An Improved YOLOX-Based Dragon Fruit Detection Method in Natural Environments (Postprint)

Authors: Shang Fengnan, Zhou Xuecheng, Liang Yingkai, Xiao Mingwei, Chen Qiao, Luo Chendi, Zhou Xuecheng

Date: 2023-02-17T00:00:00+00:00

Abstract

Precise detection of fruits in natural environments is a prerequisite for dragon fruit harvesting robots to perform harvesting operations. To improve the accuracy, robustness, and detection efficiency of fruit recognition in natural environments, this study improves the YOLOX (You Only Look Once X) network and proposes an object detection method incorporating an attention module. To facilitate deployment on embedded devices, this method uses the YOLOX-Nano network as a baseline and adds the Convolutional Block Attention Module (CBAM) to the backbone feature extraction network of YOLOX-Nano. By assigning weight coefficients to feature layers of different scales extracted by the backbone network, it learns the correlations among features in different channels, enhances the transmission of deep network information, and reduces interference from natural environment backgrounds on dragon fruit recognition. Performance evaluation and comparative experiments were conducted on this method. After training, the dragon fruit target detection network achieved an AP0.5 value of 98.9% and an AP0.5:0.95 value of 72.4% on the test set. Under the same experimental conditions, compared with other YOLO network models, this method' s average detection accuracy exceeds YOLOv3, YOLOv4-Tiny, and YOLOv5-S models by 26.2%, 9.8%, and 7.9%, respectively. Finally, real-time testing was conducted on videos collected in natural dragon fruit orchard environments at different resolutions. The experimental results show that the improved YOLOX-Nano target detection method proposed in this study has an average detection time of 21.72 ms per frame, an F1 value of 0.99, and a model size of only 3.76 MB. The detection speed, detection accuracy, and model size meet the technical requirements for dragon fruit harvesting in natural environments.

Full Text

Detection Method for Dragon Fruit in Natural Environment Based on Improved YOLOX

SHANG Fengnan[1,2,3], ZHOU Xuecheng[1,2,3]*, LIANG Yingkai[1,2,3], XIAO Mingwei[1,2,3], CHEN Qiao[1,2,3], LUO Chendi[1,2,3]

[1]College of Engineering, South China Agricultural University, Guangzhou 510642, China
[2]Guangdong Provincial Key Laboratory of Agricultural Artificial Intelligence, Guangzhou 510642, China
[3]Key Laboratory of Key Technology on Agricultural Machine and Equipment, Ministry of Education, Guangzhou 510642, China

Abstract: Accurate detection of dragon fruit in natural environments is a prerequisite for harvesting robots to perform picking operations. To improve the accuracy, robustness, and efficiency of fruit recognition under natural conditions, this study proposes an improved YOLOX (You Only Look Once X) network with an attention module for target detection. To facilitate deployment on embedded devices, the method uses YOLOX-Nano as the baseline and integrates a Convolutional Block Attention Module (CBAM) into the backbone feature extraction network. By assigning weight coefficients to feature layers of different scales extracted by the backbone network, the model learns correlations between features across different channels, enhances the transmission of deep-layer information, and reduces interference from natural background environments on dragon fruit recognition. Performance evaluation and comparative experiments demonstrate that after training, the proposed detection network achieves an $AP_{0·5}$ of 98.9% and an $AP_{0·5:0·95}$ of 72.4% on the test set. Under identical experimental conditions, the method surpasses other YOLO network models, with average detection precision exceeding YOLOv3, YOLOv4-Tiny, and YOLOv5-S by 26.2%, 9.8%, and 7.9%, respectively. Finally, real-time testing was conducted on videos captured in dragon fruit orchards under natural conditions at different resolutions. The results show that the improved YOLOX-Nano target detection method achieves an average detection time of 21.72 ms per frame with a model size of only 3.76 MB, meeting the technical requirements for dragon fruit harvesting in natural environments in terms of detection speed, accuracy, and model size.

