IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received 25 May 2024, accepted 13 June 2024, date of publication 18 June 2024, date of current version 26 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3416332

RESEARCH ARTICLE

TL-YOLOv8: A Blueberry Fruit Detection Algorithm Based on Improved YOLOv8 and Transfer Learning

RONGLI GAI[1], YONG LIU[1], AND GUOHUI XU[2]
[1]College of Information Engineering, Dalian University, Dalian 116622, China
[2]College of Life and Health, Dalian University, Dalian 116622, China

Corresponding author: Rongli Gai (gairli@sict.ac.cn)

This work was supported in part by Dalian Science and Technology Innovation Fund under Grant 2022JJ12SN052, and in part by Dalian University under Grant DLUXK-2023-YB-004.

ABSTRACT Blueberry fruit in the picking process are characterized by small fruit particles, similar color characteristics of immature fruits to leaf, and not obvious characteristics of fruits obscured by leaves, which lead to low real-time detection accuracy of blueberry fruits and fail to meet the detection standard of automatic picking. To improve the detection accuracy, we propose a blueberry identification algorithm named TL-YOLOv8 based on the YOLOv8 algorithm. By introducing an improved MPCA (Multiplexed Coordinated Attention) in the last layer of the backbone network, we enhance the feature extraction capability during the training process. By replacing the C2f module with an OREPA(Online Convolutional Re-parameterization) module, not only the training is accelerated, but also the characterization is enhanced. Meanwhile, in order to cope with the fruit occlusion problem more effectively, we introduce the MultiSEAM (Multi-scale Separation and Occlusion-Aware Module). As a means of optimizing the model parameters, we pre-trained the model using Transfer Learning to improve the generalization ability of the network. The method achieved 84.6% Precision, 91.3% recall and 94.1% mAP on the blueberry dataset, where mAP was improved by 3.4% over the original algorithm, and the experiments showed that it can be effective for blueberry fruit detection.

INDEX TERMS Blueberry detection, deep learning, object detection, occlusion, transfer learning, YOLOv8.

I. INTRODUCTION

Fruit ripeness is an important indicator in determining the optimal harvesting period for fruits. Fruits ripen at different rates in different varieties and growing environments,and fruits are harvested at the right time as raw food for fresh sale. Harvesting too early affects fruit quality and too late reduces storage time. So accurate ripeness detection is needed to determine the best time to pick the fruit [1]. In recent years, the blueberry industry has become the fastest growing new fruit tree industry by virtue of its larger business scale, specialized planting and product development. Most of the current blueberry picking still relies on traditional manual operations, which leads to huge labor costs [2]. With the development of machine vision and target detection technology, machine vision-based detection methods are gradually applied to fruit picking, becoming an emerging trend to replace traditional manual picking. Compared with the traditional manual picking, machine vision inspection has the significant advantages of fast detection speed and high efficiency [3].

Blueberry object detection in the actual agricultural production environment is often characterized by small size, dense distribution and leaf shading, coupled with the changing light and complex background of the farmland environment, resulting in the failure of existing object detection algorithms to achieve satisfactory results. Xiuming et al. [4] proposed an algorithm for scale-adaptive small target recognition in complex agricultural environments. The algorithm overcomes the influence of the complex and

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

TABLE 1. Previous research in complex agricultural environments.

Previous Research	Innovation	Discussion
[4]	This study proposed a scale-adaptive small object detection algorithm for complex agricultural environments.	The algorithm's efficiency diminishes as subgraph size decreases. The processing time of the model is multiplied.
[5]	This study proposed a blueberry fruit detection model to solve the problem of background interference in complex environments.	The algorithm overlooks the issue of occlusion between blueberry fruits or occlusion by foliage, which is crucial for accurate blueberry fruit detection.
[6]	This study proposed an algorithm for the detection and recognition of rice leaf diseases, effectively alleviating the problem of difficult identification of small-area target detection.	The algorithm shows limited improvement in addressing the issue of misidentifying similar-colored backgrounds as targets.
[7]	This study proposed a YOLOv4 deep learning algorithm combined with DenseNet to detect cherry fruits, which improves the problem of inaccurate cherry fruit detection.	The algorithm has good performance in small target detection and gives us ideas in blueberry fruit target detection.
[8]	This study proposed an improved blueberry recognition model for YOLOv5s in response to the problems of dense adhesion and severe occlusion during blueberry fruit growth.	The algorithm does not consider optimizing the time and computational resources required for training deep learning models.
[9]	This study proposed an algorithm for grape ripeness detection and visual fronting based on an improved YOLOv4 in complex agricultural environments.	The algorithm demonstrates robust performance in complex environment detection, offering a viable idea for blueberry fruit target detection.
[10]	This study applied YOLOv4 to blueberry ripeness detection with high accuracy.	The computational demands of YOLOv4 are substantial, leading to slow training speeds and low efficiency.

changeable background environment and the difficulty of feature extraction caused by the small size of the target. Zhu et al. [5] proposed to design a blueberry fruit detection model based on Faster R-CNN optimization algorithm to solve the problem of background interference in complex environments. Yang et al. [6] proposed a high-density high-level component feature pyramid network (DHLC-FPN) integrated into the detection transformer (DETR) algorithm. The DETR will effectively alleviate the problem of difficult identification for object detection in tiny areas affected by rice leaf diseases. In many cases, non-YOLO mainstream algorithms have achieved reliable results in complex environment detection, providing valuable insights for blueberry fruit detection.

The YOLO family [11], [12], [13] is deep learning-based algorithms for object detection, capable of achieving swift and precise object recognition. Many researchers have successfully applied them to crop maturity detection tasks. Gai et al. [7] proposed a YOLOv4 deep learning algorithm combined with DenseNet to detect cherry fruits, which is more suitable for the recognition of smaller cherry fruits and improves the problem of inaccurate cherry fruit detection. Yang et al. [8] proposed an improved blueberry recognition model for YOLOv5s in response to the problems of dense adhesion and severe occlusion during blueberry fruit growth. Similarly, Qiu et al. [9] proposed an algorithm for grape ripeness detection and visual fronting based on improved YOLOv4, which helps to provide spatial location data of grape clusters for fast and accurate identification and classification of grapes at different stages of ripeness. MacEachern et al. [10] used six YOLO networks for wild blueberry fruit detection, including YOLOv3 and YOLOv4.These models achieved satisfactory detection results. The YOLO series has demonstrated outstanding performance in detecting objects within complex agricultural environments. As shown in Table 1, although previous research has made many advances in complex settings, most studies have addressed single issues and do not meet our specific requirements for blueberry fruit detection. Therefore, we focused on improving the YOLOv8 algorithm to address challenges such as small fruit size, dense distribution, leaf occlusion, and slow training speed with low efficiency.

The biggest disadvantage of deep neural networks today is their high cost of use compared to traditional machine learning algorithms [14]. Especially when we are trying to deal with real-life practical problems such as image recognition, voice recognition, etc. Because models have a large number of parameters to be trained, they need massive training data to support them. With the help of transfer learning, the relationships obtained for a certain type of data in a model training task can also be easily applied to different problems in the same domain. Zhu et al. [15] proposed a small dataset-based transfer learning method for solving the problem of difficult and costly dataset production due to the large amount of sample data required for visual detection using neural network frameworks. Xu et al. [16] proposed the use of eight classical network models based on deep learning as feature extraction for transfer learning to realize the range of water content of sea buckthorn fruits that can be quickly identified by collecting images of the appearance and morphological changes during the drying process of sea buckthorn fruits. Attallah et al. [17] proposed a transfer learning algorithm based on three compact convolutional neural networks to automatically

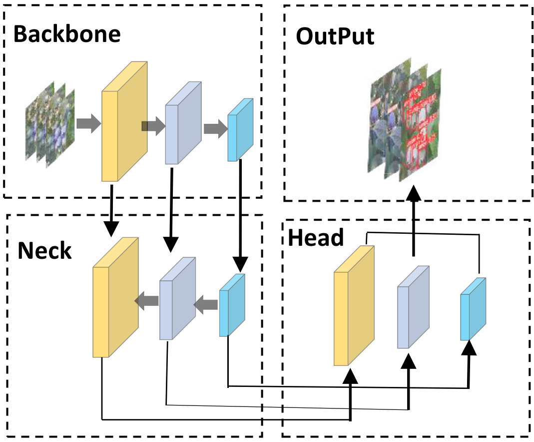

FIGURE 1. YOLOv8 algorithm.

