Candidate box fusion based approach to adjust position of the candidate box for object detection

Jie Cao¹,²,³  |  Wei Ren²  |  Hong Zhang¹,³  |  Zuohan Chen¹,³

¹ School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China
² School of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China
³ Urban Railway Transportation, Engineering Research Center of Urban Railway Transportation of Gansu Province, Lanzhou 730050, China

Correspondence
Hong Zhang, Urban Railway Transportation, Engineering Research Center of Urban Railway Transportation of Gansu Province, Lanzhou 730050, China.
Email: zhanghong@lut.edu.cn

Abstract
The method of object detection has been applied to all aspects in our lives. Although object detection methods based on deep learning have been widely used in various fields, there are still some overlooked problems in the candidate box selection stage. The detection results of traditional candidate box selection methods can only select a relatively optimal maximum candidate box. If the maximum candidate box is still not accurate enough, this type of methods will not be able to do adjust it. To solve this problem, an object detection method based on the multiple candidate box fusion is proposed. The method can not only retain the maximum candidate box and delete the non-maximum candidate box, but also adjust the position of the maximum candidate box again. Thereby a more accurate maximum candidate box can be obtained. In order to verify the generalization ability of the method, the candidate box fusion method is combined with the two object detection frameworks: faster R-CNN model and YOLOv3 model. The results of these experiments prove that the proposed method can achieve higher detection accuracy and complete the object detection task more effectively.

1 | INTRODUCTION

In recent years, as a hot research direction in the field of computer and image processing, object detection is widely used in many fields [1–4]. As object detection methods are widely used in various fields, our requirements for speed and accuracy in object detection are getting higher and higher. The research on object detection mainly focuses on the feature extraction. We can use different convolution operations repeatedly to obtain the feature information of each position in the original image, and then use bounding box to predict where the object may exist. These studies have achieved significant results.

In recent years, machine learning methods have gradually emerged. A series of deep convolutional neural networks have achieved good results in object detection tasks, such as Faster-RCNN [5] (Faster Region-CNN), YOLO [6–8] (You Only Look Once) series, SSD [9] (Single Shot Multibox Detector) and other methods. Its detection accuracy and speed are better than traditional object detection methods, such as SIFT [10] (Scale Invariant Feature Transform), HOG [11] (Histogram of Gradient), SURF [12] (Speeded up Robust Features) and DPM [13] (Deformable Parts Model). The object detection method based on deep learning can be divided into two categories according to the detection steps: one is the one-stage method with better detection speed, there are YOLOv3, SSD and other methods; and the other type is two-stage method that is widely superior in detection accuracy, there are R-CNN [14] (Region-CNN), Fast R-CNN [15], Faster-RCNN, Resnet [16] and other methods. The two frame structures are shown in Figure 1. The biggest difference is that the two-stage method generates a recommendation box through a sub-network similar to RPN (Region Proposals Network), and the one-stage method obtains the recommendation box from the feature map.

At present, most scholars’ research focus is mainly on the improvement of the backbone network [17–20]. Various improvement methods are emerging one after another, and their performance improvement is also very significant. However, the structure of the backbone network is often composed of hundreds of hidden layer structures. It should take a long time from model parameter training to performance...
verification to obtain the experimental results. Many methods mostly use non-maximum suppression methods in the candidate box selection stage. As shown in Figure 1, the common structure of the current mainstream one-stage and two-stage methods is the candidate box selection module. If an improved method with better performance can be proposed in the candidate box selection stage, in the field of object detection methods based on deep neural networks, we can improve the overall performance of most object detection methods to a new height without spending a lot of time retraining network parameters. This candidate box selection method will also be widely used.

In the object detection methods, the candidate box selection method is an indispensable part, its performance will directly affect the accuracy of the object detection method. Even if the neural network can detect many perfect candidate boxes, if there is no good candidate box selection method to filter them out, there will be a phenomenon of missed detection and false detection. Therefore, a suitable candidate box selection method is the key to good performance in object detection. In order to improve the performance of the candidate box selection algorithm, this paper will refer to the non-maximum suppression method to propose the candidate box fusion (CBF) method in the post-processing stage of the deep neural network. The CBF aims to adjust the location information of maximum candidate box by reusing the non-maximum candidate box. The candidate box selection method is suitable for the one-stage framework and two-stage object detection framework, which achieves the purpose of improving the accuracy of the object detection without retraining the model. Experimental results show that the accuracy of the target detection model with the CBF algorithm is 6.3% and 3.7% higher than other algorithms. The main contributions of this work can be summarized as:

1. Indicating the disadvantage of the traditional NMS algorithm is that it can only select the relatively optimal candidate box but cannot adjust it.
2. Proposing a candidate box fusion algorithm, which can adaptively adjust the position of the candidate box to surround the target as much as possible. The proposed candidate box fusion algorithm can effectively increase the range of the object surrounded by the candidate box.
3. Our method shows good target detection performance in COCO and VOC datasets.
4. Our method can be combined with a variety of deep learning framework models. And there is no need to retrain the model, just replace a common module to improve the model performance.