Keywords: fruit picking; natural environment; dragon fruit; object detection; YOLOX; attention mechanism; deep learning

1 Introduction

Dragon fruit possesses high nutritional, medicinal, health, and economic value, and its cultivation industry has developed rapidly in China in recent years. By

2018, China had become the world's second-largest dragon fruit producer, with a cultivation area exceeding 50,000 hectares and production surpassing 1 million tons [1, 2]. However, as domestic labor shortages intensify and labor costs continue to rise, dragon fruit production expenses have increased significantly. Statistics indicate that fruit harvesting labor costs account for 35% to 40% of total production costs in China, while the mechanized harvesting rate remains merely 2.33% [3]. Therefore, achieving mechanization and automation of dragon fruit harvesting operations is crucial for improving production technology levels and promoting sustainable industry development. The primary challenge in automating dragon fruit harvesting lies in achieving accurate recognition and rapid detection of fruits in natural orchard environments.

Deep learning-based neural network object detection methods have been widely applied in agricultural product detection [4]. Jordi et al. [5] utilized the Mask R-CNN instance segmentation network for apple detection and segmentation. Liu et al. [6] deployed a lightweight improved YOLOv5 network on drones for on-tree apple detection. Ji et al. [7] improved the MobileNetV3-Small network using whale algorithm optimization, achieving 94.43% average precision for multi-class apple recognition. Mu et al. [8] enhanced the backbone feature extraction network of Faster R-CNN, attaining 94.75% average recognition accuracy for kiwifruit. In lychee detection research, Chen et al. [9] improved the YOLOv3 network for lychee bunch detection, achieving 94.3% mean average precision. Peng et al. [10] enhanced the SSD (Single Shot MultiBox Detector) network for small-target lychee detection from UAV images, improving average precision to 55.79%—approximately 30% higher than the original method. In mango detection research, Roy et al. [11] improved the YOLOv4 network for complex orchard environments, achieving 96.2% average detection precision. Xu et al. [12] proposed an improved lightweight YOLOv3 network for green mango detection in natural environments.

In tomato detection research, Xu et al. [13] proposed an improved Mask R-CNN method that increased recognition accuracy by 11.53%. Zhou et al. [14] optimized the VGGNet structure for detecting tomato fruits, flowers, and stems, achieving average detection precisions of 81.64%, 84.48%, and 53.94%, respectively. Zheng et al. [15] improved YOLOv4 to enhance tomato detection accuracy in natural environments, reaching 94.44% average precision. Zhao et al. [16] utilized an improved YOLOv3 network for identifying different tomato flowering stages and validated it on a pollination robot.

Currently, research on target recognition and detection methods for dragon fruit in natural environments remains limited, with no mature technical methods available for application. Li et al. [17] and Wang et al. [18] improved YOLOv3 and YOLOv4 networks, respectively, proposing lightweight convolutional networks MobileNet-YOLO and YOLOv4-Tiny for dragon fruit detection, achieving 96.48% average precision. However, these studies exhibit several technical limitations: the model training process is cumbersome, requiring manual anchor box construction; anchor boxes affect detection results; small, distant

dragon fruits are frequently missed; detection performance for small targets is unsatisfactory; and the methods are significantly influenced by environment and lighting conditions, resulting in low robustness.

To address these technical challenges and explore accurate recognition and rapid detection methods for mature dragon fruit in natural environments, this study proposes a dragon fruit target detection algorithm based on an improved YOLOX network with integrated attention mechanisms. This method demonstrates improved robustness in complex backgrounds and is suitable for dragon fruit detection in natural environments, providing valuable reference for rapid identification of other fruits under complex conditions.

2 Materials and Methods

2.1 Data Acquisition

Due to the lack of publicly available dragon fruit datasets, this study captured dragon fruit images in planting orchards in Zengcheng and Conghua districts of Guangzhou City. Image collection was conducted on July 17, 2021 (sunny day) and October 17, 2021 (overcast day). The imaging device was a CANON EOS M100 camera with imaging distances ranging from 50–150 cm, automatic exposure, image resolution of 2400$\times$1344 pixels, and storage in *.JPG format. A total of 1,976 raw images were collected, including multi-fruit whole-plant and single-fruit images under sunny front-lighting, sunny back-lighting, and overcast conditions (Figure 1 [Figure 1: see original paper]). To reduce interference from duplicate images and fruitless pictures during model training, manual screening was employed for data cleaning, yielding 1,744 raw images containing dragon fruit.

