# Disease and Pest Detection for the Growing Stages of Eagle-billed Peach Using FSL Based on Transfer Learning

Yongfu Zhou, Zhi Zeng* and Yuwan Gan

*Abstract*— **Pest detection and prevention remain significant real-world challenges because of the diversity of pests that affect plants at different stages of growth, and the eagle-billed peach is no exception. Despite advancements in Internet of Things and deep learning technologies, traditional manual identification methods still rely on a substantial amount of labeled data in building robust models. This reliance leads to a time-consuming data acquisition process and raises issues related to low work efficiency and high costs. Therefore, this study proposes a few-shot learning approach for identifying diseases and pests during the growing stages of the eagle-billed peach. Our method utilizes deep residual networks to extract discriminative image features, effectively avoiding the gradient vanishing problem that can occur with increased network depth. This method achieves more precise and rapid monitoring and prevention of diseases and pests. The algorithm classified and labeled 14 categories of disease and pest characteristics during the growing stages of the eagle-billed peach. Subsequently, it employed a transfer learning method to extract features from newly added category samples. The parameters of the small-sample deep learning network were adjusted to enhance classification accuracy and provide real-time alerts. Furthermore, a graphical interface was constructed to enable image uploads and model inference. This interface provides rational suggestions for pest control. In experiments identifying and classifying several datasets, accuracy improved by 12.4%. This indicates that the proposed transfer learning-based small-sample deep learning network exhibits high image classification precision and strong generalization capabilities.**

*Index Terms*—**Few-shot learning, eagle-billed peach, pest and disease detection, transfer learning**

## I. Introduction

DURING the plant growth period, individuals typically use visual observations to monitor the status of trees and assess irregularities. Characteristics of pests and diseases, such as yellowing, curling, and leaf spotting, may indicate the presence of disease or pest issues. Apart from growth conditions, monitoring the sprouting status and mortality rate of plants is also crucial for acquiring a more comprehensive understanding of tree health. Traditional methods for diagnosing plant leaf diseases depend primarily on experience and manual observation, which are insufficient for meeting the demands of speed and efficiency. First, horticulturists have an enormous workload, making it impractical to examine each leaf individually. Furthermore, even seasoned professionals encounter problems with diagnostic accuracy including missed or incorrect diagnoses. Consequently, the use of artificial intelligence for plant leaf disease recognition has become a prominent research area.

Advancements in planting techniques and pest management remain significant challenges in enhancing the yield and quality of eagle-billed peaches. Traditional supervised learning methods for disease and pest detection require numerous labeled samples to build a model, resulting in a time-consuming and labor-intensive image acquisition process [1]. Furthermore, collecting adequate defect data from various damage scenarios is unrealistic. Although artificial intelligence (AI) has been widely applied in agriculture with notable results, supervised models can only recognize specific diseases and pest defects, necessitating further training using new examples of novel classes. Gidars et al. proposed the use of transfer learning to achieve convergence using only a few labeled samples [2]. In contrast to supervised approaches, few-shot learning (FSL) uses only one or a few labeled samples to recognize new classes and applies the generated knowledge to new images. Significant efforts have been made in this regard. A typical FSL problem aims to identify objects with minimal samples [3], thereby compensating for the limitations of supervised learning in various domains. This makes it imperative to develop an FSL approach for vision-based monitoring of existing and new diseases and pests, using weakly supervised information. This paper proposes an FSL model based on an improved prototypical network (ProtoNet) [4] for disease and pest detection, specifically targeting eagle-billed peaches. Initially, inspection images were categorized into several classes. Feature embedding was achieved through cross-domain transfer learning using ImageNet [5]. A linear classifier, $\omega^T x + b$, was added to the end of ProtoNet for classification and fine-tuned based on the support set. Subsequently, the resulting prototype and fine-tuned classifier were applied to the new inspection images.

Although the proposed method offers several advantages, it has certain limitations, including reliance on a small number of samples and the presence of similar features across

Manuscript received November 7, 2024; revised May 18, 2025.

This work was supported in part by the Key project of the Guangdong Provincial Department of Education of China under Grant 2022ZDZX3030 and Guangdong Rural Science and Technology Special Envoy Project (No. KTP20200292, KTP20200360).

Yongfu Zhou is an associate professor in the Institute of Electronic and Information Engineering at Heyuan Polytechnic, Heyuan 517000, China. (e-mail: afyours@126.com).

Zhi Zeng is a professor in the school of Computer Science and Engineering at Huizhou University, Huizhou 516007, China. (Corresponding author. Phone: 86-752-2527105; fax: 86-752-2527101; e-mail: zengzhi@hzu.edu.cn).

Yuwan Gan is an assistant lecturer in Heyuan Polytechnic at Heyuan 517000, China. (e-mail: 351672568@qq.com).

different classes. In addition, varying the support sets can alter the performance in few-shot disease and pest detection. Therefore, further research is required to enhance feature learning of the samples.

In summary, the complexity and variability of the environment present challenges to the practical application of pest and disease management, with recognition rates in intricate backgrounds remaining a significant hurdle. This makes it essential to optimize and enhance approaches that focus on specific leaf characteristics and fruit growth cycles. By introducing few-shot deep learning algorithms for image processing and adapting them to increasingly complex and diverse applications, the rate and accuracy of pest and disease identification can be improved. The contributions of this study are as follows:

1) An FSL approach based on an enhanced ProtoNet is proposed for monitoring diseases and pests in eagle-billed peaches. In this approach, feature embedding is achieved through cross-domain transfer learning from ImageNet instead of episodic training.
2) A feature extraction algorithm based on the deep residual network ResNet50 is proposed for practical applications.
3) The proposed approach was validated on actual inspection applications, demonstrating its potential for near real-time inspection of disease and pest detection as it can be implemented rapidly with few labeled samples.

The remainder of this paper is organized as follows: Section 2 reviews related work on disease and pest detection as well as few-shot learning for images; Section 3 presents the proposed approach and its architecture, along with a theoretical foundation; Section 4 covers the experiments conducted to validate the approach; and Section 5 concludes the paper.