2 | RELATED WORK

Currently, many other candidate box selection techniques have emerged. Many methods are more complex and slow in
calculation. The NMS [21] algorithm is one of the most widely used methods. The Soft-NMS algorithm only needs one line of code to greatly improve the performance of the model. NMS method is the earliest method applied to select candidate boxes. Some of the existing candidate frame screening methods have been summarized, as shown in Table 1.

| Method      | Advantages                                                                                                                                  | Disadvantages                                                                                     |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------|
| NMS         | 1. NMS can fulfill the basic requirements of candidate frame screening.  
2. NMS performs well in scenarios with sparse targets.                          | 1. The detection effect is not good when the target is dense.  
2. The probability of false detection is relatively high, resulting in low accuracy of the model. |
| Soft-NMS    | 1. Improving object detection with one line of code.  
2. Soft-NMS does not require any extra training and is simple to implement, it can be easily integrated into any object detection pipeline. | 1. Soft-NMS increases the number of calculation steps and slows down the detection speed.  
2. Soft-NMS can only choose a relatively good candidate box as output. |
| Softer-NMS  | 1. Its loss greatly improves the localization accuracies of various architectures with nearly no additional computation.  
2. The learned localization variance allows us to merge neighbouring bounding boxes during NMS, which further improves the localization performance. | 1. Softer-NMS can only improve algorithm performance after retraining the model.  
2. The model is too complicated.  
3. It cannot be flexibly transplanted to other models. |

2.1 Non-maximum suppression

The NMS aims to select the candidate box with the maximum detection score, and suppresses candidate box with non-maximum detection score. The NMS method is a hard decision method that makes decisions based on the overlap between the non-maximum candidate box and the maximum candidate box. The NMS sets the detection score of the non-maximum candidate box greater than the threshold to zero, otherwise, and keeps the candidate box detection score unchanged. The equation as follows,

\[
s_i = \begin{cases} 
    s_i \cdot \text{iou} (B_k, B_i) < N_t \\
    0, \text{iou} (B_k, B_i) \geq N_t
\end{cases}
\]  

where \( s_i \) is the detection score of the candidate box \( B_i \), \( B_k \) is the candidate box with the maximum detection score. \( B_i \) is the non-maximum candidate box. \( N_t \) is the overlap threshold. \( \text{iou}(B_k, B_i) \) is the overlap between two candidate boxes, the equation as follows,

\[
\text{iou} (B_k, B_i) = \frac{B_k \cap B_i}{B_k \cup B_i}
\]

The traditional NMS method only retains a small number of candidate box with high detection score. The biggest disadvantage is that when the object is dense in the image, many candidate boxes will be deleted. This method will greatly reduce the detection accuracy. Therefore, the Soft-NMS method is proposed which aims to select the maximum candidate box by reducing detection score.

2.2 Improved NMS

From the analysis of the NMS method, it can be seen that the fixed threshold method is difficult to adapt to object detection in complex environment. Therefore, the literature [22] proposed two Soft-NMS methods based on penalty factors. This method takes to reduce the detection score to gradually obtain the maximum candidate box. The equation as follows,

\[
s_i \begin{cases} 
    s_i \cdot \text{iou} (B_k, B_i) < N_t \\
    s_i \cdot (1 - \text{iou} (B_k, B_i)) \cdot \text{iou} (B_k, B_i) \geq N_t
\end{cases}
\]

\[
s_i = s_i \cdot e^{-\frac{(\text{iou}(B_k, B_i))^2}{\sigma}}
\]

where \( \sigma \) represents the variance of the Gaussian function.

Equation (3) is a linear penalty function, and Equation (4) is a Gaussian penalty function. The Soft-NMS method only reduces detection score of high overlap candidate box. To a certain extent, this method can reduce the probability of missed detection.

However, the linear penalty function curve is not continuous, and when the overlap degree is near the threshold value \( N_t \), it will cause a sudden change in the candidate box list. The ideal penalty function should be a smooth continuous curve, as the overlap of the candidate box increases, the detection score decreases smoothly. Therefore, we summarize several methods that have improved the penalty function of the Soft-NMS method in recent years.

The PNMS proposed in [23], whose penalty function as follows,

\[
s_i = s_i \cdot (1 - \text{iou}^3 (B_k, B_i))
\]