2.2 Dataset Preparation

The open-source LabelImg tool was used to manually annotate dragon fruits in images, creating a dataset following the Pascal VOC format with the label name "Dragon_{fruit}." The annotation principle involved manual labeling based on dragon fruit surface color information within the field of view. Specifically, fruits with red pixel area exceeding 90% of the epidermis were annotated, excluding green fruits. For occluded dragon fruits, the occluded region was estimated manually for annotation. Distant fruits with excessively small pixel areas ($<$20$\times$20 pixels) were not annotated.

2.3 Dataset Expansion

Dataset expansion effectively enhances sample diversity and ensures high model robustness across different environments. In this study, rotation, flipping, noise addition, and blurring operations were applied to expand the dataset to 5,232 images.

3 Recognition Network Model Construction

3.1 YOLOX Network Model

The YOLO (You Only Look Once) algorithm represents a leading one-stage object detection method. Due to its real-time detection speed and high accuracy, it has been widely applied in agricultural target detection and recognition. YOLOv1 through YOLOv5 are all anchor-based detectors requiring manual presetting of anchor box sizes. YOLOX, proposed by Ge et al. [19], is a single-stage object detection algorithm that integrates region prediction and category prediction into a single neural network model. It incorporates innovations including a faster-converging, higher-precision decoupled head, an anchor-free approach, and SimOTA (dynamic positive sample matching) [20-22], achieving high-precision rapid target detection and recognition. YOLOX offers multiple benchmark variants for different application scenarios, including standard versions like YOLOX-X, YOLOX-L, and YOLOX-Darknet53, as well as lightweight versions built with depthwise separable convolution.

3.1.1 Network Architecture YOLOX employs CSPDarkNet as the backbone feature extraction network. Input images undergo feature extraction in the backbone, producing feature layers that serve as the feature set for the input image. In CSPDarkNet, three effective feature layers are obtained and passed to the feature pyramid network. The feature pyramid network serves as YOLOX's enhanced feature extraction network, where the three effective feature layers from CSPDarkNet undergo feature fusion to obtain multi-scale feature information. YOLOX utilizes the Path Aggregation Network (PANet) structure from YOLOv4 [23]. PANet adds a bottom-up path aggregation module to the top-down Feature Pyramid Network (FPN) structure to better convey semantic and location information. PANet can shorten information paths and strengthen feature pyramids, which is particularly beneficial for YOLOX' s multi-scale detection, especially for small targets.

YOLOX' s decoupled head differs from previous YOLO versions, comprising a 1$×$1 convolutional layer to adjust channel numbers, followed by two parallel branches—each with two convolutional layers—dedicated to classification and regression tasks, respectively. An IoU (Intersection over Union) branch is added to the regression branch.

3.1.2 Loss Function The YOLOX loss function used in this study consists of three components, with total loss given by Equation (1):

$$L = L_{Reg} + L_{Cls} + L_{Obj}$$

where L represents total loss, L_{Reg} denotes localization error between predicted and ground-truth boxes, L_{Cls} represents object category probability loss, and

L_{Obj} denotes object confidence loss. Cls and Obj are calculated using binary cross-entropy loss functions. Equation (2) shows the IoU loss function:

$$Loss_{IoU} = 1 - IoU = 1 - \frac{|A \cap B|}{|A \cup B|}$$

where $Loss$ is the loss function, A represents the predicted bounding box location, and B represents the ground-truth bounding box location.

3.1.3 Activation Function YOLOX uses the smooth SiLU activation function, which has a lower bound but no upper bound, as shown in Equation (3):

$$SiLU(x) = x \cdot Sigmoid(x) = \frac{x}{1 + e^{-x}}$$

3.1.4 Anchor-Free Method Unlike previous YOLO networks using anchor-based approaches, YOLOX adopts an anchor-free strategy that avoids manual anchor design and parameter definition steps. The anchor-free method determines positive and negative samples based on whether the anchor box center point falls within the ground-truth box rectangle, effectively simplifying detector training and decoding processes, alleviating positive-negative sample imbalance, accelerating training, and contributing to precision improvements.

