IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 20, 2021, accepted April 13, 2021, date of publication April 19, 2021, date of current version May 10, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3073929

A Remote Sensing and Airborne Edge-Computing Based Detection System for Pine Wilt Disease

FENGDI LI[1], ZHENYU LIU[2], WEIXING SHEN[3], YAN WANG[1], YUNLU WANG[1], CHENGKAI GE[1], FENGGANG SUN[1], AND PENG LAN[1]

[1]College of Information Science and Engineering, Shandong Agricultural University, Tai'an 271018, China
[2]College of Plant Protection, Shandong Agricultural University, Tai'an 271018, China
[3]Taishan Forest Pest Management and Quarantine Station, Tai'an 271018, China

Corresponding authors: Peng Lan (lanpeng@sdau.edu.cn) and Fenggang Sun (sunfg@sdau.edu.cn)

This work was supported in part by the Shandong Agricultural Science and Technology Fund (Forestry, Science, and Technology Innovation) under Grant 2019LY003, in part by the Shandong Provincial Key Research and Development Program of China under Grant 2019GNC106106, in part by the Major Scientific and Technological Innovation Project of Shandong Province of China under Grant 2019JZZY010706, and in part by the Shandong Provincial Natural Science Foundation of China under Grant ZR2019MF026.

ABSTRACT The pine wilt disease (PWD) is one of the most dangerous and destructive diseases to coniferous forests. The rapid spread trend and strong destruction directly threaten the security of forests. The complex spread pattern and the hard labor process of diagnosis call for an effective way to detect the infected areas. In this paper, an airborne edge-computing and lightweight deep learning based system are designed for PWD detection by using imagery sensors. Unmanned aerial vehicle (UAV) is firstly utilized to realize a large-scale coverage of forests, which can substantially reduce the hard labor. Except for infected trees, a large number of irrelevant images are also acquired by the UAV, which will overload the burden of process and transmission. Then a lightweight improved YOLOv4-Tiny based method (named as YOLOv4-Tiny-3Layers) is proposed to filter these uninterested images by leveraging the computation capability of edge computing, which can realize a fast coarse-grained detection with a low missing rate. Finally, all the remaining images are transmitted to the ground workstation for the final fine-grained detection. Experimental results show that the proposed system can implement a fast detection with superior performance as compared to other methods, which helps to detect the infected pine trees in a quick manner.

INDEX TERMS Pine wilt disease, remote sensing, airborne edge computing, lightweight deep learning, two-stage detection.

I. INTRODUCTION

Pine wilt disease (PWD), caused, is one of the most dangerous and destructive diseases to coniferous forests. Once infected with the PWD, the pine trees are usually caused to severe tree wilt and ultimately die, and the entire pine forests will die within a short time if not controlled effectively. Native to North America, PWD has already spread into East Asia (Japan, China and Korea) and thereafter to Europe (Portugal and Spain) [1], where the PWD has caused massive deaths of pine trees, leading to severe economic and ecological losses. In China, the PWD in 2019 has spread to 672 counties of 18 provinces (autonomous regions, municipalities) [2]. More seriously, the PWD has a tendency to break through the previous suitable geo-environment range, and spread northward and toward high altitudes. The epidemic has expanded with a rapid growth trend, about an average annual increase of 22%, and the diseased area in 2017 reached 85,024 hectares in China [3]. The rapid spread trend and strong destruction directly threatens the security of pine forests and key ecological areas worldwide.

To detect the PWD infected trees in time is the key to control and prevent the spread of PWD. The coniferous leaves of the infected pine trees become yellowish brown or reddish brown, wilting and drooping, which is useful to identify the infected pine trees. The traditional field survey approach relies on the unaided eyes of investigators to identify, which usually consumes a large amount of labor cost but with low efficiency, especially for complex terrains. Remote sensing (RS) is regarded as an effective alternative due to the long-distance, large-scale and indirect contact characteristics, and

The associate editor coordinating the review of this manuscript and approving it for publication was Sudipta Roy.

has been widely used in forest monitoring [4]. In 2010, Dennison *et al.* uses the GeoEye-l high-resolution images to distinguish between the diseased and healthy pine trees in black pine forest areas [5]. In 2013, Johnson *et al.* applies a multiscale image classification method to discriminate discolored and healthy trees from the high-resolution satellite images [6]. Kim *et al.* uses the multi-temporal hyperspectral 1 m spatial resolution aerial data to demonstrate the change in spectral reflectance at infrared and mid-infrared wavelengths [7]. Despite the advantages in large scale monitoring, the satellite remote sensing is vulnerable to weather, temporal resolution and other factors. Therefore, it is difficult to obtain real-time and high-resolution images, and the timeliness of the observation data cannot be guaranteed. In addition, it is difficult to obtain a large number of samples of specific objects in remote sensing field, which weakens the ability of object features extraction for deep learning network. Introducing unmanned aerial vehicle (UAV) as a surveillance platform will overcome these deficiencies as one UAV can provide more flexible detection with higher spatial-temporal resolution, which has become an effective monitoring approach. By using the airborne spectral imagery, machine learning based methods are proposed in [8]–[11]to detect the PWD. Zhang *et al.* utilizes a UAV-based hyperspectral image to identify the degree of damage caused by Dendrolimus tabulaeformis Tsai et Liu (D.tabulaeformis) in the pine forest environment [12]. Since the deep learning can reveal the nonlinear features hidden in the data through multi-layer processing mechanism, and can obtain ''feature learning'' from a large number of training data sets, the deep learning based detection models have proven to be an effective detection approach. For example in [13]–[16], a UAV and deep learning based detection model was proposed to identify the dead pine trees infected with PWD, where the airborne GNSS (Global Navigation Satellite System) was used to locate the dead pine trees. In [17], Zhang *et al.* use a fixed-wing UAV to collect images of the study area and propose a deep learning network, the U-Net model, to segment the images of wilted pine trees. These results showed that the combination of UAV and deep learning method is feasible to detect dead trees of the PWD infected pine trees. However, the deep learning network topology tend to become deeper and more complex. Despite of the performance improvement, they usually require a large amount of computation power and involve ultra-high energy consumption for high performance. Limited by processing and storage resources, these complex deep learning models cannot be directly applied in UAVs. Although all these collected images can be transferred to the cloud data center (or ground workstation) for further process, which can provide substantial computational resources to process and detect the PWD infected tree. Sending all these images to the cloud data center (or ground workstation) may increase the end-to-end delay and consume more bandwidth, making it as a bottleneck for transmission. In fact, as the collected images usually include a lot of redundant frames which does not contain any infected (or suspected) pine trees, transmitting all these images to the ground workstation is meaningless but leads to severe end-to-end delay and bandwidth consumption. Therefore, how to filter the uninterested frames in a quick manner is of great importance to alleviate the following transmission and process burden.

The edge computing technique provides certain computing capability to the onboard UAV, which can be utilized to filter the uninterested frames by preprocessing and thus reduce the network traffic and end-to-end delay as a result. To realize a fast detection for edge computing, lightweight deep learning based object detection methods are required. In general, lightweight object detection method refers to the method with simplified network structure, which is mainly applicable to mobile devices, embedded devices and other devices with limited computing capabilities. Inherited from the classical deep learning models of YOLO (You Only Look Once) [18] series and SSD (Single Shot MultiBox Detector) [19] series, the recently popular lightweight models mainly include YOLO-Tiny, and MoblieNetv2-SSD [20]model. These models and their improved versions have been widely used in several fields. For example, the MoblieNetv2-SSD model has successfully used for signal light detection in unmanned vehicles [21], and intelligent diagnosis for gallstone disease [22]. An improved YOLOv3-Tiny model is used to detect kiwi fruit [23]. Han *et al.* modified the YOVOV4-Tiny model to perform detection more quickly and more accurately on devices with limited computational capabilities [24]. To the best of our knowledge, there are still few works about the application of edge computing in forest protection, which is critical to detect the PWD infected trees in time.

In this paper, we design an edge computing based detection system with a low-cost UAV to detect infected (or suspected) pine trees using limited resources onboard. Firstly, the images of the pine trees in the forest are acquired by airborne high resolution cameras of UAV, where images may contain the healthy trees, infected trees, rocks, weeds and many others. This implies that a large number of the acquired images are redundant that does not contain the PWD infected trees and will lead to tremendous computation burden and storage pressure. Thus, a real-time detection is required to identify whether an image contains the infected (or suspected) pine trees or not as quickly as possible. This real-time detection can be realized by the edge computing module. Limited by processing and storage resources, a lightweight and improved YOLOv4-Tiny based detection method (named as YOLOv4-Tiny-3Layers) is proposed to realize a coarse-grained but real-time detection with limited resources. Finally, the filtered images are transferred to the ground workstation to realize a fine-grained detection by deep learning models. In general, the main contributions of this paper are summarized as follows:

- An airborne edge-computing based detection system is designed for PWD infected trees detection, where the UAV is utilized to monitor the large-scale forest and the edge computing module is to filter the uninterested and redundant images.

