Detection of citrus in the natural environment using Dense-TRU-YOLO

Taixiong Zheng, Yilin Zhu, Siyu Liu, Yongfu Li, Mingzhe Jiang*

(*School of Advanced Manufacturing Engineering. Chongqing University of Posts and Telecommunications,
Nan'an, Chongqing 400065, China*)

Abstract: Accurate detection of citrus in the natural orchard is crucial for citrus-picking robots. However, it has become a challenging task due to the influence of illumination, severe shading of branches and leaves, as well as overlapping of citrus. To this end, a Dense-TRU-YOLO model was proposed, which integrated the Denseblock with the Transformer and used UNet++network as the neck structure. First of all, the Denseblock structure was incorporated into YOLOv5, which added shallow semantic information to the deep part of the network and improved the flow of information and gradients. Secondly, the deepest Cross Stage Partial Connections (CSP) bottleneck with the 3 convolutions module of the backbone was replaced by the CSP Transformer with 3 convolutions module, which increased the semantic resolution and improved the detection accuracy of occlusion. Finally, the neck of the original network was replaced by the combined structure of UNet++ feature pyramid networks (UNet++-FPN), which not only added cross-weighted links between nodes with the same size but also enhanced the feature fusion ability between nodes with different sizes, making the regression of the network to the target boundary more accurate. Ablation experiments and comparison experiments showed that the Dense-TRU-YOLO can effectively improve the detection accuracy of citrus under severe occlusion and overlap. The overall accuracy, recall, mAP@0.5, and F1 were 90.8%, 87.6%, 90.5%, and 87.9%, respectively. The precision of Dense-TRU-YOLO was the highest, which was 3.9%, 6.45%, 1.9%, 7.4%, 3.3%, 4.9%, and 9.9% higher than that of the YOLOv5-s, YOLOv3, YOLOv5-n, YOLOv4-tiny, YOLOv4, YOLOX, and YOLOF, respectively. In addition, the reasoning speed was 9.2 ms, 1.7 ms, 10.5 ms, and 2.3 ms faster than that of YOLOv3, YOLOv5-n, YOLOv4, and YOLOX. Dense TRU-YOLO is designed to enhance the accuracy of fruit recognition in natural settings and boost the detection capabilities for small targets at extended ranges.

Keywords: citrus, picking robot, Dense-TRU-YOLO, Denseblock, UNet++-FPN
DOI: 10.25165/j.ijabe.20251801.8866

Citation: Zheng T X, Zhu Y L, Liu S Y, Li Y F, Jiang M Z. Detection of citrus in the natural environment using Dense-TRU-YOLO. Int J Agric & Biol Eng, 2025; 18(1): 260–266.

1 Introduction

Citrus is the fruit with the largest cultivated area, the highest yield, and the largest consumption in China[1,2]. However, traditional labor-intensive manual harvesting is not only inefficient but also costly. Investigation shows that manually picking accounts for 33%-50% of the entire production cost[3-5]. Therefore, it is urgent to develop citrus-harvesting robots to cope with the increasing labor cost. Relevant studies have shown that accurate detection of citrus in complex natural environments is a key factor affecting harvesting efficiency. However, it becomes difficult in such an environment with severe occlusion and complex illumination.

Fruit detection based on traditional methods has obtained fruitful achievements in the past decades. Gan et al.[6] proposed a new Color-Thermal Combined Probability (CTCP) algorithm, which can effectively detect citrus by fusing color and thermal information. Lu et al.[7] put forward a hierarchical contour analysis (HCA) algorithm based on the light distribution on the fruit surface. Zhao et al.[8] proposed a sum of absolute transformed difference (SATD) algorithm based on the color features of images and absolute transformation. However, the traditional method was criticized for its cumbersome and manual feature extraction. Moreover, its robustness was insufficient due to environmental factors such as occlusion and illumination. With the development of artificial intelligence technology, the deep learning method represented by Deep Convolution Neural Network (DCNN) performed higher detection accuracy and faster detection speed than traditional methods in the field of target recognition. DCNN can be divided into two categories. One is the single-stage model represented by YOLO[9] and Single Shot MultiBox Detector (SSD)[10]. The other is the two-stage model represented by Fast R-CNN (Region-CNN)[11], Faster R-CNN[12], and Mask R-CNN[13]. DCNN has achieved great success in the field of fruit detection. Liu[14] proposed YOLO-tomato, which replaced R-Bbox with C-Bbox, thus improving the calculation method of NMS and IOU. Yang[15] put forward a combined fruit and branch recognition algorithm based on Mask R-CNN with recognition accuracy of fruit and branch of 88% and 96%, respectively. Research shows that compared to two-stage networks, the advantages of single-stage networks in computational efficiency make them more suitable for fruit detection. As the state of the art of You Only Look Once (YOLO), YOLOv5 attracted widespread attention because of its higher detection accuracy, faster speed, and smaller model[16-18]. Although the detection accuracy of YOLOv5 has been widely proven, it still struggles in the natural environment with severe occlusion, overlap, and so on. This is mainly because traditional convolution operations mainly focus on local regions, making it

Received date: 2024-02-17 **Accepted date:** 2024-11-03

Biographies: Taixiong Zheng, PhD, Professor, research interest: intelligent perception and control, Email: zhengtx@cqupt.edu.cn; **Yilin Zhu**, MS, research interest: computer vision, Email: S212131031@stu.cqupt.edu.cn; **Siyu Liu**, MS, research interest: computer vision, Email: 809839219@qq.com; **Yongfu Li**, PhD, Professor, research interest: intelligent vehicle environment awareness, Email: liyf@cqupt.edu.cn.