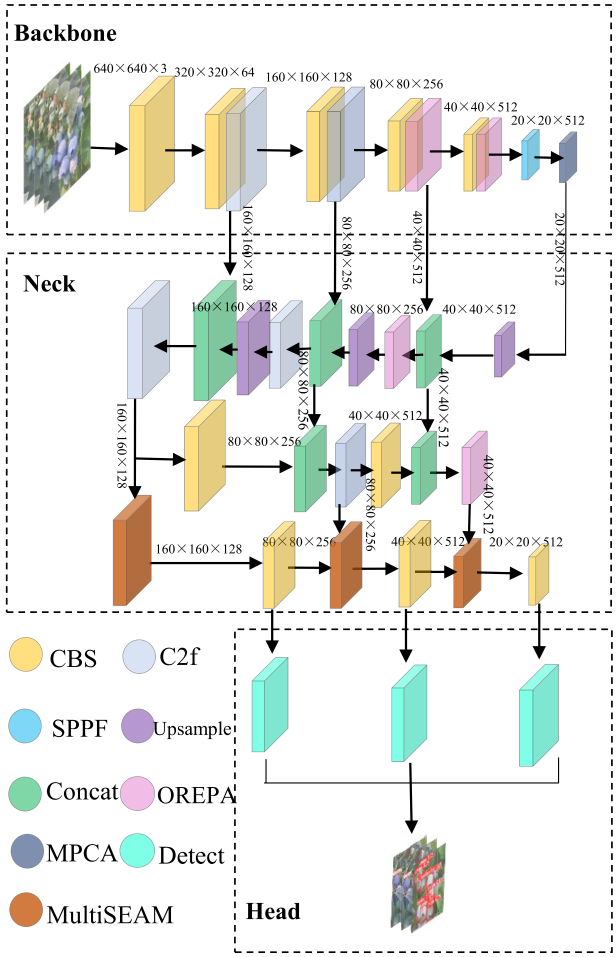

FIGURE 2. Improvement of YOLOv8 network structure.

identify tomato leaf diseases. Poyatos et al. [18] proposed an evolutionary pruning model for deep neural networks based on transfer learning, which replaces the final fully connected layer with a sparse layer optimized by a genetic algorithm, addressing the limitation of deep learning that usually requires large training datasets. Huang et al. [19] proposed a deep multi-source transfer learning algorithm applied to bearing fault diagnosis under different operating conditions to solve the data distribution shift problem.

To address the above problems, this paper proposes a blueberry fruit detection algorithm based on YOLOv8 and migration learning, aiming at adapting more effectively to the special needs of blueberry fruit detection. This improved algorithm takes advantage of YOLOv8's multi-scale feature fusion and end-to-end training as well as the feature migration benefits of transfer learning, and optimizing for the small size, dense distribution,similarity of color features of unripe fruits to leaves, and leaf occlusion of blueberry fruit. The algorithm mainly consists of: (1) The improved MPCA mechanism and OREPA module are introduced in order to improve the feature extraction capability, accelerate the training, and improve the detection accuracy. (2) Introduction of an improved MultiSEAM module for dealing with fruit occlusion. (3) Transferring training feature parameters from public fruit dataset to blueberry fruit detection by transfer learning to improve algorithm generalization ability and optimize network parameters.

II. MODEL INTRODUCTION AND IMPROVEMENT

A. YOLOv8 ALGORITHM INTRODUCTION

YOLOv8 is an improved version based on the YOLO (You Only Look Once) object detection algorithm, which introduces some new design ideas and techniques to improve the accuracy and speed of the model compared to the previous YOLO version [20]. YOLOv8 can not only perform well in general-purpose object detection tasks, but can also be used in a variety of application areas, such as smart agriculture, automated driving, and industrial detection. The algorithm has demonstrated superior detection performance in practical tasks and, through optimization of the previous version's techniques, provided a unified framework for training model execution tasks [21]. The network architecture of YOLOv8 consists of the following three main parts, as shown in Fig. 1:

Backbone: it employs a series of convolutional and anti-convolutional layers to extract features, and also uses residual joins and bottleneck structures to reduce the size of the network and improve performance. This part adopts the C2f module as the basic constituent unit, which has a smaller number of parameters and better feature extraction capability than the C3 module of YOLOv5.

Neck: it employs a multi-scale feature fusion technique to fuse different stages of feature maps from Backbone to enhance feature representation. Specifically, it includes one SPPF module, one PAA module and two PAN modules.

Head: It is responsible for the final object detection and classification task and consists of a detection head and a classification head. The detection head contains a series of convolutional and anti-convolutional layers for generating detection results, while the classification head uses global average pooling to classify each feature map.

B. IMPROVED YOLOv8 ALGORITHM

This section presents an improved framework for detection tasks based on YOLOv8. The primary aim of this

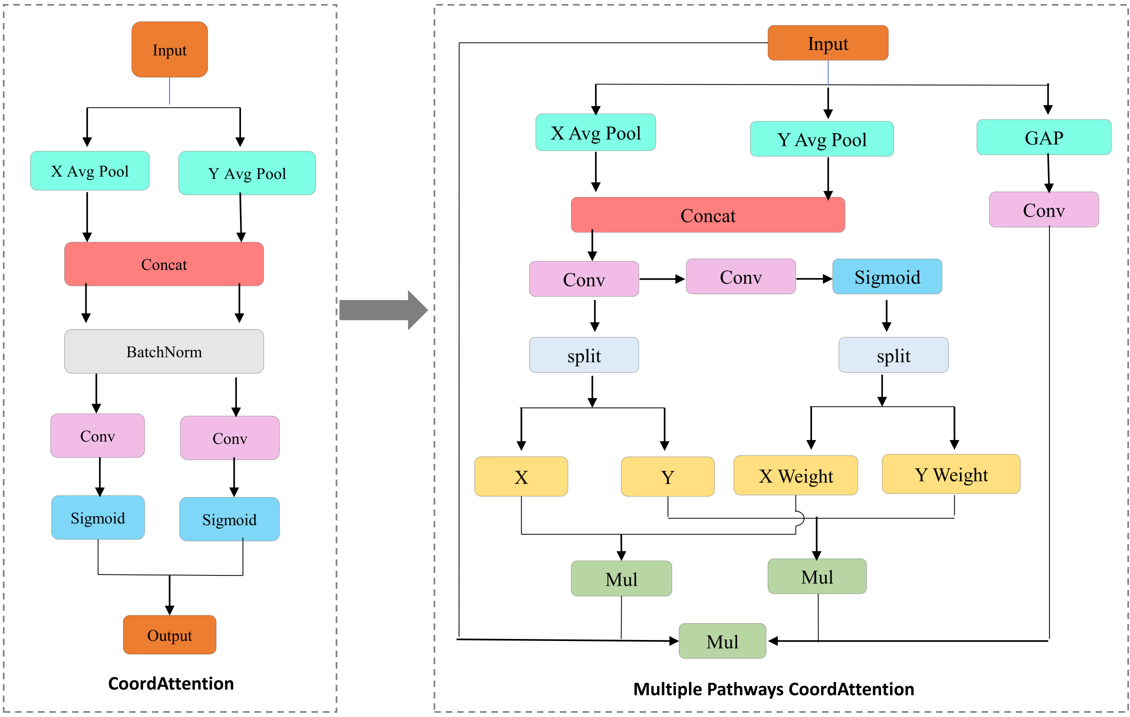

FIGURE 3. **MPCA network structure.**

investigation is to introduce a precision-oriented and efficient detection algorithm tailored for blueberry fruit recognition. Acknowledging the complexities of agricultural environments and the substantial training expenses, we have explored algorithms capable of delivering robust solutions applicable to real scenarios.