The Decay-NMS proposed in [24], whose penalty function as follows,

\[
s_i = s_i \cdot (1 - \text{iou} (B_k, B_i)) \cdot e^{-\text{iou}(B_k, B_i)}
\]

\[
\begin{aligned}
\text{TABLE 1} & \quad \text{The pros and cons of existing candidate box selection methods} \\
\text{Method} & \quad \text{Advantages} & \quad \text{Disadvantages} \\
NMS & \quad \text{1. NMS can fulfill the basic requirements of candidate frame screening.} \\
& \quad \text{2. NMS performs well in scenarios with sparse targets.} & \quad \text{1. The detection effect is not good when the target is dense.} \\
& \quad  & \quad \text{2. The probability of false detection is relatively high, resulting in low accuracy of the model.} \\
Soft-NMS & \quad \text{1. Improving object detection with one line of code.} \\
& \quad \text{2. Soft-NMS does not require any extra training and is simple to implement, it can be easily integrated into any object detection pipeline.} & \quad \text{1. Soft-NMS increases the number of calculation steps and slows down the detection speed.} \\
& \quad \text{2. Soft-NMS can only choose a relatively good candidate box as output.} & \quad \text{1. Softer-NMS can only improve algorithm performance after retraining the model.} \\
Softer-NMS & \quad \text{1. Its loss greatly improves the localization accuracies of various architectures with nearly no additional computation.} \\
& \quad \text{2. The learned localization variance allows us to merge neighbouring bounding boxes during NMS, which further improves the localization performance.} & \quad \text{2. The model is too complicated.} \\
& \quad \text{3. It cannot be flexibly transplanted to other models.} & \quad \end{aligned}
\]
The EW (Exponent Weighted) proposed in [25], whose penalty function as follows,

\[ s_i = \begin{cases} 
  s_i, \text{iou} (B_k, B_i) < N_i \\
  e^{(N_i - \text{iou}(B_k, B_i))}, \text{iou} (B_k, B_i) \geq N_i 
\end{cases} \tag{7} \]

The IGW (Improved Gaussian Weighted) proposed in [26], whose penalty function as follows,

\[ s_i = \begin{cases} 
  e^{-\frac{\sigma^2(B_k, B_i)}{\sigma^2}}, \text{iou} (B_k, B_i) < N_i \\
  -\frac{N_i^2}{e^2} (\text{iou} (B_k, B_i) - 1), \text{iou} (B_k, B_i) \geq N_i 
\end{cases} \tag{8} \]

In Equation (8), \( \sigma \) represents the variance of the Gaussian function. The above NMS-based improved methods perform well in different scenarios. Their innovation is to select unqualified candidate boxes by reducing the confidence of candidate boxes. The difference is that the attenuation function uses different smooth curves to gradually reduce the confidence of the redundant box. They verified that the algorithm performance is very good in different data sets. We will compare the performance of each algorithm in the same model and the same data set.

The candidate box selection strategies of NMS and Soft-NMS method are relatively simple. The measures adopted by this method are mainly adjusting the detection score of the candidate box. After re-scoring the detection score of the high overlap candidate box, we retain the candidate box with the highest detection score through layer-by-layer selection. Such methods are also a greedy method. If the candidate box with the maximum detection score is not enough accurate, NMS algorithm can no longer adjust it. It has two obvious problems: (1) The NMS methods are all selected based on detection score and overlap IoU values, but in most cases the overlap and detection score do not have a strong correlation. (2) The traditional method can only select relatively optimal candidate box from candidate boxes output by the neural network. Sometimes, all the boxes around a real object are detected, but they are not accurate.

### 3.1 CANDIDATE BOX FUSION (CBF)

According to the NMS method and the improved NMS method, the key factor in deciding to retain a candidate box is whether the detection score of the candidate box is greater than a pre-define threshold, or whether the overlap IoU value of the maximum candidate box is greater than the pre-define threshold. We are inspired by this to propose a candidate box fusion method based on the overlap IoU value (Overlap Candidate Box Fusion, OCBF), and a candidate box fusion method based on the detection score (Score Candidate Box Fusion, SCBF).

In the Deep Neural Network (DNN) [27–29] model of object detection, first the backbone network extracts the feature information of each position in images, and then the fully connected layer detects the candidate box. Each candidate box vector \( \text{Box} = [x, y, w, h, s, c] \) contains six position information of the object, including centre position coordinates of the candidate box \([x, y] \), border width and height of the candidate box \([w, h] \), detection score \( s \) and object category \( c \). Each parameter can directly affect the final detection performance of the method, and \([x, y, w, h] \) position information is converted to \([x1, y1, x2, y2] \) respectively correspond to the coordinate positions of the upper left corner and the lower right corner of the candidate box.

In the research process of the NMS method, first it deletes candidate boxes with detection score less than a pre-define threshold \( n_t \) to reduce the amount of calculation, the equation as follows,

\[ s_i = \begin{cases} 
  s_i, s_i \geq n_t \\
  0, s_i < n_t 
\end{cases} \tag{9} \]

We regard the non-maximum candidate box set \( C = \{C_1, C_2, \ldots, C_n\} \) that highly overlaps with the maximum candidate box as detecting the same object. In this set of candidate box, the OCBF method takes the candidate box with the maximum detection score \( B_k = [x1, y1, x2, y2] \) as the basic box. Because the probability of the candidate box containing object information is different, the OCBF method assigns a different weight to each candidate box. It is considered that the greater the overlap between some candidate box and \( B_k \), the closer it is to the real object box, so it is given a larger fusion weight \( w_i \). The function as follows,

\[ w_i = \frac{\text{iou} (B_k, B_i)}{\sum_{i=1}^{n} \text{iou} (B_k, B_i)} \tag{10} \]

The weight of each candidate box is the IoU value between the candidate box and \( B_k \) divided by the sum of the IoU values between all candidate boxes and \( B_k \). After obtaining the weight of each candidate box, we can calculate a new candidate box position information by weighting and summing the non-maximum candidate box. The calculation method is as follows,

\[ B_k = B_k - \sum_{i=1}^{n} \frac{(B_k - C_i) \cdot w_i}{\sum_{i=1}^{n} w_i} \tag{11} \]

The detection score \( s \) and category \( c \) of the new candidate box correspond to the detection score and category of the original maximum candidate box \( B_k \).

