRE 8.0.

6. Conclusions

In this paper, a mobile-end-based detection model, namely MEAN-SSD, is proposed for the real-time detection of apple leaf diseases on the mobile device. This model can be deployed on the mobile end, such as smart phones, and can automatically extract the features of apple leaf disease spots to detect 5 common apple leaf disease spots. To ensure a sufficient AppleDisease5 dataset and enhance the generalization performance of the model, 2,230 original apple leaf disease images with simple backgrounds from the laboratory and complex backgrounds from the orchards are collected, and a total of 26,767 images of disease spots for training are generated through data augmentation techniques. The MEAN-SSD model proposed in this paper improves the performance of multi-scale and minor disease spot detection on the mobile end by reconstructing the basic 3 × 3convolution and improving the Inception module.

The new mobile-based detection model is implemented in the Keras framework on the smart phone platform, which is HUAWEI Mate 30 Pro in the experiments. The detection performance of MEAN-SSD can reach 83.12% mAP and the speed is 12.53 FPS. The experiment results show that the MEAN-SSD model can accurately and efficiently detect apple leaf diseases, and it provides a practicable method for real-time detection that can be deployed on the mobile end.

Author contributions

Henan Sun and **Haowei Xu** carried out the primary work of designing the study, implementing and verifying the method, writing and revising the original manuscript. **Bin Liu** made the significant contribution of proposing idea, providing the research program, offering funding, and correcting the draft. **Dongjian He**, **Jinrong He**, **Haixi Zhang** and **Nan Geng** helped to provide constructive advice.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

We are thankful for the valuable opinions of the reviewers, which greatly help to improve the quality of this draft. This research is sponsored by the Key Research and Development Program of Shaanxi (2021NY-138, 2019ZDLNY07-06–01), by CCF-Baidu Open Fund (NO. 2021PP15002000), by the Fundamental Research Funds for the Central Universities (No. 2452019064), by National Natural Science Foundation of China (No. 61902339), by Yan'an University's Doctoral Starting up Foundation (No. YDBK2019-06), by Yan'an Special Foundation for Science and Technology (2019–01, 2019–13) and by China's National College Students Innovation and Entrepreneurship Training Program of Northwest A&F University (No. S202010712305, No. S202010712083).

References

Adeel, A., Khan, M.A., Sharif, M., Azam, F., Shah, J.H., Umer, T., Wan, S., 2019. Diagnosis and recognition of grape leaf diseases: An automated system based on a novel saliency approach and canonical correlation analysis based multiple features fusion. Sustainable Computing: Informatics and Systems 24, 100349.

Adeel, A., Khan, M.A., Akram, T., Sharif, A., Javed, K., 2020. Entropy controlled deep features selection framework for grape leaf diseases recognition. Expert Systems 167 (1), 105093.

Agarwal, M., Gupta, S.K., Biswas, K., 2020. Development of efficient cnn model for tomato crop disease identification. Sustainable Computing: Informatics and Systems 28, 100407.

Chen, J., Yin, H., Zhang, D., 2020. A self-adaptive classification method for plant disease detection using gmdh-logistic model. Sustainable Computing: Informatics and Systems 28, 1–19.

Deng, R., Jiang, Y., Tao, M., Huang, X., Bangura, K., Liu, C., Lin, J., Qi, L., 2020. Deep learning-based automatic detection of productive tillers in rice. Computers and Electronics in Agriculture 177, 105703.

Freeman, I., Roese-Koerner, L., Kummert, A., 2018. EffNet: An efficient structure for convolutional neural networks, in. In: Proceedings of the IEEE International Conference on Image Processing, pp. 6–10.

Fu, L., Feng, Y., Wu, J., Liu, Z., Gao, F., Majeed, Y., Al-Mallahi, A., Zhang, Q., Li, R., Cui, Y., 2021. Fast and accurate detection of kiwifruit in orchard using improved yolov3-tiny model. Precision Agric. 22 (3), 754–776.

M.J. Hasan, S. Mahbub, M.S. Alom, M.A. Nasim, Rice disease identification and classification by integrating support vector machine with deep convolutional neural network, in: Proceedings of the IEEE Conference on Advances in Science, Engineering and Robotics Technology, 2019, pp. 1–6.

K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861 (2017) 1–9.

Hua, L.Q., Ping, L.C., Jing, Z., Hong, C., Qing, W.S., 2019. Survey of compressed deep neural network. Computer Science 1–14.

Jadhav, S.B., Udupi, V.R., Patil, S.B., 2020. Identification of plant diseases using convolutional neural networks. International Journal of Information Technology 1–10.

Kaur, P., Pannu, H.S., Malhi, A.K., 2019. Plant disease recognition using fractional-order zernike moments and SVM classifier. Neural Comput. Appl. 31 (12), 8749–8768.

Khan, M.A., Akram, T., Sharif, M., Awais, M., Javed, K., Ali, H., Saba, T., 2018. CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep cnn features. Computers and Electronics in Agriculture 155, 220–236.

Khan, A.I., Quadri, S., Banday, S., 2020. Deep learning for apple diseases: classification and identification, arXiv preprint arXiv:2007.02980, pp. 1–15.

Khan, M.A., Akram, T., Sharif, M., Javed, K., Saba, T., 2020. An automated system for cucumber leaf diseased spot detection and classification using improved saliency method and deep features selection. Multimedia Tools and Applications 79 (1), 1–14.

Kim, W.-S., Lee, D.-H., Kim, Y.-J., 2020. Machine vision-based automatic disease symptom detection of onion downy mildew. Computers and Electronics in Agriculture 168, 105099.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 1097–1105.

W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: single shot multibox detector, in: Proceedings of the European Conference on Computer Vision, Springer, 2016, pp. 21–37.

Picon, A., Seitz, M., Alvarez-Gila, A., Mohnke, P., Echazarra, J., 2019. Crop conditional convolutional neural networks for massive multi-crop plant disease classification over cell phone acquired images taken on real field conditions. Computers and Electronics in Agriculture 167, 1–14.

Sharif, Muhammad, Khan, M. Attique, Iqbal, Zahid, Azam, M. Faisal, Lali, M.I. a. Ullah, Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection, Computers and Electronics in Agriculture 167 (2018) 1–14.

K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014) 1–14.

K.K. Singh, An artificial intelligence and cloud based collaborative platform for plant disease identification, tracking and forecasting for farmers, in: Proceedings of the IEEE International Conference on Cloud Computing in Emerging Markets, Vol. 167, 2018, pp. 49–56. doi:10.1109/CCEM.2018.00016.

C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

Venkataramanan, A., Honakeri, D.K.P., Agarwal, P., 2019. Plant disease detection and classification using deep neural networks. Int. J. Comput. Sci. Eng 11 (9), 40–46.
Wang, Q., Qi, F., 2019. Tomato diseases recognition based on faster rcnn. In: in: 2019 10th International Conference on Information Technology in Medicine and Education (ITME), pp. 772–776.
Xie, X., Ma, Y., Liu, B., He, J., Wang, H., 2020. A deep-learning-based real-time detector for grape leaf diseases using improved convolutional neural networks. Frontiers in Plant Science 11, 1–11.
X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856.
Zhang, L., Jia, J., Gui, G., Hao, X., Gao, W., Wang, M., 2018. Deep learning based improved classification system for designing tomato harvesting robot. IEEE Access 6, 67940–67950.
Zhou, Z., Song, Z., Fu, L., Gao, F., Li, R., Cui, Y., 2020. Real-time kiwifruit detection in orchard using deep learning on android smartphones for yield estimation. Computers and Electronics in Agriculture 179, 105856.