- A lightweight deep learning based detection model, YOLOv4-Tiny-3Layers, is proposed to comply with the resources-limited edge-computing module. The proposed model can realize a coarse-grained but real-time detection, which helps to distinguish the PWD infected pine trees from redundant images in a quick manner, and thus can alleviate the transmission and process burden.
- A two-stage detection procedure is performed by edge computing module (coarse-grained) and ground workstation (fine-grained), respectively. This two-stage procedure can substantially reduce the number of images needing to be transmitted after the first stage, and finally realize a fast detection, which is critical for the early detection and diagnosis of the PWD infected pine trees.

The rest of the paper is organized as follows. Section II presents the study area and introduces the airborne edge-computing detection method, where the proposed YOLOv4-Tiny-3Layers model is introduced in detail. Section III gives the experimental results for performance comparison. Section IV discusses the work of this paper and finally Section V concludes this paper.

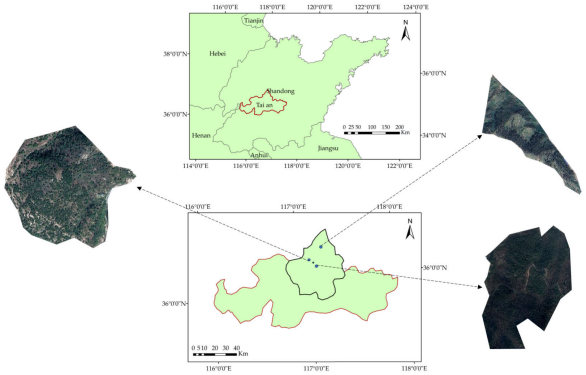

FIGURE 1. The location of the study area in Tai'an City, eastern China, with three test sites and their RGB representations.

II. MATERIALS AND METHODS

A. DATA PREPARATION

The study area is located near the Taishan mountains in Tai'an City, Shandong Province, China. The location and geomorphic characteristics of the forest areas are shown in Figure 1, where three test sites with different tree species composition and topographical features, are selected to acquire the images by UAV. PWD was confirmed to be present in this area during the general investigation in recent years, and began to spread since then. Despite efforts have been made, the disease in this area has been controlled at this stage, but there are still scattered PWD infected pine trees. As one of China's mountain parks and a natural museum of history and art, the spread of the PWD will threaten the safety of Taishan Mountain. Therefore, it is urgent to control the spread of the PWD. The data are collected between the years 2019 and 2020. The diversity of the data from three different regions and at different time can enrich the dataset and enhance the robustness of the proposed method.

B. THE AIRBORNE EDGE-COMPUTING BASED DETECTION SYSTEM

In previous works, UAVs are utilized to acquire imagery over the test sites, and then these images are transferred to the ground workstation or cloud center to identify the infected trees. However, since the images containing infected (or suspected) pine trees only cover a small proportion of the acquired images, most images are redundant for detection. If all these images are sent back for process, the transmission and processing procedures will consume a lot of time, at the cost of high communication and computing load. To alleviate the transmission and processing load, we design an airborne edge-computing based detection method in this section.

1) SYSTEM MODEL

The proposed airborne edge-computing based detection system consists of three modules, i.e., UAV-based image acquisition module, edge computing module and ground workstation module, as is shown in Figure 2.

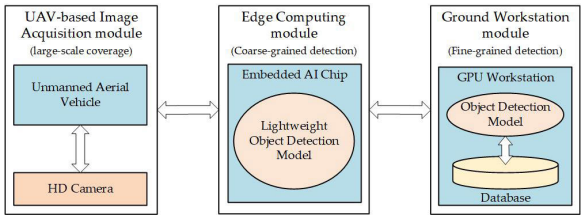

FIGURE 2. System model of the airborne edge-computing based detection system.

The UAV-based image acquisition module is composed of a low-cost UAV and airborne high resolution RGB camera. By utilizing the advantages of high flexibility, high mobility and platform scalability of UAVs, the module can acquire the images of the detected large-scale forest area. However, since a large-scale area in the forest needs to be monitored and the PWD infected trees only exist in some specific regions, a large number of irrelevant images are captured and need to be stored and transmitted, which brings a great challenge for further image transmission and processing. To alleviate the burden, a two-stage detection (a coarse-grained and fine-grained) detection procedure is performed by the following two modules in our system.

The first-stage detection is implemented by the edge computing module. This module is composed of the embedded NVIDIA Jetson TX2 chip and a lightweight detection model, which is responsible for a coarse-grained detection with less processing time and lower missing detection probability. The NVIDIA Jetson TX2 can achieve about 23 frames per second, which is 10 times faster than that of Raspberry Pi, and can satisfy the real-time processing requirements. The aim of this module is to identify the images containing the infected (or suspected) trees, and only transmit these images to the

ground workstation for a more fine-grinned detection, which can alleviate the burden of transmission and processing. The second-stage detection is performed by the ground workstation module, which is composed of image processing workstation, high precision deep learning based detection model and in-memory databases. After the first-stage coarse-grained detection, the suspected images are then transmitted to this module through the remote control transmission relay, where the second-stage fine-grained detection is performed to locate the infected trees. Finally, the detection results containing the location information are stored in the database for further access.

2) SYSTEM HARDWARE DESIGN

The hardware architecture of the airborne edge-computing based detection system is shown in Figure 3. For clarity, the specific hardware configurations of UAV, edge computing module and ground workstation module are given in Table 1.

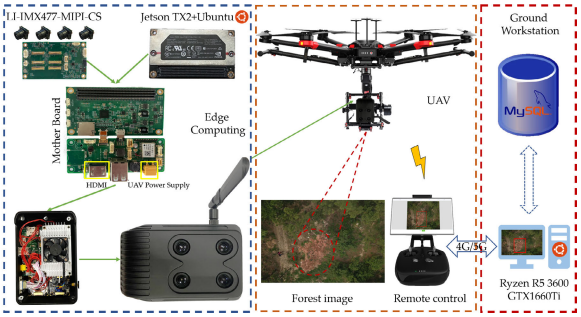

FIGURE 3. The hardware architecture of the airborne edge-computing based detection system.

TABLE 1. The computer hardware configuration of the UAV, edge computing module, and ground workstation module.

Platform	Parameter	Description
UAV	Type	DJI M600 Pro
	Max Takeoff Weight /g	15500
	Dimensions /mm	1668ÃŮ1518ÃŮ727
	Maximum Flight Altitude /m	500
	Max Speed /(kmÂůh^{-1})	65
	Hovering Time /min	32
Edge Computing	Type	NVIDIA Jetson TX2
	CPU	ARM Cortex-A57
	RAM	8 GB LPDDR4 Memory
	GPU	NVIDIA Pascal GPU
	System Environment	Ubuntu 18.04
	Camera Type	LI-IMX477-MIPI-CS
	Focal Length /mm	5
	Maximum Ground Sample Distance /cm	16.2
Ground Workstation	CPU	AMD Ryzen R5 3600
	RAM	16 GB
	GPU	NVIDIA GTX 1660Ti
	System Environment	Ubuntu 18.04 LTS
	Database	MySQL

For the UAV based image acquisition module, DJI M600Pro is chosen as the UAV platform by considering flight time, flight height and stability, etc. DJI M600Pro is equipped with an A3 flight control system, three sets of IMU (Inertial Measurement Unit), GNSS modules and Lightbridge 2 image transmission equipment, so the UAV has a strong flight control stability and simple control. The image transmission supports a maximum transmission distance of 1.7 km. The UAV has a maximum flight time of 32 minutes, a maximum altitude of 500 meters and can be observed and sent back images in real time via a multimedia remote control, as well as a wide range of mount interfaces for compatibility with a variety of devices.

The edge-computing module is then to further process the images acquired by UAV and provide a coarse-grained detection to reduce the redundant images. The low-cost Raspberry development board can only provide a frame rate of around 2 FPS (frame per second), which is too slow to detect the high-resolution aerial images. In this paper, the NVIDIA Jetson TX2 solution is used as the core of the edge computing module, which can reach a detection frame rate of around 23 FPS and can process the collected images in real time. To reduce the hardware complexity, the motherboard is designed to be powered directly by the drone, which includes four programmable camera interfaces that allow the Jetson TX2 computing core to be integrated with four LI-IMX477-MIPI-CS high-resolution imagery sensors to increase the light-sensitive area and ensure image quality. In addition, the HDMI(High Definition Multimedia Interface) port on the motherboard can be connected to the M600Pro's Lightridge2 image transfer system, making it have the real-time picture transfer capability.

For the fine-grained detection, the ground workstation module requires a high image processing and data exchange capability. Therefore, the NVIDIA GTX 1660Ti high-performance graphics computing unit and AMD Ryzen R5 3600 central processor unit are chosen for the hardware; the operating system is Ubuntu 18.04 LTS and the database is the open-source MySQL database system, which provides the module with high-performance image processing and storage capabilities.