The OCBF method can stretch or compress the position of the maximum candidate box in the direction of each non-maximum candidate box. Finally we can obtain a maximum candidate box surrounding the real object as much as possible.
which can get a higher detection score during performance evaluation.

### 3.2 SCBF

The SCBF method also selects and fuses a new candidate box from the set of candidate box output by the DNN module. Inspired by the research on the NMS method, the detection score \( s \) represents the probability containing a real object, and the same can be understood as that the candidate box contains the amount of information of the real object. The NMS method only retains the candidate box with the largest amount of information, and deletes the candidate box with a small amount of information. Even the detection score of the best candidate box is difficult to reach 1. If we can integrate the candidate box with low information into the maximum candidate box, which will effectively improve the final detection effect.

The SCBF method still adopts the way of integrating non-maximum candidate box into the position information of the maximum candidate box. The SCBF method regards the non-maximum candidate box set \( C = \{ C_1, C_2, ..., C_n \} \) that highly overlaps with the maximum candidate box as detecting the same object. In the set of candidate box, the SCBF method considers the maximum candidate box \( B_k = [x_1, y_1, x_2, y_2] \) as the basic box. Because the probability of the candidate box contains object information is different, the SCBF method assigns each candidate box a different weight in the process of candidate frame fusion. It is considered that the greater detection score \( s \), the closer to the real object, so it is given a larger fusion weight \( w_i \).

The calculation method is as follows,

\[
w_i = \frac{s_i}{\sum_{i=1}^{n} s_i} \tag{12}\]

The weight of each candidate box is the detection score \( s \) divided by the sum of the detection score of all candidate boxes. After obtaining the weight of each candidate box, we can calculate a new candidate box position information by weighting and summing the candidate box, the calculation method as follows,

\[
B_k = B_k - \sum_{i=1}^{n} \frac{(B_k - C_i) \cdot w_i}{\sum_{i=1}^{n} w_i} \tag{13}\]

The detection score \( s \) and category \( c \) of this candidate box correspond to the detection score and category of the original maximum candidate box \( B_k \).

For the object information that may exist in each candidate box, the SCBF method’s adjustment strategy is similar to the OCBF method. The CBF algorithm stretches or compresses the position of the maximum candidate box in all directions. Finally, we can obtain a new candidate box surrounding the real object as much as possible, which can get a higher detection score during performance evaluation, and achieve the purpose of improving the performance of the method.

### 3.3 Method steps

The above two CBF methods design to solve the two problems in candidate box selection: First, for the suppression of non-maximum candidate box, the CBF method draws on the idea of NMS method, which is to select and delete non-maximum candidate boxes that overlap with the maximum candidate box higher than the threshold \( N_t \); secondly, for the cases where the detection results of the maximum candidate box are not good, this method reuses non-maximum candidate box to adjust the position of the maximum candidate box by weighted summation.

Through this candidate box selection method, not only can suppress the non-maximum candidate box, but also can reserve the information of non-maximum candidate box to adjust the position of the inaccurate maximum candidate box. The new candidate box can contain more real object information to achieve the purpose of improving the detection accuracy. The method flow is shown in Figure 2. The CBF algorithm pseudo code in Table 2. On the left is the SCBF algorithm, and on the right is the OCBF algorithm. Each algorithm outputs independently and does not affect each other.

### 4 EXPERIMENT AND DISCUSSION

In order to verify the performance of two CBF methods in object detection, we select the YOLOv3 model in one-stage framework and the Faster R-CNN model in two-stage framework as basic model for verification, and we choose resnet101 structure as backbone network of the Faster R-CNN. We use the set of candidate box output by the DNN module to verify the performance of the CBF method in the candidate box selection task. The experimental results are verified on the MS COCO [30] dataset and the PASCAL VOC dataset. The computer simulation platform used in the experiment: Ubuntu 18.04, Intel CPU E5-2620, GTX 1080Ti GPU.

