



## OPEN ACCESS

## EDITED BY

Chunlei Xia,  
Chinese Academy of Sciences (CAS), China

## REVIEWED BY

Xing Xu,  
South China Agricultural University, China  
Lixiong Gong,  
Hubei University of Technology, China  
Ankush Sawarkar,  
Shri Guru Gobind Singhji Institute of  
Engineering and Technology, India

## \*CORRESPONDENCE

Ruoling Deng  
✉ dengruoling@gdou.edu.cn

RECEIVED 02 April 2025

ACCEPTED 30 June 2025

PUBLISHED 17 July 2025

## CITATION

Zhang J, Deng R, Cai C, Zou E, Liu H, Hou M,  
Chen X, Lin H and Wei Z (2025) Automatic  
detection of lucky bamboo nodes based on  
Improved YOLOv7.

*Front. Plant Sci.* 16:1604514.

doi: 10.3389/fpls.2025.1604514

## COPYRIGHT

© 2025 Zhang, Deng, Cai, Zou, Liu, Hou, Chen,  
Lin and Wei. This is an open-access article  
distributed under the terms of the [Creative  
Commons Attribution License \(CC BY\)](#). The  
use, distribution or reproduction in other  
forums is permitted, provided the original  
author(s) and the copyright owner(s) are  
credited and that the original publication in  
this journal is cited, in accordance with  
accepted academic practice. No use,  
distribution or reproduction is permitted  
which does not comply with these terms.

# Automatic detection of lucky bamboo nodes based on Improved YOLOv7

Jing Zhang<sup>1,2,3</sup>, Ruoling Deng<sup>1,2,3\*</sup>, Chengzhi Cai<sup>1</sup>, Erpeng Zou<sup>1</sup>,  
Haitao Liu<sup>1,3</sup>, Mingxin Hou<sup>1</sup>, Xinzhi Chen<sup>4</sup>, Huamin Lin<sup>1</sup>  
and Zhenye Wei<sup>1</sup>

<sup>1</sup>School of Mechanical Engineering, Guangdong Ocean University, Zhanjiang, Guangdong, China,

<sup>2</sup>Guangdong Engineering Technology Research Center of Ocean Equipment and Manufacturing, Guangdong Ocean University, Zhanjiang, Guangdong, China, <sup>3</sup>Guangdong Provincial Key Laboratory of Intelligent Equipment for South China Sea Marine Ranching, Guangdong Ocean University, Zhanjiang, Guangdong, China, <sup>4</sup>College of Mathematics and Computer Science, Guangdong Ocean University, Zhanjiang, Guangdong, China

**Introduction:** The detection of lucky bamboo (*Dracaena sanderiana*) nodes is a critical prerequisite for machining bamboo into high-value handicrafts. Current manual detection methods are inefficient, labor-intensive, and error-prone, necessitating an automated solution.

**Methods:** This study proposes an improved YOLOv7-based model for real-time, precise bamboo node detection. The model integrates a Squeeze-and-Excitation (SE) attention mechanism into the feature extraction network to enhance target localization and introduces a Weighted Intersection over Union (WIoU) loss function to optimize bounding box regression. A dataset of 2,000 annotated images (augmented from 1,000 originals) was constructed, covering diverse environmental conditions (e.g., blurred backgrounds, occlusions). Training was conducted on a server with an RTX 4090 GPU using PyTorch.

**Results:** The proposed model achieved a 97.6% mAP@0.5, significantly outperforming the original YOLOv7 (83.4% mAP) by 14.2%, while maintaining the same inference speed (100.18 FPS). Compared to state-of-the-art alternatives, our model demonstrated superior efficiency. It showed 41.5% and 153% higher FPS than YOLOv11 (70.8 FPS) and YOLOv12 (39.54 FPS), respectively. Despite marginally lower mAP ( $\leq 1.3\%$ ) versus these models, the balanced trade-off between accuracy and speed makes it more suitable for industrial deployment. Robustness tests under challenging conditions (e.g., low light, occlusions) further validated its reliability, with consistent confidence scores across scenarios.

**Discussion:** The proposed method significantly improves detection accuracy and efficiency, offering a viable tool for industrial applications in smart agriculture and handicraft production. Future work will address limitations in detecting nodes obscured by mottled patterns or severe occlusions by expanding label categories during training.

## KEYWORDS

lucky bamboo, handicraft, convolutional neural network, YOLOv7, object detection, bamboo node

## 1 Introduction

Lucky bamboo (*Dracaena sanderiana*) is a potted ornamental plant with excellent ornamental value (Chen, 2012; Akinlabi et al., 2017; Gergel and Turner, 2017). Traditional lucky bamboo is only used for flower arrangement or direct potting, with low added value (Rout et al., 2006; van Dam et al., 2018; Abdel-Rahman et al., 2020). Processing lucky bamboo into handicrafts can substantially increase its ornamental value, which is deeply loved by the public and has a large demand in the international market (Sharma et al., 2014; Amin and Mujeeb, 2019; Liu et al., 2019). Since the processing of lucky bamboo is cutting the lucky bamboo according to the bamboo nodes to meet different requirements, identifying the lucky bamboo nodes is the first and most crucial step in processing lucky bamboo. However, the existing methods of identifying lucky bamboo nodes still mainly rely on manual work, which has the disadvantages of low efficiency, high labour cost, and prone to errors. Therefore, studying a method that can automatically recognize lucky bamboo nodes with high efficiency and precision is imperative.

In recent years, traditional image processing methods have been widely used to identify bamboo-related fields. Juyal P et al. used methods such as logistic regression, support vector machine, naive Bayesian, random forest, convolutional neural network and ResNet to conduct a comparative analysis, and finally could accurately identify five common bamboos (Juyal et al., 2020). Watanabe used convolutional neural networks (CNN) to identify Japanese bamboo forest areas through Google satellite images, with an overall recognition accuracy of 93.7% (Watanabe et al., 2020). Kumar conducted research on bamboo leaf disease detection and developed a program to automatically recognize bamboo leaf diseases based on image processing and CNN (Kumar et al., 2022). Ziwei Wang used a residual neural network, original dataset, and MixUp dataset to optimize the traditional algorithm CNN to further improve the classification ability of bamboo species (Wang et al., 2022). Pankaja used Fourier descriptors to extract bamboo leaf features and used the Bayes classifier to identify bamboo leaves with an accuracy of 88.03% (Pankaja and Thippeswamy, 2017).

Existing target detection algorithms can be roughly divided into two categories. The first category is the two-stage R-CNN (He et al., 2017) series of algorithms based on region proposal, such as R-CNN, Fast R-CNN (Li et al., 2017), Faster R-CNN (Salvador et al., 2016), etc. The position of the object frame usually needs to be found first and then the category of the object frame will be determined by this algorithm. Although this type of method has high recognition accuracy, it takes a long time to calculate and is not suitable for real-time detection. The second category is the one-stage algorithm (Tian et al., 2022) represented by YOLO (Redmon, 2016) and SSD (Liu et al., 2016). This type of algorithm takes regression as the core, omitting the region proposal link of the two-stage algorithm, directly distinguishing specific categories and returning the bounding box (Redmon and Farhadi, 2017). Shilan Hong used optimized YOLOv4 to model the detection of bamboo shoots and proposed a classification and screening strategy to track each bamboo shoot (Hong et al., 2022). The experimental results

showed that the average relative error and variance of the number of bamboo shoots were 1.28% and 0.016%, respectively, and the average relative error and variance of the corresponding pixel height results were -0.39% and 0.02%, respectively. The advantage of this method is that it can perform real-time detection, which is beneficial to improving the recognition efficiency of bamboo nodes. However, there is much room for improvement in the detection accuracy and robustness of this method, and its detection ability for small targets is also relatively poor. The current target detection algorithm cannot take into account both detection accuracy and timeliness, and the detection effect for small target units such as bamboo nodes is relatively poor.