3) SYSTEM WORKFLOW

The workflow of the detection system for the PWD infected pine tree is shown in Figure 4. The UAV is utilized to monitor the large-scale forest and acquire the images, which also acquires abundant irrelevant images. These irrelevant images may overload the following transmission and process procedure. To filter these uninterested images, the acquired images are fed into the airborne edge computing module for a coarse-grained detection, where the detection results are obtained in a quick manner by a lightweight detection model. If there exist the (suspected) PWD infected tree in one image by the module, it will be automatically sent to the ground workstation for a fine-grained detection. Otherwise, the image that does not contain (suspected) PWD infected tree will be discarded. Since the suspected images only cover a small proportion of the acquired images and most images are redundant, the coarse-grained detection by edge computing module can substantially reduce the number of images needing to be transmitted. At the ground workstation, the

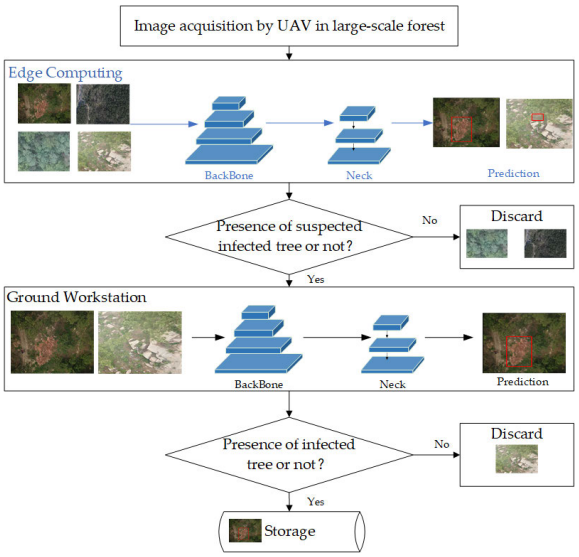

FIGURE 4. The workflow of the detection system.

suspected images are further detected by the deep learning model with a fine-grain to confirm whether it contains the infected tree or not. If the PWD infected tree is confirmed in one image by the deep learning model, the image and the corresponding location information are stored in the relational database. Otherwise, it will be discarded.

C. PROPOSED LIGHTWEIGHT DETECTION METHOD

YOLO is one of the deep learning based real time object detection frameworks, which takes both the accuracy and speed into consideration. Since the release of YOLOv3 in 2018 [25], it has been widely used in various areas. The network architecture of YOLOv3 consists of three parts: DarkNet-53 feature extraction network, feature fusion network and multiple-scale output networks (classification and location). DarkNet-53 is used for feature extraction, and feature fusion is to generate three scale feature maps. Output networks are carried out based on the feature fusion maps of three scales to realize the object detection with different sizes. Furthermore, YOLOv4 [26] is proposed by adopting mosaic data augmentation, CSPDarkNet-53, CIoU (Complete Intersection over Union) Loss function, DIoU (Distance Intersection over Union)-NMS(non maximum suppression) [27] and others, which can finally improve the detection performance in terms of accuracy and speed. However, limited by the computing capacity of the edge-computing Jetson TX2 chip, YOLOv4 cannot be directly applied. To alleviate the difficulty, a lightweight YOLOv4-Tiny method is proposed [28]. Preserving the advantages of data expansion and feature fusion network of YOLOv4, YOLOv4-Tiny simplifies the network architecture, and only utilizes a two-layer output network. As compared to YOLOv4, the network parameters of YOLOv4-Tiny are only about 10 times less [29]. Despite the advantages in detection speed, YOLOv4-Tiny cannot efficiently solve the following problems, such as multiple-scale object detection, and missing detection of the overlapped objects, which are usually the cases for infected pine tree detection. To this end, an improved YOLOv4-Tiny based lightweight detection method is required.

1) PROPOSED YOLOv4-TINY-3Layers DETECTION METHOD

To improve the detection performance, an improved YOLOv4-Tiny based detection method (named as YOLOv4-Tiny-3Layers) is proposed. The network architecture of YOLOv4-Tiny-3Layers is shown in Figure 5. As compared to YOLOv4-Tiny, the main improvements of YOLOv4-Tiny-3Layers are:

1) To adapt to the multiple-scale object detection scenarios, a third output network is added, which is used to detect the infected trees with different sizes.
2) CIoU bounding loss is adopted to accelerate the training procedure and improve the detection accuracy.
3) DIoU-NMS method is adopted to avoid NMS [30] error suppression and reduce the occurrence of missed detection when the IoU values of two different boxes are relatively large.

For clarity, the detailed improvements are given as follows.

1) A 3-layer output network architecture is adopted for multiple scale detection.

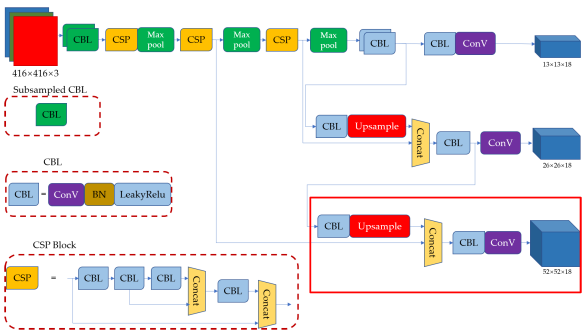

FIGURE 5. The network architecture of the YOLOv4-Tiny-3Layers model.

Affected by variable flight heights of UAVs, growth difference of pine trees, and other factors, the crown size of the PWD infected trees is different in the collected images, which requires that the detection method should adapt to the multiple-scale scenarios. To this end, a three network output layers architecture is used to enhance the ability of three-scale object detection and reduce the rate of missing detection in YOLOv4-Tiny-3Layers. The three layers output network architecture is shown in Figure 5. As compared to YOLOv4-Tiny, one another output network with the size $52 \times 52 \times 18$ is added in the YOLOv4-Tiny-3Layers model for small scale object detection. E.g., when the size of the input image is 416×416, three different output networks with different sizes are generated after convolution, upsampling and subsampling of feature extraction network, i.e., $13 \times 13 \times 18$, $26 \times 26 \times 18$, $52 \times 52 \times 18$. Among which, the $13 \times 13 \times 18$ output network is suitable for large scale object detection, the $26 \times 26 \times 18$ output network is suitable for medium scale object detection,

and the 52 × 52 × 18 output network is suitable for small scale object detection. Therefore, the proposed YOLOv4-Tiny-3Layers can realize the multiple scales detection, which is critical in the PWD infected pine trees detection.

In the proposed YOLOv4-Tiny-3Layers architecture, we set three anchor boxes [31] with different sizes for each output network. For each ground truth in the training images, if its center point falls in a certain cell, the corresponding three anchor boxes of cell are responsible for prediction. After the three-scale detection, a total of $[52\times52+26\times26+13\times13]\times3 = 10647$ prediction boxes are generated, and then IoU is calculated. By specifying the IoU threshold and performing NMS operation, most of the redundant anchor frames can be removed, as shown in Figure 6. The center of infected pine tree is the area bounded by the yellow box, and its center point falls in the red grid. Among the three corresponding anchor frames, the IoU of the blue box and the real target frame is the largest. Finally, the bounding box loss, confidence loss and classification loss between the ground truth box and the anchor box are calculated by the loss function.

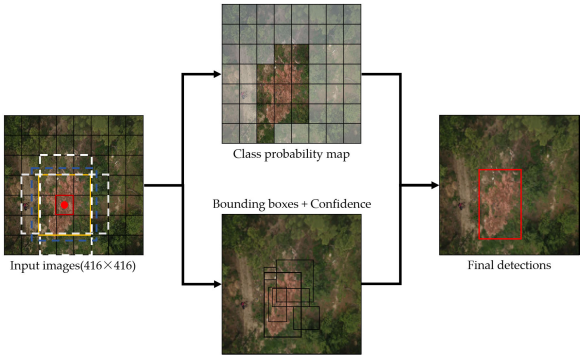

FIGURE 6. The forecasting process of YOLOv4-Tiny.

2) CIoU bounding box regression function loss is adopted to accelerate the training procedure and improve the detection accuracy.

For YOLO based models, the overall objective loss function is usually the sum of the bounding box loss, confidence loss and classification loss. The bounding box regression loss function is utilized to measure the difference between the prediction box and ground truth. In terms of evaluation metric for bounding box regression, IoU is the most popular metric, which is defined as

$$IoU = \frac{Intersection(A, B)}{Union(A, B)}, \tag{1}$$

which is the ratio of the intersection of A and B to the union of A and B. For the YOLOv4-Tiny model, the IoU loss [32] is adopted as the bounding box loss, which is defined as

$$L_{IoU} = -\ln\frac{Intersection(A, B)}{Union(A, B)}, \tag{2}$$

where L_{IoU} is the IoU loss of the ground truth A and the prediction bounding box B. Generally, the IoU loss often suffers from a lower decrease rate during model training. Especially when the ground truth A and the prediction box B have no intersection, the value of IoU equals 0, and thus the IoU loss does not exist, which cannot reflect the distance between A and B. In this case, the IoU loss function cannot be differentiable, and cannot optimize the case where the two boxes have no intersection. Moreover, even when the IoU of two boxes are the same, the IoU loss cannot reflect the form in which the two boxes intersect. To accelerate the training procedure and improve the detection accuracy, the proposed YOLOv4-Tiny-3Layers adopt the CIoU as the bounding box loss, defined as

$$L_{CIoU} = L_{IoU} + \frac{d}{g^2} + \alpha v. \tag{3}$$

As is shown in Figure 7, $d = \rho(A, B)$ denotes the Euclidean distance between the center points of the prediction box and the ground truth box. g is the diagonal length of the smallest enclosing box covering two boxes, $v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2$ denotes the consistency of the ratio of width and height between the detection frame and the real frame, $\alpha = \frac{v}{(1-IoU)+v}$ is a positive trade-off parameter. By utilizing the CIoU loss, the improved model is more inclined to optimize to the direction with dense overlapping regions, which is experimentally found to reduce the model training time and improve the model detection accuracy.