## II. Related Works

### A. Disease and pest classification and detection

The fundamental task is to determine the class to which a disease or pest belongs. Our objective was to achieve either a binary classification for each defect or a multi-defect classification. Damage detection aims to provide additional information about a disease or pest, including its location, shape, and direction. This is important because classification merely indicates the presence of defects in an image, leaving the task of locating the actual defect to inspectors [6]. A typical approach to disease and pest detection involves sliding a window over an image or splitting the image into patches, followed by classification of each window or patch. Another method utilizes bounding boxes to indicate defects, similar to object detection tasks in public datasets such as common objects in context (COCO [7] and Pascal visual object classes (VOC)) [8]. However, this approach may not always locate damage effectively because defects can exhibit varying appearances. For instance, a large bounding box may contain numerous non-defective subregions, such as an oblique crack marked by a sizeable bounding box determined by its diagonal points. Image processing methods for damage detection frequently underperform in real-world inspection images because of interference from surface textures, changing light conditions, stains, and other factors [9]. Several data-driven approaches based on AI have been developed to assist in the classification and detection of diseases and pests during visual inspections.

In traditional machine learning (ML)-based approaches, image processing is required to extract predefined features. A major issue with traditional ML methods is their dependence on handcrafted features, which often result in shallow learned representations [10]. In contrast, deep learning (DL) can automatically extract features using a multilayer neural network. Both ML and DL approaches are based on inductive supervised learning, with the performance depending on the pre-collected annotated samples available before inspection. These methods require pre-trained models to detect specific types of defects and struggle to adapt quickly to novel defects. However, data annotation is often time consuming and tedious. In addition, collecting sufficient defect images from damage scenarios is not always feasible. Traditional supervised transfer learning is expected to address this issue; however, it tends to struggle with overfitting and cannot converge with only a limited number of labeled examples.

Deep learning has emerged as a relatively new field of research, both nationally and internationally, displaying significant advancements. Liang et al. enhanced their dataset employing techniques such as flipping images and adjusting image contrast [11]. They captured images of rice pests using triangular traps and monitoring equipment, achieving a precision rate of 91.67% and recall rate of 98.30% for recognizing the rice leaf roller during testing. Wang et al. gathered 2,566 original images of grape diseases from the National Key Laboratory of Biology for Plant Diseases and Insect Pests [12]. They augmented and expanded this dataset, obtaining a comprehensive collection of up to 32,871 images. Testing this dataset achieved a precision rate of 98.60%.

Therefore, we used a high-definition camera to capture the growth conditions of the roots, leaves, and stems of plants from various angles. The growth stages of the plants were compared with those in annotated images of healthy specimens. Upon recognizing any diseased or pest-affected areas in the plant growth images, we assessed the type of growth anomaly by analyzing the detected regions. This process enabled us to identify variations in plant growth and the presence of pests and diseases. The control program subsequently directed the target light-emitting diode (LED) source to address the growth anomalies observed during plant growth. An intelligent early warning system oversaw the entire lifecycle of plant cultivation, gathered information to establish pest-monitoring models, and transformed the accumulated experience into regional planning and system management based on the specific plants being cultivated.

### B. Few-shot learning for images

The labor-intensive and time-consuming data acquisition process poses a bottleneck in applying supervised machine learning (ML) in many fields. However, FSL, encompassing few-shot classification and segmentation, can address this issue by learning from a limited number of annotated images, thereby enhancing data efficiency. This study focused on few-shot classification, which is often regarded as a type of meta learning. A meta-learner is trained through a series of related tasks (episodic training) to perform well on unseen but related tasks with only a few examples. Transduction has been widely adopted for FSL tasks in both training and inference because it is more effective at utilizing only a limited number of labeled samples than induction using supervised models [13].

Significant efforts have been made in FSL, including the development of specific image datasets [14, 15] such as Omniglot, CIFAR-FS, and mini-ImageNet, along with various approaches. Some studies use different data augmentation methods, such as self-augmentation [16], deformation [17], and deep convolutional generative adversarial network (DCGAN) [18], to address the problem of few-shot classification with limited training samples. Others focus on learning effective model initializations [19] or optimizers [20], to achieve rapid adaptation using a limited number of training examples for new classes; yet others prefer to use the Euclidean distance or cosine similarity [21].

Deep learning has been widely applied to monitor agricultural pests and diseases, achieving significant results. Ye et al. [22] utilized MobileNetV2 to identify three types of corn diseases and achieved high accuracy with low computational load. Lin et al. [23] added a dropout layer to MobileNetV2 to prevent overfitting when identifying tomato disease images. Zhang et al. [24] successfully identified four types of apple leaf diseases using the VGG16 network model and diagnosed the severity of bacterial infections in apple leaves with an accuracy of 90.4%. Bah et al. [25] proposed an unsupervised training method for convolutional neural networks to monitor plants from remote sensing images, which can potentially be extended to the agricultural sector.

For small-sample datasets, Wang et al. [26] explored the classification of plant pests and diseases using a Siamese network framework. Hu et al. [27] studied the classification and monitoring of tea disease samples using a conditional (C)-DCGAN. Melike et al. [28] applied a convolutional neural network (CNN) and learning vector quantization (LVQ) algorithm to detect the areas of tomato leaf pests and diseases, achieving an average recognition rate of 86%. Zhang et al. [29] introduced an online hard-sample mining method for pest classification and recognition that was particularly effective for small targets, resulting in noticeable improvements.

Researchers have investigated the integration of object detection networks to recognize pests and diseases. Jing et al. [30] combined a YOLO network with an attention mechanism to detect diseases and control pests on rice leaves. Zhang et al. [31] proposed a multiscale unsupervised network (MU-Net) structure for segmenting images of crop disease and pest areas, thereby enhancing feature representation and achieving an accuracy of 95.13%. Kerkech et al. [32] used SegNet to fuse multispectral image information and achieved a detection accuracy of 87% for grape leaves. Afzaal et al. [33] employed a Mask R-CNN network to recognize seven types of strawberry diseases under varying background conditions, achieving an average precision of 82.43%. Gao et al. [34] proposed a segmentation method based on a Swin Transformer, effectively resolving the issue of obscured pests with a recognition accuracy of 88 %.

In summary, traditional image recognition methods have been gradually replaced by deep learning algorithms. In practical applications, the complexity and variability of scenes pose challenges for detection tasks, and the recognition of complex backgrounds remains a significant hurdle. However, existing methods still have some limitations in terms of accuracy, feature extraction, and real-time performance, making it essential to optimize and improve these techniques according to the specific characteristics of the leaves and growth stages of the fruits. Introducing deep learning-based FSL algorithms for image processing can assist in adapting to increasingly complex and diverse application scenarios, thereby improving the identification rate and accuracy of pest and disease detection.