***Corresponding author: Mingzhe Jiang**, MS, Teaching assistant, research interest: machine vision. Chongqing University of Posts and Telecommunications, No. 2 Chongwen Road, Nan'an District, Chongqing. Tel: +86-18983841471, Email: jiangmz@cqupt.edu.cn.

difficult to capture long-distance dependencies. The purpose of this study was to further optimize YOLOv5 to improve its detection accuracy in complex environments. To this end, a Dense-TRU-YOLO (Network with Dense Transformer Structure and UNet++ Feature Pyramid) model based on YOLOv5 was proposed. In the framework, Denseblock, Transformer, and UNet++ (Nested U-Net) were organically incorporated into the model, which comprehensively improved the performance of the model. The expected contributions of this study are as follows:

Firstly, different from the backbone of the detection network in the existing literature, the Denseblock module was incorporated into the backbone of YOLOv5 to build Dense-CSPDarknet53, which promotes feature reuse and reduces semantic information loss.

Secondly, the self-attention mechanism was combined with the Cross Stage Partial Network (C3) module to construct the Cross Stage Partial Network based on the Transformer (C3TR) module, and then replaced the deepest C3 with C3TR to establish semantic relationships between different image blocks, thereby effectively capturing the dependency relationships between images.

Finally, different from existing literature, the neck of YOLOv5 was replaced by the combined structure of UNet++ and FPN (Feature Pyramid Network) to improve the fusion ability of the features with different sizes.

The research objective of this article is to address the problem of fruit target detection in natural environments. Most models have low detection accuracy for overlapping fruits, fruits obstructed by tree branches, and fruits with small targets at long distances. Dense TRU-YOLO aims to improve the recognition accuracy of fruits in natural environments, enhance the detection ability of small targets at long distances, and meet the real-time detection requirements of fruit-picking robots in natural environments.

2 Materials and methods

2.1 Sample dataset collection and preprocessing

In this study, a Canon 80D camera (Japan) was used to take citrus images from 9 a.m. to 6 p.m. in Chongqing, China. A total of 1851 citrus images were captured at different distances with different illumination. Among them, 8367 citruses were with occlusion and overlap; the other 1408 citruses were unobstructed. Furthermore, in order to improve the generalization and robustness of the proposed model, random data augmentation was used to obtain 11 106 images. For subsequent comparative experiments, the cases of fruit occlusion were divided into three categories: unobstructed, slightly occluded (occlusion rate less than 50%), and severely occluded (occlusion rate more than 50%). The dataset was further divided into a training set and a verification set, respectively, according to the ratio of 9:1.

2.2 Dense-TRU-YOLO model

The Dense-TRU-YOLO network structure was composed of a backbone, neck, and detection part. The backbone consisted of Dense-CSPDarknet53, C3TR, and SPP modules. The neck generated a feature pyramid by UNet++-FPN, a combined structure of UNet++[19] and FPN, which can fuse different levels of feature maps to obtain more context information and generate feature maps with different sizes. In the detection part, an anchor box was applied to generate a detection box, indicating the category, coordinates, and confidence. The structure of Dense-TRU-YOLO is shown in Figure 1.

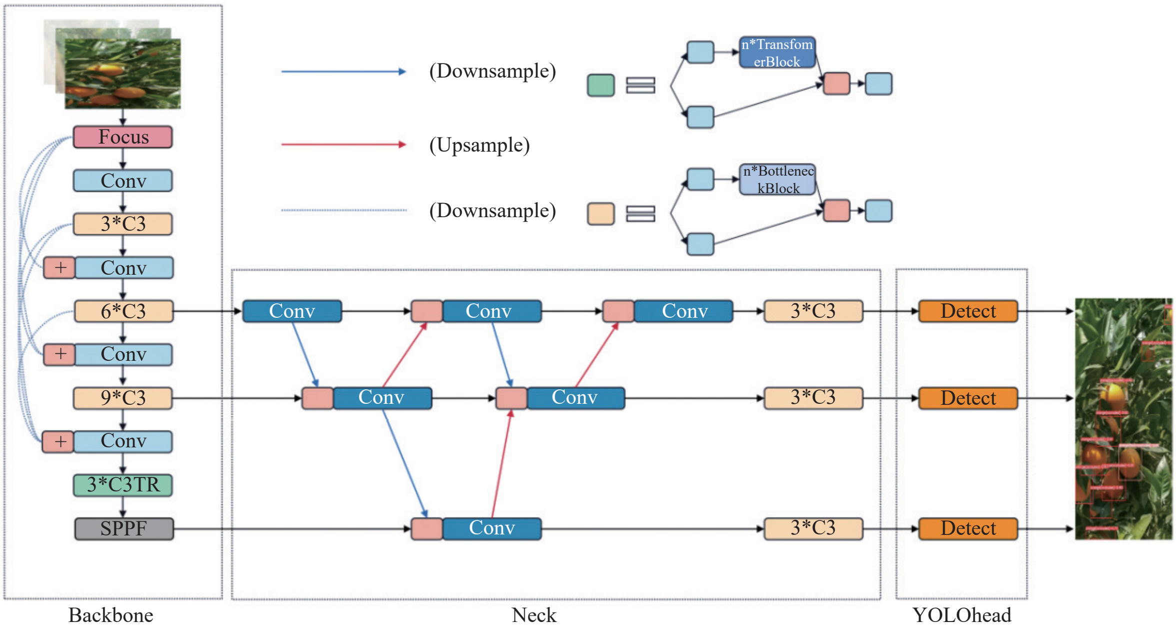


Figure 1 Structure of Dense-TRU-YOLO

2.2.1 Dense-CSPDarknet53 module

With the deepening of CNN, some features and gradients may vanish after multiple downsampling, resulting in the difficulty of convergence of the network[16]. In Denseblock, the output of each convolutional layer was used as input for subsequent layers, which can effectively alleviate the above problems[20]. Therefore, the Denseblcok module was incorporated into the backbone to form the Dense-CSPDarknet53, which contained Focus and CSP, as shown in Figure 2.