To solve the above problems, this paper proposed a method for real-time and precise detection of lucky bamboo nodes based on the improved model of YOLOv7. Firstly, an attention mechanism was introduced into the feature extraction network to enhance effective feature information and suppress invalid information. This can help the model locate and identify the lucky bamboo nodes faster and more accurately. Then, WIoU was introduced in the loss value calculation to optimize the bounding box regression process of the bamboo node through a dynamic weighting mechanism, thereby improving the model's detection ability for complex scenes and small bamboo nodes.

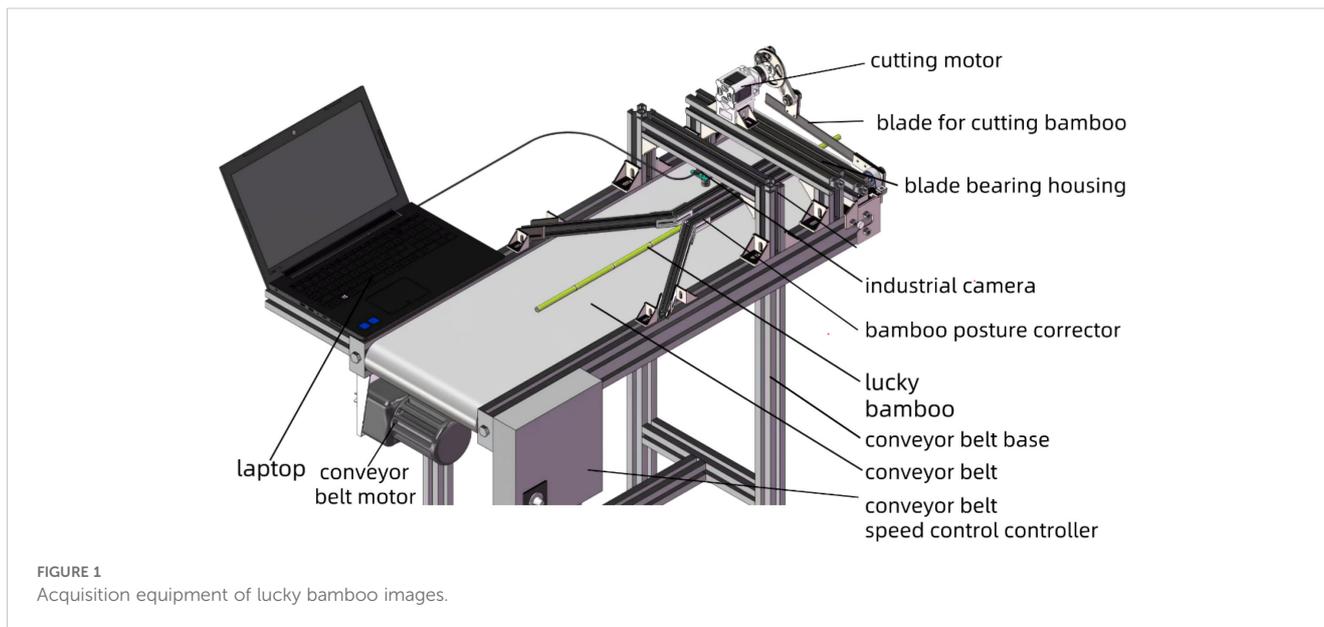
The main objectives of this research were to (a) construct the image dataset of lucky bamboo nodes using the data augmentation method, (b) establish the lucky bamboo node detection model using the improved YOLOv7, and (c) evaluate the detection stability and accuracy of the proposed method.

## 2 Materials and methods

### 2.1 Dataset construction

The image collecting equipment used in this study is shown in Figure 1. The equipment consisted of an USB industrial camera, an assembly line workbench, and a laptop. The assembly line workbench was equipped with conveyor belt, a conveyor belt motor, conveyor belt speed control controller, bamboo posture corrector, a cutting motor, a blade for cutting bamboo, a blade bearing housing, and conveyor belt base. The industrial camera was mounted on the tube, parallel to the conveyor belt.

The lucky bamboo samples were planted in the processing base of Fugui Horticultural Farm in Mazhang District, Zhanjiang City, Guangdong Province, China. The images of lucky bamboo were collected using USB industrial camera connected to a laptop on November 1, 2023. The focal length of the camera lens is 3.6 mm. Specifically, the lucky bamboo was placed on the horizontal conveyor belt and then photographed at a vertical height of 50 mm from the lucky bamboo. Since the original lucky bamboo image had a resolution of 2 million pixels, which was too large and will increase the amount of calculation, OpenCV in Python was used to process the original image. Finally, the original images were compressed to 640×640 pixels, with a horizontal and vertical resolutions of 96 dpi. Among them, the lucky bamboo image was



stored in JPG format with a size of 32 MB. In the end, a total of 1,000 lucky bamboo images were collected, as shown in Figure 2.

To improve the generalization ability and robustness of the model and avoid model overfitting, the transforms module based on the Pytorch deep learning framework was used to perform data augmentation and expansion on the collected images (Figure 3A) (Cubuk et al., 2019). Since two major data enhancement methods (geometric transformation (flip) and color space perturbation (jitter + grayscale) had been widely proven to effectively improve the generalization ability of the model (Shorten and Khoshgoftaar, 2019). Therefore, the lucky bamboo images were performed random horizontal flip (Figure 3B), random vertical flip (Figure 3C), and color jitter (Figure 3D). Also, the

RandomGrayscale function was used to convert the image to grayscale with a probability of 2.5% (Figure 3E), allowing the model to learn image features without color information and enhance the detection capabilities in complex scenes. After data enhancement, a total of 5000 images were obtained. Finally, 2000 representative images were obtained for constructing the lucky bamboo dataset. Also, the bamboo dataset was randomly divided into training set, verification set, and testing set in a ratio of 7:2:1 for model training and testing.

To enable the model to accurately locate bamboo nodes, the open-source software LabelImg was used to manually annotate all the lucky bamboo images after data augmentation (Figure 4) (Darrenl, 2017). In this study, the content of annotation was each node of lucky bamboo, that was, the collected location coordinate information. After labeling, all data were saved in the Pascal VOC dataset format. The annotation diagram is shown in Figure 4. Among them, the green frame represented the position of the bamboo nodes of lucky bamboo in the image.



## 2.2 Bamboo node detection method

### 2.2.1 Advantage of YOLOv7 model

YOLO (You Only Look Once) was a deep learning algorithm for real-time object detection that only required one forward propagation through a given neural network to detect all objects in the image (Redmon, 2016). This gave the YOLO algorithm an advantage over other algorithms in terms of speed, making it one of the most famous detection algorithms to date.