Fig. 2 shows the structure diagram of our model network for improving YOLOv8. In response to the challenges encountered during blueberry fruit detection, we implemented the following modifications in this study: (1) We added an improved MPCA mechanism to the last layer of the backbone network. (2) Replacement of the C2f module with the OREPA module for accelerated training during the training process. (3) Introduction of an improved MultiSEAM module for the fruit occlusion problem.

1) MPCA (THE MULTIPLE PATHWAYS COORDINATED ATTENTION)

In the detection process of blueberry fruits, we found that blueberry fruits have the characteristics of small target and complex background, which leads to the traditional algorithm can not play a better effect in the detection process. So we invoke the improved MPCA model to further improve the feature extraction capability. The MPCA model is an improvement of the CA (Coordinate Attention) model [22], which captures cross-channel information, as well as direction-aware and position-aware information, to achieve more accurate localization and identification of targets of interest. However, the two-channel convolution of the original CA attention is weak for feature extraction, and we consider adding another channel branch to add a convolution channel, which can fully extract the features of each path and realize a multi-branch coordinated attention mechanism.

As shown in Fig. 3, the improved MPCA attention mechanism consists of three channels, of which we retained the original CA's X channel and Y channel, and these two channels take the incoming feature maps through average pooling in the height direction and in the width direction. To enhance the fine-grained features of height and width, we conducted a subsequent convolution operation, followed by segmentation through a sigmoid function. The segmented results were then multiplied with the original convolution results individually, thus reinforcing the weights. The added third channel branch, Global Average Pooling, emphasizes performing a layer of down-sampling on overall feature information, thereby preserving more background information of the image. This process of three channels in parallel phases not only acquires inter-channel information, but also direction-related positional information, which helps the model to better localize and identify the target, which is sufficiently flexible and lightweight.

2) OREPA (ONLINE CONVOLUTIONAL RE-PARAMETERIZATION)

The YOLOv8 algorithm has the defects of slow speed and low training efficiency during the training process. Because the YOLO model consists of many layers of neural networks,

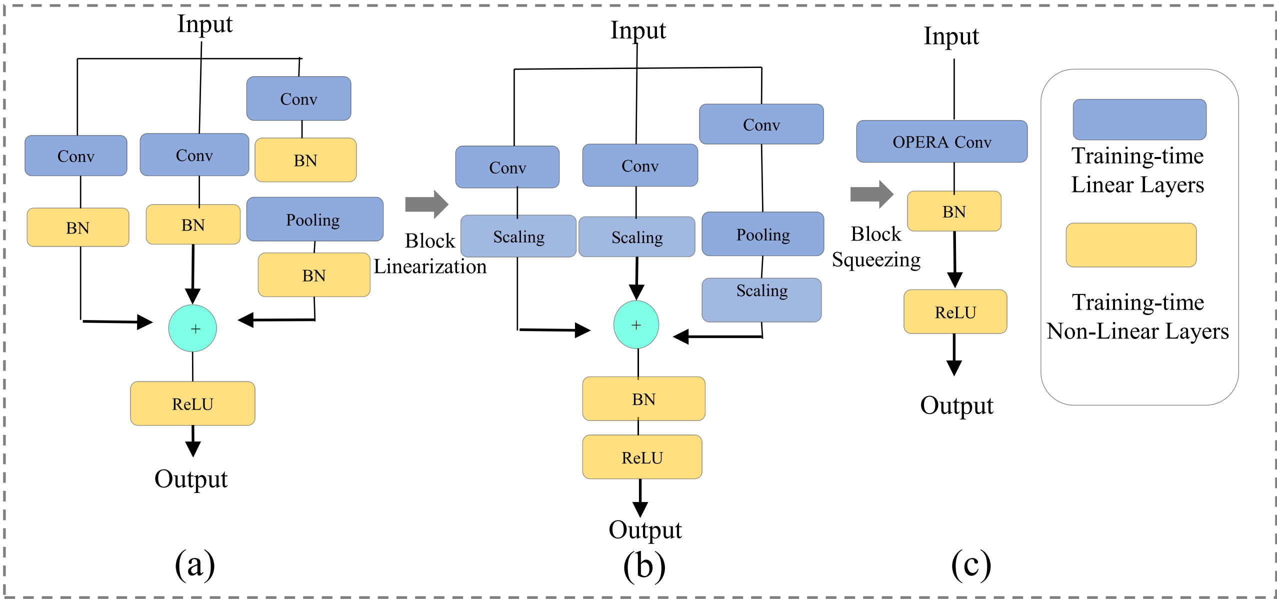

FIGURE 4. OREPA method structure diagram: (a) Prototype Block (b) Linearized Block (c) Training Block.

these layers require a large number of parameters for training and optimization, which correspondingly increases the time and computational cost of training. While the use of better hardware devices can significantly improve training speed, it still requires significant computational resources.

OREPA [23] is a model structure for online convolutional reparameterization, a two-stage pipeline that reduces the huge training overhead by compressing complex blocks of training time into a single convolution. This is essential for training large-scale neural networks with complex topologies and it further allows us to reparameterize the model in a more cost-effective way.The structure of OREPA's approach is shown in Fig. 4.

As illustrated in stages 4a to 4b of Fig. 4, the initial phase of OREPA is termed block linearization, involving the removal of all non-linear BN layers and the introduction of linear scaling layers. While these layers share similar properties with the BN layer, they contribute to optimizing diversity across different branches. 4b to 4c show the second stage of OREPA called block compression, which reduces the complex linear fast to a single convolutional layer. This phase decreases the network's complexity while preserving essential information. The second phase involves compressing multi-layer convolutions and multi-path convolutions into a single convolution. The convolution process is shown in Equation 1.

$$\mathrm{Y} = \mathrm{W}^{*}\mathrm{X} \tag{1}$$

C_i and C_o represent the number of input and output channels of a two-dimensional convolution kernel of size $\mathrm{K_H} \times \mathrm{K_W}$. $\mathrm{X} \in \mathbb{R}^{\mathrm{C}_i \times \mathrm{H} \times \mathrm{W}}$ and $\mathrm{Y} \in \mathbb{R}^{\mathrm{C}_o \times \mathrm{H'} \times \mathrm{W'}}$ denote the input tensor and the output tensor, respectively.The function description of the convolution process for multi-layer convolutions is shown in Equation 2.

$$\mathrm{Y} = \mathrm{W_N}\left(\mathrm{W_{N-1}} * \cdots (\mathrm{W_2} * (\mathrm{W_1} * \mathrm{X}))\right) \tag{2}$$

$\mathrm{W}_j \in \mathbb{R}^{C_i \times C_{i-1} \times \mathrm{K_{H_j}} \times \mathrm{K_{W_j}}}$ satisfies both $C_0 = C_i$ and $C_N = C_o$. According to the associative law, it is permissible to first convolve the convolution kernels, consolidating these layers into a single layer. The function description for compressing multiple layers of convolution is shown in Equation 3.

$$\mathrm{Y} = (\mathrm{W_N}(\mathrm{W_{N-1}} * \cdots (\mathrm{W_2} * \mathrm{W_1})) * \mathrm{X} = \mathrm{W}_e * \mathrm{X} \tag{3}$$

W_j represents the weight values of the j^{th} layer, and W_e denotes the end-to-end mapping matrix. The function description for compressing parallel convolutions is shown in Equation 4.

$$\mathrm{Y} = \sum_{m=0}^{\mathrm{M}-1} (\mathrm{W}_m * \mathrm{X}) = \left(\sum_{m=0}^{\mathrm{M}-1} \mathrm{W}_m\right) * \mathrm{X} \tag{4}$$

In accordance with the principle of convolutional linearity, the amalgamation of multiple branches into a singular entity is feasible. W_m represents the weight values of the m^{th} branch. $\sum_{m=0}^{\mathrm{M}-1} \mathrm{W}_m$ is the unified weight. OREPA employs a two-stage pipeline, reducing the computational and storage overhead caused by intermediate calculations, significantly lowering the training cost, with minimal impact on performance.