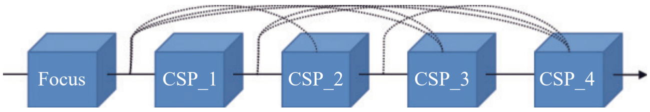


Figure 2 Dense-CSPDarknet53 structure

In Dense-CSPDarknet53 concatenation was used to fuse the residual structure of the first, third, and fifth layers after downsampling. Furthermore, a 1×1 convolution was used to extract the fused image features. Finally, the input of the eighth layer was obtained.

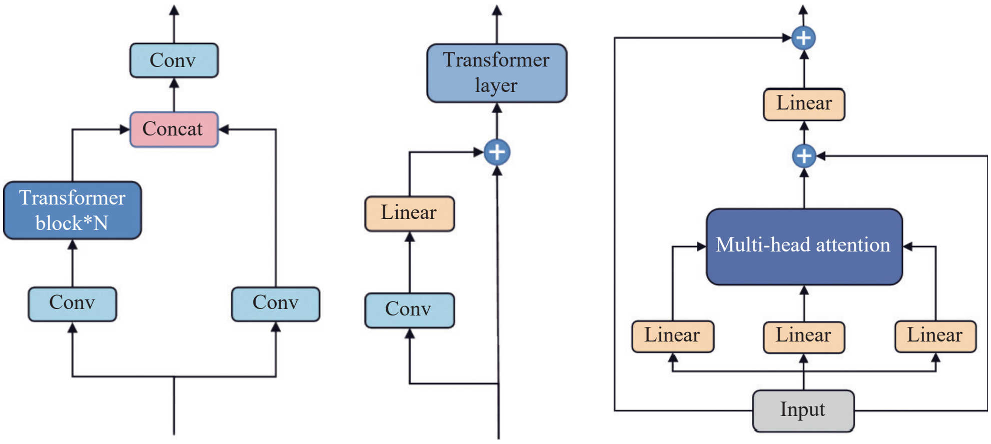
$$P_8^{input} = Conv(D(x_1) + D(x_3) + D(x_5) + x_7) \tag{1}$$

where, P_8^{input} is the input of the eighth layers, Conv is 1×1 convolution, D is the downsampling, and x_i is the output of the i layer.

2.2.2 Application of Transformer in the backbone

The YOLOv5 backbone network can effectively capture local information by using multi-layer 3×3 convolution[21]. However, due to the influence of environmental interference, occlusion, and fruit overlap, it is very important to establish global semantic information for citrus detection. Attention mechanisms allow the modeling of dependencies without regard to their distance in the input or output sequences, providing a perfect solution to the above problems. There is a consensus that using an attention mechanism can effectively improve the performance of neural networks. For example, Han et al.[22] integrated the ECA-Net attention mechanism into the backbone of YOLOv5 to address the problem of low accuracy of detection of overhead line insulators due to complex backgrounds, small targets, and overlapping targets. In 2017, the Google brain team proposed Transformer[23-26], which takes self-attention as a layer in the network structure. Transformers can pay attention to the global dependency between image feature patches, and reserve enough spatial information to detect image features through a multi-head self-attention mechanism[27]. To the best of our knowledge, the Transformer is the first transduction model relying entirely on self-attention to compute representations of its input and output without using sequence-aligned RNNs or convolution.

In visual applications, a simple way to use self-attention was to replace spatial convolution with the multi-head self-attention mechanism. Motivated by Transformer[27], the C3TR module was constructed by replacing the Bottleneck Block in the C3 module with the Transformer Block, as shown in Figure 3a. In order to make the model pay more attention to semantically related image areas, the deepest C3 module of the backbone network was replaced by C3TR, so that the backbone network was combined with the multi-head attention mechanism. In the C3TR module, image features were input to two branches respectively. In the first branch, after 1×1 convolution, the image feature was transmitted to the Transformer Block module, as shown in Figure 3b. In the second branch, only 1×1 convolution was adopted. Subsequently, the two branches were fused by concat and 1×1 convolution was used to restore channels.



a. C3TR structure b. Transformer block structure c. Transformer layer structure

Figure 3 Detailed composition of C3TR

2.2.3 UNet++-FPN

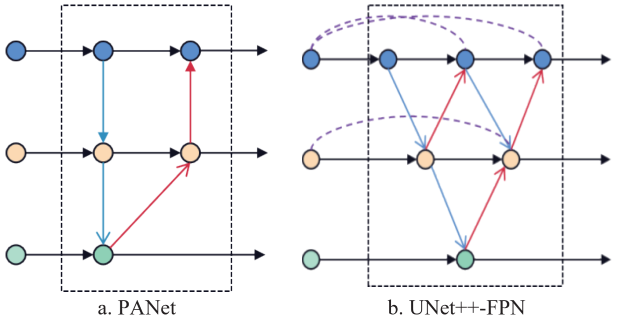
YOLOv5 used PANet[28] to create a bottom-up path enhancement and strengthen the semantic information fusion of each layer, as shown in Figure 4a. Inspired by the success of UNet++ and FPN[29], a combined structure named UNet++-FPN was proposed to replace PANet, as shown in Figure 4b, thus further strengthening the attention to the shallow semantics and fully integrating the image information of each layer. In particular, cross-layer weighted links were added between the original input and output nodes. Furthermore, upsampling and downsampling weighted feature fusion were used between nodes with different sizes.