## III. Deep Transfer Learning-based Model for Identifying Diseases and Pests

This study focused on 14 major categories of pests and diseases affecting eagle-billed peaches in Heyuan-Lianping, China. Images of leaves and fruits were collected using various devices including drones, fixed-position cameras, and smartphones to construct a comprehensive dataset. After preprocessing to exclude collected images with poor focus or unclear pest and disease characteristics, data augmentation was performed. The dataset was then randomly divided into training, validation, and testing sets in a ratio of 8:1:1, to obtain 690 images for training, 86 images for validation, and 87 images for testing. Labeling software was used to annotate the converted dataset by manually setting the categories of diseases and pests. Table I lists the 14 categories.

TABLE I
14 Categories Of Disease And Pest Classes

| Class name | No. of CLASS |
|---|---|
| Physiological fruit drop | 0 |
| Calcium deficiency | 1 |
| Mealybug infestations | 2 |
| Potassium deficiency | 3 |
| Boron and zinc deficiencies | 4 |
| Citrus fly infestations | 5 |
| Brown rot of peach (sclerotinia) | 6 |
| Peach anthracnose disease | 7 |
| Bacterial spot of peach | 8 |
| Branch canker symptoms | 9 |
| Red-brown spot disease | 10 |
| Bark beetle infestations | 11 |
| Cankers | 12 |
| Nut webworm infestations | 13 |

Finally, a recognition model was established for pest and disease identification. The proposed detection approach was derived from ProtoNet [4], which employs episodic training across a series of related tasks and prototypical transduction with the Euclidean distance for few-shot classification. This model introduces the following three key improvements:

1) Previous episodic training is replaced with cross-domain transfer learning from ImageNet for “training-free” feature embedding.
2) Embedding normalization is integrated to reduce domain variation and enhance the original ProtoNet performance based on the Euclidean distance.
3) Fine-tuning methods based on a fully connected network (FCN) and the Hadamard product can achieve better performance in fewer epochs than previous transductive fine-tuning methods [25]. The architecture of the approach is shown in Fig. 1, presenting an example of 2-way 3-shot crack detection, with detailed steps.

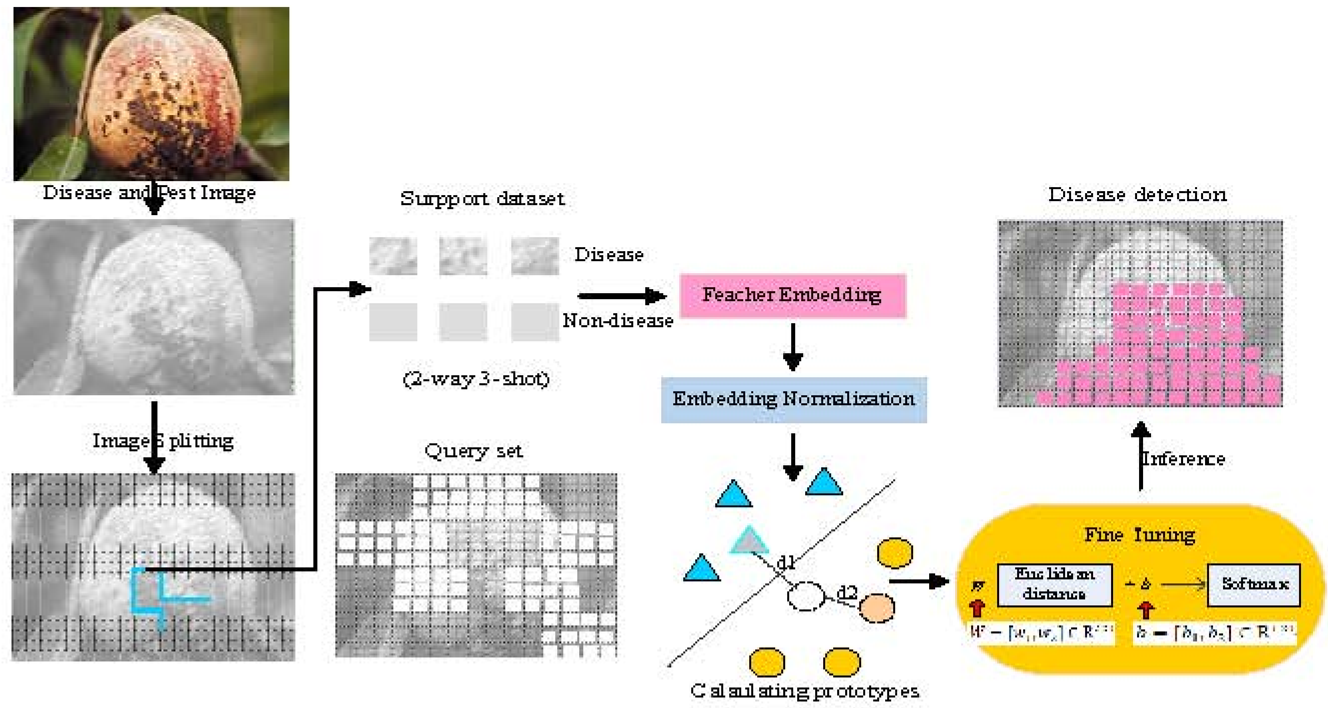



Fig. 1. Example of 2-way 3-shot improved ProtoNet framework for disease and pest identifying model

## IV. FEATURE EXTRACTION ALGORITHM BASED ON DEEP RESIDUAL NETWORKS

The residual network proposed by Microsoft Research is a type of convolutional neural network [35]. The primary characteristics of residual networks include ease of optimization and enhanced accuracy achieved through increased depth. Internal residual blocks utilize skip connections, which help mitigate the vanishing gradient problem often associated with deepening neural networks.

The residual units within the structures are a key aspect of residual networks. As illustrated in Fig. 2, the residual block features cross-layer connections that allow the input to pass directly across layers with equivalent mapping.

Assuming that the input image is denoted by x, and the output is H(x), after convolution, the output can be represented by the nonlinear function F(x). Thus, the final output can be expressed as $H(x) = F(x) + x$. This output can still undergo nonlinear transformations, where the residual refers to the "difference," specifically F(x). Consequently, the network is transformed and tasked with estimating the residual function $F(x) = H(x) - x$, which is generally easier to optimize than $F(x) = H(x)$.