3) MULTISEAM (MULTI-SCALE SEPARATION AND OCCLUSION-AWARE)

The mutual occlusion among blueberries results in some features being superimposed in the image, leading to inaccuracies in feature alignment, and certain crucial characteristics of blueberries cannot be fully captured and analyzed. The

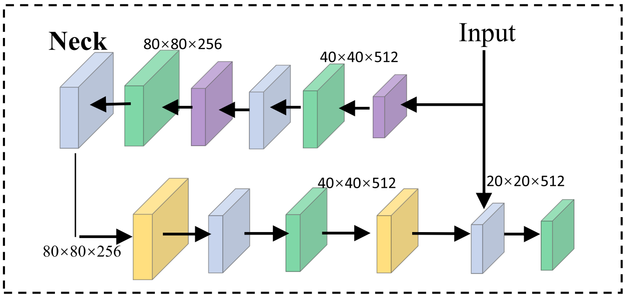

FIGURE 5. Neck structure of YOLOv8.

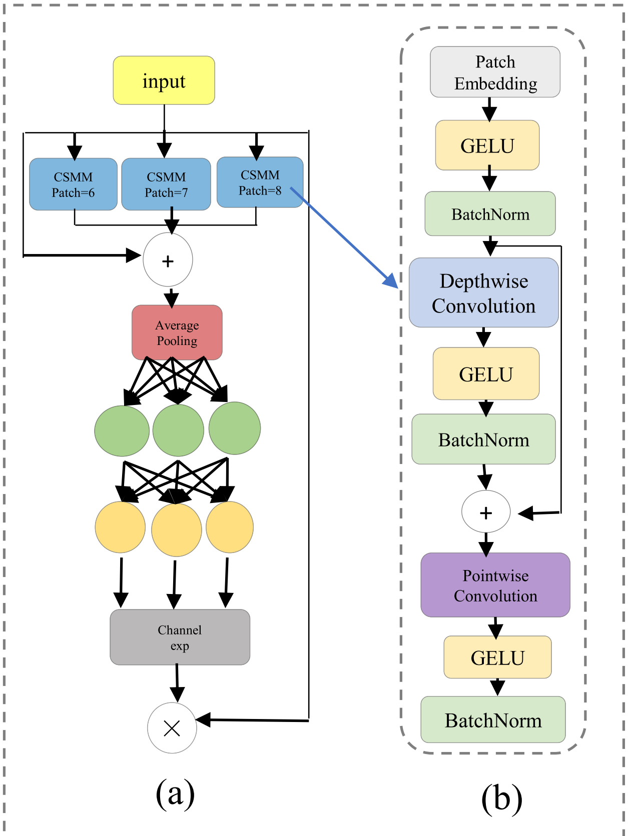

FIGURE 6. (a) Network structure of SEAM, (b) CSMM structure.

MutilSEAM is capable of focusing on the blueberry region, reducing background interference in blueberry detection, and achieving multi-scale blueberry detection, thereby enhancing the accuracy and efficiency of blueberry detection.

As shown in Fig. 6, the first part of SEAM [24] adopts depthwise separable convolutions with residual connections. Depthwise separable convolution operates channel-wise separations, effectively reducing the parameter count. While this convolutional approach learns features from individual channels, it overlooks the inter-channel information relationships.

To compensate for this deficiency, SEAM combines the outputs of different depth convolutions by point-by-point (1×1) convolution. This allows the model to mix information from various channels efficiently. The information from each channel is then fused using a two-layer fully connected network to enhance the connectivity between the channels. This process helps to capture the informational relationships between channels, compensating for the aforementioned losses in occluded scenes. Finally, the output of the SEAM module is multiplied with the original features, allowing the model to obtain finer blueberry features and effectively address the issue of blueberries being occluded by other objects in the image, ensuring the accuracy and robustness of blueberry detection.

In practical blueberry detection scenarios, there exist blueberry fruits of varying scales. Additionally, obscured targets typically occupy a small portion of the pixels in the image, leading to potential oversight or misjudgment. To address this multi-scale issue, we design the MultiSEAM module, which integrates small object detection layers with SEAM. Fig. 5 illustrates the neck structure of YOLOv8, while Fig. 2 depicts the modified neck structure of our enhanced YOLOv8. We incorporate small object detection layers into the neck to generate feature maps of different scales. These feature maps are then fed into SEAM for comprehensive coverage of small-scale feature information. We expect to optimize the performance of the model to make it more effective for various object detection tasks.

C. TRANSFER LEARNING

The core concept of transfer learning is derived from a wealth of early experiments, where knowledge and features are extracted and can be transferred to related problems. Deep learning models often necessitate a substantial volume of labeled data for effective training. However, in certain situations, acquiring a sufficient amount of labeled data can prove challenging or financially burdensome. Transfer Learning allows for the application of knowledge learned from one task to another related task, meeting expectations even if the target task has a limited amount of data [25]. Transfer learning not only addresses dataset challenges but also significantly reduces the time and computational resources required for training deep learning models, accelerating network convergence speed and training efficiency.

Due to the small number of our dataset and high similarity, in this case, we chose to use the Fruits-360 dataset sourced from kaggle for migration learning, and the migration learning flowchart is shown in Fig. 7. We first train on our improved YOLOv8 algorithm using the Fruits-360 dataset to obtain an initial weight file, and then use this weight file as the pre-training weights to train again on our Blueberry dataset. In training the blueberry dataset, we divided the training into two phases, the freezing phase and the unfreezing phase. In the freezing phase, we freeze the backbone of YOLOv8 and the feature extraction network is not changed, at this point only the network is fine-tuned. In the unfreezing phase, when the backbone of YOLOv8 is not frozen, the feature extraction network is changed and all the parameters of the network are changed so that more features can be extracted from the blueberries. Freezing the backbone network for a period of

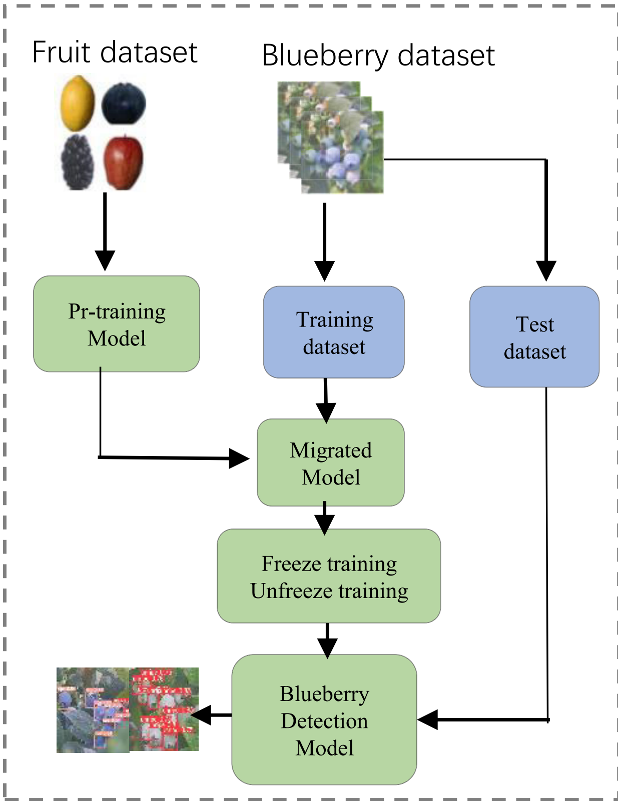

FIGURE 7. Transfer learning flowchart.

time prevents the model from over-tuning at the beginning, thus reducing the risk of overfitting. After unfreezing the backbone network, the model can be fine-tuned on a specific task and learn to adapt to the features and patterns of the task in a more targeted manner.