While combining shallow information and deep information, UNet++-FPN introduced learnable weights to learn the importance of different input characteristics. The calculation of each node was:

$$x_j^i = \begin{cases} H(Concat(input, D(x^{i-1,j}))), & j = 0 \\ H(Concat[input, x^{i,j-1}, u(x^{i+1,j-1}), D(x^{i-1,j})]), & j \neq 0 \end{cases} \tag{2}$$

where, H is the convolution, D is the downsampling,$u$is the upsampling, x^{ij} is the node output, i indicates the number of sampling layers along the bottom, and j represents the convolution layer of dense blocks along the hop index[30].

To meet the accuracy and complexity requirements of the target detection, the structure can add branches or prune the model when deepening or reducing the pyramid depth. Table 1 lists the experimental accuracy comparison of two-layer, three-layer, and four-layer UNet++-FPN in this study. It can be seen that the three-layer UNet++-FPN maintained the same mAP@0.5 as the four-layer structure while reducing the number of parameters by more than 316 000. Therefore, the three-layer UNet++-FPN was selected as the neck of the model.



a. PANet b. UNet++-FPN

Figure 4 PANet and UNet++-FPN

Table 1 Comparison of UNet++-FPN effects of different layers

Detection method	Number of parameters	mAP@0.5
2L- UNet++-FPN	12 893 808	0.882
3L-UNet++-FPN	13 871 359	0.905
4L-UNet++-FPN	14 187 668	0.905

3 Experiment results and discussion

3.1 Experimental platform and model training settings

In this study, model training and testing were conducted on a graphic workstation configured as listed in Table 2.

Table 2 Graphics workstation configuration

Configuration	Parameter
Operating system	Windows10
CPU	Intel(R) Core(TM) i7-10700
Memory	32 GB
GPU	NVIDIA GeForce RTX 3060Ti 12 GB
GPU acceleration library	CUDA11.2+Cudnn8.04
Framework version	Torch1.8.0+Python3.8

In addition, the SGD algorithm with a momentum factor of 0.937 and weight attenuation coefficient of 0.0005 was used to optimize the weight, offset, and BN in the model. Furthermore, the warm-up method was used to preheat the learning rate. The specific parameters were as follows: preheat epoch was 3 and momentum factor was 0.8. Then the cosine annealing strategy was used to dynamically adjust the learning rate. The image size for training and testing was 640×640 pixels, the batch size was 16, and the maximum training epoch was 300.

3.2 Model test and evaluation

P, *R*, F1, mAP@0.5, model reasoning speed, and model size were selected as the evaluation indices.

$$\text{mAP} = \frac{1}{n}\sum AP \tag{3}$$

$$\text{AP} = \int_{0}^{1} P(R)\text{d}R \tag{4}$$

$$P = \frac{\text{TP}}{\text{FP}+\text{TP}} \tag{5}$$

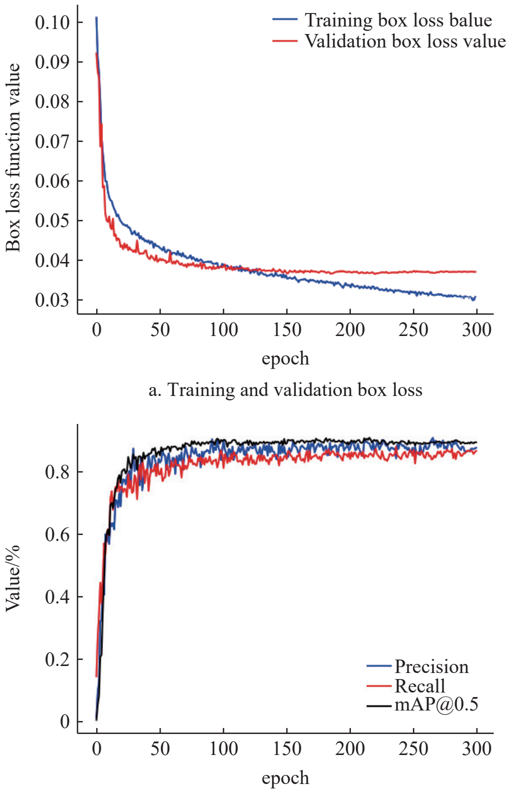
$$R = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{6}$$

where, *P* represents the accuracy rate, *R* represents the recall rate, TP represents the number of positive samples correctly predicted, AP represents the average precision, FP represents the number of negative samples detected as positive samples, and FN represents the number of positive samples detected as negative samples.

3.3 Analysis of experimental results

According to the specified super parameters, the proposed model was trained. The loss value of the training process is shown in n Figure 5a, and accuracy, recall rate, and mAP@0.5 are shown in n Figure 5b. It can be seen that the loss value tended to be stable after 250 epochs. Therefore, the model obtained after 300 epochs of training was determined as the citrus detection model. Figure 5b suggests that the proposed model achieved satisfactory results in accuracy, recall, and mAP@0.5.

