frontiers | Frontiers in Plant Science

TYPE Original Research
PUBLISHED 05 August 2022
DOI 10.3389/fpls.2022.942875

OPEN ACCESS

EDITED BY
Gregorio Egea,
University of Seville, Spain

REVIEWED BY
Jay Kant Pratap Singh Yadav,
Ajay Kumar Garg Engineering College,
India
Xiangnan Li,
Northeast Institute of Geography and
Agroecology (CAS), China

*CORRESPONDENCE
Kun Li
likun0818@wfust.edu.cn

SPECIALTY SECTION
This article was submitted to
Technical Advances in Plant Science,
a section of the journal
Frontiers in Plant Science

RECEIVED 13 May 2022
ACCEPTED 11 July 2022
PUBLISHED 05 August 2022

CITATION
Liu G, Hou Z, Liu H, Liu J, Zhao W and
Li K (2022) TomatoDet: Anchor-free
detector for tomato detection.
Front. Plant Sci. 13:942875.
doi: 10.3389/fpls.2022.942875

TomatoDet: Anchor-free detector for tomato detection

Guoxu Liu[1], Zengtian Hou[2], Hongtao Liu[1], Jun Liu[3,4], Wenjie Zhao[1] and Kun Li[3,4]*

[1]Goertek College of Science and Technology Industry, Weifang University, Weifang, China, [2]School of Intelligent Manufacturing, Weifang University of Science and Technology, Weifang, China, [3]Weifang Key Laboratory of Blockchain on Agricultural Vegetables, Weifang University of Science and Technology, Weifang, China, [4]School of Computer, Weifang University of Science and Technology, Weifang, China

The accurate and robust detection of fruits in the greenhouse is a critical step of automatic robot harvesting. However, the complicated environmental conditions such as uneven illumination, leaves or branches occlusion, and overlap between fruits make it difficult to develop a robust fruit detection system and hinders the step of commercial application of harvesting robots. In this study, we propose an improved anchor-free detector called TomatoDet to deal with the above challenges. First, an attention mechanism is incorporated into the CenterNet backbone to improve the feature expression ability. Then, a circle representation is introduced to optimize the detector to make it more suitable for our specific detection task. This new representation can not only reduce the degree of freedom for shape fitting, but also simplifies the regression process from detected keypoints. The experimental results showed that the proposed TomatoDet outperformed other state-of-the-art detectors in respect of tomato detection. The F_1 score and average precision of TomatoDet reaches 95.03 and 98.16%. In addition, the proposed detector performs robustly under the condition of illumination variation and occlusion, which shows great promise in tomato detection in the greenhouse.

KEYWORDS

tomato detection, anchor-free, CenterNet, deep learning, harvesting robots

1. Introduction

Tomato harvesting is a labor-intensive work, which needs a lot of human resources. It is also very time consuming and includes much tedious work. However, with the development of urbanization and aging of society, the people in the countryside have decreased a lot, and the labor cost continues to increase, resulting in a big labor shortage in farming work (Yue et al., 2015). On the other side, intelligent agriculture is developing fast in the past decades, which is an ideal substitute of human resources for farming work. Among the various technologies applied in the agriculture, the fruit harvesting robot is one of the prominent artificial intelligent techniques. It has huge potential efficiency in fruit harvesting, which can bring high profit as well as liberating the labor force. Thus, it is of great value and significance to develop harvesting robots.

A harvesting robot usually consists of two components—a vision system and an eye-hand coordination system (Zhao et al., 2016a). The vision system plays a key role in

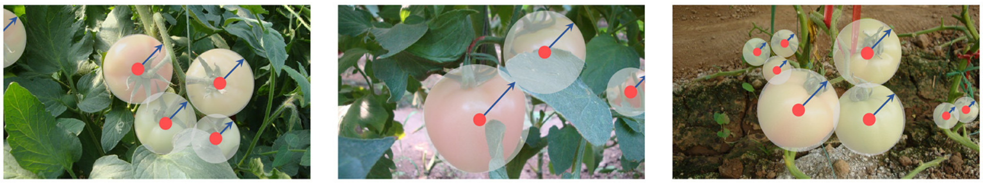

FIGURE 1
A tomato is modeled as a center point of its bounding circle. The radius of the bounding circle can be inferred from the keypoint at the center.

the whole system, since the first critical step for the harvesting robot is to detect fruits autonomously. This step determines the detection and subsequent picking accuracy of harvesting robots. Thus, it is very crucial to develop a robust fruit detection algorithm of the vision system. However, at present, no harvesting robot has been commercialized successfully due to either low detection accuracy or low detection speed. Many factors have hindered the pace of harvesting robot development such as uneven illumination, occlusion, overlap, and some other unpredictable factors (Gongal et al., 2015).

To deal with the above challenges, many researchers have studied fruit detection over the past years. In the early years, some researchers used threshold discriminant methods for fruit detection based on color, shape, texture, or fusion of them (Linker et al., 2012; Kelman and Linker, 2014; Wei et al., 2014), and achieved reasonable detection results. Bulanon et al. (2002) used an optimal threshold extracted from the intensity histogram of a red-color-difference enhanced image for apple recognition. The results showed that the success rate exceeds 88%. This method is restricted to ripe apples which present different color to the background. Okamoto and Lee (2009) employed hyperspectral imaging for detection of green citrus. The method is separated into pixel-wise segmentation process using pixel discrimination functions and fruit recognition process with thresholds selected by trial and error. This method greatly relies on the selection of several optimal thresholds, and thus is lack of robustness when the fruit environment changes. Inspired by the eigenface concept, Kurtulmus et al. (2011) proposed a novel eigenfruit feature for green citrus detection, combined with color and circular gabor texture. Although intrinsic texture features are used other than only color features, the method still confuses some fruits with background and does nothing with severe occluded fruits. Zhao C. et al. (2016) developed a cascaded pixel segmentation method for immature citrus detection in natural environment. Three color feature maps and a block matching method are adopted to identify potential fruit pixels. Finally, an SVM classifier is used to remove false detections. Nevertheless, with only color feature for segmentation in the early stage, many fruits are missed by the method due to similarity between green fruits and background. Zhao et al. (2016b) proposed a multi color feature fusion method based on wavelet transformation for mature tomato recognition. The detection accuracy reaches 93%. However, since only color features are employed, the method is inferred to be sensitive to illumination variation. These methods greatly rely on the selection of suitable thresholds, making them sensitive to the changes in the form of fruit presentation, such as illumination variation and occlusion.

With the development of machine learning, many researchers tried to apply them to fruit detection, such as adaboost, support vector machine (SVM) or other statistical classifiers (Kurtulmus et al., 2014; Lv et al., 2014; Yamamoto et al., 2014), and get better results than the threshold discriminant methods. Zhao et al. (2016c) used an adaboost classifier associated with haar features for tomato detection. An average pixel value feature is adopted for the removal of false detections. More than 96% of tomatoes are detected in their study. Li et al. (2017) proposed to use an SVM trained on histogram-based features for green and ripe tomato recognition. Prior to detection, the fast normalized cross correlation method is used to extract the potential tomato regions. Finally, the circular hough transform and color analysis are combined to obtain tomato positions. Behroozi-Khazaei and Maleki (2017) proposed to use an artificial neural network optimized by genetic algorithm for grape cluster detection. Also, the genetic algorithm is adopted for color feature selection, which subsequently serves as input to the network. A Bi-Layer schema was proposed for automatic detection of ripening tomatoes by Wu et al. (2019). In their method, a weighted relevance vector machine is used for tomato recognition based on six color-related features and five textural features. A detection rate of 94.90% is reported in the results. Liu et al. (2019) developed a coarse-to-fine method for ripe tomato detection in the greenhouse. First, a naïve bayes classifier is used to identify potential tomato area, on which an SVM classifier combined with histogram of oriented gradients is applied to recognize tomatoes. At last, a color analysis method is proposed to remove false detection. The machine learning methods usually achieve better performance than threshold discriminant methods. However, the low-level abstraction capabilities of hand-crafted features make it difficult to adapt these methods to complicated environmental change.

