IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 1, 2022, accepted May 19, 2022, date of publication May 23, 2022, date of current version May 27, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3177628

YOLO-G: A Lightweight Network Model for Improving the Performance of Military Targets Detection

LINGREN KONG[1], JIANZHONG WANG[1], AND PENG ZHAO[2]
[1]School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China
[2]China North Vehicle Research Institute, Beijing 100072, China

Corresponding author: Jianzhong Wang (cwjzwang@bit.edu.cn)

This work was supported by the Defense Industrial Technology Development Program under Grant JCKY2021602B029.

ABSTRACT Military target detection technology is the foundation and key to perceive and analyze the battlefield situation, and it is also the premise of target tracking technology. Aiming at the task of military target detection, the detection performance of traditional detection algorithms is poor in complex environment. We realized automatic detection of military targets in complex environment through deep learning. In this research, we improved the components of YOLOv3 and proposed a novel military target detection algorithm (YOLO-G). We have built a military target dataset composed of armed men with different weapons, which provides a test environment for various object detection algorithms. In the YOLOv3 network structure, by introducing the lightweight convolutional neural network GhostNet as the feature extraction network, the accuracy and speed of military target detection are improved. Then, the attention mechanism based on the coordinate attention block is introduced to enhance the representation ability of target features, suppress interference and improve the detection accuracy. Finally, the loss function of the target detector is redesigned by using DIOU loss function and Focal loss function, which further improves the detection accuracy of our detection model for military targets. We tested YOLO-G on the military target dataset. Experimental results show that our method improves the mAP by 2.9% and the detection speed by 25.9 frames/s compared with the original YOLOv3 algorithm, and the size of the proposed model is reduced to 1/6 of that of YOLOv3. In addition, we also compared our method with several state-of-the-art object detection algorithms. The results show that YOLO-G also has superior detection performance, and the mAP index obtained by our method is 1.2% higher than that of the latest YOLOv5 on the premise of meeting the application requirements. The improved network model can provide effective auxiliary technical support for battlefield situation generation and analysis.

INDEX TERMS Target detection, YOLOv3, GhostNet, coordinate attention, loss function.

I. INTRODUCTION

Battlefield situational awareness refers to the process of realizing real-time awareness of the deployment of combat troops, combat equipment and battlefield environment by using sensors. Its contents include reconnaissance, surveillance, intelligence, damage assessment, etc., which is the premise of firepower distribution [1]. The future war will be dominated by information technology, and the battlefield situational awareness will play an important role in improving the overall control of the war by both sides. The identification and positioning of military targets are the key technologies that affect battlefield situation perception [2]. At present, all military powers are strengthening the development and research of related technologies [3], [4]. Therefore, it is of great significance to carry out the research on automatic detection technology of military targets in complex environment for the generation and analysis of battlefield situation.

Military target detection belongs to the basic technology in the military field. As an important national defense

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

application subject, scholars in many countries have started a series of research work in this field. Jain *et al.* [5] put forward the method of segmentation for target detection in complex background, and applied it to military reconnaissance related fields. Nelson [6] proposed a novel method of target detection and classification using fuzzy inference system, which was used for tank and military vehicle identification. Jun *et al.* [7] proposed an Automatic Target Detection (ATD) algorithm for Charge Coupled Device (CCD) images, which was used to detect targets such as main battle tanks and armored personnel carriers in ground-to-ground scenes. Neagoe *et al.* [8] proposed a novel method of automatic target recognition in Synthetic Aperture Radar (SAR) aerial images by using neural network, with a total success rate of 97.36%. Budiharto *et al.* [9] put forward a prototype of tank military robot with target detection and tracking function based on computer vision, and simulated its turret firing.

AlexNet is a major turning point in the development of deep learning [10]. Since AlexNet was put forward, this network model has been used in many fields, including military target detection. In recent years, scholars have successively applied the method based on deep learning to battlefield situation awareness, which can better solve the problem of complex battlefield situation generation and analysis in the current information battlefield [11], [12]. R-CNN [13] is the first method to apply convolutional neural network to target detection. Compared with traditional algorithms such as Adaboost [14] and DPM [15], its performance has been greatly improved. Since then, many excellent target detection algorithms have been proposed, such as SPP-Net [16], Fast R-CNN [17], Faster R-CNN [18] and R-FCN [19]. These methods correct the defects of R-CNN, and improve the accuracy and speed of target detection. However, the above methods still have the problem of slow detection speed in practical application, so a detector aiming at improving the detection speed is put forward. SSD [20] and YOLO series [21]–[25] are representative one-stage detectors. Their most prominent feature is their fast detection speed, which can be used in real-time situations.

The purpose of this paper is to identify military targets accurately and efficiently by deep learning. YOLOv3 is a widely used algorithm of YOLO series, because it realizes the trade-off between detection speed and detection accuracy. YOLOv3 adopts the feature pyramid network (FPN) [26], [27], which can predict the target from three different scales, thus improving the detection ability of small targets. The use of ResNet also improves the detection speed and accuracy of YOLOv3 [28]. Since YOLOv3 was put forward, many researchers have made improvements based on YOLOv3, which makes the effect of target detection more suitable for specific use requirements. Reference [29], [30] added a detection layer on the basis of the original YOLOv3 detection network, thus improving the detection ability of small targets. Reference [31] realized tomato detection by circular ground truth. Reference [32] improved the feature extraction ability of the detection network by adding shortcut connection to concatenate two CBL layers between two ''residual units''. Reference [33] improved the speed and accuracy of feature extraction by simplifying the feature extraction network and adopting multiple layers concatenation. The above methods show that YOLOv3, as an excellent detector, can be improved to enhance the detection performance in accordance with the requirements of the usage scenarios. Because the battlefield situation is highly dynamic, the detection speed of military targets is required to ensure that the real-time detection of targets. In addition, in the process of target detection of battlefield image information obtained by perception system, military targets are affected by illumination, imaging angle, target size, camouflage, and occlusion of some targets. Therefore, these algorithms perform well on universal datasets, but the detection results of military targets in complex environments are not optimal. In the actual testing process, it is found that YOLOv3 is not effective enough for some small targets, and there are still quite a few false and missed detections [34]. At present, the research on military target detection technology mainly focuses on large convolutional neural networks. However, large-scale networks are difficult to deploy on devices with few hardware resources. Lower FPS and large time delay become important factors that restrict its practical application.

In view of the shortcomings of YOLOv3 in target detection and considering the characteristics of military targets, we proposed an improved YOLOv3 algorithm (YOLO-G). The main contributions of this algorithm are as follows: (1) The more excellent lightweight neural network GhostNet [35], which has better performance, is used as the feature extraction network. On the basis of linear transformation, more abundant multi-channel feature maps of targets can be obtained, which reduces the parameters and computational complexity of the network model and improves the detection accuracy and efficiency to a certain extent. (2) The coordinate attention mechanism [36] is added into the feature extraction network. By embedding the location information into the attention of the channel, the network can better obtain the information of the spatial direction features and enhance the information interaction among the features at all layers. The use of coordinate attention is more accurate for small target detection and positioning, and avoids a large amount of computational overhead. (3) In order to further improve the detection accuracy of the detection algorithm, the loss function of the target detector is redesigned based on DIOU loss function [37] and Focal loss function [38].