Before training, the parameters of the network model need to be set, in this paper, we set the number of training rounds to 300, batch-size to 16, and freeze the network backbone in the first 50 rounds of training. The initial learning rate during the freezing of the backbone was 0.01, the momentum of the learning rate was 0.937, the confidence threshold was 0.25, and the threshold for non-maximal inhibition was 0.7.

III. EXPERIMENTAL RESULTS AND ANALYSIS

A. EXPERIMENTAL DATA

This study utilized a dataset comprising two components, one of which is a self-made blueberry dataset. The dataset consists of images of blueberry fruits captured from videos recorded by multiple cameras. These videos were segmented into 2000 images depicting blueberry fruits at different stages, including germination, flowering, fruiting, and ripening phases. This study focuses on the detection of blueberry fruit ripeness, so 800 images with blueberries were selected. These images contain different blueberry fruit characteristics, such as small-target blueberry images, dense-target blueberry images, and occluded blueberry images, as shown in Fig. 8. Images are saved in JPG format with a resolution of 1920×1080 pixels. We utilized labelImg for data annotation, classifying blueberries into two categories: mature and immature. After completing the annotations, the data were divided into five equal parts for cross-validation. Each of these five subsets was assigned a unique identifier. Initially, the first subset was used as the test set, while the remaining subsets (2 to 5) were used as the training set. Subsequently, the second subset was used as the test set, and the others served as the training set. This process was repeated until each subset had been used once as the test set. This method produced 5 performance metrics, which were then averaged to obtain the final performance evaluation metric of the model.We performed 5-fold cross validation of the ablation experiments and reported the mean and standard deviation for each metric.

FIGURE 8. (a) Small target; (b) Intensive target; (c) Covered target.

The other part is the Fruits-360 dataset from the kaggle platform used to pre-train the network model, including the following fruits: apple, apricot, banana, beet red, blueberry, cactus fruit, cantaloupe, popcorn, cauliflower, cherry, tomato, walnut, watermelon, and 131 other fruits.

The total number of images: 90483, training set size: 67692 images, test set size: 22688 images. The dataset was created by photographing fruits using a Logitech C920 camera, one of the best webcams available. More information about Fruits-360 can be found in [26].

B. EVALUATION INDICATORS

In this investigation, the performance of the target detection algorithm is evaluated using Precision, Recall, and Average Precision (AP) as key metrics. Precision is the proportion of samples predicted by the model to be positive instances that are actually positive instances. Recall is the proportion of samples that are actually positive cases that are correctly predicted by the model to be positive cases. AP(Average Precision) calculates the Precision-Recall curve for each category and then calculates the area under the curve. Finally, the APs of all categories were averaged to obtain the final mAP. A high precision means that the model is less likely

TABLE 2. Performance of the YOLOv8.

Category	Precision(%)	Recall(%)	mAP50 (%)	MAP50-95(%)
Ripe blueberry	80.1 ± 3.4	88.1 ± 1.8	92.1 ± 0.4	62.1 ± 1.2
Unripe blueberry	83 ± 1.2	82.5 ± 1.0	89.4 ± 0.6	54.7 ± 0.6
All	81.6 ± 1.1	85.3 ± 0.4	90.7 ± 0.1	58.4 ± 0.9

TABLE 3. Performance of the TL-YOLOv8.

Category	Precision(%)	Recall(%)	mAP50 (%)	MAP50-95(%)
Ripe blueberry	83.6 ± 3.4	95.1 ± 1.7	95.7 ± 0.4	65.3 ± 0.9
Unripe blueberry	85.6 ± 1.6	87 ± 0.4	92.6 ± 0.6	60.1 ± 1.4
All	84.6 ± 0.9	91.3 ± 1.1	94.1 ± 0.1	62.7 ± 0.2

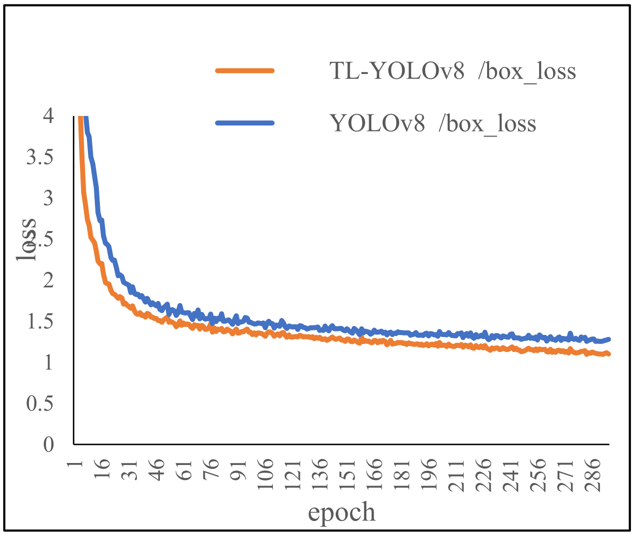

FIGURE 9. Comparison of TL-YOLOv8 and YOLOv8 box-loss curves.

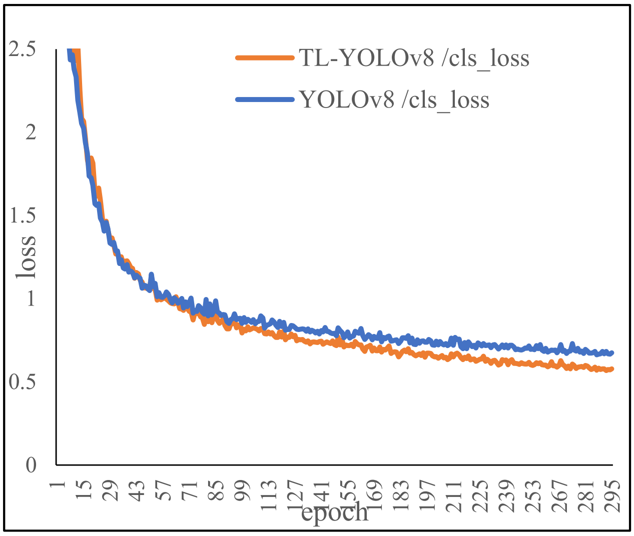

FIGURE 10. Comparison of TL-YOLOv8 and YOLOv8 cls-loss curves.

to incorrectly identify the background or other irrelevant objects as blueberry fruits. A high recall indicates that the model can detect as many blueberry fruits as possible. A high mAP signifies that the model can maintain high detection accuracy and recall at different confidence thresholds. These indicators are shown in Equation 5, Equation 6, Equation 7 and Equation 8.

$$\text{Precision} = \frac{TP}{TP + FP} \times 100\% \tag{5}$$

$$\text{Recall} = \frac{TP}{TP + FN} \times 100\% \tag{6}$$

$$AP = \int_0^1 P(R)dR \tag{7}$$

$$mAP = \frac{1}{M}\sum_{k=1}^{M} AP(K) \tag{8}$$

In this context, TP represents true positives, FP stands for false positives, TN denotes true negatives, and FN indicates false negatives. Additionally, M denotes the total number of blueberry ripeness categories. mAP50 is a specific variant of mAP, where the "50" indicates that the IoU (Intersection over Union) threshold used in the calculation is 50%. The computational steps for mAP50 are similar to those for regular mAP, with the difference that only predictions with an IoU greater than or equal to 50% are considered when calculating the area under the Precision-Recall curve.