YOLO divided the image into multiple small grids and predicted multiple bounding boxes in each grid, as well as the object categories within each bounding box. YOLO used a single neural network to predict all bounding boxes and classes in an image, instead of using multiple neural networks to predict each bounding box. The advantage of YOLO was that it can run in real time and can detect

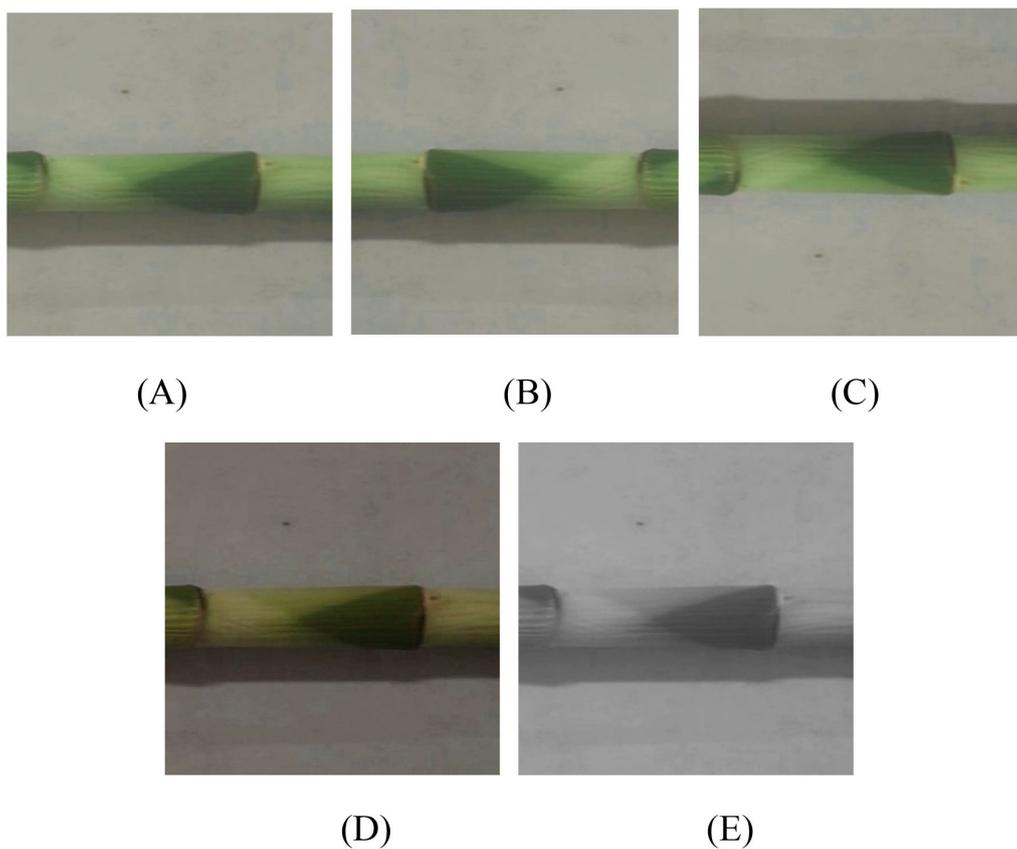


FIGURE 3 Data augmentation of lucky bamboo nodes. (A) original image; (B) random horizontal flip, (C) random vertical flip, (D) color jitter, (E) random grayscale.



FIGURE 4 The image annotation of lucky bamboo nodes (Darrenl, 2017).

more objects. YOLOv7 (Wang et al., 2023) was optimized on the basis of YOLOv4 (Bochkovskiy et al., 2020), training a better model with less training time. The input layer of YOLOv7 supported image enhancement (such as Mosaic) and adaptive anchor box calculation. Its backbone network used CSPDarknet53 and enhanced feature extraction capabilities by using CSPNet. Also, the Neck module in YOLOv7 combined Path Aggregation Network (PANet) and Spatial Pyramid Pooling (SPP) modules, which can better handle multi-scale features and enhance the accuracy of the model.

The algorithm mainly consisted of an input terminal, a feature extraction network, a feature fusion network, and an output terminal (Figure 5). Compared with YOLOv4, YOLOv7 employed a Focus operation that sampled the original image at double intervals in both horizontal and vertical directions. This reduced FLOPs value and computational complexity, thereby improving detection speed. Additionally, in the feature extraction module, YOLOv7 replaced the original CSP module with the C3 module. This modification enhanced training speed, reduced gradient redundancy, and improved learning efficiency. For network input processing, YOLOv3 (Farhadi and Redmon, 2018) and YOLOv4 required executing a separate program to compute initial anchor boxes when training on different datasets. In contrast, YOLOv7 integrated this functionality directly into the framework, enabling adaptive calculation of optimal anchor boxes for each training scenario. Also, unlike two-stage algorithms (e.g., Faster R-CNN), YOLOv7 eliminated the computationally intensive feature extraction and region proposal steps, significantly reducing inference time. Although the accuracy of YOLOv7 was slightly lower than that of Faster RCNN, its detection speed was faster and supported real-time detection. Therefore, YOLOv7 was selected as the basic framework for detecting lucky bamboo nodes.

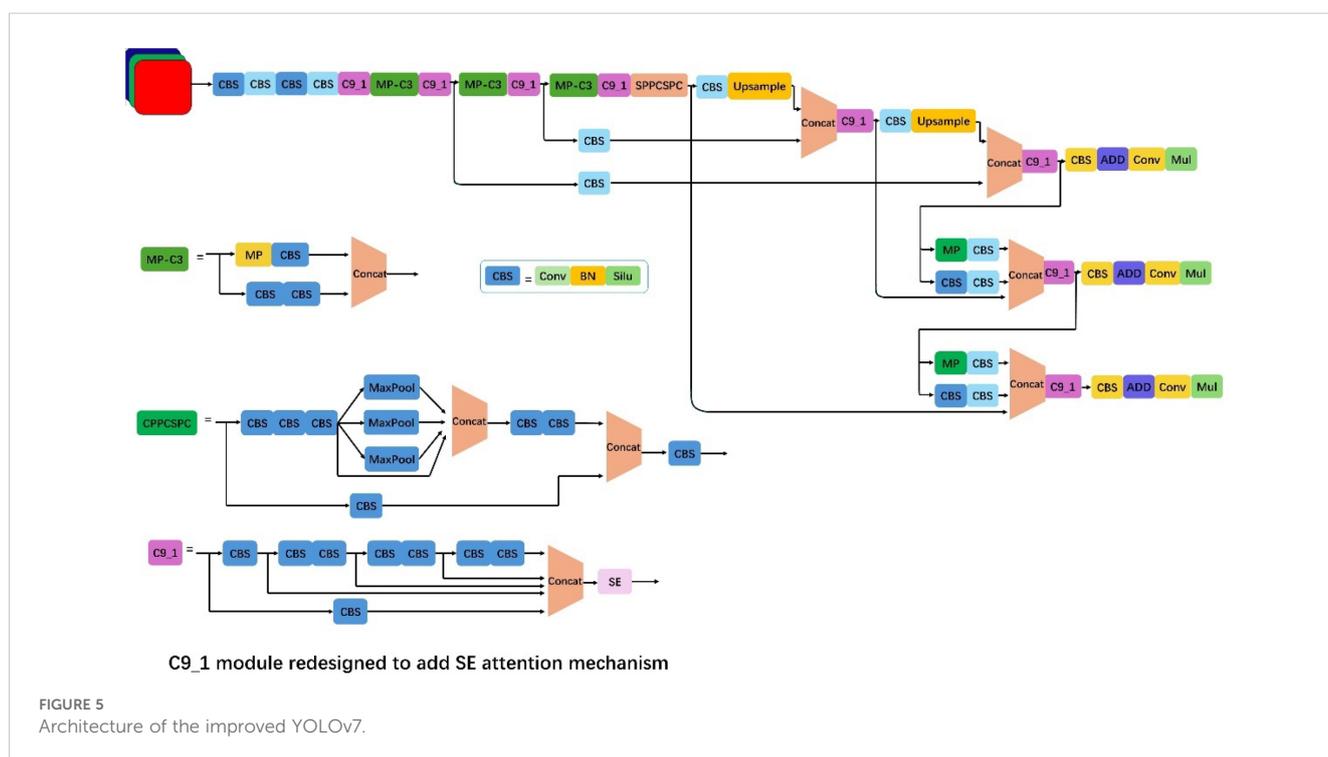
### 2.2.2 Problems caused by using YOLOv7 model

YOLOv7 was currently the best-engineered generation object detection model of the YOLO series. It achieved real-time image processing speeds while maintaining accuracy compared to state-of-the-art models. Therefore, it was widely used in real-time vision applications. However, YOLOv7 still had some shortcomings in the application environment of lucky bamboo node detection in this paper, which were mainly reflected in the following aspects.