The rest of this paper is organized as follows. In Section II, we introduce the development process of target detection technology and the framework of YOLOv3 are introduced. In Section III, the improvement methods of YOLOv3 are described in detail. Section IV introduces the experimental research. Section V gives the analysis of the experimental results. In Section VI, we give the discussion of the experimental results. Lastly, the conclusion and future work are shown in Section VII.

II. RELATED WORK

A. DEVELOPMENT OF TARGET DETECTION TECHNOLOGY

Target detection is a challenging problem in the field of computer vision. The tasks of target detection include predicting the position information and category information of the target. It has been widely used in daily life. The most common application scenarios are pedestrian detection [39], [40], defect detection [41], [42], ship target detection [43], [44], and obstacle detection in automatic driving [45], [46]. Among the traditional target detection methods, HOG (Histogram of Oriented Gradient) [47] and DPM (Deformable Parts Model) [15] are typical. Their detection process is as follows: a specific region is selected as a potential region in the image, and the selection of specific region will be framed by sliding windows of different sizes. Then, by analyzing the potential region, relevant image features are extracted. Finally, the appropriate classifier is selected to complete the classification based on the image features, but the time complexity is too high to meet the real-time requirements. In recent years, with the improvement of deep neural network and hardware computing power, the detection method of deep learning has gradually replaced the traditional method, and the convolutional neural network (CNN) is mainly used in deep learning. By training the input images in the network, convolution network can effectively extract and learn the features of the detected target. After repeated training, the performance of the training model is gradually improved, achieving excellent target detection effect.

The methods based on deep learning can be divided into two categories: one-stage approaches and two-stage approaches. The two-stage models usually include two steps: potential region extraction and category prediction. Region-based Convolutional Neural Network (R-CNN) [13] was put forward in 2014, which was a major breakthrough in the early development of two-stage detectors, and led a wave of research upsurge at that time. R-CNN extracts image features through selective search instead of the traditional sliding window, and then uses classifiers to predict targets. This method improves the detection performance, but the cost of calculation is so large that a small dataset even takes several days to train. To solve this problem, Fast R-CNN [17] was proposed to optimize the training process by simplifying the redundant calculation of overlapping candidate regions. In addition, this method abandons the idea of using multi-classifiers and bounding box regression, and achieves near real-time end-to-end training speed. On the basis of Fast R-CNN, Faster R-CNN [18] model was put forward in 2017, which introduced region proposal network (RPN) to generate possible regions, and reduced reasoning time by one order of magnitude. In the same year, in order to improve the performance of small object detection, another influential architecture called feature pyramid network (FPN) [26], [27] was proposed. FPN can learn the characteristics of different layers, and this network has become a necessary module of existing multi-scale detection methods. An excellent detection method needs both accuracy and computational efficiency. Although the two-stage methods have achieved high accuracy, the calculation speed of these methods is usually inferior to that of the one-stage methods. Therefore, the one-stage detection method represented by YOLO network is widely used in various practical tasks, especially in lightweight platforms. The first generation of YOLO [21] model was developed in 2016. It transforms the detection task into a mathematical regression problem, which greatly inspires the development of the method in the subsequent one-stage methods. In the same year, the Single Shot Multibox Detector (SSD) [20] provided a useful strategy to detect targets by combining features of different scales with default bounding boxes. After that, YOLO series has developed very rapidly. More methods are gradually added to YOLO series, and the detection accuracy is obviously improved. At present, the network based on YOLO has developed to YOLOv5.

B. THE NETWORK OF YOLOV3

The structure of YOLOv3 and its feature extraction network are shown in Fig. 1 and Table 1 respectively. YOLOv3 is a typical end-to-end target detection algorithm, so it runs very fast.

The backbone network of YOLOv3 is Darknet53, which can effectively extract the features of the input image. There are no pooling layers in the network structure of YOLOv3, and the full convolutional network (FCN) is adopted to prevent the loss of feature information. Darknet53 network is mainly composed of a series of 1×1 or 3×3 convolution layers, each of which contains a BN layer and a ReLU layer. It is called Darknet53, because it contains 53 convolution layers. The residual network is used to extract deeper features and avoid gradient fading. Five residual modules are added to Darknet53 network, each of which consists of one or more residual units.

YOLOv3 draws on the idea of FPN to detect targets of different sizes. FPN downsamples the input image five times, and predicts the target through the last three downsampling layers. The sizes of the output images corresponding to the last three down-sampling layers are 52×52, 26×26 and 13×13, respectively. The above three feature maps with different scales are used to detect small targets, medium targets and large targets respectively. Small feature maps can provide deep semantic information, while large feature maps contain a lot of fine-grained information. Therefore, YOLOv3 can not only make predictions at different scales, but also fully learn the semantics of feature maps at different scales during the prediction process.

III. IMPROVEMENTS BASED ON YOLOV3

For military target images in complex scenes, such as small targets or densely covered targets, the performance of YOLOv3 is not satisfactory. YOLOv3 model still has much room for improvement. Our method aims to achieve three goals: (1) reduce the resource occupation of the network model, so that the model is suitable for embedded

TABLE 1. The structure of Darknet53.

Number	Type	Filters	Size	Output
	Convolutional	32	3×3	416×416
	Convolutional	64	3×3/2	208×208
×1	Convolutional	32	1×1	
	Convolutional	64	3×3	
	Residual			208×208
	Convolutional	128	3×3/2	104×104
×2	Convolutional	64	1×1	
	Convolutional	128	3×3	
	Residual			104×104
	Convolutional	256	3×3/2	52×52
×8	Convolutional	128	1×1	
	Convolutional	256	3×3	
	Residual			52×52
	Convolutional	512	3×3/2	26×26
×8	Convolutional	256	1×1	
	Convolutional	512	3×3	
	Residual			26×26
	Convolutional	1024	3×3/2	13×13
×4	Convolutional	512	1×1	
	Convolutional	1024	3×3	
	Residual			13×13

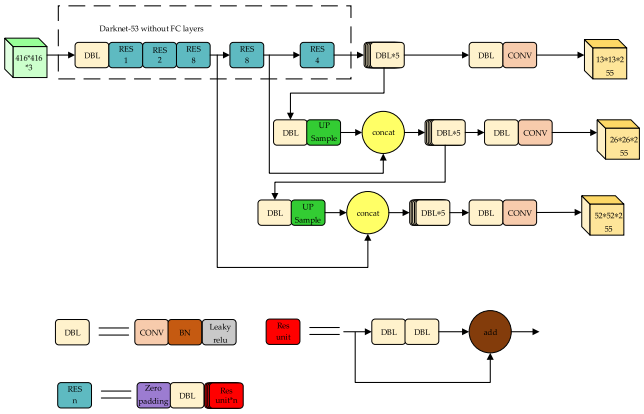


FIGURE 1. The structure of YOLOv3.

devices with limited hardware conditions; (2) compared with other advanced target detectors, further improve the detection accuracy of military targets; (3) improve the real-time performance of target detection in battlefield environment. As shown below, we explained the overall structure of our method and each improvement point in detail.