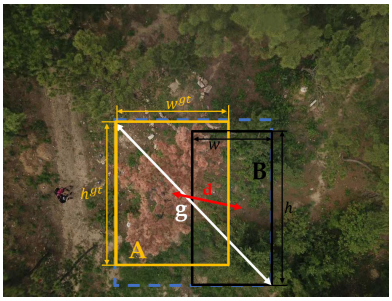

FIGURE 7. Schematic diagram of loss function.

3) DIoU-NMS method is adopted to avoid NMS error suppression and reduce the occurrence of missed detection.

For previous object detection methods, NMS is used to perform the post-processing work [33], and the traditional NMS is defined as:

$$S_i = \begin{cases} S_i, & IoU\ (\mathcal{M}, B_i) < \varepsilon \\ 0, & IoU\ (\mathcal{M}, B_i) \geq \varepsilon, \end{cases} \tag{4}$$

where B_i is the i-th prediction box, S_i is the corresponding classification score of B_i, $\mathcal{M}$ is the prediction box with the highest score, and ε is the threshold value of NMS. Usually, the bounding boxes are sorted according to their scores, and the box with the highest score is retained by the NMS method. Meanwhile the other bounding boxes that their IoU values are greater than a certain threshold are deleted [33]. Due to the dense vegetation coverage and different heights of various trees, the infected pine trees may be covered by other trees, and the NMS method will tend to filter out these overlapped

objects due to the excessive IoU value, which is one of the main reasons for a higher missing detection rate of the infected trees. To address the issue of occlusion, YOLOv4-Tiny-3Layers adopts the DIoU-NMS method to solve the occlusion problem, where the location information of the center point of the bounding box is taken as an influencing factor to further optimize the algorithm to adapt to the actual detection application. The definition of DIoU-NMS is

$$S_i = \begin{cases} S_i, & IoU - \mathcal{R}_{DIoU}(\mathcal{M}, B_i) < \varepsilon \\ 0, & IoU - \mathcal{R}_{DIoU}(\mathcal{M}, B_i) \geq \varepsilon, \end{cases} \tag{5}$$

where $\mathcal{R}_{DIoU} = \frac{d}{g^2}$ denotes the distance of the center points of the two Boxes. The Equation (5) reveals that by comparing the distance of the prediction box $\mathcal{M}$ with the highest score and the i-th box B_i, if the difference between IoU and $\mathcal{R}_{DIoU}$ is below the certain threshold ε, the score value S_i of the i-th box B_i is maintained; otherwise, if the difference exceeds the threshold ε, the score value S_i of the i-th box B_i is set to 0, implying that the i-th box B_i is filtered out. DIoU NMS adopts the DIoU as the NMS suppressing criterion, which not only considers the overlapping area, but also the center point distance between the two boxes is considered. Therefore, it can optimize the process of suppressing redundant detection frame, and reduce the missed detection of occluded targets.

III. EXPERIMENTAL RESULTS AND ANALYSIS

A. DATASET AND MODEL TRAINING

In this experiment, we use the UAV to acquire 8860 images in the study area, among which 3670 images contain the PWD infected pine trees and the rest 5190 does not contain. For the 3670 images, 2936 images (80%) are used to form the training set and the remaining 734 images (20%) are used as the test set. Once these images are acquired by UAVs, they will be directed processed by the edge-computing module and no additional operations are performed. The LabelImg software is used to label the PWD infected pine trees to produce the Pascal VOC [34] dataset. Some examples are shown in Figure 8.

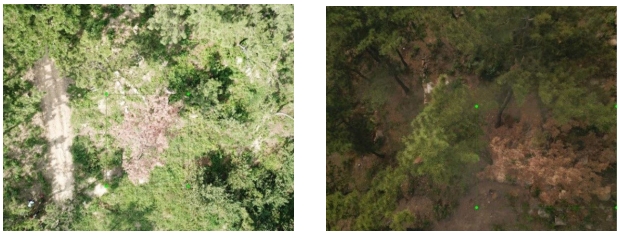

FIGURE 8. Some examples of the labeled images with the PWD infected pine trees.

In our experiments, we adopt the high performance computing platform in Shandong Agricultural University as the deep learning server for model training, where the operation system is Redhat 6.9 Enterprise, and the Darknet deep learning framework is utilized. The server is equipped with Intel Xeon E5 (128GB memory) and dual NVIDIA Tesla P100 GPU (12GB video memory). At the training stage, the size of the input images is set as 416×416, we set the parameters of training process as follows: the training steps are 50000; the step decay learning rate scheduling strategy is adopted with initial learning rate 0.0012 and multiplied with a factor 0.1 at 40000 steps and 45000 steps, respectively. The momentum and weight decay are respectively set as 0.9 and 0.0005. All architectures use GPU to execute multi-scale training in the batch size of 64 and subdivisions size of 8. In addition, we also use the Mosaic data augmentation and K-means method to optimize anchors to enhance the training process.

B. RESULTS

In this section, the performance comparisons of the proposed YOLOv4-Tiny-3Layers method with other methods are provided via experiments. The comparing methods includes MobileNetv2-YOLOv3, YOLOv3-Tiny, YOLOv3-Tiny-3Layers, and YOLOv4-Tiny. We first give the performance comparisons of different lightweight models, and then compare the detection results after the stage 1 and 2 detection procedure.

1) PERFORMANCE COMPARISONS OF DIFFERENT LIGHTWEIGHT OBJECT DETECTION MODELS

The following criteria are adopted for comparison in this part. TP (True Positive), FP (False Positive) and FN (False Negative) denote the numbers of the positive, the false positive and the false negative cases, respectively. The precision, defined as $Precision = \frac{TP}{TP+FP}$ is to evaluate the precision of model predictions. The recall, defined as $Recall = \frac{TP}{TP+FN}$ is to evaluate the ability of the model to find all positive samples. Then the average precision (AP) is adopted to measure the detection accuracy, and can be defined as the area under the precision-recall curve. The AP can be approximately calculated by the 11 point interpolation, i.e.,

$$AP = \frac{1}{11} \sum_{r \in \{0.0, \ldots\ldots, 1.0\}} AP_r, \tag{6}$$

where $r \in \{0.0, \ldots\ldots, 1.0\}$ denotes the 11 values of the recall rate, and AP_r is the maximum precision corresponding to the r-th recall rate. In general, the larger the AP is, the better the model is.

Moreover, FPS is used to measure the detect speed. Under the same hardware, FPS is mainly affected by the used models. Therefore, it can be used to measure the process speed of different models.

The loss curves of different models are given in Figure 9. As is shown, with the increase of the batch number, the loss decreases gradually, and finally tends to be stable when the number of batches is large enough. For example, when the number of batches reaches 50000, both the YOLOv4-Tiny-3Layers and MobileNetv2-YOLOv3 models are below 0.1, showing better loss performance as

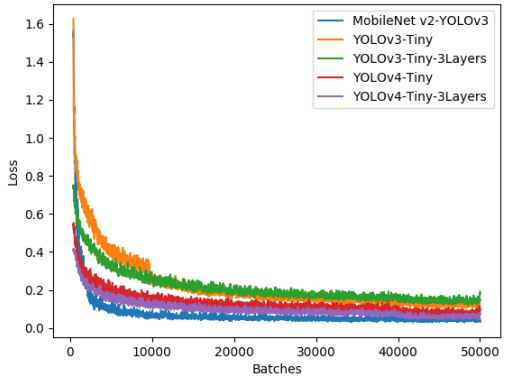

FIGURE 9. The average loss curves of different models during model training.

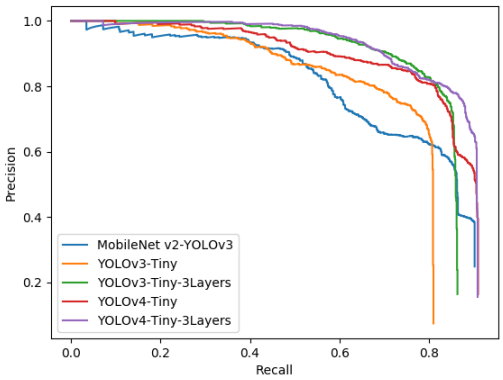

FIGURE 10. P-R curve.

compared to other models. Although the loss of the YOLOv4-Tiny-3Layers model is slightly worse than that of the MobileNetv2-YOLOv3 model, we will show later that the proposed YOLOv4-Tiny-3Layers model has a superior detection performance.