- (a) The regression idea of the YOLO algorithm was to divide the image into  $S \times S$  grids, that was, each grid could only predict at most one object. Consequently, when multiple objects occupied the same grid, the algorithm's detection performance degraded significantly, often failing to identify all objects.
- (b) The original network employed the Generalized Intersection over Union (GIoU) (Rezatofighi et al., 2019) loss function for bounding box regression. However, GIoU demonstrated limited effectiveness for small object detection, as it failed to explicitly incorporate object scale considerations. Furthermore, this loss function may induce a bias toward predicting larger bounding boxes, ultimately compromising detection accuracy.

### 2.2.3 Construction of bamboo node detection model

In response to the problems raised above, the following improvements were made to the YOLOv7 original network in this paper.



- (a) Adding the SE attention mechanism (Hu et al., 2018; Niu et al., 2021) can make the network pay more attention to the bamboo nodes to be detected, thereby improving the model accuracy. The SE attention mechanism adaptively recalibrated channel-weight feature responses through its Squeeze and Excitation operations, selectively enhancing discriminative features. This mechanism enhanced feature representation accuracy by adaptively amplifying salient features while suppression irrelevant channel responses. Furthermore, the SE module enhanced model performance in complex scenarios through learned channel-weight adaptation, dynamically optimizing feature attention to improve both robustness and detection accuracy.
- (b) To enhance model detection efficiency, we implemented the Weighted Intersection over Union (WIoU) bounding box loss function (Cho, 2021; Tong et al., 2023) at the network's output layer. The proposed method introduced an efficient IoU-based loss function that addressed the limitations of conventional approaches, achieving both accelerated convergence and enhanced regression accuracy. The WIoU loss function incorporated a dynamic non-monotonic focusing mechanism that more effectively evaluated anchor quality. This approach reduced dominance by high-quality anchors while mitigating harmful gradients from low-quality samples. This allowed the WIoU loss function to focus on anchor frames of ordinary quality and improve overall detection performance.

### 2.2.3.1 SE attention mechanism

In the traditional convolutional neural network (CNN) architecture, convolutional layers and pooling layers were the core components for building deep feature representations (Sawarkar et al., 2024). These layers formed hierarchical feature representations by gradually extracting local features in the image and reducing the spatial dimension of the data. However, there was an implicit assumption in this process. That was, each channel of the feature map was equally important to the final task. Nevertheless, in practical applications, different channels often carry different amounts of information or importance, and contribute differently to the task (Chen et al., 2017). To solve this problem, Hu et al. proposed the SE attention mechanism architecture, which improved the model's ability to express

features by adaptively recalibrating the importance of each channel (Hu et al., 2018). Firstly, global average pooling was used to capture the global information of each channel, and then a channel weight vector was generated using a fully connected layer and a sigmoid activation function. Then, this weight vector was applied to each channel of the original feature map, and the feature map was scaled by element-by-element multiplication between channels, thereby enhancing the feature representation of those important channels and weakening the influence of those irrelevant or redundant channels. This adaptive channel recalibration mechanism enabled the model to focus more on the features that contributed most to the task, thereby improving the performance and generalization ability of the network. Therefore, a three-layer SE attention mechanism was added to the backbone network of the original YOLOv7 model. The structure diagram of the SE attention mechanism is shown in Figure 6.

### 2.2.3.2 WIoU loss function

The Generalized Intersection over Union (GIoU) loss function was used in the original YOLOv7 architecture (Bochkovskiy et al., 2020; Wang et al., 2023). In most cases, GIoU can calculate IoU at a wide level. When predicted boxes perfectly coincided with ground truth boxes, their intersection area equaled their individual areas. Also, their minimum bounding rectangles were also the same. In this case, the GIoU value saturated at 1, making it unable to distinguish subtle deviations in predicted box alignment. Under this condition, GIoU reduced to standard IoU. Furthermore, the GIoU loss function suffered from two key limitations: (a) higher computational overhead, and (b) slower convergence compared to more recent alternatives. To calculate GIoU, it was necessary to find the minimum bounding rectangle for each predicted box and the true box. This approach introduced significant computational overhead and adversely impacted training convergence, particularly when processing high-volume datasets or high-resolution imagery. To solve this problem, this paper adopted the Weighted Intersection over Union (WIoU) loss function in the improved network. WIoU incorporated a dynamic non-monotonic mechanism for bounding box regression that adaptively modulated gradient distributions based on overlap states, effectively mitigating both excessive and harmful gradients from outlier samples. Through optimized gradient allocation, the WIoU loss function enhanced model performance in normal cases while demonstrating superior robustness in extreme scenarios, simultaneously accelerating convergence and improving training efficiency.

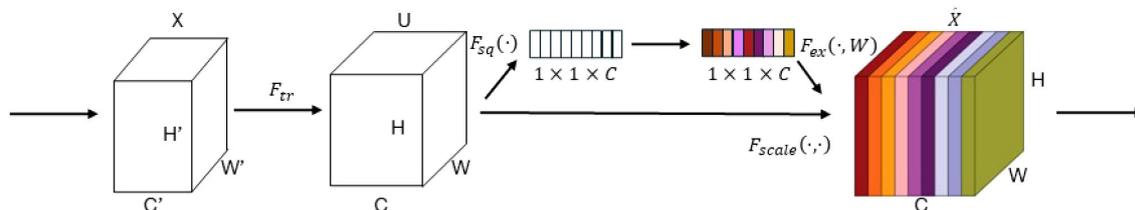


FIGURE 6  
The structure of SE attention mechanism.

Consequently, this study replaced the original YOLOv7's GIoU loss function with the WIoU variant to enhance bounding box regression performance. The calculation process of GIoU and WIoU loss function were shown in the following Equations 1–6.

$$IoU = \frac{|B_p \cap B_{gt}|}{|B_p \cup B_{gt}|} \quad (1)$$

$$GIoU = IoU - \frac{|C \setminus (B_p \cup B_{gt})|}{|C|} \quad (2)$$

$$L_{GIoU} = 1 - GIoU \quad (3)$$

$$\beta = \frac{v}{\alpha} \cdot \frac{|B_p \cap B_{gt}|}{|B_p \cup B_{gt}|} \quad (4)$$

$$WIoU = IoU \cdot \beta \quad (5)$$

$$L_{WIoU} = 1 - WIoU \quad (6)$$

Where  $B_p$  is the predicted bounding box,  $B_{gt}$  is the true bounding box,  $C$  is the smallest rectangle that can contain both the predicted box  $B_p$  and the true box  $B_{gt}$ ,  $\beta$  is dynamic weight,  $v$  is a normalization factor and  $\alpha$  is a learnable parameter.