A. METHOD OVERVIEW

Fig. 2 shows the overall structure of the military target detection model in complex environment proposed in this paper. The overall structure is based on YOLOv3 detection algorithm, and consists of GhostNet feature extraction network, attention mechanism based on coordinate attention block and YOLOv3 target detector.

B. THE FEATURE EXTRACTION NETWORK

The original feature extraction network of YOLOv3 has 53 layers, and the amounts of parameters and calculation is huge. Moreover, the traditional convolution network used by Darknet53 is insensitive to target recognition of different scales due to the limitation of convolution sampling methods. Its ability to deal with the geometric changes of features is relatively limited, and it needs a lot of images training to improve the generalization ability of the network. In the actual use of YOLOv3, if the network encounters elements that are not in the dataset, it is very likely that there will be missed detection and false detection, thus affecting the result of target detection. To solve the above problems, we improved the structure of Darknet53 network and replaced the feature extraction network with GhostNet.

At present, most convolution operations are pointwise convolution for dimension reduction, and then depthwise convolution for feature extraction, as shown in Fig. 3(a). In order to extract more feature information from the input image data, the neural network model trained by vanilla convolution (CNN) often produces many redundant feature maps after training. Although this operation can achieve better performance, it requires a lot of convolution layer calculations, which increases the consumption of computing resources and memory access. The work of some lightweight networks is precisely to make use of the redundancy of features and achieve the effect of lightening the model by cutting out some redundant features. GhostNet combines linear operation with ordinary convolution, and some redundant feature maps can be linearly transformed from the generated ordinary convolution feature maps to obtain similar feature maps, thus producing high-dimensional convolution effect and reducing model parameters and computation.

Assuming that the height and width of each feature map generated by ordinary convolution are h' and w', and the size of convolution kernel is $c \cdot k \cdot k$, the FLOPs of ordinary convolution is $n \cdot h' \cdot w' \cdot c \cdot k \cdot k$. Because the number

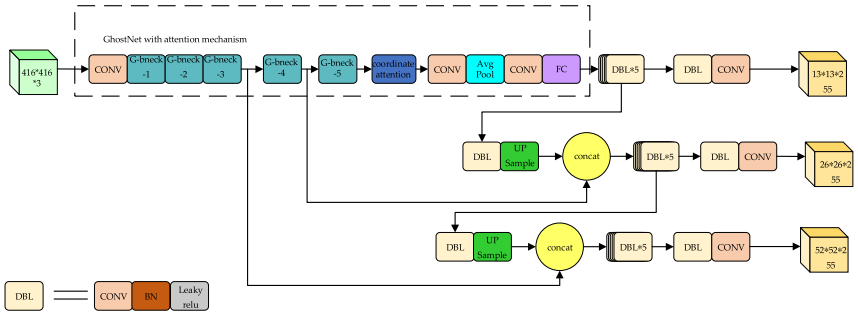
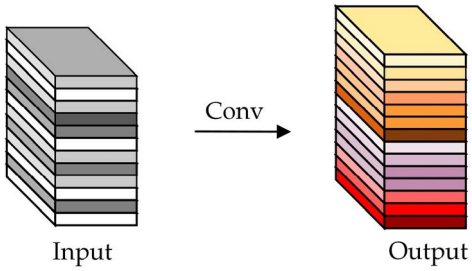
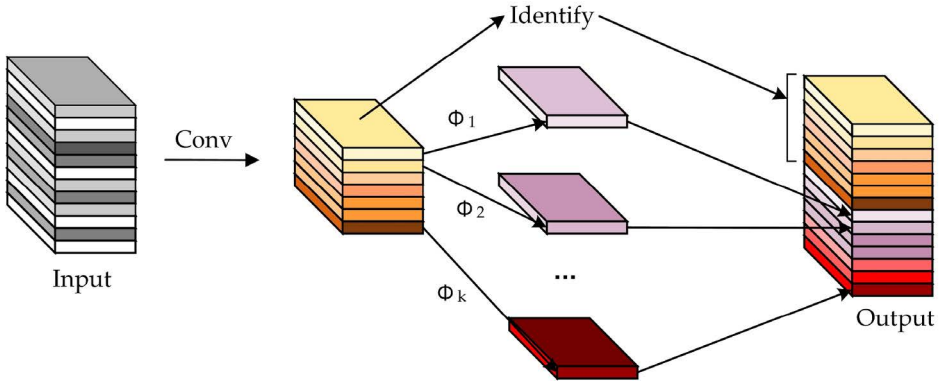


FIGURE 2. The structure of the proposed network.



(a) The convolutional layer



(b) The Ghost module

FIGURE 3. An illustration of the convolutional layer and the Ghost module for outputting the same number of feature maps.

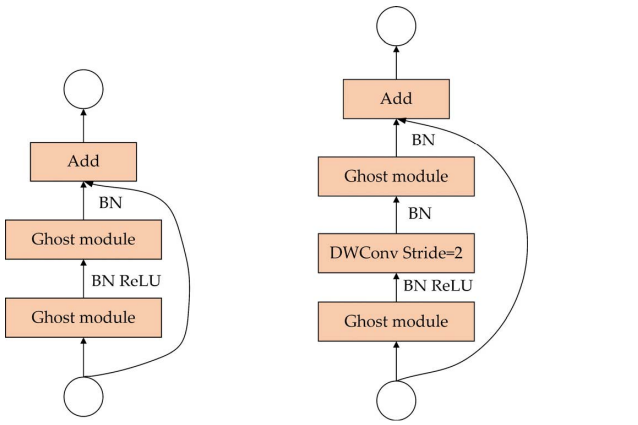
of c channel of the input data is often large, using the convolution kernel of $c \cdot k \cdot k$ to generate n feature maps will lead to a huge amount of computation. GhostNet has made great improvements in convolution processing, and established the Ghost module, as shown in Fig. 3(b). Ghost module is the core of GhostNet feature extractor. Compared with the common convolutional neural network, this module will not change the size of the output feature map, and can greatly reduce the parameters and computational complexity of the network model. Besides, Ghost module has the advantages of plug-and-play and easy transplantation.

According to the advantages and characteristics of Ghost module, the Ghost bottleneck is built based on Ghost module, as shown in Fig. 4. Ghost bottleneck draws lessons from the residual block structure in ResNet model architecture, which integrates multiple convolution layers and shortcuts. Two Ghost modules are superimposed to form G-bneck. These two Ghost modules have different effects: the first Ghost module increases the number of channels of data, while the second Ghost module decreases the number of channels of data. For G-bneck with step size of 1, BN (Batch Normalization) layer is added after two Ghost modules, and RELU (Rectified Linear Unit) activation function is used after the first Ghost module. The difference of G-bneck with step 2 is that there is a depthwise convolution with step 2 between two Ghost modules, in order to complete the downsampling operation.