#### 4.1 Experimental design

First, we trained YOLOv3 model and Faster R-CNN model on the COCO dataset and the VOC dataset respectively, and expanded the number of training set by rotating the training set images or enhancing the contrast. The pre-defined learning rate is 0.001. We use Adam optimizer to optimize the parameters. Then, we combined the 8 candidate box selection methods with YOLOv3 and Faster R-CNN models to run separately. The candidate box selection method is repeatedly tested on the dataset to find the parameter value with the highest mAP, and we can obtain the best parameters through experiments, as shown in Table 3. The linear penalty functions and Gaussian penalty functions of the Soft-NMS method are respectively denoted as Line and Gauss, the detection score threshold is denoted as \( Thres \), and the overlap threshold with the maximum candidate box is denoted as \( N_t \). Finally, we compared the evaluation metrics of each method when each method takes the best parameters.
FIGURE 2 CBF method flowchart

TABLE 2 CBF algorithm pseudo code

Algorithm: Candidate box fusion

Input: Candidate box set \( B = \{B_1, B_2, \ldots, B_N\} \); Candidate box corresponding confidence \( S = \{S_1, S_2, \ldots, S_N\} \); Overlap threshold \( N_t \).

\( B \) is an \( N \times 4 \) dimensional vector, including the coordinates of the upper left corner of the candidate box \((x_1, y_1)\) and the bottom right coordinate \((x_2, y_2)\); \( S \) is the confidence corresponding to each candidate box;

Output: Detection box and its confidence.

\( D \leftarrow \{\} \)

while \( B \neq \emptyset \) do

\( m \leftarrow \text{argmax} S \)

\( B_k \leftarrow B_m \)

\( B \leftarrow B - B_k \)

\( \text{max} \leftarrow \text{max}(B_1.B_k) \)

\( \text{idx} \leftarrow \text{idx} \geq N_t \)

\( B_k \leftarrow \text{sum}(B[\text{idx}] \times \text{sum}(\text{idx}))/\text{sum}(\text{idx}) \) //OCBF

\( B_k \leftarrow \text{sum}(B[\text{idx}] \times \text{sum}(\text{idx}))/\text{sum}(\text{idx}) \) //SCBF

\( B \leftarrow B - \text{idx} \)

\( D \leftarrow D \cup B_k \)

end while

return \( D, S \)

TABLE 3 Algorithm parameter table

| Method   | Thres | \( N_t \) | \( \sigma \) |
|----------|-------|-----------|------------|
| Line     | 0.01  | 0.45      | NA         |
| Gauss    | 0.01  | NA        | 0.3        |
| PNMS     | 0.01  | NA        | NA         |
| Decay-NMS| 0.01  | NA        | NA         |
| EW       | 0.01  | 0.005     | NA         |
| IGW      | 0.01  | 0.6       | 0.2        |
| OCBF     | NA    | 0.5       | NA         |
| SCBF     | NA    | 0.45      | NA         |

4.2 Evaluation metrics

At present, in the evaluation of object detection models, the commonly used evaluation metrics include precision, recall, AP and mAP. Among them, we hope the precision and the recall to be as high as possible, but there is a contradiction relationship...
Experimental results

YOLOv3 ablation experiment results

In this paper, we choose three evaluation metrics to evaluate models, including AP, mAP and P-R curve. When the overlap between the detection box and the real box is greater than 0.5, the detection box is regarded as correct.

\[
Precision = \frac{TP}{TP + FP}
\] (14)

\[
Recall = \frac{TP}{TP + FN}
\] (15)

\[
AP = \int_{0}^{1} \rho(r) \, dr
\] (16)

\[
mAP = \frac{1}{C} \sum_{C \in C} AP_{(C)}
\] (17)

Among them, TP represents the number of positive samples that the model predicts as positive; FP represents the number of positive samples that the model predicts as negative; FN represents the number of negative samples that the model predicts as positive; P and R represent the precision rate and the recall rate respectively; C represents the total number of classes in the data set; \(C_i\) is the i-th class.

In this paper, we choose three evaluation metrics to evaluate models, including AP, mAP and P-R curve. When the overlap between the detection box and the real box is greater than 0.5, the detection box is regarded as correct.

4.3 Experimental results

In order to fully verify the feasibility of the candidate box fusion algorithm, this paper will compare the results of two ablation experiments, and use the same neural network parameters. The first experiment was verified on the YOLOv3 and COCO data set, and the second experiment was verified on the Faster R-CNN and VOC data set. Each experiment contains 8 candidate box selection algorithms mentioned in this paper.

4.3.1 YOLOv3 ablation experiment results

Experiment 1 is verified in the COCO test data set, which contains 5000 images of any size and a total of 80 categories. In the YOLOv3 model, each image can detect up to 10,647 candidate boxes. When the IoU value of the overlap between the candidate box and the real box is greater than 0.5, the detection box is regarded as a positive sample, otherwise it is a negative sample.

Among the three indicators of mAP50, mAP60, and mAP70, mAP50 is often used to evaluate the detection accuracy of a model. As the IoU restriction increases, the detection effect becomes higher and higher. It can be seen from Table 4 that as the limit of IoU increases, the average accuracy mAP value is getting lower and lower. This is because the evaluation method requires higher and higher overlap between the candidate box and the real box. Therefore, candidate boxes whose overlap with the real box is greater than 0.5 and less than 0.6, 0.7 are regarded as negative samples. As a result, the number of positive samples decreases, and the detection accuracy also decreases.