The emergence of deep learning methods especially convolutional neural networks provides a new paradigm for computer vision tasks, including fruit detection tasks (Sa et al., 2016; Tian et al., 2019; Zheng et al., 2021). These methods can learn feature representations directly from the data and can be trained end-to-end. Nevertheless, the detection accuracy and robustness still need to be improved to enable real commercial applications under complicated conditions as discussed above.

To address the above problems, this study proposes an effective anchor-free detector called TomatoDet for tomato detection. The proposed model represents a tomato by the center point of its bounding circle, as shown in Figure 1. First, to improve the expression ability of the backbone network, an attention mechanism is introduced to guide the network to pay more attention to the region of interest (ROI), especially small tomatoes. Second, a bounding circle is adopted for tomato localization instead of the traditional bounding box, which is commonly used for general object localization.

Our main contribution is three-fold as follows:

1. The Convolutional Block Attention Module is introduced into the backbone network of CenterNet (Zhou et al., 2019) called Attentive-DLA34 to boost the representation power.
2. A circle representation for tomato detection is adopted to adapt the traditional detection methods to our specific detection task. The new circle representation not only reduces the degree of freedom for shape fitting, but also simplifies the regression process from detected keypoints.
3. Extensive experiments are conducted on tomato datasets. We show that the proposed TomatoDet achieves better performance in terms of both accuracy and robustness, compared to the original CenterNet and other state-of-the-art object detectors.

2. Related work

In recent years, deep learning methods have shown continuous performance improvements on fruit detection. A "MangoYOLO" detector was proposed for fruit detection and fruit load estimation by Koirala et al. (2019). This model combines the advantages of YOLOv2 (Redmon and Farhadi, 2017) and YOLOv3 (Redmon and Farhadi, 2018), which has both high detection speed and detection accuracy. It outperforms other methods such as Faster R-CNN (Ren et al., 2015), YOLOv2 (Redmon and Farhadi, 2017), YOLOv3 (Redmon and Farhadi, 2018), and SSD (Liu et al., 2016), on their Mango dataset. Bresilla et al. (2019) improved YOLO (Redmon et al., 2016) model for apples and pears detection. First, the grid-scale is scaled up twice to fit the size of the fruits. Second, the model is pruned to improve the detection speed while not degrading the accuracy. Afonso et al. (2020) applied Mask R-CNN to the tomato dataset for detection. Several neural networks are used as backbone for feature extraction. The best F_1 score reaches over 94% in their report. Liu G. et al. (2020) proposed a YOLO-Tomato for tomato detection based on YOLOv3 (Redmon and Farhadi, 2018). A dense architecture is incorporated to the backbone to facilitate feature reuse, and a circular bounding box is adopted to optimize the non-maximum suppression process. The model achieves a competing performance compared to state-of-the-art detection methods. Zheng et al. (2021) improved YOLOv4 (Bochkovskiy et al., 2020) for green citrus detection. First, the backbone network is trimmed to reduce detection time. Then, a novel Bi-PANet is proposed to fuse features from different layers. With the modifications, the detection accuracy is reported to be 86% on their dataset. Zhang et al. (2021) developed an edge-device oriented lightweight model for fruit detection. The structure of the original CSPNet is lightened to boost detection speed, and a deep-shallow feature fusion model is proposed to enhance the expression ability of the network. Tested on three types of edge devices, the average detection precision reaches 93, 84.7, and 85% for oranges, tomatoes, and apples, respectively. Wei et al. (2022) proposed a green fruit detection model based on D2Det. By incorporating MobileNetV2, feature pyramid networks and region proposal network structure into the original model, the detection accuracy of green fruits in orchard environments was greatly improved. Chen et al. (2022) improved YOLOv4 for the detection of citrus by incorporating an attention mechanism and a depthwise separable convolution module. In addition, a pruning algorithm was applied to remove the influence of irrelevant latent factors of the data.

Although exciting results are achieved by the above methods, there is still much room for optimization of the networks to improve detection performance. Moreover, the above methods are all anchor-based methods, which commonly perform nearly exhaustive anchor classification over the image and have many hyperparameters for anchor design, reducing the detection efficiency.

3. Materials and methods

3.1. Image acquisition

The images used in this study are captured using a digital camera (Sony DSC-W170, Tokyo, Japan) with a resolution of 3,648 × 2,056 pixels in a Tomato Production Base, which is located in Shouguang City, Shandong Province, China. The datasets are collected under various environment conditions including sunlight, shading, occlusion, and overlap, etc. Some examples captured under different conditions are shown in Figure 2.

To verify the proposed method, the datasets are split into two subsets—a training set and a test set. The training set contains 725 images, and 241 images are included in the test set. Totally, 966 images are used in this study. For data labeling, a tool

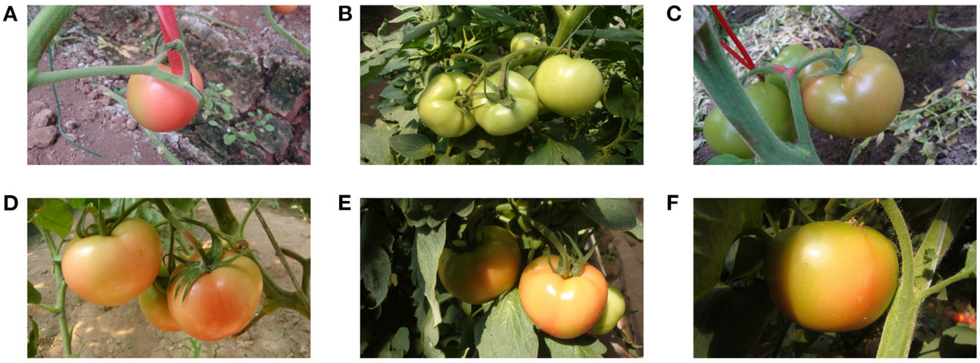

FIGURE 2
Some tomato samples with different growing circumstances: **(A)** a single tomato, **(B)** a cluster of tomatoes, **(C)** occlusion case, **(D)** overlap case, **(E)** shading case, and **(F)** sunlight case.

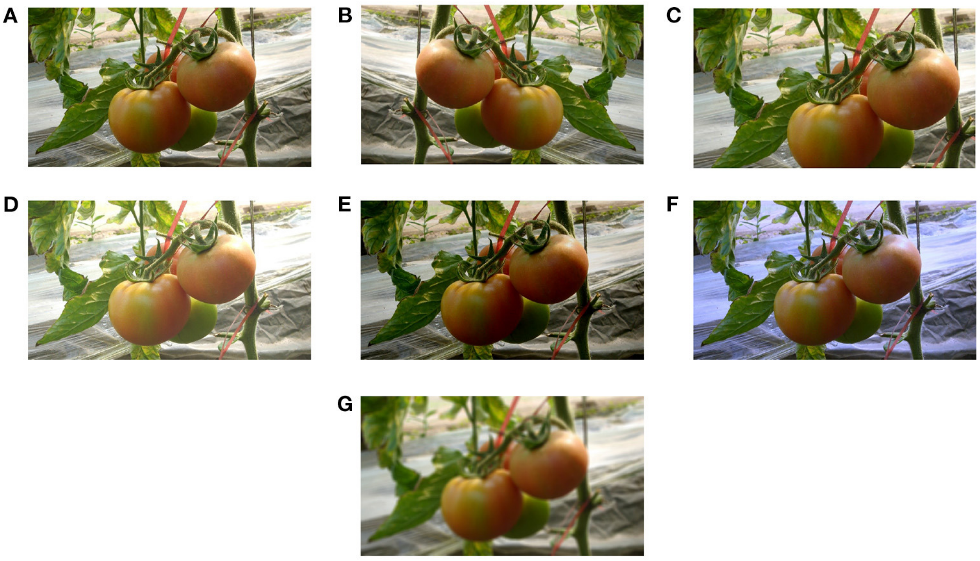

FIGURE 3
Data augmentation of tomato images: **(A)** original image, **(B)** horizontal flip, **(C)** scaling and cropping, **(D)** high brightness, **(E)** low brightness, **(F)** color balancing, and **(G)** blur processing.

called Label-Tomato has been developed to annotate images with proposed bounding circles based on Python. The output format of Label-Tomato is txt files, which include the numbers and locations of tomatoes for each image.