3.2 Improved YOLOX Network Model Design

The YOLOX backbone network contains multiple Cross Stage Partial (CSP) layers with numerous residual networks. CSP layers directly connect input features after minimal processing with output features from multiple residuals, creating large residual connections. While this operation effectively mitigates gradient vanishing issues in deep networks, it also transmits feature information along with noise to deeper network layers, adversely affecting backbone feature extraction. To address YOLOX' s limitations in natural environment dragon fruit detection, this study introduces the Convolutional Block Attention Module (CBAM) [24] into the backbone feature extraction network. CBAM structures are added after each of the three effective feature layers extracted by the backbone network, enabling the network to focus on important features while suppressing unnecessary ones. The improved YOLOX network structure is shown in Figure 2 [Figure 2: see original paper].

CBAM, proposed by Woo et al. [24], processes input feature layers through both channel and spatial attention mechanisms, enhancing the representational capacity of extracted features. Its structure is illustrated in Figure 3 [Figure 3: see original paper].

3.2.1 Channel Attention Mechanism The channel attention mechanism primarily focuses on meaningful information in input images. For an input feature layer of size $C \times H \times W$, average pooling and max pooling are performed to obtain two feature layers of size $C \times 1 \times 1$. These pass through a two-layer MLP (Multilayer Perceptron) with C/r neurons in the first layer using ReLU activation and C neurons in the second layer, where C is the input feature layer channel number and r is the reduction ratio. The two results are added and passed through a Sigmoid function to obtain weight values for each channel of the input feature layer. Finally, the obtained weight values are multiplied by the original input feature layer to produce new features. For an input feature F, the feature after channel attention is given by Equation (4):

$$F' = M_c(F) \otimes F$$

where F represents the input feature matrix, F' denotes the feature map output from the channel attention mechanism, and M_c is the channel compression weight matrix, with $\otimes$ representing element-wise multiplication.

3.2.2 Spatial Attention Mechanism The spatial attention mechanism primarily focuses on target location information. For the feature layer output from channel attention, max pooling and average pooling are performed to obtain two feature layers of size $1 \times H \times W$, which are then stacked. A 7×7 convolution with one output channel and a Sigmoid function produce weight values, which are multiplied by the input feature layer to obtain the final feature. For the channel attention output feature F', the feature after spatial attention F'' is given by Equation (5):

$$F'' = M_s(F') \otimes F'$$

where F'' represents the feature matrix output from the spatial attention mechanism and M_s is the spatial compression weight matrix.

3.3 Training Configuration

3.3.1 Experimental Platform The training platform used in this study was a desktop computer equipped with Windows 10 64-bit operating system, Intel i9-10900X @ 3.75 GHz CPU, NVIDIA GeForce GTX 3090 GPU, 128 GB RAM, PyTorch v1.7.1, CUDA v11.1, and CUDNN v8.0.2. Training and testing environments were identical.

3.3.2 Network Training The 5,232 images in the dataset were randomly divided into training, validation, and test sets at a ratio of 8:1:1, as detailed in Table 1 . Stochastic Gradient Descent (SGD) with linear scaling (lr × BatchSize/64) was used for optimization. The initial learning rate was set to 0.001

with cosine LR scheduling. Weight decay was set to 0.0005, and SGD momentum was 0.937. Official recommended pretrained weights were used for 300 epochs of training, with the first 50 epochs as frozen training (BatchSize = 64) and subsequent unfrozen training (BatchSize = 32). Since Mosaic augmentation can reduce training effectiveness when samples deviate from actual conditions, Mosaic and MixUp data augmentation were disabled for the final 30 epochs.