Citrus detection in the case of occlusion and overlap is a major difficulty for the fruit-picking robot. Therefore, the detection accuracy of citrus in an occluded environment was an important evaluation index. To this end, single-fruit and multiple-fruit citrus detection experiments were carried out under the conditions of no obstruction, slight occlusion, and severe occlusion, and the statistics of the experimental results are listed in Table 3.



a. Training and validation box loss

b. Precision and recall and mAP@0.5 value

Figure 5 Network training results of training and validation box loss and precision and recall and mAP@0.5 value

Table 3 Model occlusion target detection effect

Item	No occlusion		Slight occlusion		Severe occlusion	
	Single fruit	Multiple fruits	Single fruit	Multiple fruits	Single fruit	Multiple fruits
Number of samples	25	25	50	50	50	50
Recognition rate/%	100	100	100	98	100	94

As can be seen from Table 3, the detection rate of a single fruit without occlusion can reach 100%. Excellent detection results were also shown in the case of multiple fruits, with a detection accuracy of 100% for a single fruit and 98% for multiple fruits. Even in the case of severe occlusion, the detection rate of multiple fruits was 94% and the detection rate of single fruit even reached 100%.

Furthermore, a comparative experiment with YOLOv5, YOLOv3, YOLOv4-tiny, YOLOv4, YOLOv5-s, YOLOX, and YOLOF on the same test datasets was conducted. As shown in Table 4, the accuracy of Dense-TRU-YOLO was the highest, which was 4.2%, 5.8%, 1.9%, 7.4%, 3.2%, 4.9%, and 9.9% higher than that of YOLOv5, YOLOv3, YOLOv5-n, YOLOv4-tiny, YOLOv4, YOLOX, and YOLOF, respectively. In other words, the innovation effectively improved the detection accuracy. In addition, the reasoning speed was 9.2 ms, 1.7 ms, 10.5 ms, and 2.3 ms faster than that of YOLOv3, YOLOv5-n, YOLOv4, and YOLOX, respectively. Compared with allied models, Dense-TRU-YOLO showed competitive accuracy while maintaining faster reasoning speed and lower parameters, which shows that the model can meet the requirements of overlapping and occluded citrus fruit detection.

Table 4 Comparison experiment with mainstream models

Network model	Type	*P*/%	*R*/%	F1	mAP/%	Time/ms	Size of model/MB
YOLOv3	Occluded	77.5	84.8	0.81	84.7	17.6	236.0
	Unobstructed	76.1	78.2	0.77			
YOLOv5-n	Occluded	86.7	82.4	0.84	88.6	10.1	42.1
	Unobstructed	87.7	85.1	0.86			
YOLOv4-tiny	Occluded	79.9	75.2	0.77	83.1	6.4	23.0
	Unobstructed	83.6	78.4	0.81			
YOLOv4	Occluded	83.9	80.5	0.82	87.3	18.9	245.0
	Unobstructed	85.9	86.6	0.86			
YOLOv5-s	Occluded	88.8	74.2	0.81	86.3	8.2	14.0
	Unobstructed	90.3	85.1	0.87			
YOLOX	Occluded	88.6	89.7	0.89	85.6	10.7	68.5
	Unobstructed	85.6	82.4	0.84			
YOLOF	Occluded	83.2	79.8	0.81	80.6	/	483.0
	Unobstructed	88.4	85.5	0.87			
Model proposed in this study	Occluded	89.8	84.2	0.87	90.5	8.4	26.0
	unobstructed	91.2	90.5	0.91			

To demonstrate the effectiveness of the innovations in this study, three groups of ablation experiments were conducted on the basis of YOLOv5 with the same super parameters, training set, and test set. Ablation experiments are listed in Table 5.

In Table 5, "+" represents the introduced module. mAP@0.5 was used to measure the model performance. It can be seen that after replacing the C3 module with C3TR, mAP@0.5 increased by 0.8% and the parameter quantity only increased by 256, which proved that the Transformer module can enhance the feature extraction ability of the model with a small increase in network parameters and computational burden. Although the model parameters increased, mAP@0.5 increased by 2% after incorporating the Denseblock module, which proved that the introduction of the Denseblock module can improve the feature extraction and the regression ability to the target boundary at the expense of certain real-time. After replacing PANet with UNet++-FPN, mAP@0.5 further increased by 1.3%, which effectively improved the feature extraction capability of the model. The average inference time increased by 9.5 ms compared to the YOLOv5 model, which is completely acceptable in practical detection environments. In summary, all the innovations made in this study can effectively strengthen the performance of the model.

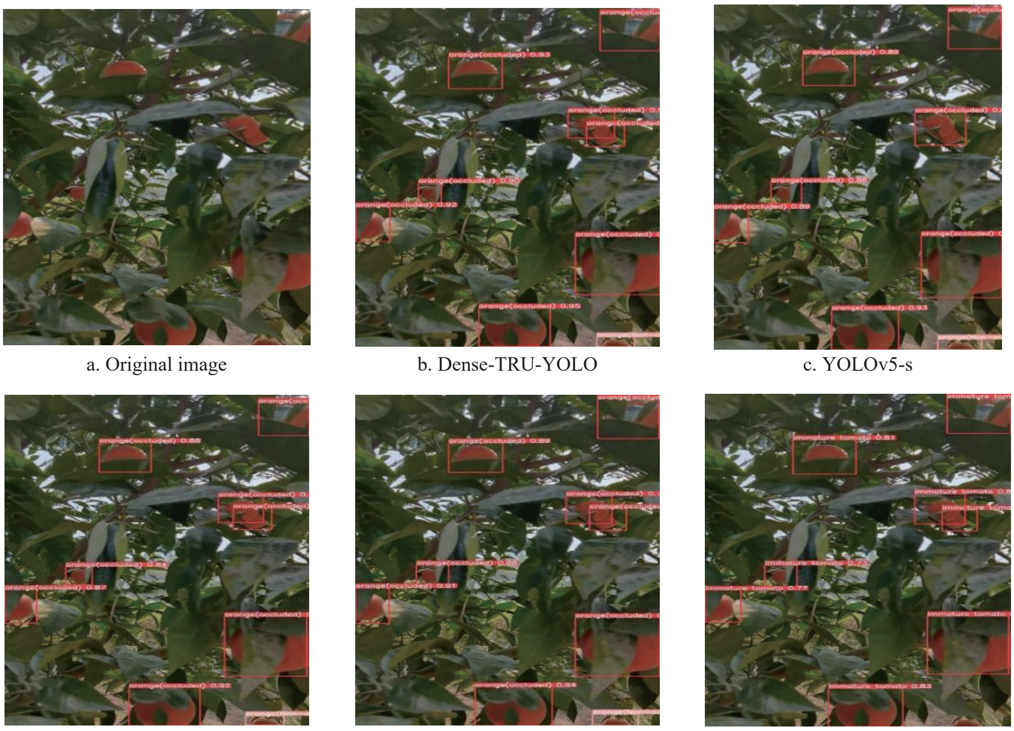
Table 5 Model structure ablation experiment