3.2. Data augmentation

To avoid over-fitting of the model in the training process, the data augmentation is used in this study to simulate real-life interference and enhance the richness of the collected datasets. Several image processing technologies are adopted for augmentation - horizontal flip, scaling and cropping, brightness transformation, color balancing and image blurring, as shown in Figure 3. For the brightness transformation, we use a factor falling in the range [0.6, 1.4] to change the intensity of the pixels in the image randomly. This process can simulate different weather factors on the image intensity. For the scaling and cropping operation, we follow the same process as in Liu G. et al. (2020). To eliminate the effect of lighting on color rendering, we adopt the gray world algorithm (Lam, 2005) for color balancing. Finally, we randomly blur the augmented images by flip, scaling and cropping, brightness transformation, and color balancing to simulate indistinct images caused by camera movement. After data augmentation, the whole number of resultant images is shown in Table 1.

TABLE 1 The number of training images after data augmentation.

	Original	Flip	Scaling and cropping	Brightness	Color	Blur	Total
No. of tomato images	725	725	725	1,450	725	725	5,075

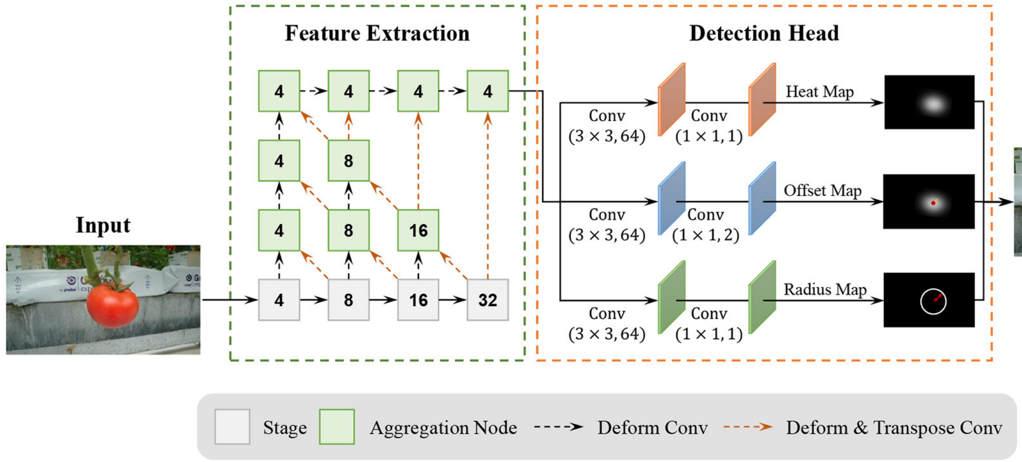

FIGURE 4
An overview of the proposed model.

3.3. Overview of tomatoDet

Our tomato detection model, called TomatoDet, pools several concepts from the past work with our novel idea to improve the detection performance. An overview of the proposed model is shown in Figure 4. The proposed TomatoDet is based on CenterNet and consists of two modules. The first module is used for feature extraction. It adopts Deep Layer Aggregation-34 (DLA34) (Yu et al., 2018) as the backbone and incorporates Convolutional Block Attention Module (CBAM) (Woo et al., 2018) to improve the feature expression ability and guide the network to focus on small-scale tomato targets. The second module is the detection head. The architecture of the detection head is like that of CenterNet, except that we use a radius head instead of the height and width head for bounding circle prediction. More details are presented in Sections 3.4 and 3.5.

3.4. The proposed attentive-DLA34 backbone

In this study, an attentive Deep Layer Aggregation network (Attentive-DLA34) is proposed as the base backbone for feature extraction. The DLA is inspired by dense connection and feature pyramid and has two main structures: the iterative deep aggregation (IDA) and the hierarchical deep aggregation (HDA). The IDA is mainly used for feature fusion across resolutions and scales while the HDA focuses on semantic fusion, i.e., aggregating features from different channels and depths in a tree-based structure. Based on these two structures, the DLA could make better use of spatial and semantic information for recognition and localization. However, the complicated conditions make it challenging to detect tomatoes in a natural environment, not to mention the existence of a large number of small tomatoes. To mitigate this problem, we introduce an attention mechanism—Convolutional Block Attention Module (CBAM)—into the backbone network to guide it to pay more attention to the region of interest (ROI). The architecture of the proposed Attentive-DLA34 model is shown in Figure 5.

As shown in Figure 5, we replace the original layers in each stage with CBAM to focus its attention on tomato areas. For CBAM, it is divided into a channel attention module and a spatial attention module in a sequential manner. First, the channel attention module takes the input and infers a 1D channel attention map. Then, the multiplication output of the input and the attention map is inputted to the spatial attention module to get the final output feature map in the same way. The detailed operation can be depicted in Equations (1) and (2):

$$F' = M_c(F) \otimes F \tag{1}$$

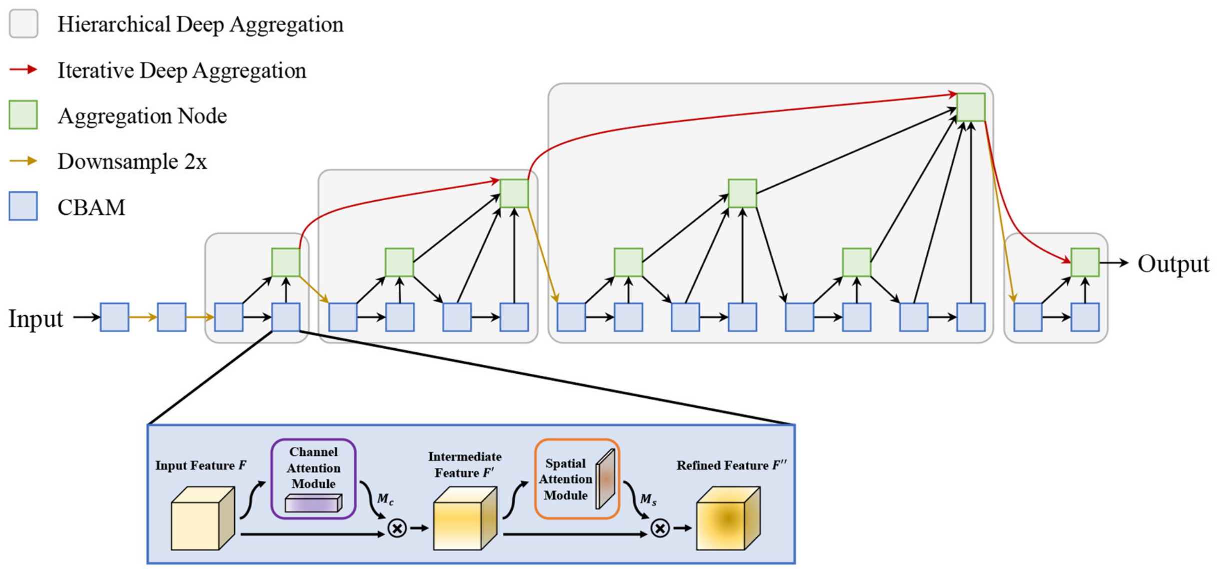

FIGURE 5
The proposed attentive-DLA34 model.

$$F'' = M_s\left(F'\right) \otimes F' \tag{2}$$

where $\otimes$ indicates element-wise multiplication, $F \in R^{C\times H\times W}$ is the input feature map, $M_C \in R^{C\times 1\times 1}$ denotes the generated channel attention map, and $M_s \in R^{1\times H\times W}$ denotes the generated spatial attention map. F'' is the final output by CBAM.

3.5. Circle representation

For general object detection, a bounding box is usually adopted for object localization. However, this type of detection representation is not optimal for specific objects which have a particular shape. In this study, since our detection target is tomato, which is roughly circular, it is better to use bounding circles instead of bounding boxes for localization. It has three folds of advantages. Firstly, compared with bounding boxes, bounding circles could better match the shape of tomatoes. Secondly, the representation of a circle is simpler than that of a box, which makes it easier for the network to learn. Lastly, the circle is invariant to rotation.