GhostNet model architecture is based on G-bneck, and its network structure design draws lessons from MobileNetV3,

TABLE 2. Overall architecture of GhostNet.

Input	Operator	#exp	#out	SE	Stride
$224^2 \times 3$	Conv2d 3×3	-	16	-	2
$112^2 \times 16$	G-bneck	16	16	-	1
$112^2 \times 16$	G-bneck	48	24	-	2
$56^2 \times 24$	G-bneck	72	24	-	1
$56^2 \times 24$	G-bneck	72	40	1	2
$28^2 \times 40$	G-bneck	120	40	1	1
$28^2 \times 40$	G-bneck	240	80	-	2
$14^2 \times 80$	G-bneck	200	80	-	1
$14^2 \times 80$	G-bneck	184	80	-	1
$14^2 \times 80$	G-bneck	184	80	-	1
$14^2 \times 80$	G-bneck	480	112	1	1
$14^2 \times 112$	G-bneck	672	112	1	1
$14^2 \times 112$	G-bneck	672	160	1	2
$7^2 \times 160$	G-bneck	960	160	-	1
$7^2 \times 160$	G-bneck	960	160	1	1
$7^2 \times 160$	G-bneck	960	160	-	1
$7^2 \times 160$	G-bneck	960	160	1	1
$7^2 \times 160$	Conv2d 1×1	-	960	-	1
$7^2 \times 960$	AvgPool 7×7	-	-	-	-
$1^2 \times 960$	Conv2d 1×1	-	1280	-	1
$1^2 \times 1280$	FC	-	1000	-	-



(a) Ghost bottleneck with stride=1 (b) Ghost bottleneck with stride=2

FIGURE 4. Ghost bottleneck.

replacing bottleneck in MobileNetV3 with G-bneck. In order to improve the efficiency, GhostNet abandons the use of Hard-swish activation function and uses ReLU to complete nonlinear operation. The first layer of the whole network is an ordinary convolution layer with a convolution kernel of $3 \times 3 \times 16$, followed by a series of G-bnecks, which are divided into different stages according to the size of the input feature map. The step size of the last G-bneck in each stage is 2 to complete the downsampling operation. After the last G-bneck, 960 pointwise convolutions are connected, followed by global average pooling and 1280 pointwise convolutions to convert the feature map into 1280-dimensional vector to complete the subsequent classification task. GhostNet network architecture is shown in Table 2.

At present, there are many lightweight neural networks, such as Mobilenet [48] and Shufflenet [49]. Compared with these lightweight neural networks, GhostNet pays more attention to the redundancy and correlation between feature maps. Compared with the vanilla convolution, GhostNet is divided into two parts. Firstly, it is the ordinary convolution operation, through which the feature map with smaller channel can be generated with less computation. Then, based on these feature maps with few parameters, more feature maps are generated by a series of simple linear operations. Finally, the feature maps of the two steps are spliced together through concat layers to get the final output feature map.

The specific steps are as follows:

(1) Any convolution layer generates n feature maps:

Assuming that the input data is $X \in \mathbb{R}^{c \times h \times w}$, where c is the number of channels of the input data, h and w are the length and width of the input data, respectively, and its operation expression is:

$$Y = X * f + b \tag{1}$$

where $*$ represents convolution operation, b is the bias term, Y denotes the output feature map of the channel, and f is the convolution filters of this layer.

The number n of filters and the number c of channels are usually very large, and there is a lot of redundancy in the middle feature map. The original output feature maps have some intrinsic features, and the number is very small.

(2) Generate ghosting features:

$$Y' = X * f' \tag{2}$$

where $Y \in \mathbb{R}^{h' \times w' \times m}$ is the ordinary convolution output, and $f' \in \mathbb{R}^{c \times k \times k \times m}$ is the used filter. As $m \leq n$, the bias term is simplified. The obtained intrinsic feature map is only m-dimensional. Under the condition that the spatial scale is the same, it is necessary to obtain n-dimensional feature maps and perform a series of simple linear transformations on these intrinsic feature maps:

$$y_{ij} = \Phi_{ij}\left(y_i'\right), \forall i = 1, \ldots, m, j = 1, \ldots, s \tag{3}$$

where $y_i^{'}$ is the i-th intrinsic feature map in $Y^{'}$, and Φ_{ij} is the function of the j-th linear transformation of the i-th feature map. Finally, an identity mapping $\Phi_{i,s}$ is added. In order to keep the intrinsic feature map, which is superimposed on the feature map obtained by linear transformation.

Assuming that the Ghost module contains an intrinsic feature map and $m \cdot (s-1) = \frac{n}{s} \cdot (s-1)$ linear transformation operations, each operation kernel size is $d \times d$, and the theoretical acceleration ratio of upgrading ordinary convolution by Ghost module is as follows:

$$\begin{aligned} r_s &= \frac{n \cdot h' \cdot w' \cdot c \cdot k \cdot k}{\frac{n}{s} \cdot h' \cdot w' \cdot c \cdot k \cdot k + (s-1) \cdot \frac{n}{s} \cdot h' \cdot w' \cdot d \cdot d} \\ &= \frac{c \cdot k \cdot k}{\frac{1}{s} \cdot c \cdot k \cdot k + \frac{s-1}{s} \cdot d \cdot d} \\ &\approx \frac{s \cdot c}{s + c - 1} \approx s \end{aligned} \tag{4}$$

The amplitude of $d \times d$ is similar to that of $k \times k$. Since $s \ll c$, the theoretical parameter compression ratio is:

$$\begin{aligned} r_c &= \frac{n \cdot c \cdot k \cdot k}{\frac{n}{s} \cdot c \cdot k \cdot k + (s-1) \cdot \frac{n}{s} \cdot d \cdot d} \\ &\approx \frac{s \cdot c}{s + c - 1} \approx s \end{aligned} \tag{5}$$

Therefore, theoretically, changing Darknet53 into GhostNet can speed up and reduce the running time.

C. ATTENTION MECHANISM BASED ON COORDINATE ATTENTION BLOCK

Researches on lightweight network show that channel attention has the ability to bring significant performance improvement to the model. The channel attention usually ignores the location information which is very important for generating the spatial selective attention map. In order to improve the ability of models to extract useful features, this paper added a novel coordinate attention block to the last layer of feature extraction network, as shown in Fig. 5.

We used the coordinate attention (CA) block to embed the location information into the channel attention, so that the network can pay attention to a larger area. Unlike channel attention, which transforms input into a single feature vector through two-dimensional global pooling, CA block decomposes channel attention into two one-dimensional feature coding processes that aggregate features in different directions. This has the advantage of capturing long-range dependence along one spatial direction and keeping accurate position information along another spatial direction. Then, the generated feature maps are coded separately to form a pair of directional-aware and position-sensitive feature maps, which can be applied to the input feature maps complementarily to enhance the representation of interested objects.