## 2.3 Experimental environment

The experiment platform in this research was an autonomously configured server running in the deep learning framework. The hardware environment is Intel Core i9-14900KF processor with 64GB running memory, and the graphics card is Nvidia GeForce RTX 4090 with 24GB memory. The software environment is a virtual environment built using Anaconda under the Ubuntu20.04 operating system. The virtual environment consisted of Pytorch 2.1.0, CUDA 12.3, and Python 3.8.6. The Python language was used as the main language for writing program codes. Also, the numpy, pandas, OpenCV and other required libraries were called to implement the training and testing of the lucky bamboo node detection model. The hardware and software configurations for the established model were listed in Table 1.

TABLE 1 The hardware and software configurations for the established model.

Project	Content
Operating system	Ubuntu20.04
CPU	Intel Core i9-14900KF
GPU	NVIDIA GeForce RTX 4090
RAM	64GB
Compiled language	Python3.8.6
Deep learning framework	CUDA12.3 Pytorch 2.1.0

## 2.4 Evaluation of bamboo node detection model

In this study, objective indicators such as precision, recall, and loss function convergence curve would be used to evaluate the performance of bamboo node detection model. Among them, Intersection over Union (IOU) represents the ratio of the intersection and union between the detected bounding box and the real bounding box, which is a common indicator for evaluating the performance of object detection model. The higher the IOU value, the better the model detection performance. Precision refers to the ratio of the number of correctly detected bamboo nodes to the total number of detected bamboo nodes. Recall is the ratio of the number of correctly detected bamboo nodes to the number of actual true bamboo nodes. The precision and recall were computed by Equations 7 and 8. True Positive (TP) indicates the number of correctly detected lucky bamboo nodes when the IOU is greater than or equal to the selected threshold. False Positive (FP) indicates the number of misjudged bamboo nodes when the IOU is smaller than the selected threshold. False Negative (FN) represents the number of undetected bamboo nodes. Average Precision (AP) refers to the area under the Precision-Recall curve. The higher the AP value, the better the performance of the model in detecting bamboo nodes. Mean Average Precision (mAP) refers to the mean value of AP for all categories. The AP and mAP were computed by Equations 9 and 10.

$$P = \frac{TP}{TP + FP} \quad (7)$$

$$R = \frac{TP}{TP + FN} \quad (8)$$

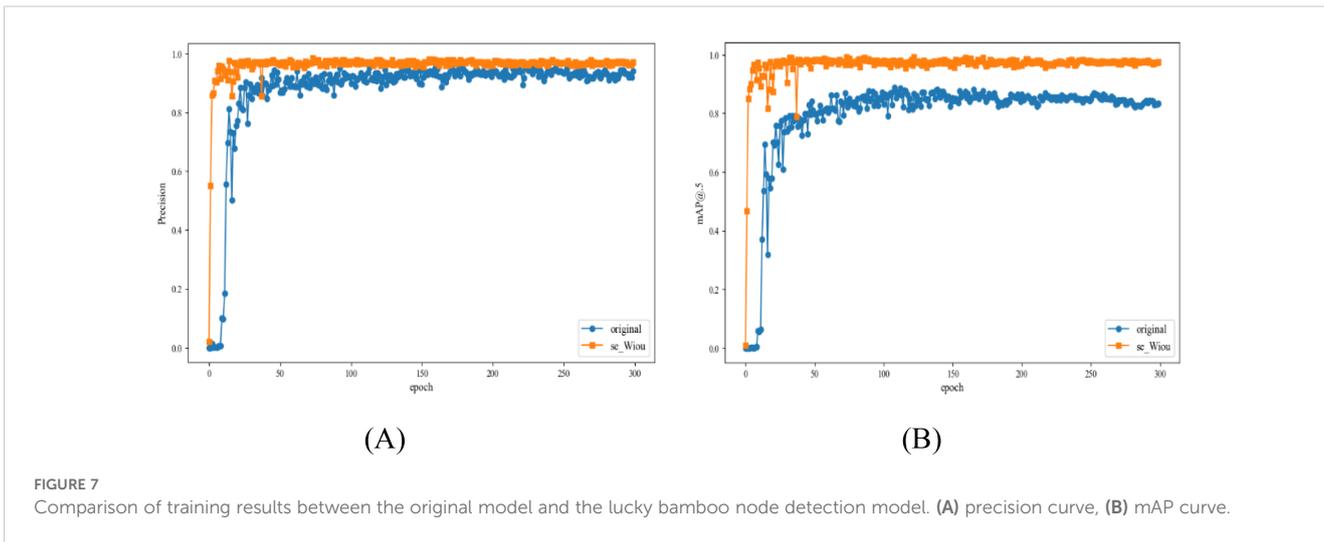
$$AP = \int_0^1 P(R) dR \quad (9)$$

$$mAP = \frac{1}{N} \sum_1^N AP_i \quad (10)$$

## 3 Results and discussion

### 3.1 Performance of lucky bamboo node detection model

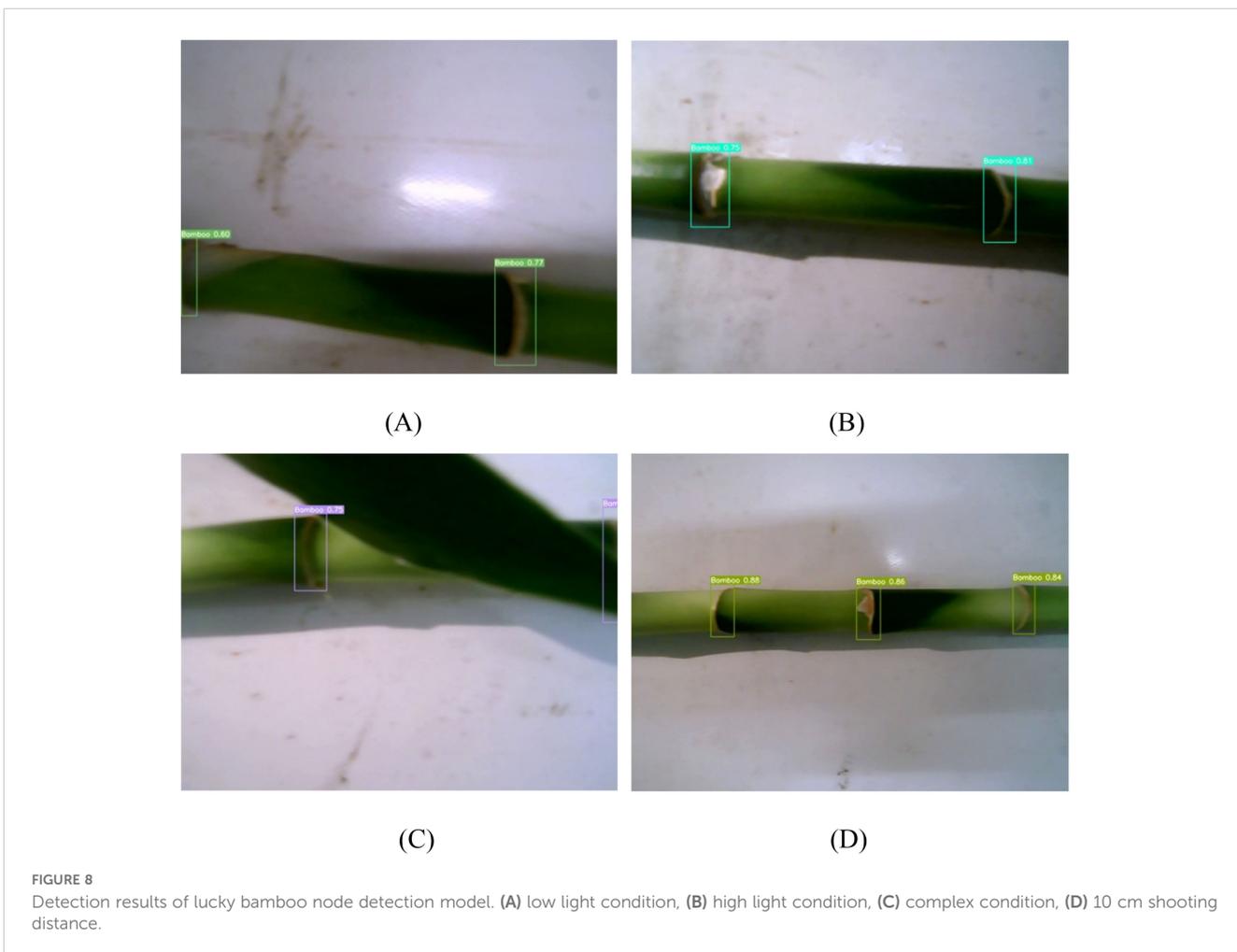
The training results of lucky bamboo node detection model are shown in Figure 7. With the increase of training epochs, the precision value, and mAP value of lucky bamboo node detection model gradually increased. The details are as follows. During epochs 0–20, the precision values of the model increased rapidly. Afterwards, the precision values of the model remained stable between 0.95 and 0.97. Similarly, the mAP of the model increased rapidly during epochs 0–20 and then remained stable between 0.96 and 0.99. Conclusively, the training was suspended at 300 epochs.