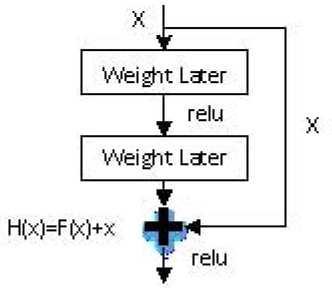



Fig. 2. Example of residual units in residual networks

The ResNet-50 network consists of 49 convolutional layers and one fully connected layer. As shown in Fig. 3, the ResNet-50 architecture can be divided into seven parts. The first part does not include residual blocks and primarily performs convolution, regularization, activation, and max pooling on the input. The second, third, fourth, and fifth parts contain the residual blocks. The green components in the diagram do not alter the sizes of the residual blocks; they are used only to change the dimensions of these blocks. In the ResNet-50 structure, each residual block contains three convolutional layers. Therefore, the total number of convolutional layers is calculated as $1 + 3 \times (3 + 4 + 6 + 3) = 49$. With the addition of a final fully connected layer, the total number of layers reaches 50; hence, the name ResNet-50 is used. The input to the network undergoes convolutional computations in the first five parts, and the pooling layer transforms it into a feature vector. Finally, the classifier processes the feature vector and outputs the category probabilities.

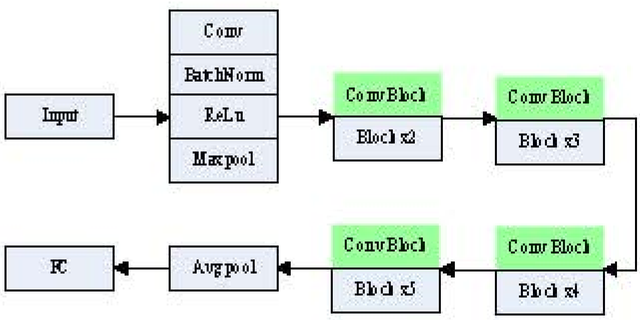



Fig. 3. ResNet-50 network structure diagram

## V. Transfer Learning-Based Model Training Strategies

In deep learning networks, one- and few-shot learning are critical theoretical foundations for visual detection. One-shot learning falls within the broader category of transfer learning [36]. Transfer learning, which is a widely used approach in the field of computer science, involves applying a pretrained model to a specific task. This generally entails training the model on certain datasets such as ImageNet, followed by fine-tuning and transferring the learned information to another dataset. The rationale for knowledge transfer originates from the fact that machine learning, particularly supervised learning, usually requires extensive labeled data, which can be massive and complex. Transfer learning enables the already acquired powerful knowledge to be adapted to a new context, thereby helping the network acquire relevant features more effectively and quickly.

However, small sample sizes can lead to CNNs learning features that are overly coarse and lack fine granularity, making it difficult to articulate the essential characteristics for pest and disease management. The learning process of the model is typically lengthy. Therefore, transfer learning can significantly reduce the time required for model training, making it suitable for the training and testing of complex CNNs. Given sufficient diversity and quantity of sample data, the network model can learn discriminative feature representations.

Transfer learning allows the knowledge acquired in the source domain to be conveyed to the target domain through feature-mapping methods, thereby enhancing the feature expression capabilities in the target domain. In general, the source domain consists of datasets with numerous high-quality labeled samples, with complex CNNs designed to learn distinguishing feature representations. The target domain corresponds to practical application scenarios where the number of training samples is limited, making it challenging to construct a large-scale labeled dataset within a short timeframe. Moreover, real-time applications may require lightweight CNN models for deployment. A fundamental requirement is that these models possess the capability to represent the knowledge learned from complex models of the source domain.

If only one image is used as the input to a CNN and a softmax unit outputs multiple labels corresponding to different targets, the actual performance is often unsatisfactory. This is because a small training dataset is insufficient to train a robust neural network. Adding new sample sets inevitably requires retraining the neural network. Therefore, for one-shot learning, the neural network must learn a similarity function that calculates the difference values, as shown in (1).

$$d(img1, img2) = degree\ of\ diff.\ between\ images. \quad (1)$$

During the recognition process, a threshold is set as the hyperparameter. If distance $d$ exceeds this threshold, the objects are considered to be the same target. The system can continue to function normally regardless of whether new samples are added to the dataset. However, this type of learning does not achieve high accuracy, necessitating a transition to FSL to recognize new targets.

In practical applications, collecting and labeling a large number of samples in a short period is infeasible. Although increasing the complexity of the network model can enhance recognition accuracy, deploying such models in resource-constrained environments poses a challenge. The key issue is to transfer the knowledge learned from complex models to simpler ones, thereby enabling the latter to possess the feature-representation capabilities of the former.

To address these challenges, we developed a deep residual network algorithm based on transfer learning to identify pests and diseases in peach trees (Fig. 4).

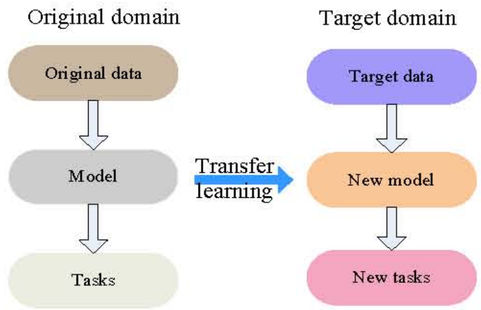



Fig. 4. Process of cross-domain transfer learning

## VI. Testing And Experiments

### A. Dataset Preprocessing

Considering the major pests and diseases affecting eagle-billed peaches in the Heyuan-Lianping area, a dataset comprising 14 images of these pests and diseases was created (Fig. 5). The CNN design, training, and optimization were closely integrated with the tasks of identifying pests and diseases, to assess their severity. The identification results were sent to an early warning platform for real-time monitoring.

The purpose of data preprocessing is to enhance the diversity of the dataset samples enabling the deep CNN to learn discriminative semantic information, thereby improving its generalization capability and adaptability to various scenarios. Common data preprocessing techniques include cropping, horizontal flipping, grayscale conversion, Gaussian smoothing, contrast equalization, contrast enhancement or reduction, random noise addition, random background blending, and sample fusion.


Fig. 5. Sample of 14-categories of pest and disease datasets for eagle-billed peach

*B. Testing Environment*

The experimental environment was built on a personal computer, and the hardware configuration details are listed in Table II. The software environment was configured using the Windows 10 operating system, PyCharm platform, CUDA11.6, cuDNN8.9.7, and PyTorch 2.1.0, deep learning framework using Python 3.8.8 as the programming language.