Some images in the dataset have low resolution, the targets in the images are small and the details are fuzzy, so the network is easy to lose features when extracting features. Therefore, we added CA mechanism to the detection network to help the network select the information that is more critical

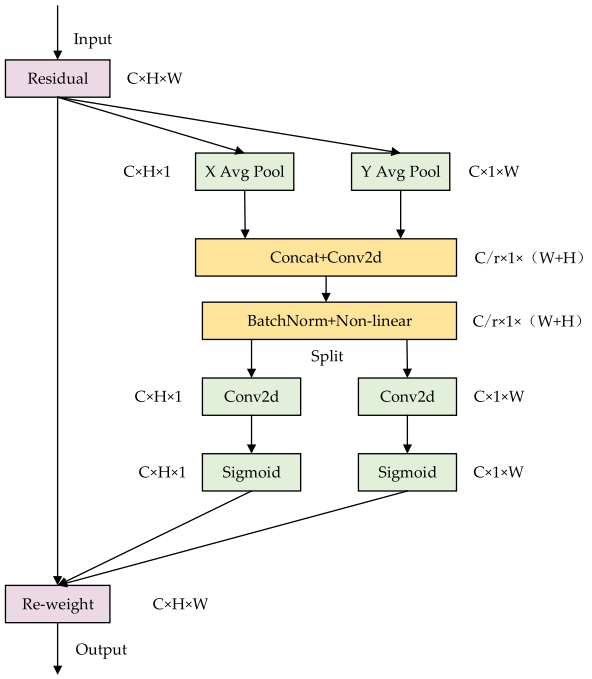


FIGURE 5. Coordinate attention structure, including coordinate information embedding and attention generation.

to the current detection task. By using the CA mechanism, the model can pay more attention to important features and suppress unnecessary features such as background information, which significantly reduces the missed detection rate of small target detection, thus improving the expressive force of the model.

The operation process of coordinate attention consists of two steps: coordinate information embedding and coordinate attention generation.

(1) Coordinate information embedding

Global pooling is often used in channel attention to globally encode spatial information as channel descriptors, so it is difficult to save location information. In order to promote the attention block to capture spatial long-range dependence with accurate location information, it decomposes global pooling into two feature encoding operations. Specifically, for the input X, we first encode each channel along the horizontal and the vertical coordinate direction with the pooling kernels $(H, 1)$ or $(1, W)$ of two spatial extents, so the output of the c-th channel with the height h is expressed as follows:

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i \le W} x_c(h, i) \tag{6}$$

similarly, the output of the c-th channel with width w is expressed as follows:

$$z_c^w(w) = \frac{1}{H} \sum_{0 \le j \le H} x_c(j, w) \tag{7}$$

The above two transformations aggregate features along two spatial directions, and return a pair of directional

awareness attention maps. This is completely different from the method of SE block [50] to generate a feature vector. These two transformations also allow attention block to capture the long-range dependence along one spatial direction and save the accurate position information along the other direction, which is helpful for the network to locate the target of interest more accurately. This coordinate information embedding operation corresponds to the parts of X Avg Pool and Y Avg Pool in Fig. 5.

(2) Coordinate attention generation

Coordinate attention generation can make full use of the captured position information, accurately locate the region of interest, and effectively capture the relationship among channels. The operation is to concatenate the two feature maps generated by the previous blocks first, and then use a shared 1×1 convolution to transform F_1. The generated $f \in \mathbb{R}^{C/r \times (H+W)}$is an intermediate feature map of spatial information in the horizontal and vertical directions, where r means the downsampling ratio and is used to control the size of the block just like SE block. Then, f is divided into two separate tensors $\boldsymbol{f}^h \in \mathbb{R}^{C/r \times H}$ and $\boldsymbol{f}^w \in \mathbb{R}^{C/r \times W}$along the spatial dimension, and then two 1×1 convolution F_h and F_w are used to transform the feature maps f^h and f^w into the same channel as the input X. The results are as follows:

$$\boldsymbol{g}^h = \sigma\left(F_h\left(\boldsymbol{f}^h\right)\right) \tag{8}$$

$$\boldsymbol{g}^w = \sigma\left(F_w\left(\boldsymbol{f}^w\right)\right) \tag{9}$$

then, $\boldsymbol{g}^h$ and $\boldsymbol{g}^w$ are expanded. As attention weight, the final output of coordinate attention block can be expressed as the following formula.

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j) \tag{10}$$

Through the above process, coordinate attention block completed the attention in the horizontal and vertical directions, and it is also a kind of channel attention. Similar to coordinate attention, SENet and CBAM [51] are prevalent attention mechanisms in deep learning. SE module only pays attention to the coding of information between channels, but ignores the important position information. CBAM block can only capture the local correlation but not the dependency of larger regions. The coordinate attention block embeds the location information into the channel attention, and decomposes the channel attention into two parallel one-dimensional feature codes. Compared with SE and CBAM, coordinate attention block can efficiently integrate the spatial coordinate information into the generated feature maps, and each feature map captures the long-distance dependency of the input feature map along one spatial direction, so the extraction of feature information is more comprehensive.

D. OPTIMIZATION OF LOSS FUNCTION

The loss function L_s is composed of the bounding box loss function L_i, the confidence loss function L_d and the category loss function L_c, as shown in formula (11):

$$L_s = L_i + L_d + L_c \tag{11}$$

In the regression prediction of bounding box by IOU, if there is no overlap between the prediction box and the ground truth box, the gradient of loss function will be zero. To solve this problem, we introduced DIOU loss function for the regression prediction of bounding box, which can effectively solve the problems of inaccurate regression and slow convergence. The calculation formula is as follows:

$$L_i = 1 - \frac{\left|B_p \cap B_g\right|}{\left|B_p \cup B_g\right|} + \frac{\rho^2\left(b_p, b_g\right)}{d^2} \tag{12}$$

where b_p and b_g denote the center points of predicted box $B_p = (x, y, w, h)$ and target box $B_g = \left(x_g, y_g, w_g, h_g\right)$, respectively. $\rho(\cdot)$ represents the Euclidean distance between b_p and b_g, and d means the diagonal length of the smallest rectangle covering the target box and the prediction box.

There are some sample data that are easy to classify in the training dataset, so the optimization of the model is contrary to the expected direction. In order to reduce the influence of negative examples on model optimization, we introduced Focal loss function to reduce the weight of negative examples, so that it can focus on more difficult and complicated classified samples. The calculation formula is as follows:

$$\begin{aligned} L_d = &-\left(y_g - y_p\right)^2 \times y_g \ln y_p - \left(y_g\right. \\ &\left. - y_p\right)^2 \left(1 - y_g\right) ln\left(1 - y_p\right) \end{aligned} \tag{13}$$

where y_p and y_g are the predicted value of the target confidence and the ground truth, respectively.