Table 1 Distribution of the dataset

Dataset	Images	Targets
Training	4,176	11,865
Validation	523	1,460
Test	523	1,601

3.4 Model Evaluation Metrics

The trained model was evaluated using F1-score, Recall, Precision, Average Precision (AP), detection speed (Frames per Second, FPS), average detection time, and model size. F1, Precision, and Recall are calculated as follows:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

where TP (True Positive) represents correctly segmented positive samples, FP (False Positive) represents incorrectly segmented positive samples, and FN (False Negative) represents incorrectly segmented negative samples. $F_{0.5}$ is the harmonic mean of precision and recall at IoU = 0.5.

AP is calculated using Equation (9):

$$AP = \int_0^1 Precision(Recall)\, dr$$

where r is the integration variable representing the product of recall and precision integrated over recall values. AP represents the area under the PR (Precision-Recall) curve, ranging from 0 to 1. $AP_{0.5}$ is the average precision at IoU = 0.5, while $AP_{0.5:0.95}$ calculates the mean AP across 10 IoU values from 0.5 to 0.95 with a step size of 0.05, providing a more comprehensive reflection of detection performance. AP_S, AP_M, and AP_L represent average precision for

small, medium, and large targets at IoU = 0.5, respectively, where small targets are defined as pixel area $< 32^2$, medium targets as $32^2 <$ Area $< 96^2$, and large targets as Area $> 96^2$.

4 Results and Analysis

4.1 Comparison of Different Input Resolutions

To analyze model convergence on the validation set, training was conducted and compared using two input resolutions (640$×640*pixelsand*416×$416 pixels). The loss curves for training and validation sets are shown in Figure 4 [Figure 4: see original paper]. The red and green curves represent training loss for high-resolution and low-resolution inputs, respectively. Both models exhibit similar convergence characteristics, with rapid loss reduction in early stages that gradually stabilizes as epochs increase. At approximately 250 epochs, the models converge, with the high-resolution curve achieving lower final loss values. The blue and yellow curves represent validation loss for high-resolution and low-resolution inputs, respectively, showing similar convergence patterns, with the high-resolution validation loss ultimately lower than the low-resolution version.

These results demonstrate that YOLOX-Nano networks with different resolution inputs both achieve good convergence, while high-resolution input makes dragon fruit features clearer and facilitates better feature learning, consistent with findings in similar literature [25]. Therefore, this study adopts 640$×$640 pixel input resolution as the final detection model.

4.2 Analysis of Different Model Sizes

Using identical training methods, three lightweight YOLOX network models (YOLOX-Nano, YOLOX-Tiny, and YOLOX-S) were trained and tested on the same test set to analyze performance differences. As shown in Table 2 , the three detection networks exhibit similar F1 values. At IoU = 0.5, YOLOX-Nano' s average precision is 2.3 and 1.6 percentage points higher than YOLOX-Tiny and YOLOX-S, respectively. YOLOX-Nano' s $AP_{0.5:0.95}$ reaches 70.2%, exceeding YOLOX-Tiny and YOLOX-S by 3.9 and 2.6 percentage points, respectively. The YOLOX-Nano model size is only 3.7 MB, substantially smaller than YOLOX-Tiny (19.4 MB) and YOLOX-S (34.3 MB). Notably, the smallest model (YOLOX-Nano) demonstrates optimal performance, likely because all three models maintained similar learning progress and optimization parameters during training with identical augmentation strategies (including the same epoch for disabling Mosaic and MixUp). However, appropriate augmentation strategies vary across model sizes, with larger models benefiting from stronger augmentation [19]. Therefore, to facilitate deployment on embedded and mobile devices while maintaining satisfactory detection accuracy, this study selected the smaller YOLOX-Nano as the baseline for research.