Methods	Model			
	YOLOv5			Model proposed in this study
C3TR	-	+	+	+
Denseblock	-	-	+	+
UNet++-FPN	-	-	-	+
mAP@0.5	0.864	0.872	0.892	0.905
Time/ms	28.0	25.0	35.0	37.5
Parameter	7 025 023	7 025 279	12 280 575	13 871 359

Note: "+" represented the introduced module.

3.4 Comparative test under different conditions

As is well-known, besides occlusion and overlap, illumination also affects citrus detection performance. Therefore, comparative experiments with YOLOv5-s, YOLOX, YOLOv4, and YOLOF were conducted under different illumination. The experiment results are shown in Figures 6-8.



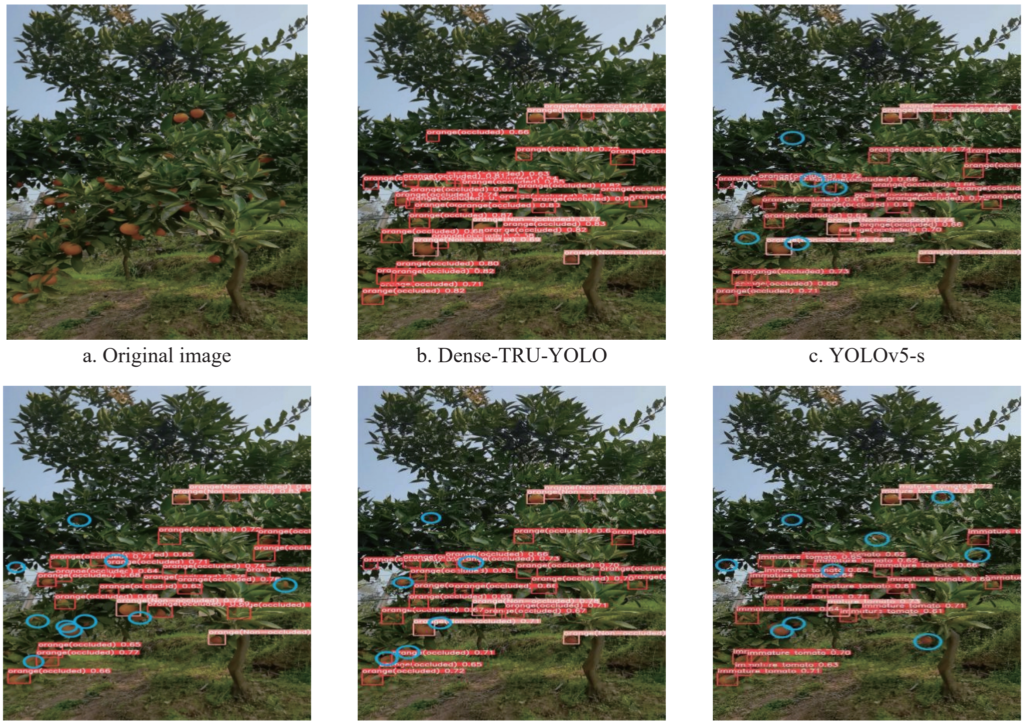
a. Original image b. Dense-TRU-YOLO c. YOLOv5-s