The classification loss can be calculated by the cross-entropy loss function. The input image is divided into $S \times S$ grid cells, and each grid corresponds to three target prediction boxes. The classification loss is only valid for grids with identified objects. The calculation formula is as follows:

$$L_c = \sum_{i=0}^{S \times S} \sum_{j=0}^{3} \sum_{rin\ classes} I_{ij}\, BC\left(p_i(r), \hat{p}_i(r)\right) \tag{14}$$

$$BC\left(a, \hat{a}\right) = -\left[a\, ln\, \hat{a} + (1-a) \ln\ \left(1 - \hat{a}\right)\right] \tag{15}$$

where r and *classes* respectively represent the category of detection target and the category of target contained in dataset. I_{ij} indicates whether the target is detected in the j-th prediction box associated with the i-th grid. If the target is detected, the value of I_{ij} is 1, otherwise, the value is 0. $p_i(r)$ and $\hat{p}_i(r)$ are the ground truth and predicted values of the classification probability of the detected target, respectively. Formula (15) is the Cross-entropy cost function, which corresponds to the solution method of $BC\left(p_i(r), \hat{p}_i(r)\right)$ in formula (14), where a represents $p_i(r)$ in formula (14).

To ensure the recognition accuracy of military targets in complex environment, it is necessary to detect military targets on multiple scales. So, the final loss function L is as follows:

$$L = \sum_{i=1}^{3} L_i^i + L_d^i + L_c^i \tag{16}$$

where i is the scale of the detection feature.

TABLE 3. Analysis of K-means clustering results.

K	Anchors	Avg IoU
1	(239,291)	0.35
2	(49,105), (154,290)	0.52
3	(35,75), (90,200), (209,330)	0.59
4	(23,52), (58,127), (111,251), (237,341)	0.63
5	(23,50), (54,127), (136,163), (100,275), (234,344)	0.65
6	(19,44), (45,98), (73,198), (149,154), (125,301), (257,348)	0.68
7	(19,44), (43,101), (105,132), (70,210), (123,305), (198,207), (260,360)	0.69
8	(14,35), (32,68), (53,123), (73,209), (142,151), (113,289), (185,326), (301,357)	0.71
9	(12,31), (28,58), (44,108), (61,182), (105,121), (98,265), (189,197), (159,336), (286,359)	0.72

E. OTHER IMPROVEMENTS

YOLOv3 draws on the idea of anchor box in Faster-RCNN, and the use of anchor box can predict bounding boxes more accurately. The anchor box is used to make the bounding box match the size of the target before detection. As the anchor box in YOLOv3 is obtained from VOC dataset, we re-generate the anchor box by K-means based on the characteristics of the self-built dataset. K-means algorithm is an iterative clustering algorithm. According to the image samples in the training dataset, the function of the K-means algorithm is conducting latitude clustering to make anchor boxes and adjacent ground truth have larger IoU values, which is not directly related to the size of anchor boxes. In order to obtain the best detection effect, the size of the anchor box should be as close as possible to the ground truth of the target. Therefore, their IoU values must be as large as possible, so as to reduce the error caused by the anchor box. The distance function between the ground truth and the bounding box is:

$$d\,(box, centroid) = 1 - IoU\,(box, centroid) \tag{17}$$

IoU indicates the intersection ratio, and its definition is shown in the formula:

$$IoU = \frac{S_{overlap}}{S_{union}} \tag{18}$$

$S_{overlap}$ is the overlapping area between the predicted box and the ground truth, while S_{union} means the union area between them. The pseudo code of K-means in this paper is shown in algorithm 1.

Generally speaking, the generated anchor box results are related to the number of cluster center points. The more cluster points, the larger the IOU between the ground truth and the anchor box. The variation trend of average IOU with the increase of cluster points is shown in Fig. 6, from which it can be seen that the curve gradually becomes gentle. According to the image samples in the training dataset, the targets are clustered by K-means algorithm, and nine anchor boxes are obtained, which are (12,31), (28,58), (44,108), (61,182), (105,121), (98,265), (189,197), (159,336) and (286,359). Among them, (12,31), (28,58) and (44,108) are the anchor boxes of Scale 3. (61,182), (105,121) and (98,265) are the anchor boxes of Scale 2. (189,197), (159,336) and (286,359) are the anchor boxes of Scale 1. We use these anchor boxes to carry out target detection experiments. The clustering results are shown in Table 3:

Algorithm 1 Clustering Anchor Boxes With K-Means

Step 1: Set k cluster center points randomly: (W_i, H_i), $i \in \{1, 2, \ldots, k\}$. W_i and H_i denote the width and height of the anchor box respectively, which are proportional to the whole image.

Step 2: Calculate the distance between each cluster center and each ground truth by the following formula: $d\,(box, centroid) = 1 - IoU\,(box, centroid)$. Because the location of the anchor box is not fixed, the center point of each ground truth coincides with the clustering center.

Step 3: Recalculate the cluster center of each cluster: $W_i^{'} = \frac{1}{N_i}\sum w_i, H_i^{'} = \frac{1}{N_i}\sum h_i$

Step 4: Repeat Step 2 and Step 3 until clustering converges.

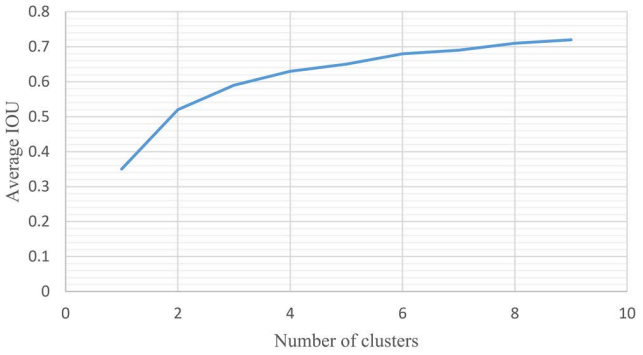


FIGURE 6. The relationship between clustering number and average IOU by K-means clustering.

In non-maximal suppression (NMS), the redundant boxes caused by bounding box regression are usually suppressed based on the intersection-union ratio (IoU). Because only the influence of overlapping areas is considered, false suppression is often generated in the case of occlusion. The formula of NMS is as follows:

$$s_i = \begin{cases} s_i, & IoU\,(H, B_i) < \varepsilon \\ 0, & IoU\,(H, B_i) \geq \varepsilon \end{cases} \tag{19}$$

where H is the bounding box with the highest classification score, B_i is the compared bounding box, ε is the threshold of NMS, and s_i is the classification score.

In order to suppress redundant boxes more accurately and improve the accuracy of detection, on the basis of IoU, the distance between the center points of two bounding boxes is considered as DIoU, and the following penalty items are established:

$$P_{DIoU} = \frac{\rho^2(b, h)}{r^2} \tag{20}$$

where h is the center point of the bounding box with the highest detection score, b is the center point of the bounding box compared with it, $\rho(\cdot)$ represents the Euclidean distance between the two center points, and r is the diagonal distance of the minimum circumscribed rectangle of the two bounding boxes.