### 3.2 Detection result of lucky bamboo node detection model

The lucky bamboo node detection model was tested on additional 400 RGB images collected later. Figure 8 shows some examples of the

detected bamboo nodes in different environmental conditions. The confidence scores were indicated beside each detected node. The high scores (up to 1.000) demonstrated that the results were quite reliable. In Figure 8A, even though the lucky bamboo plant was in low-light environment and blurry, the bamboo nodes thereon can still be



correctly detected. In images taken in high-light conditions, the bamboo nodes appeared brighter and the bamboo nodes were similar in color to the background and surrounded by black shadows. It would be hard to recognize them manually, but the developed model was able to detect all the nodes in the image with high confident scores (Figure 8B). In Figure 8C, the bamboo nodes were occluded by a bamboo leaf, and one of them was occluded by more than 50%. But they were all successfully detected. In images taken at a high shooting distance condition (Figure 8D), the bamboo nodes accounted for a small proportion of pixels in the image, which would be difficult to be recognized. Nevertheless, every bamboo node was recognized with a high confidence score.

Among the 400 images, a total of 1385 bamboo nodes were manually counted from the captured images by two people. With the same 400 images, the lucky bamboo node detection model detected 1376 true positives, 84 false positives, and 9 false negatives. Comparing two sets of results, the lucky bamboo node detection model was in good agreement with the manual counting, as indicated

by the low errors, for example, 0.346 for RMSE. The minor discrepancies between the lucky bamboo node detection model and the manual counting could attribute to the following reasons. When training the CNN model, the white mottled rings of lucky bamboo (Figure 9A), the blocky mottled patches (Figure 9B), and the blurred bamboo body with scratches (Figure 9C) were not labeled when manually annotating the bamboo nodes. This meant that the model was not trained for these cases. This may lead to false positives (FP) in the detection. Avoiding these cases would improve the accuracy of bamboo node detection. In some cases, dry bamboo leaves that were not completely removed severely blocked the bamboo nodes, causing the CNN model to mistakenly classify the nearby nodes as background (Figure 9D). All these would explain the higher nodes counts of the bamboo node detection model. The resulting discrepancies could be minimized by labeling white mottled rings, blocky mottled patches, and blurred bamboo body with scratches as additional categories during image preprocessing, which would be explored in future studies. Overall, the low error showed that the

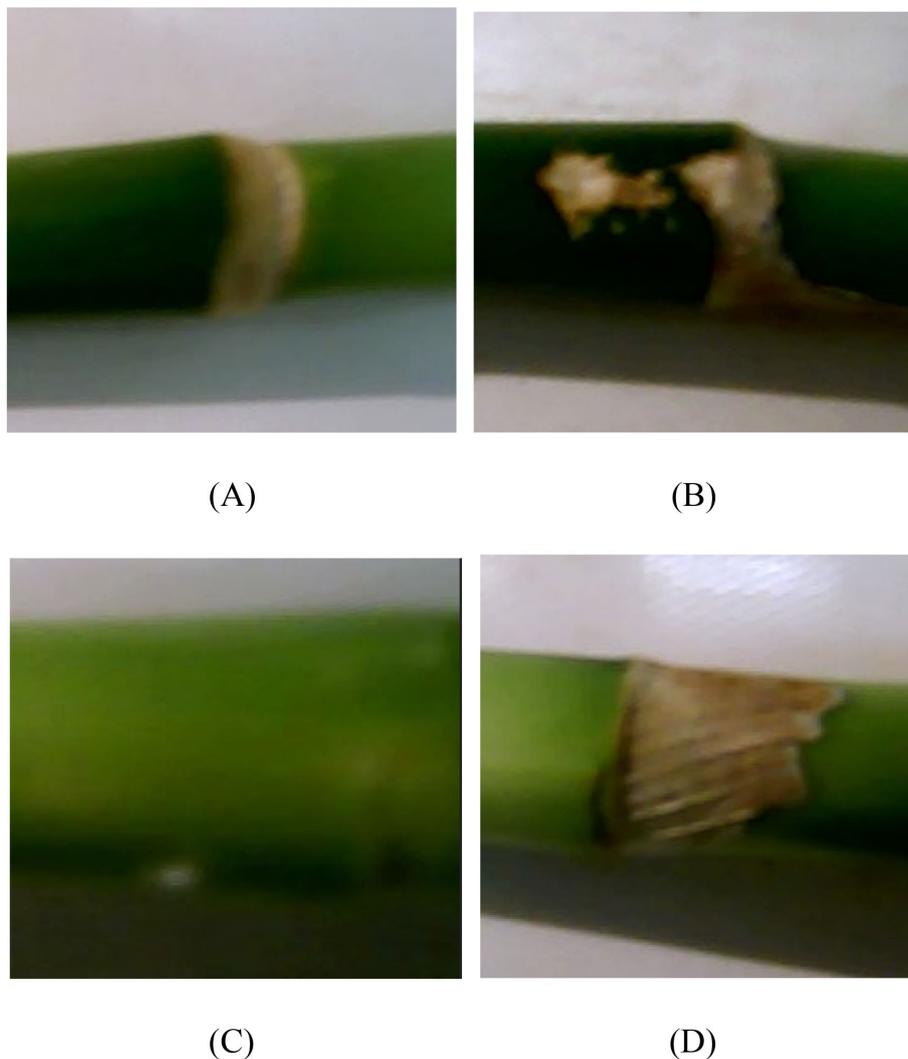


FIGURE 9

Examples of images which might have caused bamboo nodes detection errors. (A) blurred conditions, (B) similarities of bamboo spot and node, (C) sick bamboo node, (D) dried bamboo leaf.

model could successfully detect most of the bamboo nodes under various environmental conditions.