C. EXPERIMENTAL RESULTS AND ANALYSIS

1) PERFORMANCE ANALYSIS OF THE TL-YOLOv8

The present study focuses on improving the YOLOv8 algorithm and integrating it with transfer learning to obtain the TL-YOLOv8 algorithm. In order to comprehensively analyze the improvement effect of TL-YOLOv8, we will conduct comparisons with the YOLOv8 algorithm, ensuring consistency in experimental conditions for each iteration. Training will be terminated upon achieving convergence, and the best-performing training weights will be selected for testing. The comparative data presented in Table 2 and 3 illustrates that, regardless of mature blueberries, immature blueberries, or the entirety of blueberries, the detection performance of TL-YOLOv8 exhibits a noticeable improvement over YOLOv8. Specifically, Precision has increased by 3%, and mAP50 has improved by 3.4%. The experimental results indicate that the TL-YOLOv8 algorithm demonstrates a higher detection accuracy for blueberry fruits.

The loss curve illustrates the variation of the loss values throughout the training process of the model. We observed that the convergence speed of the loss curves in Fig. 9 and 10 is fast and exhibits minor fluctuation, indicating the stability of model training and its enhanced generalization capability. In comparison to the YOLOv8 algorithm, TL-YOLOv8 exhibits a faster descent rate and approaches loss values

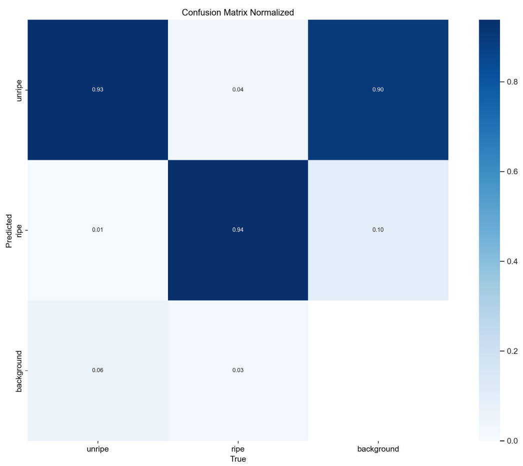

FIGURE 11. The confusion matrix of the YOLOv8.

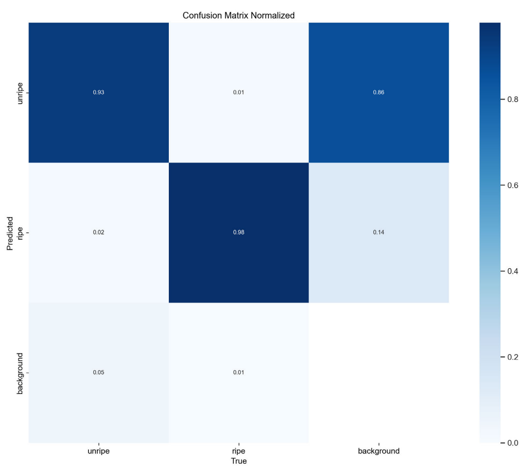

FIGURE 12. The confusion matrix of the TL-YOLOv8.

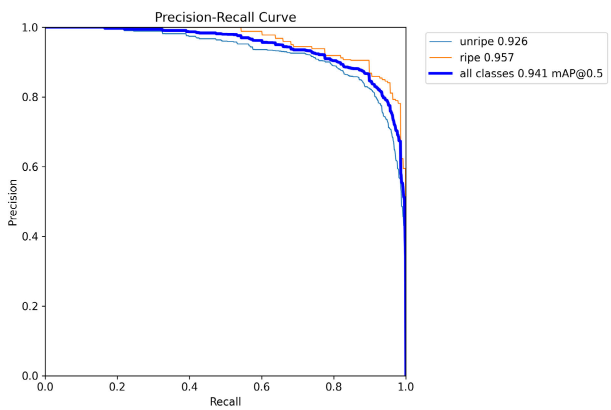

FIGURE 13. Precision-Rall curve of TL-YOLOv8.

closer to 0, indicating the superior learning capability of our algorithm.

In the preliminary observations of the experimental subjects, we observed that immature blueberry fruit exhibit a green color, which closely resembles the color of blueberry

TABLE 4. Performance comparison between MPCA and other attention mechanisms.

Model	Precision(%)	Recall(%)	mAP50 (%)	MAP50-95(%)
SimAM	83.6	86.3	91.7	57.9
LSKBlock	83.2	88	91.5	58.7
TripletAttention	85.4	85.5	91.7	58.8
SEAttention	82.8	88.9	92	58.6
EfficientAttention	83	84.6	91.2	57.5
BiLevelRoutingAttention	86.8	84.3	91.3	57.3
BAMBlock	82.9	86.5	91.4	57.9
EMA	84.6	84.2	91.1	58.9
CoordAttention	86.3	85.7	91.6	59
MPCA	86.2	87.8	92.2	60.2

leaves.Therefore, in practical detection, there is a tendency to confuse blueberry fruits with leaves. The confusion matrix in Fig. 11 and 12 illustrates instances where the model incorrectly assigns instances to other categories. The figures indicate that although significant background confusion exists in both images, the performance of TL-YOLOv8 has shown a substantial improvement over the YOLOv8 algorithm.

Fig. 13 demonstrates the correlation between precision and recall during the training process for blueberry fruit detection. It is observed that the curve predominantly resides in the upper-right quadrant of the chart, with minor fluctuations, indicating the model's robustness and generalization capability.In all categories of detection, the mAP50 has reached 94.1%, indicating that the TL-YOLOv8 algorithm exhibits superior performance.

2) ABLATION EXPERIMENT

We conducted a comparative analysis of the effectiveness in terms of accuracy among our proposed MPCA, the CoordAttention, and other attention mechanisms. To ensure the accuracy of the comparative experiments, each attention mechanism was separately incorporated into the final layer of the YOLOv8 backbone network. Experimental results demonstrate that our proposed MPCA attention mechanism exhibits superior accuracy during training on the blueberry dataset.This is shown in Table 4.

The improved YOLOv8 algorithm includes the original YOLOv8 module, MultiSEAM module, OREPA module and MPCA module. The contribution of each module was analyzed by ablation experiments and the results are shown in Table 5. In this study, the MultiSEAM module was first added to the YOLOv8 baseline algorithm, and the results showed that the mAP50 increased by 0.7%, which indicates that the occlusion-aware part of the module effectively solves the blueberry occlusion problem. Subsequently, the OREPA module was added alone, and the results showed that online convolutional reparameterization worked better in this experiment, with a 2.7% increase in mAP50. Finally,

TABLE 5. Ablation experiment.

YOLOv8	MultiSEAM	OREPA	MPCA	Precision(%)	Recall(%)	mAP50 (%)	MAP50-95(%)
✓				81.6 ± 1.1	85.3 ± 0.4	90.7 ± 0.1	58.4 ± 0.9
✓	✓			86.7 ± 1.2	88.6 ± 1.5	91.4 ± 0.4	60.1 ± 0.5
✓		✓		89.1 ± 1.5	86.8 ± 1.3	93.4 ± 0.1	60.8 ± 0.2
✓			✓	86.2 ± 1.0	87.8 ± 0.8	92.2 ± 0.4	60.2 ± 0.3
✓	✓	✓		87 ± 1.6	88.4 ± 1.7	93.2 ± 0.3	62.3 ± 1.1
✓	✓	✓	✓	86.2 ± 0.9	89.7 ± 0.8	93.7 ± 0.1	62 ± 0.2

TABLE 6. Comparison of transfer learning experiments:①YOLOv8②YOLOv8+ transfer learning ③Improved YOLOv8 ④Improved YOLOv8+ transfer learning.