TABLE 2. The performance comparisons of different lightweight detection models.

Detection model	TP	FP	FN	AP	FPS	Model size
MobileNetv2-YOLOv3	938	1504	102	74.09%	19.5	**17.4MB**
YOLOv3-Tiny	833	603	207	72.99%	13.4	34.7MB
YOLOv3-Tiny-3Layers	892	427	148	78.56%	**11.8**	36.0MB
YOLOv4-Tiny	942	961	98	81.46%	26.2	23.5MB
YOLOv4-Tiny-3Layers	**946**	**367**	**94**	**84.88%**	24.4	24.5MB

The detection results of different lightweight detection models are shown in Table 2. As can be observed, the YOLOv4-Tiny-3Layers model shows superior performance as compared to other four models in terms of AP, TP, FP, FN. This is because that the YOLOv4-Tiny-3Layers model has been modified in several aspects, such as network architecture, loss function, and NMS. Specifically, the proposed YOLOv4-Tiny-3Layers model achieves the maximum TP (946) and AP (84.88%), the minimum FP and FN. Moreover, the FPS of the proposed model achieves 24.4, which is larger than that of MobileNetv2-YOLOv3, YOLOv3-Tiny, and YOLOv3-Tiny-3Layers, and is only slightly slower than that of YOLOv4-Tiny. Therefore, the proposed model can achieve a superior detection performance, while the detection speed can guarantee the real-time detection of the PWD.

Further, the precious-recall (P-R) curve is shown in Figure 10, which is used to synthetically evaluate model performance and generalization capability. As can be observed, the YOLOv4-Tiny-3Layers model achieves the maximum area under the P-R curve, implying that this model can guarantee a high precision while maintaining a high recall. Therefore, the proposed YOLOv4-Tiny-3Layers model can achieve the best detection performance among these methods, and can be utilized to undertake the task in the airborne edge computing platform.

C. DETECTION PERFORMANCE COMPARISONS

For the detection system, a two-stage detection procedure is executed, i.e., a coarse-grained and fine-grained detection. The former stage is performed with the lightweight object detection models, and the aim is to filter the uninterested images that do not contain infected (or suspected) pine trees as quickly as possible, which thus can reduce the transmission and processing burden. Meanwhile, the latter stage is performed by the ground workstation and to realize a fine-grained detection with high precision, which provides the final detection results.

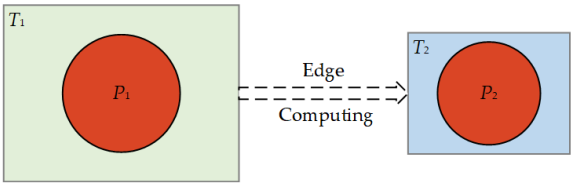

FIGURE 11. The filtering effect of the edge computing module (the coarse-grained detection).

For the coarse-grained detection, the filtering effect is one important factor since it determines the images requiring to transmitting to the ground workstation. To measure the filtering effect, we use Figure 11 to define the related metrics, where T_1 and T_2 denote the total number of images before and after the edge-computing module, P_1 and P_2 denote the number of images actually containing the PWD infected trees before and after the edge-computing module. From the definitions, $T_2 - P_2$ denote the falsely detected images that do not contain any PWD infected pine trees, and $P_1 - P_2$ denote the missing detected images that contain the PWD infected pine trees, but not identified by the edge computing module. The following metrics are used for evaluation.

1) Missing detection probability (MP): a criterion to measure the probability of the missing detected PWD infected trees by the coarse-grained detection in the edge-computing module, which is defined as the ratio between the number of missing detected PWD infected

trees and the number of all the PWD infected trees, i.e.,

$$MP = \frac{P_1 - P_2}{P_1} \times 100\%. \quad (7)$$

2) Filtering coefficient (FC): a criterion to measure the filtering effects of the edge-computing module, and can be defined as

$$FC = \frac{P_2}{T_2} \times 100\%. \quad (8)$$

From (8), there are T_1 images acquired by UAV for further process at the edge-computing module, among which P_1 images contain infected trees. By the coarse detection of the edge-computing module, T_2 images are remained and only P_2 images contain infected images.

According to the definitions of MP and FC, we can see that MP varies between 0 and 1, and is expected to be the smaller the better. When MP approaches to zero, it means that almost all the images containing infected trees can be detected by the coarse detection. However, in some extreme cases, e.g., when $T_1 = T_2, P_1 = P_2$, the MP criterion reaches its best, zero. But FC equals 1, implying that no redundant images are removed by the coarse-grained detection. Therefore, to measure the effects of edge-computing module, the two criterions MP and FC should be considered jointly.

1) DETECTION PERFORMANCE COMPARISONS OF THE LIGHTWEIGHT DETECTION MODELS (THE COARSE-GRAINED DETECTION STAGE)

In this section, we will compare the detection performance of the different lightweight object detection models in stage 1. Due to the complex background of forests when acquiring images by UAVs, a large number of redundant images are collected by UAVs. To alleviate the transmission and processing burden, a coarse-grained detection is required to filter these redundant images.

The total number of the images acquired by the UAV is 8860 to cover the study area, of which 3670 images contain the PWD infected pine trees, and 5190 images does not. To reflect the filtering effect, we expand these images by data augmentation to obtain an image set with 90,000 images that does not contain the PWD infected pine trees (denoted as non-PIS). The 734 images that contain the PWD infected pine trees are used as the image set that contain the PWD infected pine trees (denoted as PIS). We then randomly select from the two image sets and investigate the filtering effect, where the following two cases are considered.

Case 1: each group contains $T_1 = 6007$ images in the training set, among which $P_1 = 5$ images contain the PWD infected trees, and the rest 6002 images does not.

Case 2: each group contains $T_1 = 6007$ images in the training set, among which $P_1 = 50$ images contain the PWD infected trees, and the rest 5957 images are not.

These images under the two cases are selected randomly and sent to the considered lightweight models, such as MobileNetv2-YOLOv3, YOLOv3-Tiny, YOLOv3-Tiny-3Layers, YOLOv4-Tiny and YOLOv4-Tiny-3Layers for test, respectively. We perform 20 trials for each case to verify the robustness of proposed models. Moreover, since the confidence threshold is an important factor to affect the detection results, we further give the results when the confidence thresholds are set as 0.5, 0.6, and 0.75, respectively. The detection results for Case 1 are given in Figure 12, and the detection results for Case 2 are given in Figure 13. In Figures 12 and 13, T_2 denotes the number of remaining images after the coarse-grained detection in the edge computing module, P_2 denotes the number of images detected that actually contain the PWD infected pine trees. Therefore, $T_2 - P_2$ denotes the number of the falsely detected images that do not contain any PWD infected pine trees. $P_1 - P_2$ denotes the number of the missing detected images that contain the PWD infected pine trees, but not identified by the edge computing module.

TABLE 3. The average detection results of the five lightweight models during the first stage detection (coarse-grained detection) under two experiment cases ($P_1 = 5$ and $P_1 = 50$).

models		MobileNetv2-YOLOv3	YOLOv3-Tiny	YOLOv3-Tiny-3Layers	YOLOv4-Tiny	YOLOv4-Tiny-3Layers
$P_1 = 5$	T_2	930.80	75.75	36.60	112.35	**13.85**
	P_2	5.0	4.8	4.85	4.95	5.0
	$T_2 - P_2$	925.8	70.95	31.75	107.30	**8.85**
	$P_1 - P_2$	**0.00**	0.20	0.15	0.05	**0.00**
	FC	0.54%	6.38%	13.53%	4.44%	**37.56%**
	MP	0.00%	4.00%	3.00%	1.00%	**0.00%**
$P_1 = 50$	T_2	963.85	120.7	83.4	161.45.	**57.25**
	P_2	**49.7**	47.8	47.15	49.55	48.90
	$T_2 - P_2$	914.15	72.9	36.25	111.90	**8.35**
	$P_1 - P_2$	**0.3**	2.2	2.85	0.45	1.10
	FC	5.16%	39.7%	56.71%	30.80%	**85.37%**
	MP	**0.60%**	4.40%	5.70%	0.90%	2.20%

From Figures 12 and 13, we can see that with the increase of the confidence threshold, the values of T_2, P_2 and $T_2 - P_2$ decrease gradually, meanwhile the values of $P_1 - P_2$, FC and MP increase. This is because that a higher confidence threshold will remove more prediction box and less are remained, which may lead to the increase of a falsely detection. Therefore, the selection of confidence threshold is a critical factor to affect the detection results, and should be considered carefully. Since the PWD is a vital disease to the pine trees, and a lower missing detection required. In the following we set the confidence threshold as 0.5 for analysis. And the detection results of the different lightweight models in Table 3, where the two cases $P_1 = 5$ and $P_1 = 50$ are considered. In Table 3, we can see that the proposed YOLOv4-Tiny-3Layers model can achieve a superior detection performance as compared to the other four comparing methods. For case 1 with $P_1 = 5$, the proposed model provides the minimal T_2 and all the infected pine trees are successfully detected, i.e., $P_1 = 5$. Besides, the falsely detected ($T_2 - P_2$) and missing detected ($P_1 - P_2$, MP) are also the optimal among these comparing methods. In terms of FC, the proposed model provides much larger FC than that of the other methods, which implies that it can minimize the transmission burden, and enhance the process efficiency.