d. YOLOX e. YOLOv4 f. YOLOF

Figure 6 Citrus recognition effect under backlighting

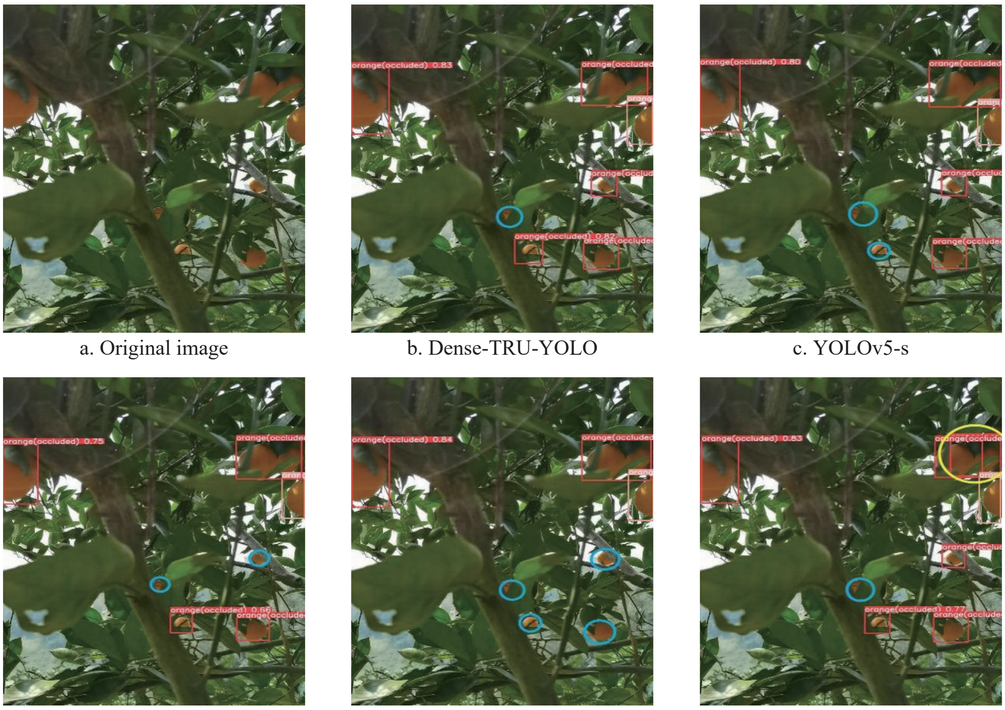
It can be seen from Figure 6 and Table 6 that although all five models can detect the citrus fruits in the case of backlighting, the detection confidence of other models was lower than that of Dense-TRU-YOLO. Moreover, the detection box of the comparative model also shifted due to the influence of leaf occlusion. In addition, YOLOv5-s experienced missed recognition when fruits overlapped, while the YOLOF model may also miss recognition when the image edges were incomplete. In other words, with the help of a self-attention mechanism, Dense-TRU-YOLO can effectively eliminate the influence of leaf occlusion.

As can be seen from Figure 7a, in the case of long distance, citrus looked relatively small and some citrus was seriously


a. Original image b. Dense-TRU-YOLO c. YOLOv5-s

d. YOLOX e. YOLOv4 f. YOLOF

Figure 7 Citrus recognition at long distance


a. Original image b. Dense-TRU-YOLO c. YOLOv5-s

d. YOLOX e. YOLOv4 f. YOLOF

Figure 8 Citrus recognition at close-range occlusion

Table 6 Confidence level detection under backlight conditions

Network model	Dense-TRU-YOLO	YOLOv5-s	YOLOX	YOLOv4	YOLOF
Average confidence	0.926	0.870	0.842	0.868	0.786

obstructed. Figure 7b shows the detection results of Dense-TRU-YOLO. Figure 7c, Figure 7d, Figure 7e, and Figure 7f show that YOLOv5-s, YOLOX, YOLOv4, and YOLOF all had missed detection, where the missed citrus fruits are marked with blue circles in the figure. As shown in the figures, even some citrus with less serious occlusion were not detected. In contrast, due to the introduction of Denseblock and attention mechanism, Dense-TRU-YOLO detected all citrus accurately with high confidence, even those with serious occlusion.

As can be seen from Figure 8a, there is a huge obstruction in front of the camera, causing the background to blur and most of the fruits to be obstructed. Figure 8c, Figure 8d, Figure 8e, and Figure 8f show that YOLOv5-s, YOLOX, YOLOv4, and YOLOF all had missed detection, where the missed citrus fruits are marked with blue circles in the figure, and misidentified fruits marked with yellow circles. Dense-TRU-YOLO detected most of the fruits without any misidentification.

4 Conclusions

The above results suggest that Dense-TRU-YOLO was superior to other algorithms in recognizing occluded and overlapping fruits under different environmental conditions. By introducing Denseblock and Transformer to YOLOv5, the Dense-TRU-YOLO model was proposed to detect citrus in orchards. The experiment results suggested that the overall accuracy, recall, mAP, and F1 of model recognition were 90.8%, 87.6%, 90.5%, and 87.9%, respectively. The average detection speed of each image was 8.4 ms. Compared with the current popular similar algorithms, Dense-TRU-YOLO had the fastest detection speed and the highest detection accuracy, which was 4.2%, 5.8%, 1.9%, 7.4%, 3.2%, 4.9%, and 9.9% higher than that of YOLOv5, YOLOv3, YOLOv5-n, YOLOv4-tiny, YOLOv4, YOLOX, and YOLOF, respectively. The experimental results show that Dense-TRU-YOLO can effectively identify citrus fruits with severe occlusion or overlap in various complex environments.nts.

Acknowledgments

This research is supported by the special research project of the Innovation and Development Center for Ideological and Political Work in Colleges and Universities (Wuhan Donghu University) under the Ministry of Education in 2024 (Practice Research on the Whole-process Fine Cultivation of New Engineering Talents with New Qualities in the Context of New Productive Forces, Grant No. WHDHSZZX2024085), the Humanities and Social Sciences Research project of the Chongqing Education Commission in 2024 (Theory and Practice Research on Digital Portraits Enabling Comprehensive Quality Evaluation of College Students, Grant No. 24SKGH100), and the general project of the “14th Five-Year Plan” for Education Science in Chongqing in 2024 (Research and Practice on the Construction of an Intelligent Recommendation Employment System for Person-Job Matching Enabled by Digital Portraits, Grant No. K24YG2060081).

Some of the datasets that were used and analyzed in this study have been uploaded to the website https://github.com/GDzhu01/Fruit-Dataset. All the homemade datasets in this study can be obtained by contacting the corresponding author.

[References]

[1] Ross J, Davis V, Foste C, Ray T. Agricultural Statistics. 2020. Available: http://www.nass.usda.gov. Accessed on [2023-05-14].