TABLE II
HARDWARE CONFIGURATION FOR TESTING

| DEVICE | NAME of MODEL |
|---|---|
| Processor | Intel Core i9-10900 CPU |
| Display adapter | NVIDIA GEFORCE RTX4070Ti |
| Memory | Kingston 2666MHz 64GB |
| Hardware Disk | WD Blue SN570 1TB SSD |

The training parameters were set to 300 epochs, a batch size of 4, and a learning rate of 0.003, using adaptive moment estimation (Adam) as the optimizer and a rectified linear unit (ReLU) activation function. Initially, the model was pre-trained on a large ImageNet-1k dataset. The classifier obtained after training, which recognized 1,000 categories, was modified to correspond to the 14 categories relevant to mango pest and disease identification. The dataset consisted of 6,769 images and was divided into training, validation, and testing sets in a 7:2:1 ratio. The images were then fed into the ResNet-50 model. The experiment employed a transfer learning technique commonly used in contemporary deep learning, to enhance the generalizability of the model and mitigate the overfitting resulting from the relatively small dataset.

*C. Evaluation Metrics*

Various quantitative metrics were used to evaluate the performances of the model.

*1) Loss function*

The loss function reflects the error rate between the predicted and ground truth values. This is helpful for iterative optimization during model training and effectiveness evaluation of the model for detection. The model loss in the object detector comprises a combination of disease and pest classification losses, as well as a positive sample prediction bounding box bias loss.

$$\mathrm{L}_{detec} = L_{cls} + \mu L_{reg}. \tag{2}$$

where

$$L_{cls} = \sum_{n=1}^{N} -w_{pos}^{n} \times \ln(s^{n}) - w_{neg}^{n} \times \ln(1 - s^{n}) + \sum_{m=1}^{M} FL(s^{m},0)$$
$$\mathrm{L}_{reg} = \sum_{n=1}^{N} w_{pos}^{n} \times GIoU(loc, loc')$$
$$\mathrm{FL} = \begin{cases} -a \times (1 - s^{m})^{\eta} \times \log s^{m} & s^{m} = 1 \\ -a\ (1 - a)[1 - (1 - s^{m})]^{\eta} \times \log(1 - s^{m}) & others \end{cases}. \tag{3}$$

$$\begin{aligned}\mathrm{GIoU}_{loss} &= 1 - GIoU(loc, loc') \\ &= 1 - [IoU - \frac{Bbox_{min} - Uion(loc, loc')}{Bbox_{min}}].\end{aligned} \tag{4}$$

In (2), the object detector loss $\mathrm{L}_{detec}$ comprises $\mathrm{L}_{cls}$ and $\mathrm{L}_{reg}$, where $\mathrm{L}_{cls}$ denotes the predicted category loss; $\mathrm{L}_{reg}$ represents the predicted regression loss; and $\mu$ is a modulation factor. $N$ and $M$ represent the number of detected frames in the candidate set and number of detected boxes outside the candidate set, respectively. FL stands for focal loss, whereas generalized intersection over union (GIoU) is an important computational metric used to measure the degree of difference between the predicted and target boxes and is a key factor for improving model performance. $loc$ and $loc'$ refer to the positions of the predicted and actual boxes. In (3), $\alpha$ is responsible for balancing the importance between positive and negative samples, and $\eta$ regulates the rate of weight reduction for simple samples. In (4), $Bbox_{\mathbf{min}}$ is the smallest enclosing convex object of $loc$ and $loc'$.

*2) Average precision (AP)*

Relying solely on either accuracy or recall for model assessment is inadequate; therefore, a composite metric, AP, was introduced to gauge model performance. In this study, we specifically focused on detection precision for various pests and diseases. AP is calculated as follows:

$$AP = \int_0^1 P(R)\mathbf{dr}. \tag{5}$$

Here, $P$ measures the precision of the model. Accuracy refers to the ratio of correctly predicted instances to the total number of predictions, which in this study pertains to the proportion of correctly identified pest and disease counts relative to the total number predicted by the model. $R$ denotes the ratio of correctly predicted samples to the total sample size; in this context, it indicates the proportion of correctly identified pest and disease counts to the overall number of pests and diseases.

*3) Mean average precision (mAP)*

As a comprehensive metric that assesses the overall performance of an algorithm, mAP refers to the average detection accuracy across all categories of pests and diseases in this study and is calculated as follows:

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i. \tag{6}$$

The larger the mAP value, the better is the performance of the algorithm. In addition, under different conditions, mAP can take various forms, with mAP@50 and mAP50-95 commonly used in object detection. mAP@50 refers to the average precision across all classes when IoU reaches 0.5 in an object detection task. In contrast, mAP@50-95 refers to the average precision across all classes as IoU varies from 0.5 to 0.95. In this study, mAP@50 was used as the standard.

*D. Testing*

The ProtoNet dataset, created using annotation software, typically consists of the following elements: [class_id × y × h]. Here, class_id refers to the ID number of the category; x is the x coordinate (horizontal) of the target center point relative to the total width of the image; y is the y coordinate (vertical) of the target center point relative to the total width of the image, w is the width of the bounding box relative to the total width of the image, and h is the height of the bounding box relative to the total height of the image. In the lower-right corner of Fig. 6, the results are shown in five columns of data for each row of the 1.txt file. Before training, the dataset was divided into training and validation datasets in an 8:2 ratio.

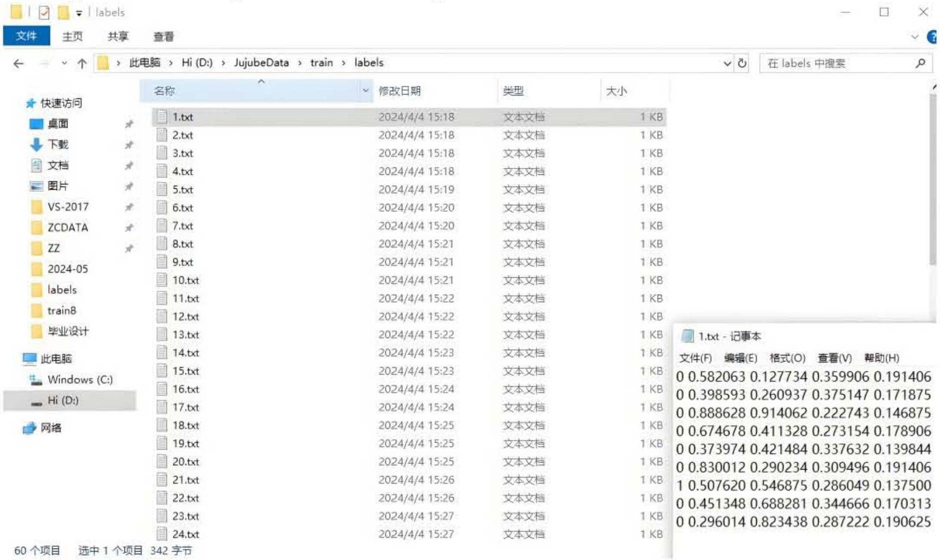



Fig. 6. Sample of labeled image data

In deep learning, the model-training process is typically monitored using a loss-function curve. In the context of YOLOv8, ProtoNet incurs three types of losses during training: localization (box_loss), classification (cls_loss), and dynamic feature (dfl_loss) losses.