3.5.1. From point to bounding circle

For an input image $I \in R^{W\times H\times 3}$ with width W and height H, the target is to produce a keypoint heatmap $\hat{Y} \in [0,1]^{\frac{W}{K}\times\frac{H}{K}\times C}$, where K is the downsampling ratio of output and C is the number of classes. A prediction from the heatmap $\hat{Y}_{x,y,c} = 1$ denotes a detected keypoint, and $\hat{Y}_{x,y,c} = 0$ denotes background. Following Law and Deng (2018), the ground truth of the keypoints is mapped onto a heatmap Y using a 2D Gaussian kernel as in Equation (3):

$$Y_{x,y,c} = \exp\left(-\frac{\left(x-\tilde{p}_x\right)^2+\left(y-\tilde{p}_y\right)^2}{2\sigma_p^2}\right) \tag{3}$$

where $\tilde{p}_x$ and $\tilde{p}_y$ are the equivalent groundtruth keypoints of prediction, and they are downsampled by the factor K from the original keypoint p and are then discretized. σ_p is a kernel standard deviation.

After getting the peaks of the heatmap for tomatoes, the top N peaks are selected among all the detected responses whose value is greater or equal to its eight-connected neighbors. We define $\hat{\mathcal{P}} = \left\{\left(\hat{x}_i, \hat{y}_i\right)\right\}_{i=1}^{N}$ as the set of N detected center points. The confidence of the detected bounding circle is represented by the keypoint values $\hat{Y}_{x_i,y_i,c}$, and the center point $\hat{p}$ and radius $\hat{r}$ of the bounding circle is denoted as follows:

$$\hat{p} = \left(\hat{x}_i + \Delta\hat{x}_i, \hat{y}_i + \Delta\hat{y}_i\right) \tag{4}$$

$$\hat{r} = \hat{R}_{\hat{x}_i,\hat{y}_i} \tag{5}$$

where $\left(\Delta\hat{x}_i, \Delta\hat{y}_i\right) = \hat{O}_{\hat{x}_i,\hat{y}_i} \in R^{\frac{W}{K}\times\frac{H}{K}\times 2}$ is the offset prediction and $\hat{R}_{\hat{x}_i,\hat{y}_i} \in R^{\frac{W}{K}\times\frac{H}{K}\times C}$ is the radius prediction.

3.5.2. Bounding circle IOU

The intersection-over-union (IOU) is commonly used to evaluate the similarity of two bounding boxes. In this study,

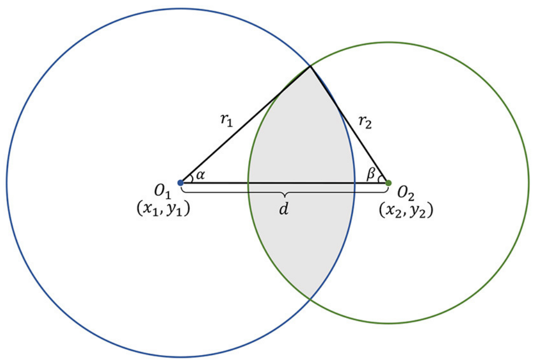

FIGURE 6
The schematic diagram of cIOU.

we introduce a circle IOU (cIOU) for evaluation of two bounding circles.

As shown in Figure 6, denoting the center coordinates of two intersected circles O_1 and O_2 be (x_1, y_1) and (x_2, y_2), respectively, the distance between two centers d can be represented in Equation (6) and satisfies the condition $|R - r| \leq d \leq |R + r|$.

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{6}$$

The angles α and β can be calculated as:

$$\alpha = \cos^{-1} \frac{r_1^2 + d^2 - r_2^2}{2 r_1 d} \tag{7}$$

$$\beta = \cos^{-1} \frac{r_2^2 + d^2 - r_1^2}{2 r_2 d} \tag{8}$$

Then, the intersection area $A_{O_1 \cap O_2}$ and union area $A_{O_1 \cup O_2}$ of circles O_1 and O_2 can be derived as in Equations (9) and (10).

$$A_{O_1 \cap O_2} = \alpha r_1^2 + \beta r_2^2 - \frac{1}{2} r_1^2 \sin 2\alpha - \frac{1}{2} r_2^2 \sin 2\beta \tag{9}$$

$$A_{O_1 \cup O_2} = \pi r_1^2 + \pi r_2^2 - A_{O_1 \cap O_2} \tag{10}$$

Consequently, the cIOU can be represented as follows:

$$\text{cIOU} = \frac{(2\alpha - \sin 2\alpha) r_1^2 + (2\beta - \sin 2\beta) r_2^2}{(2\pi - 2\alpha + \sin 2\alpha) r_1^2 + (2\pi - 2\beta + \sin 2\beta) r_2^2} \tag{11}$$

3.6. Loss function

The loss function of TomatoDet in the training stage consists of three parts, i.e., the keypoint heatmap loss, bounding circle radius loss and center offset loss. The keypoint heatmap loss L_{hm} is based on focal loss (Lin et al., 2017) as in Equation (12).

$$L_{hm} = -\frac{1}{N} \sum_{x,y,c} \begin{cases} \left(1 - \hat{Y}_{x,y,c}\right)^{\alpha} \log \hat{Y}_{x,y,c} & \text{if } Y_{x,y,c} = 1 \\ \left(1 - Y_{x,y,c}\right)^{\beta} \left(\hat{Y}_{x,y,c}\right)^{\alpha} \log\left(1 - \hat{Y}_{x,y,c}\right) & \text{otherwise} \end{cases} \tag{12}$$

where N is the number of keypoints in an image, and α and β are hyper-parameters for the focal loss. In this study, α and β are set to be 2 and 4 following Zhou et al. (2019).

To rectify the keypoint location error resulting from the discretization of downsampling, an offset loss L_{off} is designed to measure the difference between the predicted offset $\hat{O}$ and the groundtruth O based on L1 loss.

$$L_{off} = \frac{1}{N} \sum_{p} \left| \hat{O}_{\tilde{p}} - O_{\tilde{p}} \right| \tag{13}$$

The tomato radius is regressed from the center points optimized by the radius loss L_r in Equation (14).

$$L_r = \frac{1}{N} \sum_{k=1}^{N} \left| \hat{R}_{p_k} - r_k \right| \tag{14}$$

where $\hat{R}_{p_k}$ and r_k denotes the predicted and groundtruth radius of the kth tomato, and N represents the number of results.

Above of all, the total loss of TomatoDet is denoted as in Equation (15).

$$L_{det} = L_{hm} + \lambda_{off} L_{off} + \lambda_r L_r \tag{15}$$

where $\lambda_{off} = 1$ and $\lambda_r = 0.1$ are used in our experiment to balance different losses, referring to Zhou et al. (2019).

3.7. Experimental setup

The experiments are performed on a Ubuntu 16.04 with an Intel(R) Core(TM) i7-9700 K CPU@3.60 GHz. It is accelerated by an NVIDIA GeForce GTX 1080Ti GPU. The proposed TomatoDet model is implemented on Pytorch.

The model is trained on an input resolution of 512 × 512 pixels. It is trained with a batch size of 8 and an initial learning rate of 1.25e-4 for 140 epochs. The learning rate is then dropped 10 at 90 and 120 epochs, respectively.

To evaluate the performance of the proposed method, recall (R), precision (P), and F_1 score are used as the criterion indexes. They are defined in Equations (16)–(18):

$$P = \frac{TP}{TP + FP} \tag{16}$$

$$R = \frac{TP}{TP + FN} \tag{17}$$

$$F_1 = \frac{2 \times P \times R}{P + R} \tag{18}$$

where TP, FP, and FN represent true positives (correct detections), false positives (false detections), and false negatives (missing detections), respectively.

Besides, the average precision (AP) is adopted in this study to evaluate the overall detection performance. AP is defined as follows:

TABLE 2 Ablation study on the major components of TomatoDet.