[2] Guo J, Gao Z, Xia J, Ritenour M A, Li G, Shan Y. Comparative analysis of chemical composition, antimicrobial and antioxidant activity of citrus essential oils from the main cultivated varieties in China. Lebensmittel-Wissenschaft & Technologie, 2018; 97: 825–839.

[3] Gonzalez-de-Santos P, Fernández R, Sepúlveda D, Navas E, Emmi L, Armada M. Field robots for intelligent farms - Inhering features from industry. Agronomy, 2020; 10(11): 1638.

[4] Mehta S S, MacKunis W, Burks T F. Robust visual servo control in the presence of fruit motion for robotic citrus harvesting. Computers and Electronics in Agriculture, 2016; 123: 362–375.

[5] Mehta S S, Burks T F. Vision-based control of robotic manipulator for citrus harvesting. Computers and Electronics in Agriculture, 2014; 102: 146–158.

[6] Gan H, Lee W S, Alchanatis V, Ehsani R, Schueller J K. Immature green citrus fruit detection using color and thermal images. Computers and Electronics in Agriculture, 2018; 152: 117–125.

[7] Lu J, Hu X W. Detecting green citrus fruit on trees in low light and complex background based on MSER and HCA. Transactions of the CSAE, 2017; 33(19): 196–201. (in Chinese)

[8] Zhao C Y, Lee W S, He D J. Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove. Computers and Electronics in Agriculture, 2016; 124: 243–253.

[9] Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016; pp.779–788. doi: 10.1109/CVPR.2016.91.

[10] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C Y, et al. SSD: Single shot multibox detector. In: Computer Vision - ECCV 2016. Lecture Notes in Computer Science, Springer, 2016; pp.21–37. doi: 10.1007/978-3-319-46448-0_2.

[11] Girshick R. Fast R-CNN. In: IEEE International Conference on Computer Vision, Santiago, 2015; pp.1440-1448. doi: 10.1109/ICCV.2015.169.

[12] Ren S Q, He K M, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017; 39(6): 1137–1149.

[13] He K M, Gkioxari G, Dollar P, Girshick R. Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020; 42(2): 386–397.

[14] Liu G X, Nouaze J C, Touko Mbouembe P L, Kim J H. YOLO-tomato: A robust algorithm for tomato detection based on YOLOv3. Sensors, 2020; 20(7): 2145.

[15] Yang C H, Xiong L Y, Wang Z, Wang Y, Shi G, Kuremot T, et al. Integrated detection of citrus fruits and branches using a convolutional neural network. Computers and Electronics in Agriculture, 2020; 174: 105469.

[16] Zheng T X, Jiang M Z, Li Y F, Feng M C. Research on tomato detection in natural environment based on RC-YOLOv4. Computers and Electronics in Agriculture, 2022; 198: 107029.

[17] Jocher G, Stoken A, Borovec J, NanoCode012, Stan C, Liu C Y, et al. ultralytics/yolov5: v3.1 - Bug fixes and performance improvements. 2020. Available: https://zenodo.org/records/4154370. Accessed on [2023-06-21].

[18] Yan B, Fan P, Lei X Y, Liu Z J, Yang F Z. A real-time apple targets detection method for picking robot based on improved YOLOv5. Remote Sensing, 2021; 13(9): 1619.

[19] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015. MICCAI 2015, Springer, 2015; pp.234–241. doi: 10.1007/978-3-319-24574-4_28.

[20] Huang G, Liu Z, Van Der Maaten L, Weinberger K Q. Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu: IEEE, 2017; pp.2261–2269. doi: 10.1109/CVPR.2017.243.

[21] Srinivas A, Lin T Y, Parmar N, Shlens J, Abbeel P, Vaswani A. Bottleneck transformers for visual recognition. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville: IEEE, 2021; pp.16514–16524. doi: 10.1109/CVPR46437.2021.01625.

[22] Han G J, He M, Gao M Z, Yu J Y, Liu K P, Qin L. Insulator breakage detection based on improved YOLOv5. Sustainability, 2022; 14(10): 6066.

[23] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, et al. Attention is all you need. In: NIPS’17: Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, 2017; 30(4): 6000–6010. doi: 10.5555/3295222.3295349.

[24] Devlin J, Chang M W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North, Minneapolis, Minnesota, 2019; 1: 1423.doi: 10.18653/v1/n19-1423.

[25] Brown T B, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, et al. Language models are few-shot learners. arXiv: Computation and Language, 2005; In Press. doi: 10.48550/arXiv.2005.14165.

[26] Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X H, Unterthiner T, et al. An image is worth 16×16 words: Transformers for image recognition at scale. arXiv:2010.11929, 2020; doi: 10.48550/arXiv.2010.11929.

[27] Zhang Z X, Lu X Q, Cao G J, Yang Y T, Jiao L C, Liu F. ViT-YOLO: Transformer-based YOLO for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal: IEEE, 2021; pp.2799–2808. doi: 10.1109/ICCVW54120.2021.00314.

[28] Liu S, Qi L, Qin H F, Shi J P, Jia J Y. Path aggregation network for instance segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018; pp.8759–8768. doi: 10.1109/CVPR.2018.00913.

[29] Lin T Y, Dollar P, Girshick R, He K M, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, 2017; pp.936–944. doi: 10.1109/CVPR.2017.106.

[30] Zhou Z W, Siddiquee M M R, Tajbakhsh N, Liang J M. UNet++: Redesigning skip connections to exploit multiscale features in image segmentation. IEEE Transactions on Medical Imaging, 2020; 39(6): 1856–1867.