During this calculation, the target box was scaled to the size of the feature map by dividing it by the corresponding stride and compared with the predicted bounding box to compute the complete intersection over union (CIoU) loss. Simultaneously, the distance from the center of the predicted anchors to the edges was used to calculate the regression DFLLoss. As shown in Fig. 7, this process is part of the ProtoNet training workflow, and the DFLLoss allows more accurate adjustments to the predicted box location.

The precision-recall (PR) curve is typically used to illustrate the relationship between precision and recall. The PR curve for the training results of this study is shown in Fig. 8. mAP represents the area under the PR curve, with "m" indicating the mean. The number following "@" represents the threshold used to determine the positive and negative samples based on the IoU. For instance, mAP@0.5 indicates the average mAP when the threshold is greater than 0.5. The average mAP@0.5 for the two object detection classes in this model was 0.790, which is considered remarkably good. Fig. 9 shows an example of a real-time detection interface for eagle-billed peaches.

From the object detection results, the location, category, and confidence levels were extracted and passed to the interface display module to draw the detected bounding boxes on the interface.

Performance curve comparisons of the mAP, parameter counts, and latency results for the proposed model, ProtoNet, and YOLOv8, as tested on the dataset, are shown in Fig. 10. Evidently, our model significantly improves the accuracy compared to both ProtoNet and YOLOv8. However, the corresponding parameter counts for the N/S/M models also increase considerably, resulting in slower inference speeds for most models compared with ours.

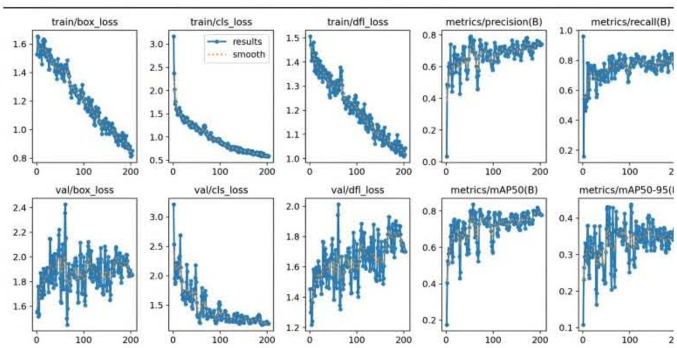



Fig. 7. Model training results

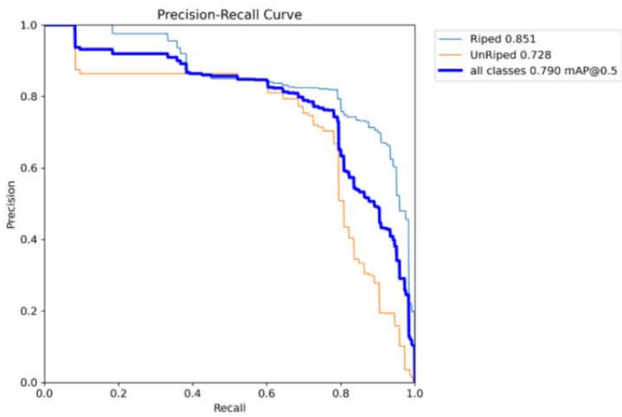



Fig. 8. PR curve of the training results

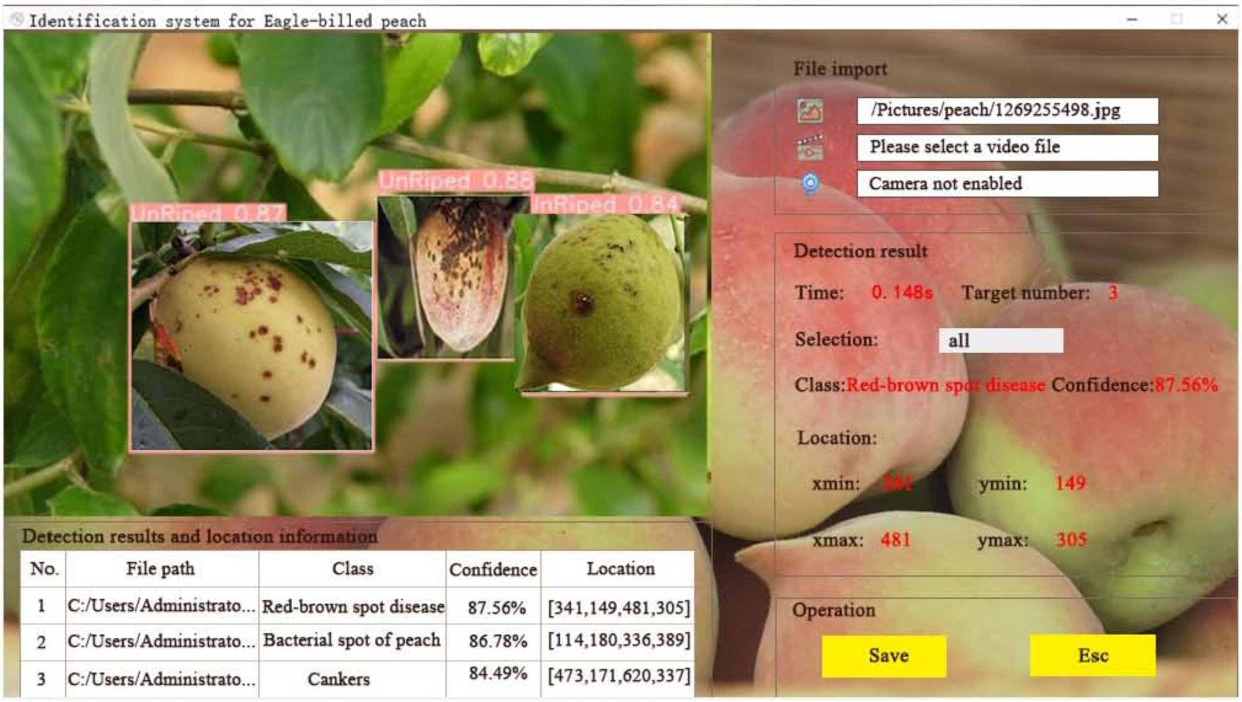
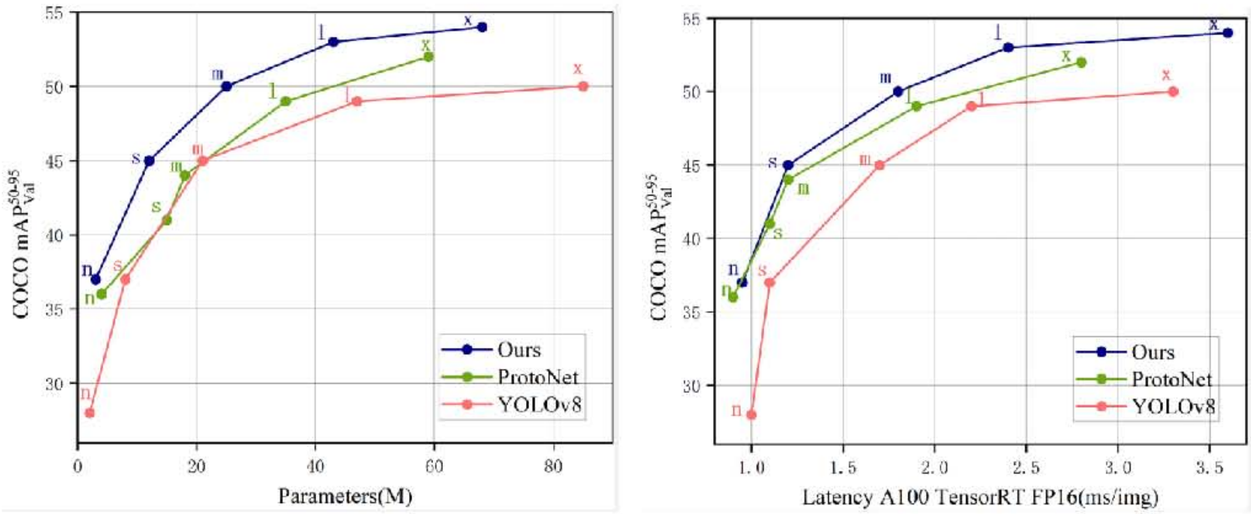