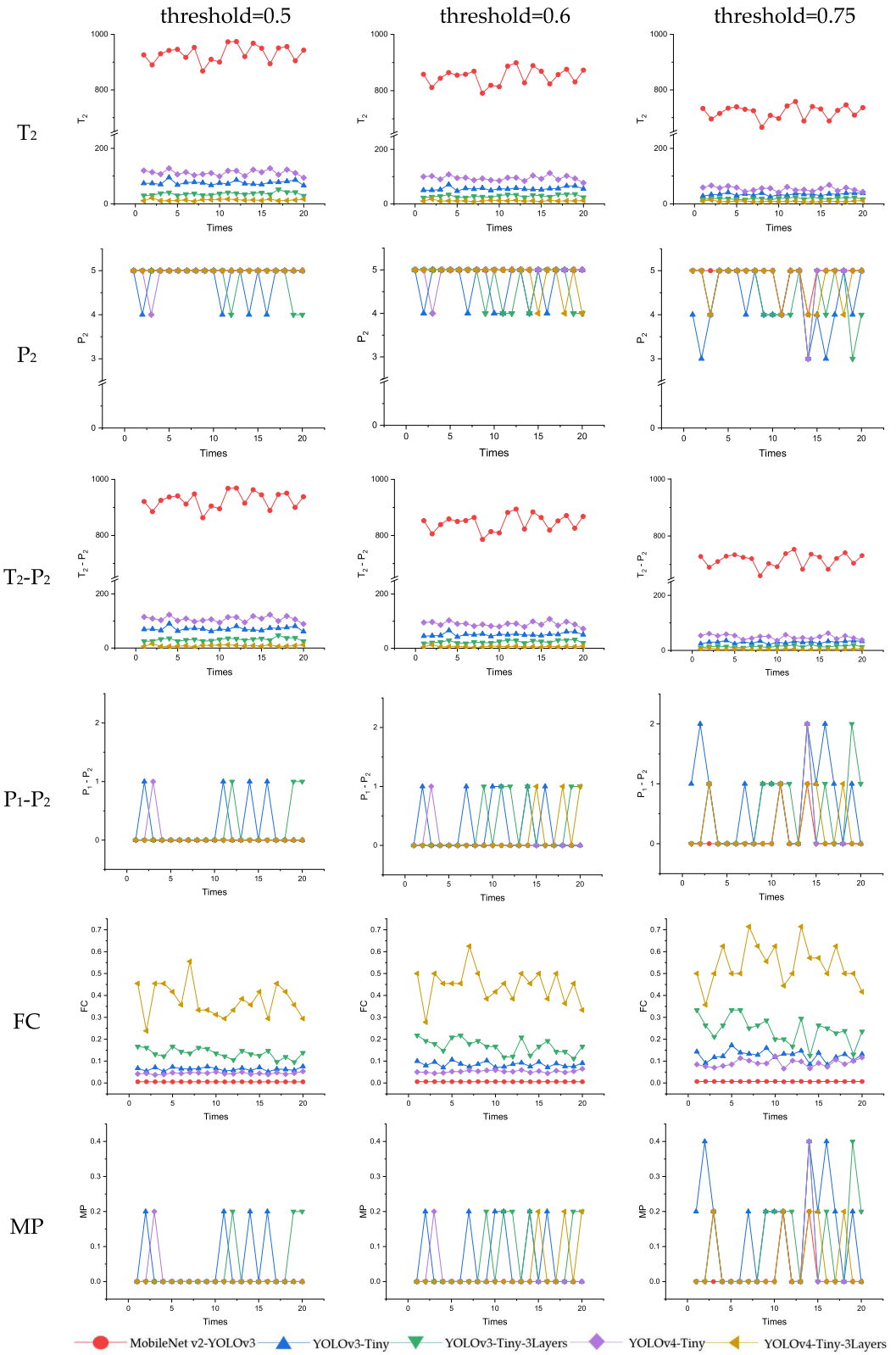

FIGURE 12. The detection performance of the different lightweight detection models over the first 20 trials, with $T_1 = 6007$, $P_1 = 5$, and the confidence thresholds are set as 0.5, 0.6, and 0.75 respectively.

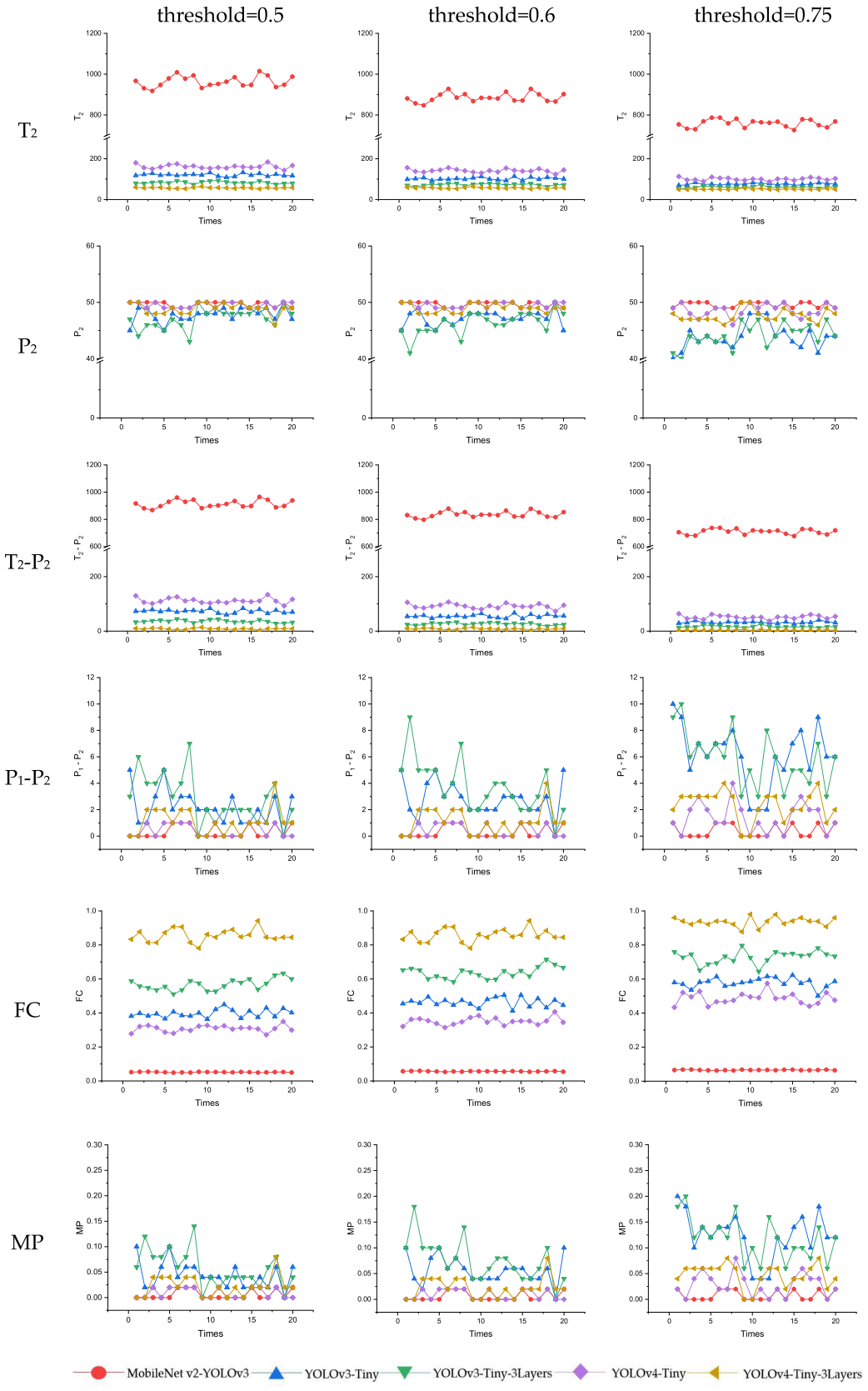

FIGURE 13. The detection performance of the different lightweight detection models over the first 20 trials, with $T_1 = 6007$, $P_1 = 50$, and the confidence thresholds are set as 0.5, 0.6, and 0.75 respectively.

For case 2 with $P_1 = 50$, the YOLOv4-Tiny-3Layers model still provides the minimal T_2 and falsely detected ($T_2 - P_2$), i.e., the filtered images needing to transmit are minimized. In terms of P_2, $P_1 - P_2$, and *MP*, the YOLOv4-Tiny-3Layers model is better than that of the YOLOv3-Tiny and YOLOv3-Tiny-3Layers, and has a slight performance loss as compared to MobileNetv2-YOLOv3 and YOLOv4-Tiny. Despite the slight performance loss, the YOLOv4-Tiny-3Layers model provides much better FC performance, which is quite critical to filter these uninterested images and alleviate the transmission and process burden. Therefore, by considering these metrics jointly, the proposed YOLOv4-Tiny-3Layers model can achieve a better tradeoff for detection.