Model	Precision(%)	Recall(%)	mAP50 (%)	MAP50-95(%)
①	81.6 ± 1.1	85.3 ± 0.4	90.7 ± 0.1	58.4 ± 0.9
②	86.4 ± 0.6	87.4 ± 0.4	91.6 ± 0.2	58.8 ± 0.3
③	86.2 ± 0.9	89.7 ± 0.8	93.7 ± 0.1	62 ± 0.2
④	84.6 ± 0.9	91.3 ± 1.1	94.1 ± 0.1	62.7 ± 0.2

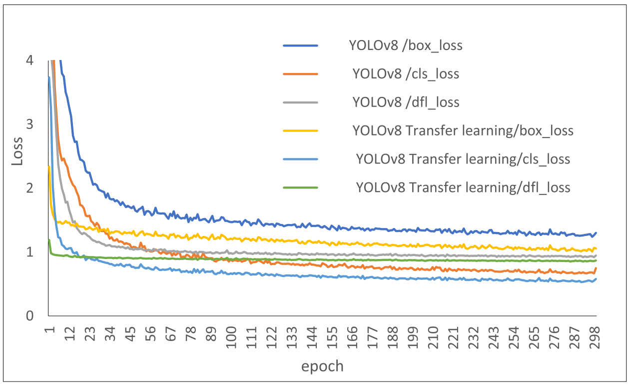

FIGURE 14. Comparison of YOLOv8 and YOLOv8 + transfer learning loss curves.

the MPCA module was added alone and the results showed a 1.5% increase in mAP50. It is further shown that three channels in parallel can increase the feature extraction capability. Subsequently, we added the MultiSEAM module to the baseline network along with the OREPA module and showed a 2.5% increase in mAP50. Finally, we added all three modules to the baseline network, and the experimental results showed that the mAP50 increased by 3%, and the improved YOLOv8 algorithm could significantly increase the accuracy of blueberry fruit detection.

3) PERFORMANCE ANALYSIS OF THE TRANSFER LEARNING

In order to be able to fully demonstrate the advantages of Transfer Learning, we do a comparative analysis of two sets of experimental data respectively, starting with the first set of YOLOv8 combined with transfer learning

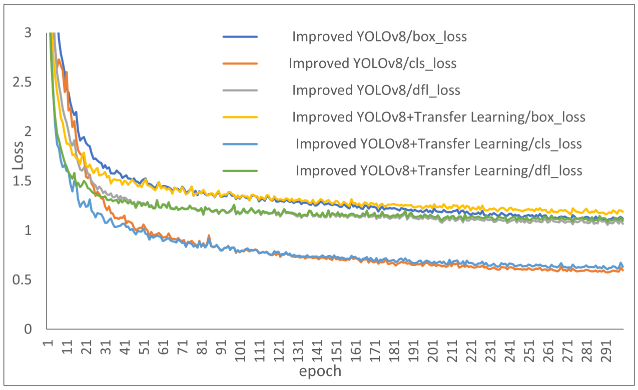

FIGURE 15. Comparison of improved YOLOv8 and improved YOLOv8 + transfer learning loss curves.

comparing to the original YOLOv8 algorithm. As can be seen in Tables 6, the Precision, Recall and mAP are all improved, with a 1.1% improvement in Precision and a 0.9% improvement in mAP50. In the second group, the improved YOLOv8 combined with transfer learning was compared with the improved YOLOv8 algorithm. The results indicate that, except for a slight decrease in accuracy, other metrics have improved. Specifically, the recall rate increased by 1.6%, and mAP50 increased by 0.4%. This confirms that Transfer Learning plays a significant positive role in improving algorithm performance.

Fig. 14 and 15 illustrate the decreasing trend of the three losses throughout the training process, both before and after the implementation of Transfer Learning techniques. The results show that the loss function curve decreases faster with Transfer Learning than without Transfer Learning, especially in the initial phase of training. So Transfer Learning performs better for accelerating the training process. This strategy of knowledge transfer not only helps the model to converge faster and speeds up the whole training process, but also improves the model's ability to generalize to blueberry fruits.

4) COMPARE WITH THE MAINSTREAM ALGORITHM

To be able to evaluate the advantages of TL-YOLOv8, we compare TL-YOLOv8 with other mainstream network

FIGURE 16. Comparison of YOLOv8 and TL-YOLOv8 detection results.

TABLE 7. Model performance comparison.

Model	Size of Model (MB)	Recall(%)	mAP50 (%)	MAP50-95(%)
Faster RCNN	108	51.5	85.2	41
Cascade R-CNN	527	57.8	88.0	47.0
DETR	474	62.2	88.6	53.4
Deformable-DETR	461	50.9	74.2	37.3
ATSS	244	61.6	87.1	50.0
TOOD	244	60.1	84.9	48.0
DINO	543	64.0	59.8	34.3
YOLOv7	284	79.2	82	45.7
YOLOv5	14	85.5	88.7	57.5
TL-YOLOv8	5.9	91.3	94.1	62.7

algorithms using our blueberry dataset. We used model Size, Recall, mAP50 and mAP50-95 as evaluation metrics.

As shown by the results in Tables 7, our TL-YOLOv8 algorithm improves 8.9%, 6.1%, 5.5%, 19.9%, 7.0%, 9.2%, 34.3%, 12.1%, and 5.4% (mAP50) over Faster RCNN, Cascade R-CNN, DETR, Deformable-DETR, ATSS, TOOD, DINO, YOLOv7, and YOLOv5, respectively. Compared to non-YOLO models, TL-YOLOv8 exhibits significant improvements in mAP, Recall, and model size. Notably, our model is substantially smaller than other models in terms of size. Thus, it is evident that TL-YOLOv8 not only ensures the accuracy of object detection but also is more suitable for deployment on resource-constrained devices. In summary, the data show that TL-YOLOv8 has a high blueberry fruit detection performance, to meet the daily detection needs.

Fig. 16 illustrates the comparison of blueberry fruit detection performance across different scenes. The left and right images of (a) are the comparison images of the original YOLOv8 algorithm and TL-YOLOv8 effect on small target blueberry detection, respectively. The left and right images of (b) show the comparison of the effectiveness of the original YOLOv8 algorithm and TL-YOLOv8 for dense blueberry target detection, respectively. The left and right images of (c) show the comparison of the detection effect of the original YOLOv8 algorithm and the TL-YOLOv8 occlusion blueberry target, respectively. The white ellipses in the figure indicate detected blueberry recognition frames and the black ellipses indicate undetected blueberry recognition frames.

IV. CONCLUSION AND PROSPECTS

In order to solve the problems encountered in blueberry detection, such as small fruit size, dense distribution, leaf occlusion, similarity of color features of unripe fruits to leaves, small dataset, and high training cost, we propose a new algorithm of TL-YOLOv8. By introducing an improved MPCA mechanism at the end layer of the backbone network, we have successfully enhanced the efficacy of feature extraction to more accurately identify and localize blueberry fruits in complex backgrounds. Next, we replace the C2f module with the OREPA module, aiming to rapidly accelerate the training process and further enhance the characterization capabilities. To solve the fruit occlusion problem and reduce the leakage rate of small objects, we introduce the MultiSEAM module. Finally, we perform pre-training on the Fruits-360 dataset and subsequently transfer the weight files to the Blueberry dataset for fine-tuning in order to optimize the network parameters and improve the model's generalization ability.