Since the CBF algorithm can adjust the position of the maximum candidate box, its detection box is closer to the real candidate box and obtained higher indicators in the performance evaluation. As the limit of IoU increases, CBF is more advantageous. The accuracy of mAP50 is as high as 54.9%, which is 3.4–6.6% higher than the improved NMS algorithm, the accuracy of mAP60 is 48.8%, which is 2.2–4% higher than the improved NMS algorithm, and the accuracy of mAP70 is 38.4%, which is higher than the improved NMS algorithm by 0.3–0.9%.

Since the number of candidate boxes in the YOLOv3 model is as high as 10,647, we need to calculate the weight values of a large number of candidate boxes when the candidate box fusion. Therefore, the detection accuracy is greatly improved at the expense of a part of the detection speed. It can still achieve the real-time detection effect of the one-stage framework. When the IoU threshold is 0.5, the detection accuracy of some categories in the COCO data set is shown in Table 5.

The detection accuracy of each category of the OCBF and SCBF algorithms in Table 5 is between 43.4% and 82.9%, which is higher than the NMS improved algorithm by 0.9–14.1%. However, the OCBF algorithm and the SCBF algorithm proposed in this paper are almost equal to each other in terms of their calculation methods, whether it is a single category AP value or multiple category mAP values. This small difference can be ignored in object detection and evaluation, and compared with other algorithms CBF have obvious advantages.

According to the above experimental results and data analysis, the performance of Gauss method and IGW in the improved NMS algorithm is relatively better. In order to verify the recall rate of the algorithm in this paper, we randomly selected four categories from the COCO data set: Bicycle,
TABLE 5  Accuracy AP value of some categories in the COCO test set

| Method | Person | Bike | Car | mbike | Aero | Bus | Train | Truck | Boat | Traffic light |
|--------|--------|------|-----|-------|------|-----|-------|-------|------|--------------|
| Line   | 71.3   | 41.9 | 59.3| 55.3  | 77.4 | 76.2| 77.5  | 35.2  | 39.6 | 47.8         |
| Gauss  | 72.5   | 42.2 | 60.0| 55.6  | 78.7 | 75.5| 78.1  | 35.9  | 40.2 | 48.0         |
| PNMS   | 69.1   | 38.1 | 57.6| 51.4  | 74.8 | 72.9| 76.2  | 33.7  | 37.6 | 46.3         |
| Decay-NMS | 71.1 | 42.3 | 58.9| 55.8  | 78.4 | 77.6| 78.5  | 36.1  | 39.7 | 47.7         |
| EW     | 68.7   | 40.5 | 57.5| 51.8  | 74.6 | 75.1| 76.5  | 34.0  | 38.4 | 46.7         |
| IGW    | 72.4   | 42.8 | 59.6| 56.3  | 78.9 | 75.7| 78.2  | 35.9  | 40.7 | 47.9         |
| OCBF   | 74.1   | 49.6 | 61.8| 64.6  | 81.9 | 82.9| 81.1  | 43.8  | 43.4 | 48.9         |
| SCBF   | 74.1   | 49.3 | 62.5| 65.5  | 82.3 | 82.9| 80.1  | 44.5  | 42.7 | 49.2         |

FIGURE 3  P-R curves of different algorithms of YOLOv3 model. (a) Bicycle P-R curve. (b) Bus P-R curve. (c) Surfboard P-R curve. (d) Dining table P-R curve.

Bus, Dining table and Surfboard. The P-R curves comparison result of these two algorithms and the CBF algorithm shown in Figure 3.

It can be clearly seen from Figure 3 that the candidate box fusion algorithm proposed in this paper is verified on four randomly selected categories, and its P-R curve can completely surround other algorithms, and the curves of the OCBF and SCBF algorithms proposed in this paper can almost overlap. It can be known that the CBF algorithm can not only increase the proportion of positive samples in the detection results, but also increase the probability of positive samples being detected in the sample, and also reduce the probability of missed detection and false detection of the object. From the related evaluation method theory introduced in this section, it can be concluded that the candidate box fusion algorithm proposed in this paper can greatly increase the recall rate under the premise of ensuring the improvement of the precision rate.

Figure 4 shows the partial detection effect of the CBF algorithm combined with the YOLOv3 model on the COCO test data set. Most of the objects in the picture have been detected. Because of some shortcomings of the neural network model, a small number of objects in the picture cannot be detected.
4.3.2 Faster R-CNN ablation experiment results

The second experiment is verified on the Faster R-CNN model + VOC dataset of the two-stage algorithm, which contains 4952 images of any size, with a total of 20 categories. The experiment process is similar to experiment 1, and we will compare the four performance indicators of AP, mAP, P-R curve and FPS to verify the effectiveness of the algorithm in this paper.