Fig. 9. Example of the real-time detection interface for the Eagle-billed peach




(a) Performance curve comparisons of parameter count

(b) Performance curve comparisons of latency results

Fig. 10. Performance comparison curve between improved ProtoNet and other frameworks

## VII. Conclusions

The trained model was saved for prediction purposes, and the experimental results demonstrated strong performance on the test dataset with an identification rate of 98.4%. The model accurately predicted the outcomes in test experiments using images of peach pests and diseases from other regions, effectively classifying 14 categories of pests and diseases, thereby demonstrating outstanding generalization ability.

The experimental results confirmed that FSL combined with deep transfer learning can achieve excellent recognition accuracy for identifying pests and diseases in peach trees. This approach accelerates network training and convergence, further improving defect detection rates while reducing errors and missed detections. It aids in recognizing the occurrence of pests and diseases in orchards, rapidly identifying affected areas and their severity, providing reasonable solutions based on insights from intelligent big data analysis systems. This study presents new ideas for digital and intelligent management of agricultural production.

Compared to traditional detection methods, DL has shown significant results in the field of plant pest and disease identification, with deep residual networks extracting more discriminative and comprehensive feature information.

Transfer learning can address the challenge of data collection, thereby significantly enhancing the generalization capability of the model. In summary, deep learning offers robust solutions for pest and disease detection with a broad application potential that can be extended to various types of crop pests, disease recognition, and early warning systems. This supports the development of unique rural industries and provides a practical demonstration of smart agricultural production.

ACKNOWLEDGMENT

The authors thank all the survey participants. In particular, our appreciation goes to the Shangping eagle-billed peach industry base in the Heyuan-Lianping area for selfless assistance and guidance.