2) DETECTION PERFORMANCE COMPARISONS OF THE LIGHTWEIGHT DETECTION MODELS AFTER STAGE 2 DETECTION (THE FINE-GRAINED DETECTION)

The edge-computing module performs a coarse detection (stage 1) and the filtered images are then transmitted to the ground workstation for a fine-grained detection (stage 2). To verify the final detection results, the ground workstation can adopt a high performance deep learning based detection model for the second stage detection. For consistency, in this experiment, we adopt the YOLOv4 model to detect these filtered images. That is, the images filtered by the previous edge-computing module will be transmitted to the ground workstation and detected by the YOLOv4 model. Except for the above mentioned lightweight detection models, we also consider the case when no edge computing is applied for fair comparison. In this experiment, we set the confidence threshold as 0.5 and the final average detection results of different models after the two-stage detection over 60 trials are shown in Table 4, where the final *MP* and the consuming time are provided.

TABLE 4. The average detection results of different models after the two-stage detection over 60 trials.

model	Case 1 ($P_1 = 5$)		Case 2 ($P_2 = 50$)	
	Time	MP	Time	MP
No edge computing	618.72s	**0.010**	624.72s	**0.010**
MobileNetv2-YOLOv3	95.84s	**0.010**	99.28s	**0.010**
YOLOv3-Tiny	7.80s	0.050	12.55s	0.044
YOLOv3-Tiny-3Layers	3.76s	0.043	8.59s	0.056
YOLOv4-Tiny	11.56s	0.010	16.79s	**0.010**
YOLOv4-Tiny-3Layers	**1.41s**	0.010	**5.88s**	0.020

In Table 4, we present the final average detection results of different methods after the two-stage detection over 60 trials. As can be observed, in case 1 ($P_1 = 5$), the proposed YOLOv4-Tiny-3Layers model provides the best *MP*, which is the same as no edge-computing approach, MobileNetv2-YOLOv3, and YOLOv4-Tiny, and is better than that of the YOLOv3-Tiny and YOLOv3-Tiny-3Layers. In case 2 ($P_1 = 50$), the proposed YOLOv4-Tiny-3Layers model has a slight MP loss, but is still better than that of YOLOv3-Tiny and YOLOv3-Tiny-3Layers.Table 4 also gives the average consuming time of different methods in the second detection stage. The YOLOv4-Tiny-3Layers uses the least time but provides almost the same detection *MP* compared to other methods for both cases. This further verifies the proposed lightweight model's effectiveness since the least number of images are remained and need to transfer to the ground workstation. It also seems that the no edge-computing approach and MobileNetv2-YOLOv3 also provide a better *MP* value. But notice that, the proposed YOLOv4-Tiny-3Layers model has the highest *FC* in the two cases as is shown in Table 3, which implies that the proposed model just needs to transmit the filtered images as less as possible, but still provide superior detection performance as compared to other models. E.g., the no edge-computing approach needs to transmit all the images (6007) to the ground workstation for the two cases, and MobileNetv2-YOLOv3 needs to transmit 930.80 and 963.85 on average for the two cases. Meanwhile, the proposed model just needs to transmit 13.85 and 57.25 on average, respectively. In complex forests with poor communication quality, the no edge-computing approach and MobileNetv2-YOLOv3 will suffer from a long transmission latency, even a transmission failure since so many images are required to transmit. In contrast, the YOLOv4-Tiny-3Layers model just needs to transmit the minimum amount of images. This great reduction is extremely useful for the communication-limited forests, and helps to detect the PWD infected pine trees in time for early decision.

IV. DISCUSSION

The focus of this paper is to design an airborne edge computing platform for the PWD infected tree detection, where a lightweight detection method is required for edge computing platform. By collaborating with the ground workstation, a two-stage detection procedure is performed, i.e., a coarse-grained and fine-grained detection.

Various challenges arise during the design of airborne platform. The first problem confronted is the selection of UAV type and hardware part of edge computing module. As for the UAV selection, it is necessary to comprehensively evaluate its size, working environment, flight endurance, payload capacity, etc. In the aspect of hardware selection of edge computing module, it is necessary to evaluate the processing ability of embedded AI chip, the sensitive area of camera and the compatibility with UAV platform.

In addition, considering the limitation of embedded AI chip in processing capacity and detection performance, a lightweight detection method becomes a necessary, which needs to provide a relatively high detection performance with limited resources. To achieve this goal, there are many problems needing to be addressed. For example, 1) how to simplify the structure of object detection method to make it suitable for the embedded AI chip with limited computing capacity; 2) how to improve detection accuracy and reduce *MP* as far as possible in the case of lightweight method; 3) how to find the compromise between "lightweight" and "accuracy" in practical applications.

To this end, a lightweight airborne edge-computing based detection platform is designed in this paper, which consists of three modules, i.e., UAV based image acquisition module, edge computing module and ground workstation module. Once the design scheme is determined, the hardware implementation should be determined according to the practical working environment, equipment miniaturization and integration requirement. The camera is integrated with the edge-computing module, which is then carried on the DJI M600Pro UAV and implanting the images acquisition. Due to the limited computation resources of the airborne equipment, a coarse-grained detection is performed to filter the uninterested images. After the coarse-grained detection, the filtered images are then transmitted to the ground workstation module through the airborne data transmission equipment, where a fine-grained detection is performed to further improve the detection results. To ensure the model lightweight and reduce the rate of missed detection and false detection, this paper proposes an improved YOLOv4-Tiny based detection method, named as YOLOv4-Tiny-3Layers, by making several improvements, e.g., adding a network output layer for multi-scale objection detection, improving the regression loss algorithm for optimize the training process and using DIoU non-maximum suppression algorithm to reduce the missing detection of overlapping objects. Experimental results show that the proposed method provides a superior detection performance, as compared to other methods.

The aim of the proposed YOLOv4-Tiny-3Layers detection method is to filter the uninterested images as accurately as possible, and reduce the transmit amount. Therefore, it requires that the missing detection is as low as possible meanwhile with high transmission efficiency. To reasonably evaluate the system performance, in the Results part, we use three different confidence thresholds and two sets of randomly selected images containing different number of infected pine trees. Experimental results show that when the threshold is set to be 0.5, the detection result can obtain a reasonable tradeoff, and achieve the aim of low missing detection rate.

After the coarse-grained detection, the filtered images are then transmitted to the ground workstation for a fine-grained detection. The overall performance of the models with/without edge-computing are compared in terms of detection accuracy, missing rate, etc. Experiments show the superiority of the proposed method. Therefore, the proposed airborne edge-computing based detection model can provide advantages of strong stability and high reliability, and is a new solution for monitoring pine wood nematode disease in forest areas.

V. CONCLUSION

In this paper, an airborne edge-computing based detection platform is developed with low missing and high precision, to monitor the pine wilt disease by UAV in forest area, which mainly consists of three modules, i.e., UAV based image acquisition module, edge computing module and ground workstation module. The detection platform can detect the acquired images as quickly as possible and find the PWD infected trees. Once these infected trees are detected and verified by our proposed method, the forestry staffs need to make an early decision on the strategy to dispose of these infected trees.

To adapt to the research region, the hardware part is approximately selected firstly and then complete the overall system design. Since the images acquired by the UAV may contain a large number of redundant images that does not contain infected images, the detection procedure is performed by two steps, a coarse-grained detection to filter uninterested images by the edge computing module and a fine-grained detection by the ground workstation module. Since the edge computing module is critical to affect the detection performance, in this paper, an improved YOLOv4-Tiny based lightweight detection method, named as YOLOv4-Tiny-3Layer, is proposed for the edge computing module. Due to the low miss detection rate requirement, we compare the proposed method with other detection models under different confidence thresholds, and found that when the confidence threshold is 0.5, the proposed YOLOv4-Tiny-3Layers method can achieve the expected performance requirement. Further, the filtered images are transmitted to the ground workstation module, and a comparative experiment is performed with/without the edge computing module, showing that the images collected by UAV can achieve high detection accuracy by the two-stage detection. Compared with the detection method without edge computing module, the proposed method can achieve almost the same detection performance; meanwhile, much less images are required to be processed by the ground workstation module. This is because that a large number of uninterested images are filtered by the edge computing module, and only suspected images are transmitted to the ground workstation module. Therefore, the proposed method can effectively filter these uninterested images and then can reduce the burden of transmission and processing, and helps to detect the PWD infected pine trees in time for an early decision on the strategy to dispose of these infected trees.

Moreover, the proposed method be applied in other regions, where it just needs a fine-tune according to the images acquired in the new region, and then can be used to detect the PWD with high accuracy in the new region.

ACKNOWLEDGMENT

The authors would like to thank the HPC center of Shandong Agricultural University for providing the computing support.