Attention module	Circle representation	Recall (%)	Precision (%)	F_1 (%)	AP (%)
		91.56	92.98	92.26	95.75
✓		92.87	94.32	93.59	97.11
	✓	92.98	94.43	93.70	96.98
✓	✓	94.30	95.77	95.03	98.16

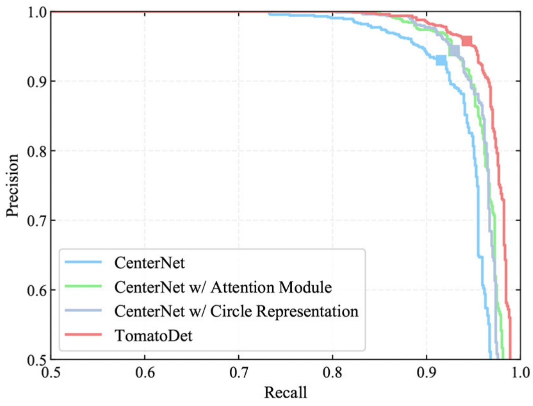

FIGURE 7
PR curves of the major components of TomatoDet for ablation study. The markers indicate the points where recall and precision are obtained when the prediction confidence threshold equals 0.6.

TABLE 3 Tomato detection results of different algorithms.

Methods	Recall (%)	Precision (%)	F_1 (%)	AP (%)	(ms) (ms)
YOLOv2	86.18	87.24	86.71	88.46	30
YOLOv3	90.89	91.60	91.24	94.06	45
YOLO-Tomato	93.09	94.75	93.91	96.40	54
YOLOv4	92.76	94.11	93.43	96.59	25
Faster R-CNN	91.78	92.89	92.33	94.37	231
CenterNet	91.56	92.98	92.26	95.75	32
TomatoDet	94.30	95.77	95.03	98.16	35

$$AP = \sum_{n} \left(r_{n+1} - r_n\right) p_{interp}\left(r_{n+1}\right) \tag{19}$$

$$p_{interp}\left(r_{n+1}\right) = \max_{\tilde{r}:\tilde{r} \geq r_{n+1}} p(\tilde{r}) \tag{20}$$

where $p(\tilde{r})$ is the measured precision at recall $\tilde{r}$.

4. Results and discussion

4.1. Ablation study

In this study, an attention mechanism and a circle representation are incorporated to the proposed detector. In order to evaluate the effectiveness of each component, an ablation study is performed on the tomato dataset. The results of the ablation experiments are shown in Table 2 and Figure 7.

From Table 2, we can see that the incorporation of the attention mechanism brought a significant improvement of all the indexes including the recall, precision, F_1 score and average precision (AP). The F_1 score and AP increases by 1.33 and 1.36%, respectively. This verifies the advantages of the proposed attentive-dla34 backbone, which optimizes the focus of the network and boosts the representation power. We also performed a contrast experiment to verify the effectiveness of the circle representation. With circle representation, the F_1 score and AP increases by 1.44 and 1.23%, respectively, as shown in Table 2. This benefits from the intrinsic shape fitting of the new circle representation to tomatoes, which can reduce the degree of freedom of the rectangle representation and simplify the regression process from detected keypoints. We also show the precision-recall (PR) curves of different components in Figure 7. The markers indicate the points where recall and precision are obtained when the confidence threshold equals 0.6. It can be

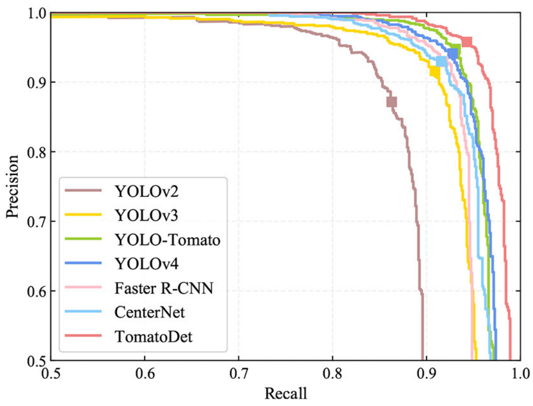

FIGURE 8
PR curves of different detection algorithms.

seen that the detection performance improves significantly with the incorporation of different components.

4.2. Comparison of different methods

To verify the performance of the proposed TomatoDet model, we designed a comparative experiment of the state-of-the-art detection algorithms, including YOLOv2 (Redmon and Farhadi, 2017), YOLOv3 (Redmon and Farhadi, 2018), YOLO-Tomato (Liu G. et al., 2020), YOLOv4 (Bochkovskiy et al., 2020), Faster R-CNN (Ren et al., 2015), CenterNet (Zhou et al., 2019), and the proposed model. Among all of these algorithms, the Faster R-CNN is a two-stage detector, and the others are one-stage detectors. Moreover, CenterNet and the proposed TomatoDet are anchor-free detectors, while the remaining are all anchor-based methods.

The recall, precision, F_1 score, average precision (AP), and average detection time are the evaluation indicators, as shown in Table 3. The precision-recall (PR) curves of different detection models are shown in Figure 8. In terms of detection performance, one can see that the proposed TomatoDet is superior to the other five methods. The F_1 score of TomatoDet is 95.03%. It is 1.12% higher than that of YOLO-Tomato,

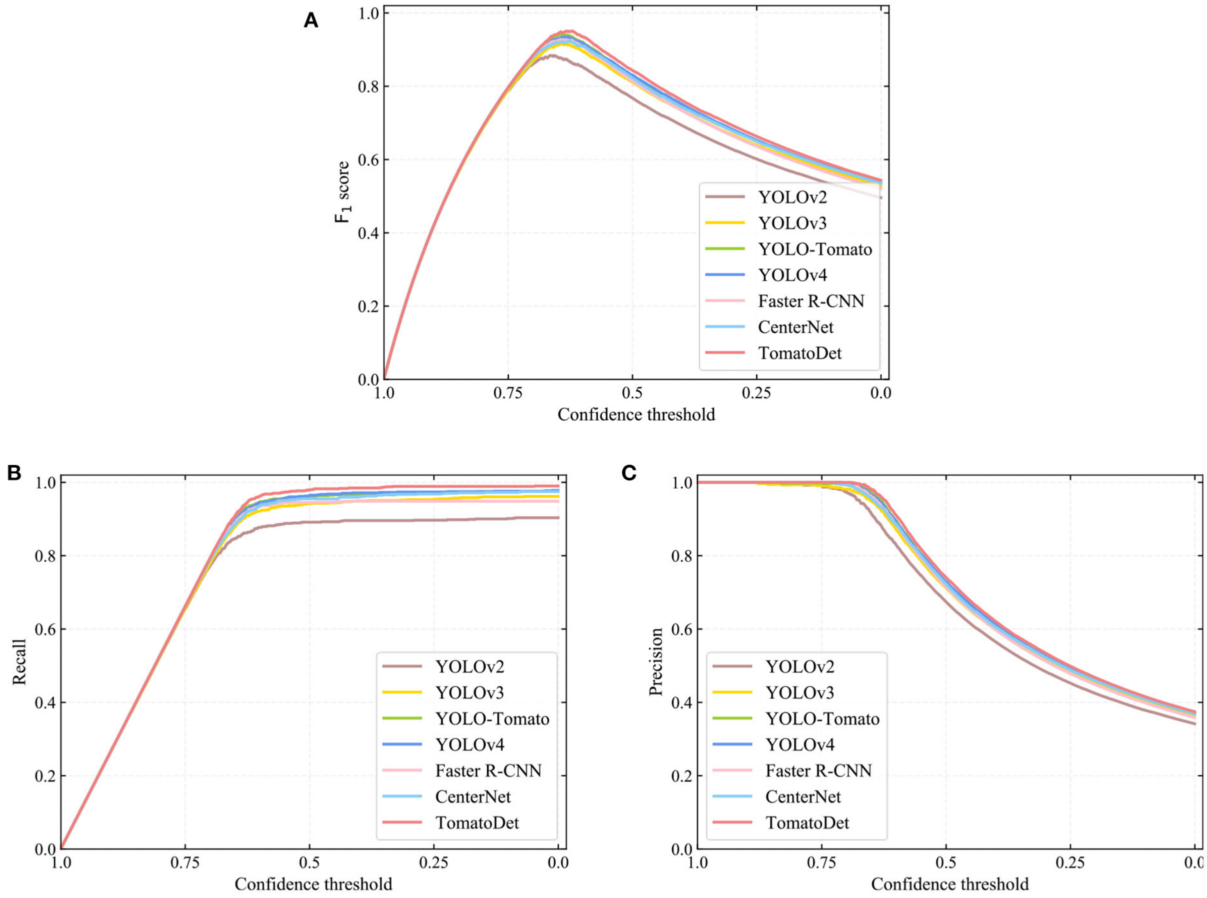

FIGURE 9
The **(A)** F_1, **(B)** recall, and **(C)** precision curves of different detection algorithms.