REFERENCES

[1] Nuthalapati, V, and Tunga A, "Multi-domain few-shot learning and dataset for agricultural applications," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2021, 19-25 June, 2021, Kuala Lumpur, Malaysia, pp1399-1408.
[2] Gidaris S, Komodakis N, Paristech P, and Komodakis N, "Dynamic few-shot visual learning without forgetting," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2018, 18-22 June, 2018, Salt Lake, USA, pp4367-4375.
[3] Lake B, Salakhutdinov R, and Gross J, "One shot learning of simple visual concepts," Proceedings of the 33rd Annual Meeting of the Cognitive Science Society, 20-23 July, 2011, Boston, USA, pp2568-2573.
[4] Jake Snell, Kevin Swersky, and Richard Zemel, "Prototypical networks for few-shot learning," Proceedings of the 30th Neural Information Processing Systems Conference and Workshop (NIPS) 2017, 4-9 December, 2017, California, USA, pp175-187.
[5] Deng J, Dong W, and Socher R, "ImageNet: a Large-Scale Hierarchical Image Database," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009, 20-25 June, 2009, Florida, USA, pp520-532.
[6] Konig J, Jenkins M, Mannion M "What's cracking? A Review and Analysis of Deep Learning Methods for Structural Crack Segmentation, Detection and Quantification. Arxiv Preprint Arxiv, vol. 22, no.2, pp1-18, 2022.
[7] Lin T Y, Maire M, and Belongie S, "Microsoft coco: common objects in context," Proceedings of the 13th European Conference on Computer Vision (ECCV) 2014, 6-12 September, 2014, Zurich, Switzerland, pp740-755.
[8] Everingham M, Van Gool L, and Williams C, " Unk Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision, vol. 88, no.2, pp303-338, 2010.
[9] Fu H, Meng D, Li W, "Bridge crack semantic segmentation based on improved deeplabv3+," Journal of Marine Science and Engineering, vol. 9, no.6, pp671-684, 2021.
[10] Hsieh Y, et al.: "Machine learning for crack detection: review and model performance comparison," Journal of Computing in Civil Engineering, vol. 34, no.5, pp1943-5487, 2020.
[11] Liang Yong, Qiu Rongzhou, and Li Zhipeng, "Identification Method of Major Rice Pests Based on YOLO v5 and Multi-source Datasets," Transactions of the Chinese Society for Agricultural Machinery, vol. 53, no.7, pp250-258, 2022.
[12] Wang CX, Qi X, Ma G, Zhu L, Wang BX, and Ma CS, "Intelligent identification system of grape diseases based on YOLO V3," Plant Protection, vol. 48, no.6, pp278-288, 2022.
[13] Sain, R. Stephan, "The Nature of Statistical Learning Theory," Technometrics, Vol. 38, no.4, pp409-420, 1999.
[14] Lake, Brenden M, Salakhutdinov, R., Tenenbaum, J., "The Omniglot challenge: a 3-year progress report," Current Opinion in Behavioral Sciences, vol. 29, no.4, pp97-104, 2019.
[15] Bertinetto L, Torr S, Henriques J, and Vedaldi A, "Meta-learning with differentiable closed-form solvers," Proceedings of the 7th International Conference on Learning Representations, (ICLR) 2019, 6-9 May, 2019, Louisiana, USA, pp1-15.
[16] Chen Z, Fu Y, Chen K, and Jiang Y, "Image block augmentation for one-shot learning," Proceedings of the 33rd International Conference on Artificial Intelligence (CVPR) 2019, 27-31 January, 2019, Hawaii, USA, pp3379-3386.
[17] Chen Z, Fu Y, Wang X, Ma L, Liu W, and Hebert M, "Image deformation meta-networks for one-shot learning," Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2019, 15-21 June, 2019, Los Angeles, USA, pp8672-8681.
[18] Fu H, Meng D, Li W, "Bridge crack semantic segmentation based on improved deeplabv3+," Journal of Marine Science and Engineering, vol. 9, no.6, pp671-683, 2021.
[19] Rusu, and Andrei A, "Meta-learning with latent embedding optimization," Proceedings of the 7th International Conference on Learning Representations, (ICLR) 2019, 6-9 May, 2019, Louisiana, USA, pp1-17.
[20] Ravi S, and Larochelle H, "Optimization as a model for few-shot learning," Proceedings of the 5th International Conference on Learning Representations, (ICLR) 2017, 24-26 April, 2017, Toulon, France, pp1-11.
[21] Chen Y, Liu Z, and Xu H, "Meta-baseline: exploring simple meta-learning for few-shot learning," Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2021, 19-25 June, 2021, Kuala Lumpur, Malaysia, pp9042-9051.
[22] NIE Ganggang, RAO Honghui, and KANG Lichun, "Research on Camellia oleifera disease recognition based on lightweight convolutional neural network," Acta Agriculturae Universitatis Jiangxiensis, vol. 2, no.11, pp502-515, 2024.
[23] Lin Jianwu, Zhang Xin, and Chen Xiaoyulong et al, "Tomato Disease Image Recognition Based on Lightweight Convolutional Neural Network," Radio Engineering, vol. 52, no.8, pp1347-1353, 2022.
[24] K. Zhang, Y. Guo, X. Wang, J. Yuan, Q. Ding, "Multiple feature reweight densenet for image classification," IEEE Access, Vol. 7, no.11, pp9872-9880, 2019.
[25] Bah M D, Hafiane A, Canals R, "Deep learning with unsupervised data labeling for weed detection in line crops in UAV images," Remote Sensing, vol. 10, no.11, pp1690-1704, 2018.
[26] Wang B, and Wang D, "Plant leaves classification: a few-shot learning method based on Siamese network," IEEE Access, vol. 7, no.151, pp754-763, 2019.
[27] Hu Gensheng, Wu Haoyu, Zhang Yan, and Wan Mingzhu, "A low shot learning method for tea leaf's disease identification," Computers and Electronics in Agriculture, vol. 163, no.104, pp852-863, 2019.
[28] Sardogn M, Tuncer A, Ozen Y, "Plant leaf disease detection and classification based on CNN with LVQ algorithm," Proceedings of the 3rd International Conference on Computer Science and Engineering (UBMK) 2018, 20-23 September, 2018, Sarajevo, Bosnia, and Herzegovina, pp382-385.
[29] Miaohui Zhang, Yunzhong Chen, Bo Zhang, Kangning Pang, and Bingxue Lv, "Recognition of Pest Based on Faster RCNN," Lecture Notes in Engineering and Computer Science: Proceedings of the 6th International Conference on Signal and Information Processing, Networking and Computers (ICSINC) 2020, 13-16 August, 2020, Guiyang, China, pp62-69.
[30] V Senthil Kumar, M Jaganathan, A Viswanathan, M Umamaheswari, and J Vignesh, "Rice leaf disease detection based on bidirectional feature attention pyramid network with YOLO v5 model," Environmental Research Communications, vol. 5, no.6, pp14-32, 2023.
[31] Zhang, Shanwen, and C. Zhang, "Modified U-Net for plant diseased leaf image segmentation," Computers & Electronics in Agriculture, vol. 204, pp107511, 2023.
[32] Kerkech, Mohamed, A. Hafiane, and R. Canals, "Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation approach," Computers and Electronics in Agriculture, vol. 174, pp105446, 2020.
[33] Afzaal, U., Bhattarai, B., Pandeya, Y. R., & Lee, J., "An instance segmentation model for strawberry diseases based on mask R-CNN," Sensors, vol. 21, no.19, pp6565-6584, 2021.
[34] GAO Jiajun, ZHANG Xu, GUO Ying, LIU Yukun, GUO Anqi, SHI Mengmeng, WANG Peng, YUAN Ying, "Research on the optimized pest image instance segmentation method based on the Swin Transformer model," Journal of Nanjing Forestry University (Natural Science Edition), 47(3): 1-10, 2023.
[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016, 26-30 June, Las Vegas, USA, pp770-778, 2016.
[36] Sinno Jialin Pan, and Qiang Yang, "A Survey on Transfer Learning," IEEE Transactions on Knowledge and Data Engineering, vol. 22, no.10, pp1345-1359, 2010.

**Yongfu Zhou** was born in 1979. He received his B.E. degree in Computer Science from Jiangxi Normal University in 2001 and M.E. degree in Computer Science from Huazhong University of Science and Technology in 2005. His research interests include AI with a focus on intelligent detection in agriculture. In recent years, he has published more than 10 papers.

**Zhi Zeng** was born in 1971 and is currently a Professor at the College of Computer Science and Engineering, Huizhou University, Guangdong, China. He received his B.E. in Computer Science from Jiangxi Normal University in 1995, M.E. in Computer Science from Zhejiang University in 2005, and Ph.D. in GIS from Zhejiang University in China in 2012. His research interests include AI, GIS, and SLAM, with a focus on visual tracking and navigation. In recent years, he has published over 30 papers.

**Yuwan Gan** was born in 1996. She received her B.S. degree in Internet of Things engineering from Harbin University of Science and Technology in 2018 and an M.E. degree in Electronics and Communications Engineering from Guilin University of Electronic Technology in 2021. His research interests include machine learning and intelligent image processing.