Table 2 Comparison of YOLOX test results across different model sizes

Model	Size (MB)	$F1_{0.5}$	$AP_{0.5}$ (%)	$AP_{0.5:0.95}$ (%)
YOLOX-Nano	3.7	0.98	98.9	70.2
YOLOX-Tiny	19.4	0.97	96.6	66.3
YOLOX-S	34.3	0.97	97.3	67.6

4.3 Comparison Before and After Improvement

To validate the performance of the CBAM-integrated YOLOX-Nano network, comparative analysis was conducted between the original and improved networks. As shown in Figure 5 [Figure 5: see original paper], the original YOLOX-Nano exhibits false detection under backlit conditions due to sunlight interference, incorrectly identifying geometric light spots as dragon fruit, whereas the improved YOLOX-Nano avoids this error. In Figure 6 [Figure 6: see original paper], both networks correctly identify dragon fruits in the scene, but the improved YOLOX-Nano achieves higher confidence scores (0.82, 0.81, and 0.74 from left to right) compared to the original (0.73, 0.79, and 0.69). Detailed evaluation metrics are presented in Table 3 .

Table 3 Comparison of dragon fruit detection results for improved YOLOX-Nano network

Network Model	Size (MB)	Avg Time (ms)	$F1_{0.5}$	$AP_{0.5}$ (%)	$AP_{0.5:0.95}$ (%)	AP_S (%)	AP_M (%)	AP_L (%)
YOLOX-Nano	3.7	18.46	0.98	98.0	70.2	54.4	96.3	99.2
YOLOX-Nano+CBAM	3.76	21.72	0.99	98.9	72.4	56.2	98.8	99.4

The improved YOLOX-Nano target detection network achieves an $F1$ value of 0.99, with $AP_{0.5}$ 0.9 percentage points higher and $AP_{0.5:0.95}$ 2.2 percentage points higher than the original YOLOX-Nano, despite a 3.26 ms increase in detection time. After adding CBAM, model size increases slightly, but detection precision for large, medium, and small dragon fruit targets improves by 1.8%, 2.5%, and 0.8%, respectively. These results demonstrate that CBAM integration enhances model robustness for dragon fruit detection in complex natural backgrounds. The channel attention mechanism assigns different weight values to different channels in feature maps, while the spatial attention mechanism assigns different weights to feature points at different positions, collectively refining extracted dragon fruit features and improving overall detection precision while meeting embedded device deployment requirements.

4.4 Comparison with Different YOLO Network Models

To objectively evaluate the improved YOLOX-Nano network performance, this study trained other lightweight YOLO models and the standard YOLOv3 model under identical settings for comparison. Table 4 presents comparative results across five target detection networks.

Table 4 Comparison of dragon fruit detection results across different networks

Network	Size (MB)	FPS	Avg Time (ms)	$F1_{0.5}$	$AP_{0.5}$ (%)	$AP_{0.5:0.95}$ (%)	AP_S (%)	AP_M (%)	AP_L (%)
YOLOX-Nano+CBAM	3.76	46	21.72	0.99	98.9	72.4	56.2	98.8	99.4
YOLOv5-S	14.3	48	20.83	0.96	91.0	59.5	29.6	82.1	98.3
MobileNetV3-YOLOv4	30.2	45	22.22	0.95	81.9	55.4	40.1	75.3	95.2
YOLOv4-Tiny	11.4	52	19.23	0.94	80.9	54.4	40.2	73.8	94.1
YOLOv3	236	41	24.39	0.92	72.7	46.2	1.7	65.4	91.3

The improved YOLOX-Nano network achieves an $AP_{0.5}$ of 98.9%, surpassing all other models. Its $AP_{0.5:0.95}$ exceeds YOLOv5-S, MobileNetV3-YOLOv4, and YOLOv4-Tiny by 12.9, 17.0, and 18.0 percentage points, respectively. For large targets in close-range fields of view, the improved YOLOX-Nano shows average detection precision approximately 12% higher than other lightweight YOLO networks. As distance increases and dragon fruit pixel areas decrease, detection precision declines for all models, but the improved YOLOX-Nano' s medium-target performance improves by approximately 17% compared to other lightweight models. For small targets, its AP_S of 56.2% is approximately 1.9× that of YOLOv5-S, 1.4× that of MobileNetV3-YOLOv4, and 2.1× that of YOLOv4-Tiny. The original YOLOv3 model shows inferior detection precision across all metrics, with an AP_S of only 1.7%, essentially failing to detect distant dragon fruits.