TABLE 4 Performance of the proposed TomatoDet under different lighting conditions.

Illumination	Tomato count	Correctly identified		Falsely identified		Missed	
		Amount	Rate(%)	Amount	Rate (%)	Amount	Rate (%)
Sunlight	487	460	94.46	22	4.56	27	5.54
Shading	425	400	94.12	16	3.85	25	5.88

which obtains the second-best performance. In terms of AP, TomatoDet performs 1.76 and 1.57% better than YOLO-Tomato and YOLOv4, respectively. Compared to CenterNet, the proposed TomatoDet is about 2.8 and 2.4% higher in terms of F_1 score and AP, respectively. We also show the F_1, recall and precision curves in Figure 9, separately. In accordance with the PR curves, they demonstrate the superiority of the proposed TomatoDet over other methods. This verifies the effectiveness of the proposed modifications. The introduction of CBAM guides the model to pay more attention to the ROI and thus improves the feature expression ability of the network. Besides, the adoption of bounding circles makes it easier to regress from center points to the size as the bounding circle only has one parameter, i.e., radius. Furthermore, bounding circles could match the shape of tomatoes better in nature and improve the IOU. The average detection time of the proposed model reaches 0.036 s per image. It is about 0.2 s less than Faster R-CNN and almost the same as the YOLOv2 model. The experimental results show that the proposed TomatoDet could detect tomatoes in complex environments in real-time with strong robustness.

4.3. Qualitative analysis

To better understand the prediction ability of our proposed TomatoDet, the output feature is visualized. Figure 10 shows

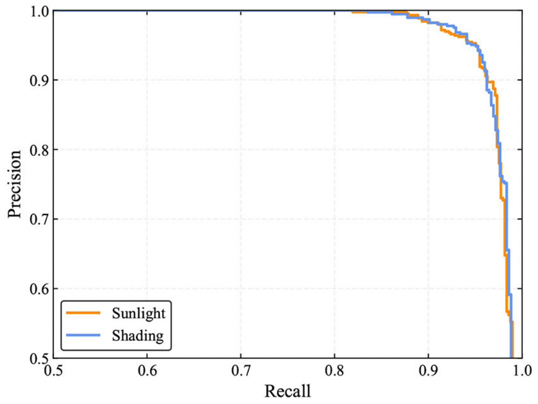

FIGURE 11
PR curves of the proposed method under different lighting conditions.

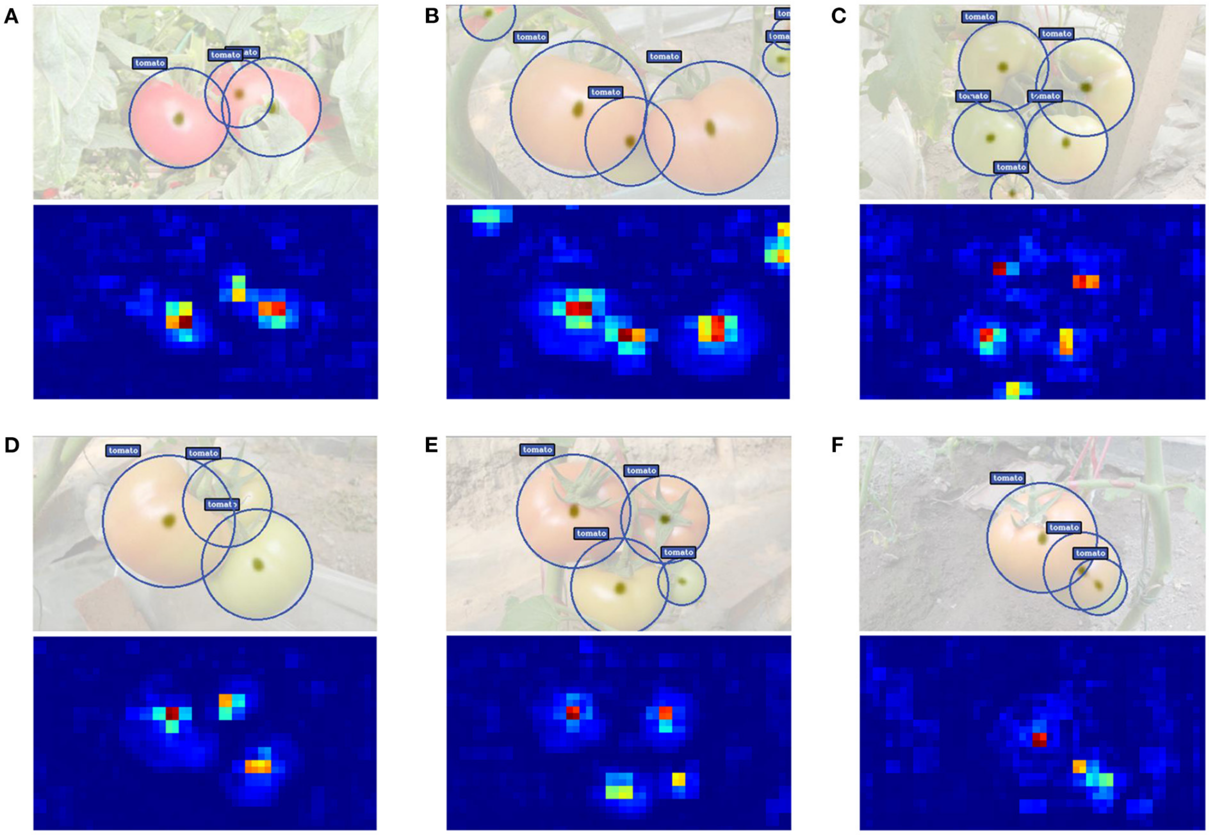

FIGURE 10
(A–F) Some examples of detection results along with the output heatmap.

some examples of detection results along with the output heatmap. From the second row of the subfigures, one can see that through the proposed attentive-DLA34 backbone, the heatmap almost only fires at the area of tomatoes, including small and severe occluded ones. This benefits from the combination of CBAM and DLA34, which emphasizes the meaningful features throughout the network and thus boosts the representation power. Further, the keypoints for tomatoes are extracted from the peaks of the heatmap and are then regressed to the radius of the proposed bounding circle, which reduces the degree of freedom of fitting compared to the traditional bounding boxes, as is shown in the first row of the subfigures.

4.4. Performance of the proposed model under different lighting conditions

In the natural environment, tomatoes may be exposed to different lighting conditions due to uneven illuminations. The performance of the proposed TomatoDet under different lighting conditions is evaluated in this study. Among all the tomatoes in the test set, 425 tomatoes are in shading conditions, while 487 tomatoes are in sunlight conditions. The correct identification rate (or recall), false identification rate and missing rate are used as evaluation indicators.

As shown in Table 4, 460 out of 487 tomatoes are correctly identified by the TomatoDet under sunlight conditions. The counterpart is 400 out of 425 for the shading conditions. The correct identification rates are comparable. The false identification rates are 4.56 and 3.85% for sunlight and shading conditions, respectively. This means that some of the detections are falsely recognized as tomatoes, which in fact are leaves, branches, or other backgrounds. This occurs when the background presents similar color and shape to tomatoes. The above results show that the proposed method is robust under different lighting conditions in real scenes. From Figure 11, one can see that the PR curves under sunlight and shading conditions are comparable, showing the robustness of the proposed method to different lighting conditions. Some examples are shown in Figure 12.