### 3.3 Ablation experiment

To further verify that the bamboo node detection model with added SE attention mechanism and WIoU loss function performs better than the original YOLOv7 model, four ablation experiments were conducted. The same data set and the same training parameters and methods were used to complete the training of each set of experiments. The experimental results are shown in Table 2, where “×” represents that the corresponding improvement strategy is not used in the network model, and “√” represents that the improvement strategy is used. Among them, Model 1 is the original YOLOv7 model, and model 2, model 3 and model 4 are models that add the SE attention mechanism, WIoU loss function, and SE attention mechanism-WIoU loss function to the original YOLOv7 model, respectively. The results showed that the detection accuracy (mAP) of the model 1 without any improvement strategy was 83.4%. Model 2, which introduced the SE

attention mechanism based on the original YOLOv7 model, embedded spatial position information into channel attention. This enabled the model to achieve better prediction results when detecting bamboo nodes that relied on position information, thereby improving the detection accuracy (mAP) by 15.5%. Model 3 introduced a new bounding box loss function MIOU to reduce the overlap error between the predicted box and the true box, thereby improving the detection accuracy and stability of the model. Therefore, the detection accuracy (mAP) of model3 was improved by 12.7%. Model 4 (i.e. our developed model), which introduced the SE attention mechanism and MIOU loss function based on the original model, optimized the prediction accuracy of the YOLOv7 model for the recognition and localization of bamboo nodes, thereby improving the model detection accuracy (mAP) by 14.2%. Over all the four pretrained models, the model 4 had the best mAP (97.6%).

To further evaluate the improved models, the P-R curves for the four models were plotted. The P-R curves showed that the model 4 had the highest precision consistently over the recall range of 0.90 to 1.00 (Figure 10). All these performance indicators proved that our proposed lucky bamboo node detection model was superior than the other three models.

TABLE 2 Results of ablation experiment.

Model name	Improvement strategies		mAP (%)
	SE-mechanism	WIoU	
Model 1	×	×	83.4
Model 2	√	×	98.9
Model 3	×	√	96.1
Proposed model	√	√	97.6

### 3.4 Comparison with other latest models

To better demonstrate the excellent performance of the proposed model, the proposed model was compared with the current mainstream YOLOv11 and YOLOv12 models. Compared to the baseline YOLOv7, our proposed model demonstrated comparable inference speed but achieved a significant 17.03% improvement in mAP (0.976). When benchmarked against Model 2, YOLOv11, and

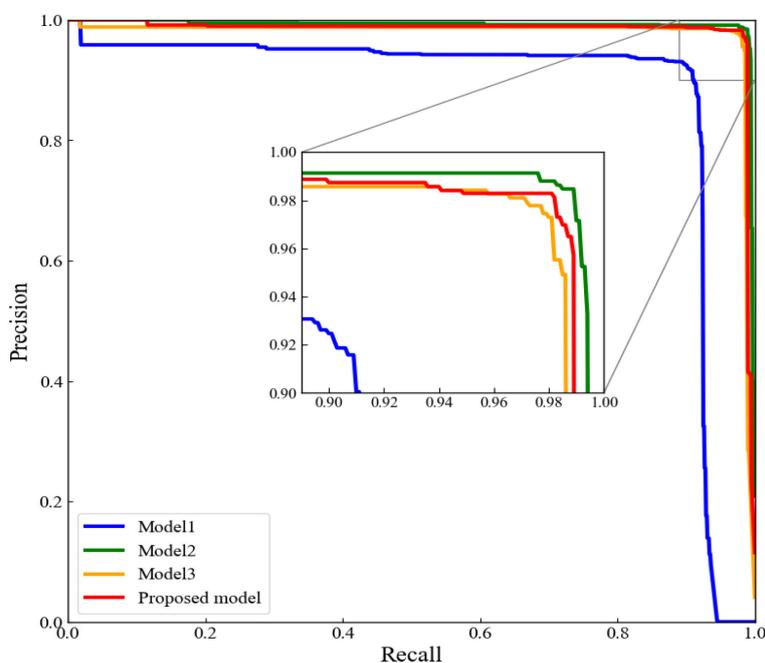


FIGURE 10 P-R curve of ablation experiment.

TABLE 3 Comparison results with other latest models.

Model name	Parameter quantity (MB)	Average inference time (ms)	Frames Per Second (FPS)	mAP@0.5 (%)
<b>Proposed model</b>	136.04	<b>9.98</b>	<b>100.18</b>	0.976
YOLOv7_original	135.52	9.98	100.18	0.834
Model 2	135.52	13.30	75.20	<b>0.989</b>
YOLOv11	<b>109.11</b>	14.12	70.8	0.988
YOLOv12	227.44	25.29	39.54	0.980

Bold values indicate the best performance achieved in each column (i.e., lowest parameter quantity, fastest inference time, highest FPS, and highest mAP@0.5).

YOLOv12, our proposed model showed moderate mAP reductions of 1.3%, 1.22%, and 0.41% respectively. However, it delivered substantial Frame Per Second (FPS) enhancements-increasing FPS from 75.20 to 100.18 against Model 2, 70.8 to 100.18 versus YOLOv11, and 39.54 to 100.18 compared to YOLOv12 (Table 3). These computational efficiency gains represent critical advantages for industrial deployment scenarios. Consequently, our proposed model exhibited reduced inference time and higher processing throughput, demonstrating particular advantages for video stream analysis and enhanced suitability for real-world deployment scenarios.

## 4 Conclusion

In this study, a high-precision and high-efficiency method was proposed based on a deep learning CNN mode for automatic detection of lucky bamboo node on the bamboo plant. Using the method, a high-throughput and low-cost application was developed and evaluated using lucky bamboo plant samples. The following conclusions were drawn. The CNN-based lucky bamboo node detection model was capable of recognizing and locating bamboo node in the lucky bamboo plant. The model was found to be the most efficient model for recognizing and locating bamboo nodes on the lucky bamboo plant structure. When compared to manual detection of bamboo node, the developed method had an estimated accuracy of 97.6%. The accuracy of the developed method was not affected by complex environment. The developed method shows great promise as a robust tool for computer-aided detecting of bamboo nodes on the lucky bamboo plant structure, which will help artisans rapidly and accurately identify lucky bamboo nodes to speed up the processing of lucky bamboo. However, more tests may be required to further verify the developed method.

## Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

## Author contributions

JZ: Conceptualization, Funding acquisition, Methodology, Project administration, Resources, Software, Supervision, Writing – original draft, Writing – review & editing. RD: Data curation, Formal Analysis, Software, Validation, Writing – original draft, Conceptualization,

Funding acquisition, Methodology, Project administration, Supervision, Writing – review & editing. CC: Formal Analysis, Software, Visualization, Writing – original draft. EZ: Data curation, Formal Analysis, Software, Writing – original draft, Visualization. HaL: Data curation, Formal Analysis, Funding acquisition, Writing – review & editing. MH: Data curation, Funding acquisition, Writing – review & editing. XC: Data curation, Validation, Writing – original draft. HuL: Writing – review & editing, Validation. ZW: Validation, Writing – review & editing.

## Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This research was funded by Guangdong Basic and Applied Basic Research Foundation (No. 2022A1515110468), the Science and Technology Projects of Zhanjiang (No. 2024B01044), the Innovation Team Project for Ordinary Universities in Guangdong Province (No. 2024KCXTD041), Program for scientific research startup funds of Guangdong Ocean University (No. 060302062106 and 060302062202), Natural Science Foundation of Guangdong Province (No. 2025A1515012901), and Guangdong Provincial Science and Technology Special Fund (No. A24333), Zhanjiang Key Laboratory of Modern Marine Fishery Equipment (No. 2021A05023).