To validate detection effectiveness in natural environments, different models were compared under front-lighting, shading, and back-lighting conditions. As shown in Figures 7 [Figure 7: see original paper] and 8 [Figure 8: see original paper], the improved YOLOX-Nano network produces superior detection results under front-lighting and shading conditions without false detections, successfully identifying even distant small targets and heavily occluded fruits. Under severe back-lighting conditions (Figure 9 [Figure 9: see original paper]), all networks exhibit missed detections, with only the improved YOLOX-Nano and

YOLOv3 detecting nearby fruits, and the improved YOLOX-Nano achieving slightly higher confidence scores.

In summary, the proposed method effectively detects dragon fruit in natural environments. However, severe back-lighting conditions still affect detection performance, requiring further research to address environmental lighting impacts on camera imaging.

4.5 Analysis of Different Video Resolution Inputs

To verify whether the improved YOLOX-Nano target detection network meets real-time detection requirements in natural environments, video data at different resolutions was captured using a ZED 2i stereo camera. The average detection frame rate across 0–100 frames was calculated for each resolution, with results shown in Table 5 . At $4416\$ \times 1242 resolution, the average detection frame rate is 5.77 f/s, increasing to 20.94 f/s at 1340 \times 376 resolution. see original paper] shows detection results at frame 50 from the left camera perspective at 3840 \times 1080 and 1340 \times \$37$ resolutions, with detection frame rates of 6.63 f/s and 20.41 f/s, respectively. In conclusion, the improved YOLOX-Nano target detection network achieves real-time detection performance, providing valuable reference for fruit recognition in other natural environments.

Table 5 Comparison of dragon fruit detection frame rates at different resolutions

Resolution	Camera FPS (f/s)	Avg Detection FPS (f/s)
4416$×1242	15	5.77
3840×1080	30	6.63
2560×720	60	12.82
1340×$376		

5 Conclusion

This study presents an improved YOLOX-based dragon fruit detection method. By introducing an attention mechanism into the YOLOX-Nano target detection network, the trained model achieves an AP value of 98.9% at IoU = 0.5, an $AP_{0.5:0.95}$ value of 72.4%, and an AP_S of 56.2% for small targets. The improved network model accurately detects dragon fruit under various lighting and occlusion conditions. With an average single-image detection time of 21.72 ms and a lightweight model size of only 3.76 MB, the method is suitable for deployment on embedded devices and mobile terminals. Real-time testing on video streams at different input resolutions demonstrates that at 1340$×$376 resolution, the average detection frame rate reaches approximately 21 f/s, meeting real-time detection requirements. In summary, the improved YOLOX-Nano target detection model satisfies the requirements for rapid dragon fruit detection in natural environments in terms of both real-time performance and accuracy, representing significant progress for intelligent dragon fruit harvesting equipment development and providing valuable insights for intelligent detection technology research on other fruits.

References

[1] ZENG Xi, HU Guibing, QIN Yonghua. Current status and countermeasures of dragon fruit industry development in Guangdong Province[J]. China Fruit News, 2019, 36(9): 9-12.

[2] XU Leilei, JIN Yan, HOU Yuanyuan, et al. Investigation and analysis report on China' s dragon fruit market and industry[J]. Agricultural Products Market, 2021(8): 43-45.

[3] LU Huazhong, LI Jun, LI Can. Research progress in orchard mechanization production technology[J]. Guangdong Agricultural Sciences, 2020, 47(11): 226-235.

[4] ZHENG Taixiong, JIANG Mingzhe, FENG Mingchi. Vision-based target recognition and location for picking robot: A review[J]. Chinese Journal of Scientific Instrument, 2021, 42(9): 28-51.

[5] JORDI G-M, RICARDO S-C, JOAN R R-P, et al. Fruit detection and 3D location using instance segmentation neural networks and structure-from-motion photogrammetry[J]. Computers and Electronics in Agriculture, 2020, 169: 105165.

[6] LI Zhijun, YANG Shenghui, SHI Deshuai, et al. Yield estimation method of apple tree based on improved lightweight YOLOv5[J]. Smart Agriculture, 2021, 3(2): 100-114.