The experiments proved that our algorithm made significant progress in the metrics of detection Precision, Recall, mAP50 and mAP95, achieving 84.6%, 91.3%, 94.1% and 62.7%, respectively. TL-YOLOv8 not only meets the stringent requirements for blueberry fruit detection, but also provides precise localization for automated blueberry picking, thereby simplifying the harvesting process and ensuring increased operational efficiency. Furthermore, it contributes to greater economic benefits for the blueberry industry and drives technological innovation in the field of agricultural automation.

We consider that variations in blueberry fruit color and shadow under different lighting conditions can affect detection accuracy. Additionally, weather conditions (such as cloudy or rainy days) and seasonal changes (such as variations in leaf density and color across different seasons) can also impact the performance of the model. Future work should incorporate data from various lighting conditions, weather,and seasonal scenarios during training to ensure the model's robustness under diverse environmental conditions. Moreover, the introduction of improved MPCA mechanisms and MultiSEAM modules may increase computational complexity, potentially requiring additional computational resources for algorithm deployment, which is disadvantageous for operation on embedded systems or edge devices. Future research should explore model pruning techniques to reduce model size while maintaining accuracy, facilitating deployment on embedded devices.

REFERENCES

[1] M. Cvetković, M. Kočić, D. D. Zagorac, I. Ćirić, M. Natić, D. Hajder, A. Životić, and M. F. Akšić, "When is the right moment to pick blueberries? Variation in agronomic and chemical properties of blueberry (*Vaccinium corymbosum*) cultivars at different harvest times," *Metabolites*, vol. 12, no. 9, p. 798, Aug. 2022.

[2] D. Ding, S. Yu, and Y. Han, "Analysis on the nutritional value and production development status of blueberries," *Field Crop*, vol. 6, no. 2, 2023.

[3] D. A. Yeh, J. Kramer, L. Calvin, and C. E. Weber, "The changing landscape of us strawberry and blueberry markets: Production, trade, and challenges from 2000 to 2020," U.S. Dept. Agricult., Econ. Res. Service, Washington, DC, USA, Tech. Rep. EIB-257, 2000.

[4] G. Xiuming, Z. Yeping, L. Shijuan, Z. Jie, L. Chunyang, and L. Shengping, "Scale adaptive small objects detection method in complex agricultural environment: Taking bees as research object," *Smart Agricult.*, vol. 4, no. 1, p. 140, 2022.

[5] X. Zhu, H. Ma, J.-T. Ji, X. Jin, K.-X. Zhao, and K. Zhang, "Detecting and identifying blueberry canopy fruits based on faster R-CNN," *J. Southern Agricult.*, 2020, doi: 10.3969/j.issn.2095-1191.2020.06.032.

[6] H. Yang, X. Deng, H. Shen, Q. Lei, S. Zhang, and N. Liu, "Disease detection and identification of rice leaf based on improved detection transformer," *Agriculture*, vol. 13, no. 7, p. 1361, Jul. 2023.

[7] R. Gai, N. Chen, and H. Yuan, "A detection algorithm for cherry fruits based on the improved YOLO-v4 model," *Neural Comput. Appl.*, vol. 35, no. 19, pp. 13895–13906, Jul. 2023.

[8] W. Yang, X. Ma, W. Hu, and P. Tang, "Lightweight blueberry fruit recognition based on multi-scale and attention fusion NCBAM," *Agronomy*, vol. 12, no. 10, p. 2354, Sep. 2022.

[9] C. Qiu, G. Tian, J. Zhao, Q. Liu, S. Xie, and K. Zheng, "Grape maturity detection and visual pre-positioning based on improved YOLOv4," *Electronics*, vol. 11, no. 17, p. 2677, Aug. 2022.

[10] C. B. MacEachern, T. J. Esau, A. W. Schumann, P. J. Hennessy, and Q. U. Zaman, "Detection of fruit maturity stage and yield estimation in wild blueberry using deep learning convolutional neural networks," *Smart Agricult. Technol.*, vol. 3, Feb. 2023, Art. no. 100099.

[11] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[12] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[13] C.-Y. Wang, A. Bochkovskiy, and H.-Y.-M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 7464–7475.

[14] N. C. Thompson, K. Greenewald, K. Lee, and G. F. Manso, "The computational limits of deep learning," 2020, *arXiv:2007.05558*.

[15] X. Zhu, F. Jiang, C. Guo, D. Xu, Z. Wang, and G. Jiang, "Surface morphology inspection for directed energy deposition using small dataset with transfer learning," *J. Manuf. Processes*, vol. 93, pp. 101–115, May 2023.

[16] Y. Xu, J. Kou, Q. Zhang, S. Tan, L. Zhu, Z. Geng, and X. Yang, "Visual detection of water content range of seabuckthorn fruit based on transfer deep learning," *Foods*, vol. 12, no. 3, p. 550, Jan. 2023.

[17] O. Attallah, "Tomato leaf disease classification via compact convolutional neural networks with transfer learning and feature selection," *Horticulturae*, vol. 9, no. 2, p. 149, Jan. 2023.

[18] J. Poyatos, D. Molina, A. D. Martinez, J. D. Ser, and F. Herrera, "EvoPruneDeepTL: An evolutionary pruning model for transfer learning based deep neural networks," *Neural Netw.*, vol. 158, pp. 59–82, Jan. 2023.

[19] M. Huang, J. Yin, S. Yan, and P. Xue, "A fault diagnosis method of bearings based on deep transfer learning," *Simul. Model. Pract. Theory*, vol. 122, Jan. 2023, Art. no. 102659.

[20] S. Madhukara and P. R. D. Reddy, "An improved fire detection approach based on YOLO-v8 for smart cities," *Int. J. Adv. Res. Sci., Commun. Technol.*, vol. 35, no. 28, pp. 358–361, May 2024.

[21] J. Terven and D. Cordova-Esparza, "A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS," 2023, *arXiv:2304.00501*.

[22] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 13713–13722.
[23] M. Hu, J. Feng, J. Hua, B. Lai, J. Huang, X. Gong, and X. Hua, "Online convolutional reparameterization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 558–567.
[24] Z. Yu, H. Huang, W. Chen, Y. Su, Y. Liu, and X. Wang, "YOLO-FaceV2: A scale and occlusion aware face detector," 2022, *arXiv:2208.02019*.
[25] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Int. Conf. Artif. Neural Netw.*, vol. 11141, Rhodes, Greece. Berlin, Germany: Springer, Oct. 2018, pp. 270–279.
[26] Kaggle. (2020). *Fruits-360*. [Online]. Available: https://www.kaggle.com/datasets/moltean/fruits

RONGLI GAI received the M.S. and Ph.D. degrees from the University of Science and Technology of China. She is an External Master's Tutor with the University of Chinese Academy of Sciences and the Master's Tutor with Dalian University. From 2017 to 2018, she was a Visiting Scholar with The University of Auckland. Her research interests include machine learning, data mining and network security technology, intelligent agriculture, and other fields of research. She is a member of the CCF Computer Application Professional Committee; the Secretary-General of the Intelligent Agriculture Professional Committee, Dalian Computer Society; and a Reviewer of journals, such as *Small Microcomputer*.

YONG LIU received the bachelor's degree from Zaozhuang College, in 2022. He is currently pursuing the master's degree with the School of Information Engineering, Dalian University. His research interests include computer vision and deep learning.

GUOHUI XU received the M.S. and Ph.D. degrees from Harbin Normal University. She is currently the Director of horticulture with the College of Life and Health, Dalian University; and the Master's Tutor and a Visiting Scholar with the Plant and Food Research Institute (PFR), New Zealand. Her main research interest includes blueberry genetic breeding.