It can be seen from Table 6, the accuracy of mAP50 of the CBF algorithm is as high as 80.5%, which is higher than the improved NMS algorithm by 0.7–3.7%, the accuracy of mAP60 is 75.8%, which is higher than the improved NMS algorithm by 1.4–2.6%, and the accuracy of mAP70 is 66.5%, which is higher than the improved NMS algorithm by 1.3–2.6%. With the increase of the IoU threshold, the average precision of the CBF algorithm decreases while still at a high level, which verifies the effectiveness of the candidate box fusion algorithm on the Faster R-CNN model. Compared with the YOLOv3 model, the Faster R-CNN model only selects the top 100 candidate boxes for selecting, which greatly reduces the computational complexity. Therefore, the detection speed of the CBF algorithm is only slightly lower than that of the improved NMS algorithm, even faster than some algorithms. When the IoU threshold is 0.5, the detection accuracy of some categories in the VOC data set is shown in Table 7.

The detection accuracy of each category of OCBF and SCBF algorithms in Table 7 is between 69.1% and 87.7%, which is higher than the improved NMS algorithm by 0.3–9.1%. However, the OCBF algorithm and the SCBF algorithm proposed in this paper are almost equal to each other in terms of their calculation methods, whether it is a single category AP value or multiple category mAP values. This small difference can be ignored in target detection and evaluation, and compared with other algorithms, have obvious advantages.

From the above experimental results and data analysis, it can be seen that the various performances of IGW in the NMS improved algorithm are relatively good. In order to verify the recall rate of the algorithm in this paper, we randomly selected four categories from the VOC data set: Cow, Dining table, Person and Bicycle. The P-R curves comparison result of Method4 and the CBF algorithm is shown in Figure 5.

It can be clearly seen from Figure 5 that the P-R curve of candidate box fusion algorithm can completely surround the Method4 with better detection accuracy. The P-R curves of OCBF and SCBF are almost the same. It can be known that the CBF algorithm can not only increase the proportion of positive samples in the detection results, but also increase the probability of positive samples being detected in the sample, and also reduce the probability of missed detection and false detection of the target. From the related evaluation method theory introduced in this section, it can be concluded that the candidate box fusion algorithm proposed in this paper can greatly improve the precision rate under the premise of ensuring the recall rate in the two-stage model.

Figure 6 shows the partial detection results of the two CBF algorithms combined with the Faster R-CNN model on the VOC test data set. Most of the objects in the picture have been detected. Because of some shortcomings of the neural network model, a small number of objects in the picture cannot be detected.
TABLE 7 Faster R-CNN model different algorithm AP value

| Method   | Bike  | Bottle | Car   | Cow   | Table | Person | Plant | Sheep | Sofa | Train |
|----------|-------|--------|-------|-------|-------|--------|-------|-------|------|-------|
| Line     | 84.1  | 67.6   | 86.9  | 81.9  | 64.6  | 82.4   | 51.4  | 78.4  | 71.7 | 83.3  |
| Gauss    | 84.1  | 67.6   | 86.9  | 81.9  | 64.6  | 82.4   | 51.4  | 78.4  | 71.7 | 83.3  |
| PNMS     | 84.1  | 67.4   | 86.9  | 81.9  | 64.5  | 82.4   | 51.2  | 78.3  | 71.7 | 83.3  |
| Decay-NMS| 84.1  | 67.6   | 86.9  | 81.9  | 64.6  | 82.4   | 51.4  | 78.4  | 71.7 | 83.3  |
| EW       | 84.2  | 67.1   | 87.4  | 83.7  | 66.0  | 82.4   | 51.3  | 78.2  | 73.4 | 84.8  |
| IGW      | 86.4  | 68.8   | 87.1  | 85.8  | 71.4  | 82.8   | 53.4  | 80.2  | 78.0 | 84.6  |
| OCBF     | 87.5  | 69.1   | 87.7  | 86.9  | 72.4  | 83.7   | 54.6  | 82.8  | 77.7 | 86.4  |
| SCBF     | 87.5  | 69.1   | 87.7  | 87.0  | 73.6  | 84.0   | 54.3  | 82.4  | 79.2 | 86.4  |

FIGURE 5 P-R curves of different algorithms of the Faster R-CNN model. (a) Cow P-R graph. (b) Cow P-R graph. (c) Person P-R graph. (d) Bicycle P-R graph

5 CONCLUSIONS

This paper proposes two CBF algorithms based on the object detection method of deep convolutional neural network. In the candidate box selection stage, the CBF algorithm can not only retain local maximum candidate boxes and delete redundant boxes, but also adjust the position of the maximum candidate box through the reuse of redundant boxes. Making the maximum candidate box contain more real object information, which can increase the overlap between the candidate box and the real box. CBF effectively improve the problem of low precision and recall of the candidate box selection algorithm in multi-object detection tasks. The experimental results on many indicators of CBF algorithm are better than other methods, including P-R curve value, AP value and mAP under different restriction conditions. In addition, the proposed method can improve the accuracy without reducing the recall. It should be pointed out that the method in this paper still has shortcomings. In the future, we will continue to conduct further research on the speed of candidate box fusion, in order to achieve the goal of improving the accuracy of the algorithm without reducing the detection speed.
FIGURE 6  Partial test results of Faster R-CNN model combined with CBF method