[7] JI J, XU Z, MA H, et al. Apple fruit recognition based on a deep learning algorithm using an improved lightweight network[J]. Applied Engineering in Agriculture, 2021, 37(1): 123-134.

[8] MU Longtao, GAO Zongbin, CUI Yongjie, et al. Kiwifruit detection of far-view and occluded fruit based on improved AlexNet[J]. Transactions of the CSAM, 2019, 50(10): 24-34.

[9] CHEN Yan, WANG Jiasheng, ZENG Zeqin, et al. Vision pre-positioning method for litchi picking robot under large field of view[J]. Transactions of the CSAE, 2019, 35(23): 61-68.

[10] PENG Hongxing, LI Jing, XU Huiming, et al. Litchi detection based on multiple feature enhancement and feature fusion SSD[J]. Transactions of the CSAE, 2022, 38(4): 153-160.

[11] ROY ARUNABHA M, JAYABRATA BHADURI. Real-time growth stage detection model for high degree of occultation using DenseNet-fused YOLOv4[J]. Computers and Electronics in Agriculture, 2022, 193: 106694.

[12] XU Z, JIA R, SUN H, et al. Light-YOLOv3: Fast method for detecting green mangoes in complex scenes using picking robots[J]. Applied Intelligence, 2020, 50(12): 4670-4687.

[13] XU P, FANG N, LIU N, et al. Visual recognition of cherry tomatoes in plant factory based on improved deep instance segmentation[J]. Computers and Electronics in Agriculture, 2022, 197: 106991.

[14] ZHOU Yuncheng, XU Tongyu, ZHENG Wei, et al. Classification and recognition approaches of tomato main organs based on DCNN[J]. Transactions of the CSAE, 2017, 33(15): 219-226.

[15] ZHENG T, JIANG M, LI Y, et al. Research on tomato detection in natural environment based on RC-YOLOv4[J]. Computers and Electronics in Agriculture, 2022, 198: 107029.

[16] ZHAO Chunjiang, WEN Chaowu, LIN Sen, et al. Tomato florescence recognition and detection method based on cascaded neural network[J]. Transactions of the CSAE, 2020, 36(24): 143-152.

[17] LI X, QIN Y, WANG F, et al. Pitaya detection in orchards using the MobileNet-YOLO model[C]//Technical Committee on Control Theory. Beijing, China: Chinese Association of Automation, 2020.

[18] WANG Jinpeng, GAO Kai, JIANG Hongzhe, et al. Method for detecting dragon fruit based on improved lightweight convolutional neural network[J]. Transactions of the CSAE, 2020, 36(20): 218-225.

[19] GE Z, LIU S, WANG F, et al. YOLOX: Exceeding YOLO series in 2021[J/OL]. arXiv: 2107.08430, 2021.

[20] ZHANG J, KE S. Improved YOLOX fire scenario detection method[J]. Wireless Communications and Mobile Computing, 2022, 2022: 1-8.

[21] LIU B, HUANG J, LIN S, et al. Improved YOLOX-S abnormal condition detection for power transmission line corridors[C]//2021 IEEE 3rd International Conference on Power Data Science (ICPDS). Piscataway, NY, USA: IEEE, 2021: 13-16.

[22] LIU M, ZHU C. Residual YOLOX-based Ship Object Detection Method[C]//2022 2nd International Conference on Consumer Electronics and Computer Engineering (ICCECE). Piscataway, NY, USA: IEEE, 2022: 427-431.

[23] WANG J, TANG C, LI J. Towards real-time analysis of marine phytoplankton images sampled at high frame rate by a YOLOX-based object detection algorithm[C]//OCEANS 2022-Chennai. Piscataway, NY, USA: IEEE, 2022: 1-9.

[24] WOO S, PARK J, LEE J Y, et al. CBAM: Convolutional block attention module[J/OL]. arXiv: 1807.06521, 2018.

[25] FU L, WU F, ZOU X, et al. Fast detection of banana bunches and stalks in the natural environment based on deep learning[J]. Computers and Electronics in Agriculture, 2022, 194: 106800.