In our research, DIoU-NMS [37] algorithm is adopted, and DIoU is used to replace the original IoU. DIoU considers the influence of the overlapping area and the center distance between two bounding boxes, and reduces the suppression of errors, thus improving the detection performance of the model. The formula is as follows:

$$s_i = \begin{cases} s_i, & IoU - P_{DIoU}(H, B_i) < \varepsilon \\ 0, & IoU - P_{DIoU}(H, B_i) \geq \varepsilon \end{cases} \tag{21}$$

IV. EXPERIMENTAL STUDIES

In order to verify the validity of YOLO-G model in military target detection, we made a dataset based on armed men with different weapons, and carried out the following two experiments on the dataset:

(1) Ablation experiment of YOLO-G

The YOLO-G proposed in this paper has taken many improvement measures. In order to verify the contribution of each improvement strategy to the detection performance, ablation experiment was carried out in the training dataset constructed in this paper.

(2) Comparative experiment among YOLO-G and other detection algorithms

In order to further confirm the detection effect of YOLO-G in the military target dataset, four state-of-the-art target detection algorithms (Faster-RCNN, SSD, YOLOv3, YOLOv5s) are selected for detection experiment.

A. DATASET

The military targets detected in this paper are armed men equipped with different weapons. According to the different weapons, the threat degree of the target is determined, that is, not only the armed men but also the weapons equipped by armed men need to be detected. In the real combat environment, military action units usually use military camouflage or camouflage similar to the complex natural environment, thus greatly improving their battlefield survivability and effectively reducing the probability of being discovered by various detection means. As the training samples belong to sensitive military resources, there is no public dataset can be directly used. In view of the above-mentioned task of military target detection in complex environment, we obtained 5,000 images of armed men with different weapons in complex background through google and bing search engines, and constructed the military target dataset. Our dataset contains two kinds of targets: armed men with rifles (Soldier_R) and armed men with rocket launchers (Soldier_RPG). The dataset includes desert, jungle, town and other battlefield environments in total. At the same time, the factors affecting the target detection results, such as foreground occlusion, smoke, target size, and imaging angle, are considered. An example of partial images of the dataset is shown in Fig. 7. The dataset is divided into training set, test set and verification set according to the ratio of 7:2:1. Then, the image labeling software LabelImg is used to lable the sample data in the dataset, and the dataset labeling format is consistent with PASCAL VOC dataset. There are 5,000 images in the dataset, including 10,157 target annotation boxes.

B. EXPERIMENTAL PROCESS

1) EXPERIMENTAL PLATFORM

All the experiments in this article were carried out on the workstation with Ubuntu16.04 operating system. The graphics processor is GeForce RTX 2080ti×2, and the memory is 16GB. The neural network is built with Pytorch1.7.1 as the basic framework and programmed with Python language. The training and testing data are the military target dataset constructed in this paper.

2) IMPLEMENTATION DETAILS

During the training process, the stochastic gradient descent (SGD) method is used to train the detection model. The batch size is set to 32, and the training round epoch is set to 100. The initial learning rate, weight attenuation coefficient and momentum factor are set to 0.001, 0.0005 and 0.9 respectively. When the number of iterations reaches 60,000 and 80,000, the learning rate is adjusted by dividing by 10 respectively. In addition, we generate more military target samples for network training through image rotation, scaling, contrast adjustment and saturation adjustment, so as to enhance the generalization of the network and improve the accuracy and stability of military target detection.

In the ablation experiment, we used the original YOLOv3 as the baseline, and then added GhostNet network, coordinate attention mechanism and improved loss function one by one. After each experiment, the AP of each category, the mAP of all targets and the detection speed of the network model in the dataset are obtained. By observing the values of each index, the promotion of the model detection performance by each improvement strategy is evaluated. In the comparative experiment of several detection algorithms, we trained and tested YOLO-G and several advanced detection algorithms on self-built dataset, and the running environment of each algorithm is consistent. Then, the superiority of YOLO-G model was further verified by AP, mAP, FPS and model size.

FIGURE 7. Examples of partial dataset images. (a) Soldiers with rifles; (b) Soldiers with individual rocket launchers; (c) Soldiers with two weapons.

3) PERFORMANCE METRICS

In this paper, the average precision (AP) is selected to evaluate the detection accuracy of the detection model for a single target category. AP is related to precision (Pr) and recall (Re), which can effectively evaluate the detection performance of the model for a single target category. The mean of average precision (mAP) is used to evaluate the comprehensive detection performance of the detection model for multiple target categories. Generally, the higher the AP and mAP values, the better the detection performance of the model, and vice versa. The detection speed of the detection model is measured by the number of images per second (FPS).

$$AP = \int_0^1 precision\,(recall)\,d\,(recall) \tag{22}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{23}$$

$$Pr = \frac{TP}{TP + FP} \tag{24}$$

$$Re = \frac{TP}{TP + FN} \tag{25}$$

$$FPS = \frac{1}{t} \tag{26}$$

where the true positive (TP) is the number of positive samples predicted to be positive, false positive (FP) means the number of samples predicted to be positive but actually negative, and false negative (FN) is the number of samples predicted to be negative but actually positive. FPS means the pictures that can be processed per second, and t represents the time required to process a picture.

V. RESULTS

A. ABLATION EXPERIMENT

The YOLO-G network proposed in this paper uses the lightweight convolutional neural network GhostNet as the feature extraction network. The model uses the coordinate attention mechanism to further improve the detection accuracy. We also redesigned the loss function of multi-scale target detector. In order to verify the effectiveness of each improvement point, we conducted ablation experiment on the self-built dataset to complete the evaluation. To ensure the fairness of the evaluation, we set the same parameters for each variable.

When the input resolution is 416 × 416, the detection results are shown in Table 4. It can be seen that the improvement strategies of each module in this paper are helpful to

TABLE 4. Comparison of metrics in ablation experiment.

Methods	a	b	c	d
GhostNet		√	√	√
Coordinate Attention			√	√
Improved Loss Function				√
AP(Soldier_R)	80.9%	81.6%	81.9%	85.6%
AP(Soldier_RPG)	91.8%	92.9%	94.6%	93.0%
mAP@0.5	86.4%	87.3%	88.3%	89.3%
FPS	71.9	104.2	98.0	97.8

improve the detection performance. Method (a) is the original YOLOv3 with basic Darknet53 backbone, serving as the baseline, which achieves 80.9% and 91.8% AP in Soldier_R and Soldier_RPG, respectively.

Method (b) is the improved YOLOv3, which is replaced with the feature extraction network. The model size of GhostNet is 9.5MB, far less than YOLOv3 with 235.1MB, so its number of network parameters is much smaller than Darknet53. Ghost model uses cheap operations, so the amount of calculation is only 7.14% of that of the original YOLOv3. The use of GhostNet makes the model has faster detection speed and higher detection accuracy. Experimental results show that, compared with the original YOLOv3 model, the detection result of mAP is improved by 0.9% and FPS is improved by 32.3 frames /s, which effectively increase the detection performance.