## Acknowledgments

The authors thank their partners, Zhanjiang Key Laboratory of Modern Marine Fishery Equipment, for providing the platform for research.

## Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

## Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated

organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

## References

- Abdel-Rahman, T. F., El-Morsy, S., and Halawa, A. (2020). Occurrence of stem and leaf spots on lucky bamboo (*Dracaena sanderiana* hort. ex. mast.) plants in vase and its cure with safe means. *J. Plant Prot. Pathol.* 11, 705–713. doi: 10.21608/jppp.2020.170648
- Akinlabi, E. T., Anane-Fenin, K., and Akwada, D. R. (2017). Bamboo. *Multipurpose. Plant* 268, 262. doi: 10.1007/978-3-319-56808-9
- Amin, M., and Mujeeb, A. (2019). Callus induction and synthetic seed development in *Dracaena sanderiana* Sanderex Mast: Lucky Bamboo. *Biotechnol. J. Int.* 23, 1–8. doi: 10.9734/bji/2019/v23i330082
- Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv. preprint. arXiv:10934*. doi: 10.48550/arXiv.2004.10934
- Chen, G. (2012). *Landscape architecture: planting design illustrated* (Virginia, USA: ArchiteG, Inc).
- Chen, Y., Li, J., Xiao, H., Jin, X., Yan, S., and Feng, J. (2017). Dual path networks. *Adv. Neural Inf. Process. Syst.* 30, 1–11. doi: 10.48550/arXiv.1707.01629
- Cho, Y.-J. (2024). Weighted intersection over union (wIoU): a new evaluation metric for image segmentation. *Pattern Recognition Letters*. 185, 101–107. doi: 10.1016/j.patrec.2024.07.011
- Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. (2019). "Autoaugment: Learning augmentation strategies from data," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. (California, USA). 113–123.
- Darreln (2017). Labeling: Labeling Is a Graphical Image Annotation Tool and Label Object Bounding Boxes in Images. Available online at: <https://github.com/tzutalin/labelImg> (Accessed 21 January 2025).
- Farhadi, A., and Redmon, J. (2018). "Yolov3: An incremental improvement," in *Computer vision and pattern recognition* (Springer Berlin, Heidelberg, Germany), 1–6.
- Gergel, S. E., and Turner, M. G. (2017). *Learning landscape ecology: a practical guide to concepts and techniques* (New York, USA: Springer).
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*. (Venice, Italy) 2961–2969.
- Hong, S., Jiang, Z., Zhu, J., Rao, Y., Zhang, W., and Gao, J. (2022). A deep learning-based system for monitoring the number and height growth rates of moso bamboo shoots. *Appl. Sci.* 12, 7389. doi: 10.3390/app12157389
- Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Utah, USA) 7132–7141.
- Juyal, P., Kulshrestha, C., Sharma, S., and Ghanshala, T. (2020). Common bamboo species identification using machine learning and deep learning algorithms. *Int. J. Innovative. Technol. Explor. Eng.* 9, 3012–3017. doi: 10.35940/ijitee.D1609.029420
- Kumar, K., Sharma, S., Pandey, P., and Goyal, H. R. (2022). "Identification of various bamboo diseases using deep learning approach," in *2022 IEEE Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*. 1–6 (Gwalior, India: IEEE).
- Li, J., Liang, X., Shen, S., Xu, T., Feng, J., and Yan, S. (2017). Scale-aware fast R-CNN for pedestrian detection. *IEEE Trans. Multimedia*. 20, 985–996. doi: 10.1109/TMM.2017.2759508
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "Ssd: Single shot multibox detector," in *Computer Vision—ECCV 2016: 14th European Conference*, Amsterdam, The Netherlands, October 11–14, 2016, Vol. 14. 21–37 (Amsterdam, Netherlands: Springer), Proceedings, Part I.
- Liu, Y., Lu, J., and Zhou, Y. (2019). First report of *Colletotrichum truncatum* causing anthracnose of lucky bamboo in Zhanjiang, China. *Plant Dis.* 103, 2947. doi: 10.1094/PDIS-05-19-1122-PDN
- Niu, Z., Zhong, G., and Yu, H. (2021). A review on the attention mechanism of deep learning. *Neurocomputing* 452, 48–62. doi: 10.1016/j.neucom.2021.03.091
- Pankaja, K., and Thippeswamy, G. (2017). "Survey on leaf recognition and classification," in *2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*. 442–450 (Bangalore, India: IEEE).
- Redmon, J. (2016). "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. (Nevada, USA).
- Redmon, J., and Farhadi, A. (2017). "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7263–7271. (Hawaii, USA).
- Rezatofghi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., and Savarese, S. (2019). "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 658–666. (California, USA).
- Rout, G., Mohapatra, A., and Jain, S. M. (2006). Tissue culture of ornamental pot plant: A critical review on present scenario and future prospects. *Biotechnol. Adv.* 24, 531–560. doi: 10.1016/j.biotechadv.2006.05.001
- Salvador, A., Giró-i-Nieto, X., Marqués, F., and Satoh, S.i. (2016). "Faster r-cnn features for instance search," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. (Nevada, USA). 9–16.
- Sawarkar, A. D., Shrimankar, D. D., Ali, S., Agrahari, A., and Singh, L. (2024). Bamboo plant classification using deep transfer learning with a majority multiclass voting algorithm. *Appl. Sci.* 14, 1023. doi: 10.3390/app14031023
- Sharma, K., Merritt, J. L., Palmateer, A., Goss, E., Smith, M., Schubert, T., et al. (2014). Isolation, characterization, and management of *Colletotrichum* spp. causing anthracnose on lucky bamboo (*Dracaena sanderiana*). *HortScience* 49, 453–459. doi: 10.21273/HORTSCI.49.4.453
- Shorten, C., and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *J. Big. Data* 6, 1–48. doi: 10.1186/s40537-019-0197-0
- Tian, Z., Chu, X., Wang, X., Wei, X., and Shen, C. (2022). Fully convolutional one-stage 3d object detection on lidar range images. *Adv. Neural Inf. Process. Syst.* 35, 34899–34911. doi: 10.48550/arXiv.2205.13764
- Tong, Z., Chen, Y., Xu, Z., and Yu, R. (2023). Wise-IoU: bounding box regression loss with dynamic focusing mechanism. *arXiv. preprint. arXiv:10051*. doi: 10.48550/arXiv.2301.10051
- van Dam, J. E., Elbersen, H. W., and Montaña, C. M. D. (2018). "Bamboo production for industrial utilization," in *Perennial grasses for bioenergy bioproducts*, 175–216. (Amsterdam, Netherlands: Academic Press, Elsevier)
- Wang, C.-Y., Bochkovskiy, A., and Liao, H.-Y. M. (2023). "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7464–7475. (Vancouver, Canada).
- Wang, Z., Li, J., Yue, X., Yi, W., Wang, H., Chen, H., et al. (2022). A phenomic approach of bamboo species identification using deep learning. *Appl. Sci.* 12 (3), 1023. doi: 10.21203/rs.3.rs-1644335/v1
- Watanabe, S., Sumi, K., and Ise, T. (2020). Identifying the vegetation type in Google Earth images using a convolutional neural network: a case study for Japanese bamboo forests. *BMC Ecol.* 20, 1–14. doi: 10.1186/s12898-020-00331-5