REFERENCES

[1] D. N. Proença, G. Grass, and P. V. Morais, "Understanding pine wilt disease: Roles of the pine endophytic bacteria and of the bacteria carried by the disease-causing pinewood nematode," *MicrobiologyOpen*, vol. 6, no. 2, Apr. 2017, Art. no. e00415.

[2] *Announcement of the National Forestry and Grassland Bureau*, Nat. Forestry Grassland Bur., Beijing, China, 2020, no. 2.

[3] J. Zhao, J. Huang, J. Yan, and G. Fang, "Economic loss of pine wood nematode disease in mainland China from 1998 to 2017," *Forests*, vol. 11, no. 10, p. 1042, Sep. 2020.

[4] M. Pause, C. Schweitzer, M. Rosenthal, V. Keuck, J. Bumberger, P. Dietrich, M. Heurich, A. Jung, and A. Lausch, "*In situ*/remote sensing integration to assess forest health—A review," *Remote Sens.*, vol. 8, no. 6, p. 471, Jun. 2016.

[5] P. E. Dennison, A. R. Brunelle, and V. A. Carter, "Assessing canopy mortality during a mountain pine beetle outbreak using GeoEye-1 high spatial resolution satellite data," *Remote Sens. Environ.*, vol. 114, no. 11, pp. 2431–2435, Nov. 2010.

[6] B. A. Johnson, R. Tateishi, and N. T. Hoan, "A hybrid pansharpening approach and multiscale object-based image analysis for mapping diseased pine and oak trees," *Int. J. Remote Sens.*, vol. 34, no. 20, pp. 6969–6982, Oct. 2013.

[7] S.-R. Kim, W.-K. Lee, C.-H. Lim, M. Kim, M. Kafatos, S.-H. Lee, and S.-S. Lee, "Hyperspectral analysis of pine wilt disease to determine an optimal detection index," *Forests*, vol. 9, no. 3, p. 115, Mar. 2018.

[8] J. P. Dash, M. S. Watt, G. D. Pearse, M. Heaphy, and H. S. Dungey, "Assessing very high resolution UAV imagery for monitoring forest health during a simulated disease outbreak," *ISPRS J. Photogramm. Remote Sens.*, vol. 131, pp. 1–14, Sep. 2017.

[9] X. Zhou and X. Zhang, "Individual tree parameters estimation for plantation forests based on UAV oblique photography," *IEEE Access*, vol. 8, pp. 96184–96198, 2020.

[10] G. Lindner, K. Schraml, R. Mansberger, and J. Hübl, "UAV monitoring and documentation of a large landslide," *Appl. Geomatics*, vol. 8, no. 1, pp. 1–11, Mar. 2016.

[11] H. Santoso, H. Tani, and X. Wang, "A simple method for detection and counting of oil palm trees using high-resolution multispectral satellite imagery," *Int. J. Remote Sens.*, vol. 37, no. 21, pp. 5122–5134, Nov. 2016.

[12] N. Zhang, X. Zhang, G. Yang, C. Zhu, L. Huo, and H. Feng, "Assessment of defoliation during the *Dendrolimus tabulaeformis Tsai et Liu* disaster outbreak using UAV-based hyperspectral images," *Remote Sens. Environ.*, vol. 217, pp. 323–339, Nov. 2018.

[13] X. Deng, Z. Tong, Y. Lan, and Z. Huang, "Detection and location of dead trees with pine wilt disease based on deep learning and UAV remote sensing," *AgriEngineering*, vol. 2, no. 2, pp. 294–307, May 2020.

[14] A. Rajagopal, A. Ramachandran, K. Shankar, M. Khari, S. Jha, Y. Lee, and G. P. Joshi, "Fine-tuned residual network-based features with latent variable support vector machine-based optimal scene classification model for unmanned aerial vehicles," *IEEE Access*, vol. 8, pp. 118396–118404, 2020.

[15] H. Tao, C. Li, D. Zhao, S. Deng, H. Hu, X. Xu, and W. Jing, "Deep learning-based dead pine tree detection from unmanned aerial vehicle images," *Int. J. Remote Sens.*, vol. 41, no. 21, pp. 8238–8255, Nov. 2020.

[16] M. Syifa, S.-J. Park, and C.-W. Lee, "Detection of the pine wilt disease tree candidates for drone remote sensing using artificial intelligence techniques," *Engineering*, vol. 6, no. 8, pp. 919–926, Aug. 2020.

[17] R. Zhang, L. Xia, L. Chen, C. Xie, M. Chen, and W. Wang, "Recognition of wilt wood caused by pine wilt nematode based on U-Net network and unmanned aerial vehicle images," *Trans. Chin. Soc. Agricult. Eng.*, vol. 36, no. 12, pp. 61–68, Jul. 2020.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," Dec. 2016, *arXiv:1512.02325*. [Online]. Available: http://arxiv.org/abs/1512.02325

[20] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[21] Y. Cai, C. Li, S. Wang, and J. Cheng, "DeLTR: A deep learning based approach to traffic light recognition," in *Image and Graphics* (Lecture Notes in Computer Science), Y. Zhao, N. Barnes, B. Chen, R. Westermann, X. Kong, and C. Lin, Eds. Cham, Switzerland: Springer, 2019, pp. 604–615.

[22] S. Pang, S. Wang, A. Rodríguez-Patón, P. Li, and X. Wang, "An artificial intelligent diagnostic system on mobile Android terminals for cholelithiasis by lightweight convolutional neural network," *PLoS ONE*, vol. 14, no. 9, Sep. 2019, Art. no. e0221720.

[23] L. Fu, Y. Feng, J. Wu, Z. Liu, F. Gao, Y. Majeed, A. Al-Mallahi, Q. Zhang, R. Li, and Y. Cui, "Fast and accurate detection of kiwifruit in orchard using improved YOLOv3-tiny model," *Precis. Agricult.*, early access, pp. 1–23, Sep. 2020, doi: 10.1007/s11119-020-09754-y.

[24] B.-G. Han, J.-G. Lee, K.-T. Lim, and D.-H. Choi, "Design of a scalable and fast YOLO for edge-computing devices," *Sensors*, vol. 20, no. 23, p. 6779, Nov. 2020.

[25] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," Apr. 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[26] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," Apr. 2020, *arXiv:2004.10934*. [Online]. Available: http://arxiv.org/abs/2004.10934

[27] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," Nov. 2019, *arXiv:1911.08287*. [Online]. Available: http://arxiv.org/abs/1911.08287

[28] Z. Jiang, L. Zhao, S. Li, and Y. Jia, "Real-time object detection method based on improved YOLOv4-tiny," 2020, *arXiv:2011.04244*. [Online]. Available: https://arxiv.org/abs/2011.04244

[29] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Goncalves, W. R. Schwartz, and D. Menotti, "A robust real-time automatic license plate recognition based on the YOLO detector," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–10.

[30] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, vol. 3, Aug. 2006, pp. 850–855.

[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2015, pp. 91–99.

[32] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: An advanced object detection network," in *Proc. 24th ACM Int. Conf. Multimedia (MM)*, Oct. 2016, pp. 516–520.

[33] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS—Improving object detection with one line of code," Apr. 2017, *arXiv:1704.04503*. [Online]. Available: http://arxiv.org/abs/1704.04503

[34] S. Vicente, J. Carreira, L. Agapito, and J. Batista, "Reconstructing Pascal VOC," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 41–48.

FENGDI LI received the B.E. degree from the College of Computer Information Engineering, Jiangxi Normal University, in 2018. He is currently pursuing the M.S. degree with Shandong Agricultural University. His main research interests include computer science, deep learning, and object detection.

ZHENYU LIU is currently a Professor with Shandong Agricultural University. His research interests include plant protest and sustainable management of plant disease.

WEIXING SHEN received the Ph.D. degree from Shandong Agricultural University, in 2012. He is currently with the Taishan Forest Pest Management and Quarantine Station. His research interests include plant protest and plant pest control.

YAN WANG received the B.E. degree from the College of Business, Anhui University of Technology, in 2019. She is currently pursuing the M.S. degree with Shandong Agricultural University. Her main research interests include agriculture information service technology and engineering.

YUNLU WANG received the B.E. degree from the College of Information Science and Engineering, Shandong Agricultural University, in 2020, where he is currently pursuing the M.S. degree. His main research interests include computer systems, image processing, and object detection.

CHENGKAI GE received the B.E. degree from the College of Information Engineering, Hangzhou Dianzi University, in 2017. He is currently pursuing the M.S. degree with Shandong Agricultural University. His main research interests include image processing and object detection.

FENGGANG SUN received the B.Sc. and M.Sc. degrees from the College of Information Science and Engineering, Shandong University, in 2006 and 2009, respectively, and the Ph.D. degree from the Institute of Communications Engineering, PLA University of Science and Technology. He is currently with Shandong Agricultural University. His research interests include signal processing, compressive sensing, and remote sensing.

PENG LAN received the B.Sc. and Ph.D. degrees from Shandong University, in 2004 and 2009, respectively. He is currently an Associate Professor with Shandong Agricultural University. His research interests include compressive sensing, optimization methods, signal processing, compressive sensing, and remote sensing.