4.5. Performance of the proposed model under different occlusion conditions

In the greenhouse, tomatoes are inevitably obscured by leaves or branches and overlap with each other. This will have a certain impact on tomato detection. In this study, we also evaluate the performance of the proposed method under different occlusion conditions. As in YOLO-Tomato (Liu G. et al., 2020), depending on the degree of occlusion or overlap,

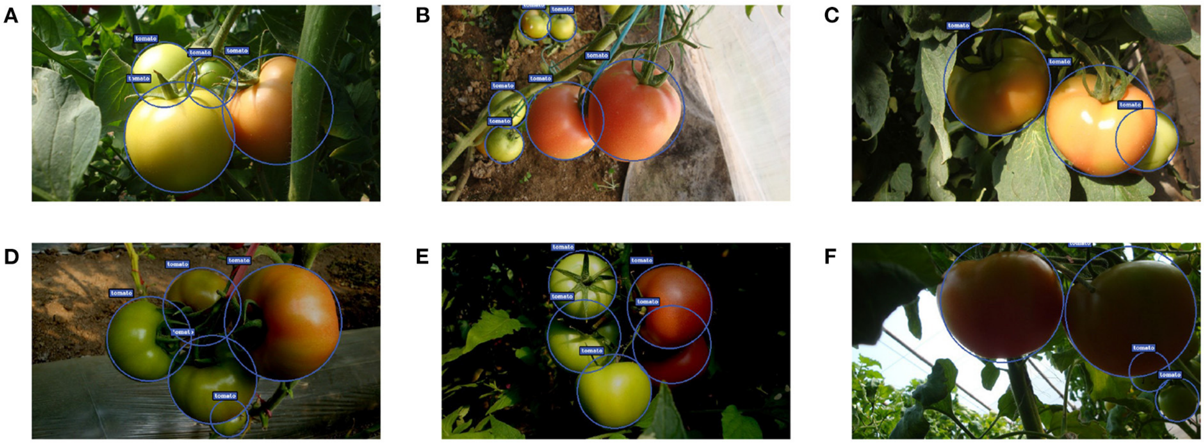

FIGURE 12
Some examples of the detection results under different lighting conditions: **(A–C)** sunlight conditions, and **(D–F)** shading conditions.

TABLE 5 Performance of the proposed TomatoDet under different occlusion conditions.

Occlusion condition	Tomato count	Correctly identified		Falsely identified		Missed	
		Amount	Rate (%)	Amount	Rate (%)	Amount	Rate (%)
Slight case	609	576	94.58	22	3.68	33	5.42
Severe case	303	284	93.73	16	5.33	19	6.27

we classify tomatoes as slight and severe occlusion cases. Severe cases refer to tomatoes being blocked by leaves, branches, or other tomatoes by more than 50% degrees. Conversely, tomatoes are regarded as slight cases. The detection results are shown in Table 5 and Figure 13.

Based on the above experiments, one can see that the detection performance for tomatoes under slight occlusion cases is marginally better than that of tomatoes under severe cases. This shows that occluded and overlapped tomatoes cause inaccurate detections. Nevertheless, most of the occluded and overlapped tomatoes can be detected by our model correctly. This is achieved by the accurate keypoints estimation resulting from the implicit contextual information utilization of the convolutional neural networks since the networks learn hierarchical features through multiple levels of abstraction. However, it is believed that the detection performance of occluded tomatoes can be further improved by exploiting contextual information explicitly (Liu L. et al., 2020). Figure 14 shows some examples of detection results for both cases.

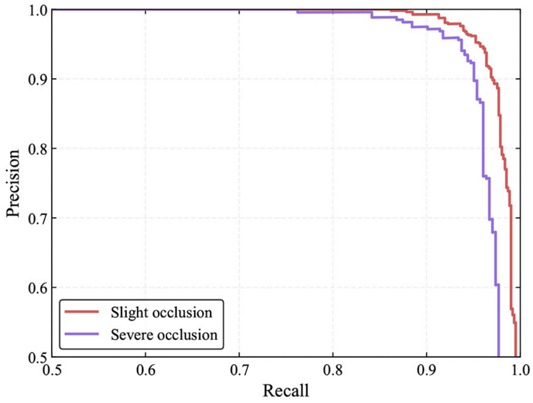

FIGURE 13
PR curves of the proposed method under different occlusion conditions.

5. Conclusions and future work

In this study, we propose TomatoDet, an improved anchor-free detector for tomato detection based on CenterNet. The proposed detector incorporates an attention mechanism to optimize the focus of the network and thus boost the representation power. In addition, a circle representation is introduced to adapt the detector to our specific detection task. With circle representation, the degree of freedom for tomato fitting is reduced and the regression process from keypoints to the size is simplified.

The experimental results show that the proposed TomatoDet is superior to other state-of-the-art detectors for tomato detection in the greenhouse. It can also detect tomatoes under different lighting and occlusion conditions with strong robustness.

Although the proposed model has achieved a good performance on the tomato datasets, there is still much space for further development. They can be summarized as follows:

When the overlap or occlusion area is high, the detection rate will drop. One possible solution is to incorporate contextual information such as branches or leaves to improve the detection accuracy.

The experimental dataset is relatively small and more data are needed for training and verification in the future study.

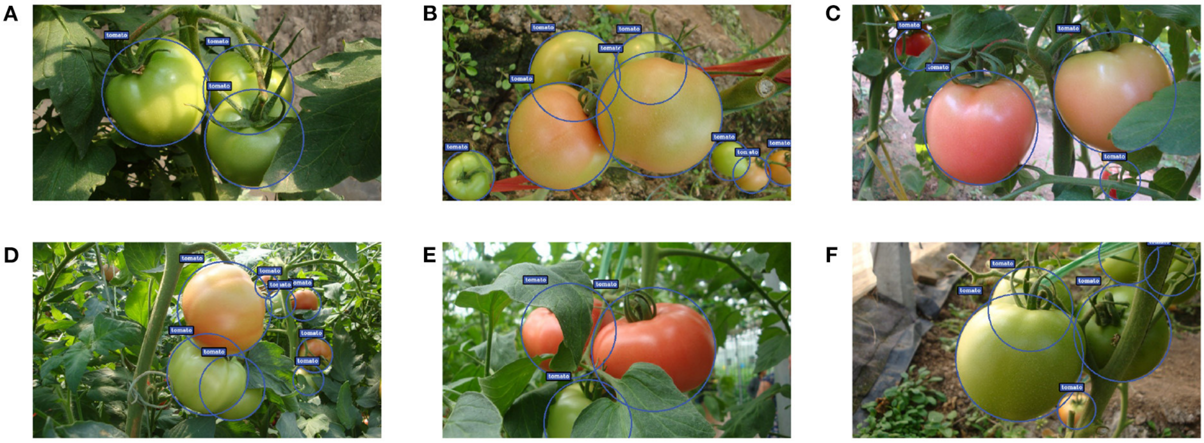

FIGURE 14
Some examples of detection results under different occlusion conditions: **(A–C)** slight cases and **(D–F)** severe cases.

Moreover, the characteristics of tomatoes in different growing stages will be analyzed to realize multi-stage tomato detection.

Data availability statement

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

Author contributions

GL conceived the research idea. GL and ZH designed the methodology. JL and HL performed the experiments and analysis. GL and KL wrote the original draft. WZ and KL revised the manuscript. KL supervised the experiments. All authors contributed to the article and approved the submitted version.

Funding

This study was supported by the Weifang Science and Technology Development Plan (2021GX054), Doctoral Research Foundation of Weifang University (2022BS70), and Natural Science Foundation of Shandong Province (ZR2021QC173).

Conflict of interest

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

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Afonso, M., Fonteijn, H., Fiorentin, F. S., Lensink, D., Mooij, M., Faber, N., et al. (2020). Tomato fruit detection and counting in greenhouses using deep learning. *Front. Plant Sci.* 2020, 1759. doi: 10.3389/fpls.2020.571299

Behroozi-Khazaei, N., and Maleki, M. R. (2017). A robust algorithm based on color features for grape cluster segmentation. *Comput. Electron. Agric.* 142, 41–49. doi: 10.1016/j.compag.2017.08.025

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934.* doi: 10.48550/arXiv.2004.10934

Bresilla, K., Perulli, G. D., Boini, A., Morandi, B., Corelli Grappadelli, L., and Manfrini, L. (2019). Single-shot convolution neural networks for real-time fruit detection within the tree. *Front. Plant Sci.* 10, 611. doi: 10.3389/fpls.2019.00611

Bulanon, D. M., Kataoka, T., Ota, Y., and Hiroma, T. (2002). AE–automation and emerging technologies: a segmentation algorithm for the automatic recognition of fuji apples at harvest. *Biosyst. Eng.* 83, 405–412. doi: 10.1006/bioe.2002.0132