ORCID

Wei Ren  🌐 https://orcid.org/0000-0002-2357-0204

REFERENCES

1. Meng, Z., Zhang, M., Wang, H.: CNN with pose segmentation for suspicious object detection in MMW security images. Sensors 20, 4974 (2020)
2. Yu, H., Gong, J., Chen, D.: Object detection using multi-scale balanced sampling. Appl. Sci. 10, 6053 (2020)
3. Osorio, K., et al.: A deep learning approach for weed detection in lettuce crops using multispectral images. Agric. Eng. 2, 471–488 (2020)
4. Shin, S.-J., et al.: Hierarchical multi-label object detection framework for remote sensing images. Remote Sens. 12, 2734 (2020)
5. He, Z., Zhang, L.: Multi-adversarial faster-rcnn for unrestricted object detection. In: Proceedings of the IEEE International Conference on Computer Vision. Seoul, Korea, pp. 6668–6677 (2019)
6. Braun, M., et al.: EuroCity persons: A novel benchmark for person detection in traffic scenes. IEEE Trans. Pattern Anal. Mach. Intell. 41(8), 1844–1861 (2019)
7. Zhou, Y., Tuzel, O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, pp. 4490–4499 (2018)
8. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
9. Li, M., et al.: Agricultural greenhouses detection in high-resolution satellite images based on convolutional neural networks: Comparison of faster R-CNN, YOLO v3 and SSD. Sensors 20, 4938 (2020)
10. Lowe, D.G.: Distinctive image features from scale-invariant key points. Int. J. Comput. Vision 60(2), 91–110 (2004)
11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. San Diego, CA, pp. 886–893 (2005)
12. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: Proceedings of the European Conference on Computer Vision. pp. 404–417. Springer, Berlin, Heidelberg (2006)
13. Felzenszwalb, P.F., et al.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. 32, 1627–1645 (2009)
14. He, K., et al.: Rethinking imagenet pre-training. In: Proceedings of the IEEE International Conference on Computer Vision. Seoul, Korea, pp. 4918–4927 (2019)
15. Qi, C.R., et al.: Frustum pointnets for 3d object detection from rgb-d data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, pp. 918–927 (2018)
16. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, pp. 770–778 (2016)
17. Yang, J., et al.: A feature-enhanced anchor-free network for UAV vehicle detection. Remote Sens. 12, 2729 (2020)
18. Chen, S., et al.: FCC-Net: A full-coverage collaborative network for weakly supervised remote sensing object detection. Electronics 9, 1356 (2020)
19. Lee, H., et al.: A deep learning-based fragment detection approach for the arena fragmentation test. Appl. Sci. 10, 4744 (2020)
20. Huang, Y.-J., et al.: SSD-TSEFFM: New SSD using trident feature and squeeze and extraction feature fusion. Sensors 20, 3630 (2020)
21. Zhao, J., et al.: A cascade coupled convolutional neural network guided visual attention method for ship detection from SAR Images. Proc. IEEE Access 6(10), 50693–50708 (2018)
22. Bodla, N., et al.: Soft-NMS–improving object detection with one line of code. In: Proceedings of the IEEE International Conference on Computer Vision. Venice, Italy, pp. 5561–5569 (2017)
23. L. Zhen-dong, et al.: PNMS algorithm based on penalty factors for face detection and alignment. Adv. Eng. Sci. 50(06), 225–231 (2018)
24. C. Jin, H. Mengna, L. Yi, Z. Shuai: Segmentation of impurity rice grain images based on U-Net model. J. Agric. Eng. 36(10), 174–180 (2020)
25. Z. Ye, et al.: Research on faster RCNN object detection based on hard example mining. J. Electron. Inf. Technol. 41(6), 1496–1502 (2019)
26. Z. Hai-ya, C. Ji-hong: Research on night traffic flow detection method based on deep learning. Software Guide 18(9), 33–37 (2019)
27. Ramalingam, B., et al.: A human support robot for the cleaning and maintenance of door handles using a deep-learning framework. Sensors 20(20), 3543
28. Sugimori, H., et al.: Development of a deep learning-based algorithm to detect the distal end of a surgical instrument. Appl. Sci. 10, 4245 (2020)
29. Kwan, C., et al.: Real-time and deep learning based vehicle detection and classification using pixel-wise code exposure measurements. Electronics 9, 1014 (2020)
30. Zhou, B., et al.: Semantic understanding of scenes through the ade20k dataset. Int. J. Comput. Vision 127(3), 302–321 (2019)
31. Z. Zhi-hua: Machine Learning. Tsinghua University Press (2016)

How to cite this article: Cao J, Ren W, Zhang H, Chen Z. Candidate box fusion based approach to adjust position of the candidate box for object detection. IET Image Process. 2021;1–11.
https://doi.org/10.1049/iptr2.12264.