Chen, W., Lu, S., Liu, B., Chen, M., Li, G., and Qian, T. (2022). CitrusYOLO: a algorithm for citrus detection under orchard environment based on YOLOV4. *Multim. Tools Appl.* 1–27. doi: 10.1007/s11042-022-12687-5

Gongal, A., Amatya, S., Karkee, M., Zhang, Q., and Lewis, K. (2015). Sensors and systems for fruit detection and localization: a review. *Comput. Electr. Agric.* 116, 8–19. doi: 10.1016/j.compag.2015.05.021

Kelman, E. E., and Linker, R. (2014). Vision-based localisation of mature apples in tree images using convexity. *Biosyst. Eng.* 118, 174–185. doi: 10.1016/j.biosystemseng.2013.11.007

Koirala, A., Walsh, K., Wang, Z., and McCarthy, C. (2019). Deep learning for real-time fruit detection and orchard fruit load estimation: benchmarking of "mangoyolo". *Precis. Agric.* 20, 1107–1135. doi: 10.1007/s11119-019-09642-0

Kurtulmus, F., Lee, W. S., and Vardar, A. (2011). Green citrus detection using "eigenfruit," color and circular Gabor texture features under natural outdoor conditions. *Comput. Electr. Agric.* 78, 140–149. doi: 10.1016/j.compag.2011.07.001

Kurtulmus, F., Lee, W. S., and Vardar, A. (2014). Immature peach detection in colour images acquired in natural illumination conditions using statistical classifiers and neural network. *Precis. Agric.* 15, 57–79. doi: 10.1007/s11119-013-9323-8

Lam, E. Y. (2005). "Combining gray world and retinex theory for automatic white balance in digital photography," in *Proceedings of the Ninth International Symposium on Consumer Electronics, 2005 (ISCE 2005)* (Macau), 134–139. doi: 10.1109/ISCE.2005.1502356

Law, H., and Deng, J. (2018). "CornerNet: detecting objects as paired keypoints," in *Proceedings of the European Conference on Computer Vision (ECCV)* (Munich), 734–750. doi: 10.1007/978-3-030-01264-9_45

Li, H., Zhang, M., Gao, Y., Li, M., and Ji, Y. (2017). Green ripe tomato detection method based on machine vision in greenhouse. *Trans. Chinese Soc. Agric. Eng.* 33, 328–334. doi: 10.11975/j.issn.1002-6819.2017.z1.049

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision* (Venice), 2980–2988. doi: 10.1109/ICCV.2017.324

Linker, R., Cohen, O., and Naor, A. (2012). Determination of the number of green apples in RGB images recorded in orchards. *Comput. Electr. Agric.* 81, 45–57. doi: 10.1016/j.compag.2011.11.007

Liu, G., Mao, S., and Kim, J. H. (2019). A mature-tomato detection algorithm using machine learning and color analysis. *Sensors* 19:2023. doi: 10.3390/s19092023

Liu, G., Nouaze, J. C., Touko Mbouembe, P. L., and Kim, J. H. (2020). YOLO-tomato: a robust algorithm for tomato detection based on YOLOV3. *Sensors* 20:2145. doi: 10.3390/s20072145

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., et al. (2020). Deep learning for generic object detection: a survey. *Int. J. Comput. Vis.* 128, 261–318. doi: 10.1007/s11263-019-01247-4

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "SSD: single shot multibox detector," in *European Conference on Computer Vision* (Amsterdam: Springer), 21–37. doi: 10.1007/978-3-319-46448-0_2

Lv, Q., Cai, J., Liu, B., Deng, L., and Zhang, Y. (2014). Identification of fruit and branch in natural scenes for citrus harvesting robot using machine vision and support vector machine. *Int. J. Agric. Biol. Eng.* 7, 115–121. doi: 10.3965/j.ijabe.20140702.014

Okamoto, H., and Lee, W. S. (2009). Green citrus detection using hyperspectral imaging. *Comput. Electr. Agric.* 66, 201–208. doi: 10.1016/j.compag.2009.02.004

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). “You only look once: unified, real-time object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV), 779–788. doi: 10.1109/CVPR.2016.91

Redmon, J., and Farhadi, A. (2017). “YOLO9000: better, faster, stronger,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu), 7263–7271. doi: 10.1109/CVPR.2017.690

Redmon, J., and Farhadi, A. (2018). YOLOV3: an incremental improvement. *arXiv preprint arXiv:1804.02767*. doi: 10.48550/arXiv.1804.02767

Ren, S., He, K., Girshick, R., and Sun, J. (2015). “Faster r-CNN: towards real-time object detection with region proposal networks,” in *Advances in Neural Information Processing Systems* (Montreal, QC), 28.

Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., and McCool, C. (2016). Deepfruits: a fruit detection system using deep neural networks. *Sensors* 16:1222. doi: 10.3390/s16081222

Tian, Y., Yang, G., Wang, Z., Wang, H., Li, E., and Liang, Z. (2019). Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Comput. Electr. Agric.* 157, 417–426. doi: 10.1016/j.compag.2019.01.012

Wei, J., Ding, Y., Liu, J., Ullah, M. Z., Yin, X., and Jia, W. (2022). Novel green-fruit detection algorithm based on D2D framework. *Int. J. Agric. Biol. Eng.* 15, 251–259. doi: 10.25165/j.ijabe.20221501.6943

Wei, X., Jia, K., Lan, J., Li, Y., Zeng, Y., and Wang, C. (2014). Automatic method of fruit object extraction under complex agricultural background for vision system of fruit picking robot. *Optik* 125, 5684–5689. doi: 10.1016/j.ijleo.2014.07.001

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). “CBAM: Convolutional block attention module,” in *Proceedings of the European Conference on Computer Vision (ECCV)* (Munich), 3–19. doi: 10.1007/978-3-030-01234-2_1

Wu, J., Zhang, B., Zhou, J., Xiong, Y., Gu, B., and Yang, X. (2019). Automatic recognition of ripening tomatoes by combining multi-feature fusion with a bi-layer classification strategy for harvesting robots. *Sensors* 19:612. doi: 10.3390/s19030612

Yamamoto, K., Guo, W., Yoshioka, Y., and Ninomiya, S. (2014). On plant detection of intact tomato fruits using image analysis and machine learning methods. *Sensors* 14, 12191–12206. doi: 10.3390/s140712191

Yu, F., Wang, D., Shelhamer, E., and Darrell, T. (2018). “Deep layer aggregation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT), 2403–2412. doi: 10.1109/CVPR.2018.00255

Yue, Z., Li, S., and Feldman, M. W. (2015). *Social Integration of Rural-Urban Migrants in China: Current Status, Determinants and Consequences, Vol. 13*. World Scientific. doi: 10.1142/9428

Zhang, W., Liu, Y., Chen, K., Li, H., Duan, Y., Wu, W., et al. (2021). Lightweight fruit-detection algorithm for edge computing applications. *Front. Plant Sci.* 12:740936. doi: 10.3389/fpls.2021.740936

Zhao, C., Lee, W. S., and He, D. (2016). Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove. *Comput. Electr. Agric.* 124, 243–253. doi: 10.1016/j.compag.2016.04.009

Zhao, Y., Gong, L., Huang, Y., and Liu, C. (2016a). A review of key techniques of vision-based control for harvesting robot. *Comput. Electr. Agric.* 127, 311–323. doi: 10.1016/j.compag.2016.06.022

Zhao, Y., Gong, L., Huang, Y., and Liu, C. (2016b). Robust tomato recognition for robotic harvesting using feature images fusion. *Sensors* 16:173. doi: 10.3390/s16020173

Zhao, Y., Gong, L., Zhou, B., Huang, Y., and Liu, C. (2016c). Detecting tomatoes in greenhouse scenes by combining adaboost classifier and colour analysis. *Biosyst. Eng.* 148, 127–137. doi: 10.1016/j.biosystemseng.2016.05.001

Zheng, Z., Xiong, J., Lin, H., Han, Y., Sun, B., Xie, Z., et al. (2021). A method of green citrus detection in natural environment using a deep convolutional neural network. *Front. Plant Sci.* 12:705737. doi: 10.3389/fpls.2021.705737

Zhou, X., Wang, D., and Krähenbühl, P. (2019). Objects as points. *arXiv preprint arXiv:1904.07850*. doi: 10.48550/arXiv.1904.07850