Method (c) adds the coordinate attention mechanism to method (b). Coordinate attention improves the feature extraction ability of the network, captures more accurate location information and target features, and suppresses the influence of interference factors on detection results, thus strengthening the characterization ability of target detection features. From the experimental results, it can be seen that the use of attention mechanism can improve the mAP from 87.3% to 88.3%. The detection accuracy can be effectively improved when the FPS changes little.

Method (d) optimizes the loss function based on method (c). The loss function of YOLOv3 model can improve the accuracy of the model. Modifying the loss function usually only affects the training process, but will not or rarely affect the inference time of the network. In this paper, the loss function is redesigned by introducing DIOU loss function and Focal loss function, which further improve the positioning and classification ability of the detection model. In the process of selecting the detection box, DIOU-NMS is introduced to make the results more reasonable and effective. The experimental results show that the mAP increases from 88.3% to 89.3% with little change in FPS.

In order to explain the improvement of detection effect more intuitively, Figs. 8-10 show the processing results of several typical scene examples. (a), (b), (c), (d) and (e) respectively correspond to the detection results of the original image, baseline, (b)+GhostNet, (c)+coordinate attention and (d)+improved loss function. As can be seen from Fig. 8, although YOLOv3 can detect distant and different scale targets, it mistakenly detects the interferent in the background as the target. After replacing GhostNet, the false detection situation disappears, and the detection performance of the model is relatively improved. With the addition of the coordinate attention mechanism, the detection accuracy of the model is further improved. By replacing the improved loss function on the basis of the previous model, the detection accuracy of the model is improved to a greater extent, and all targets can be detected completely and correctly. Fig. 9 shows a typical occlusion scene. From the detection results, it can be seen that YOLOv3 has missed detection in this situation. After the introduction of GhostNet, the missed detection disappears. With the addition of the coordinate attention mechanism, the detection accuracy of the model is greatly improved. Finally, by adding the newly designed loss function, the detection accuracy of the model is further improved, and all targets can be detected with high accuracy. Fig. 10 is a scene with dense targets and small targets, from which it can be seen that YOLOv3 has missed many small targets in the distance. After replacing GhostNet, the number of missed targets is reduced, and the detection accuracy is relatively improved. With the addition of the coordinate attention mechanism, the number of missed detections is further reduced and the detection accuracy is further improved. After replacing the improved loss function, the model only loses one target in the figure, and the other targets can be correctly identified with high accuracy. To sum up, our method achieves better detection effect than YOLOv3.

B. COMPARISON WITH OTHER MODELS

In order to further confirm the overall detection effect of the improved YOLOv3 in the military target dataset, the proposed YOLO-G is compared with the four state-of-the-art target detection algorithms. The comparison algorithms are: (1) the original YOLOv3, (2) Faster R-CNN, which is famous for its high accuracy based on regional algorithm, (3) SSD, which is close to YOLOv3 detection speed based on regression algorithm, and (4) YOLOv5s, the latest one-stage target detection algorithm. In order to effectively compare the performance of improved YOLOv3, the training environments and dataset of the four algorithms are completely consistent.

Fig. 11 shows the P-R curves generated by the five algorithms after running in the dataset. From the figure, we can see that the improved algorithm (ours) completely surrounds the curves of the other four algorithms, which is closer to the point (1,1) numerically. This shows that compared with other algorithms, the improved algorithm has obvious advantages and higher accuracy. The performance of YOLOv5s is second only to Yolo-G, which shows that it has good performance in identifying targets in the dataset. The result of YOLOv3 is better than Faster R-CNN and SSD, while SSD is the worst among all detection results, which is caused by the attribute of the algorithm itself. Fig. 12 shows the mAP curves of the five models, which also shows that the improved algorithm

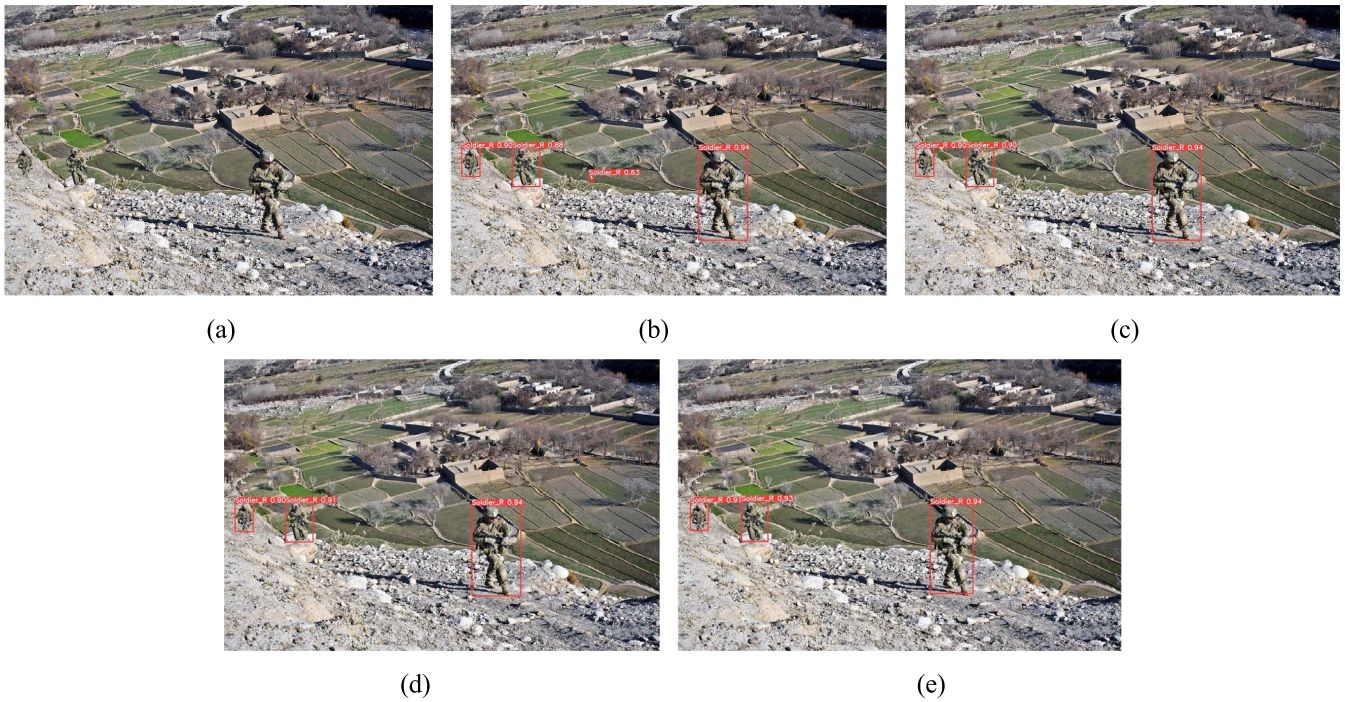


FIGURE 8. The detection results of the improved strategy of the algorithm in scene 1. (a) the original image; (b) baseline; (c) baseline + GhostNet; (d) baseline + GhostNet + coordinate attention; (e) baseline + GhostNet + coordinate attention + improved loss function.

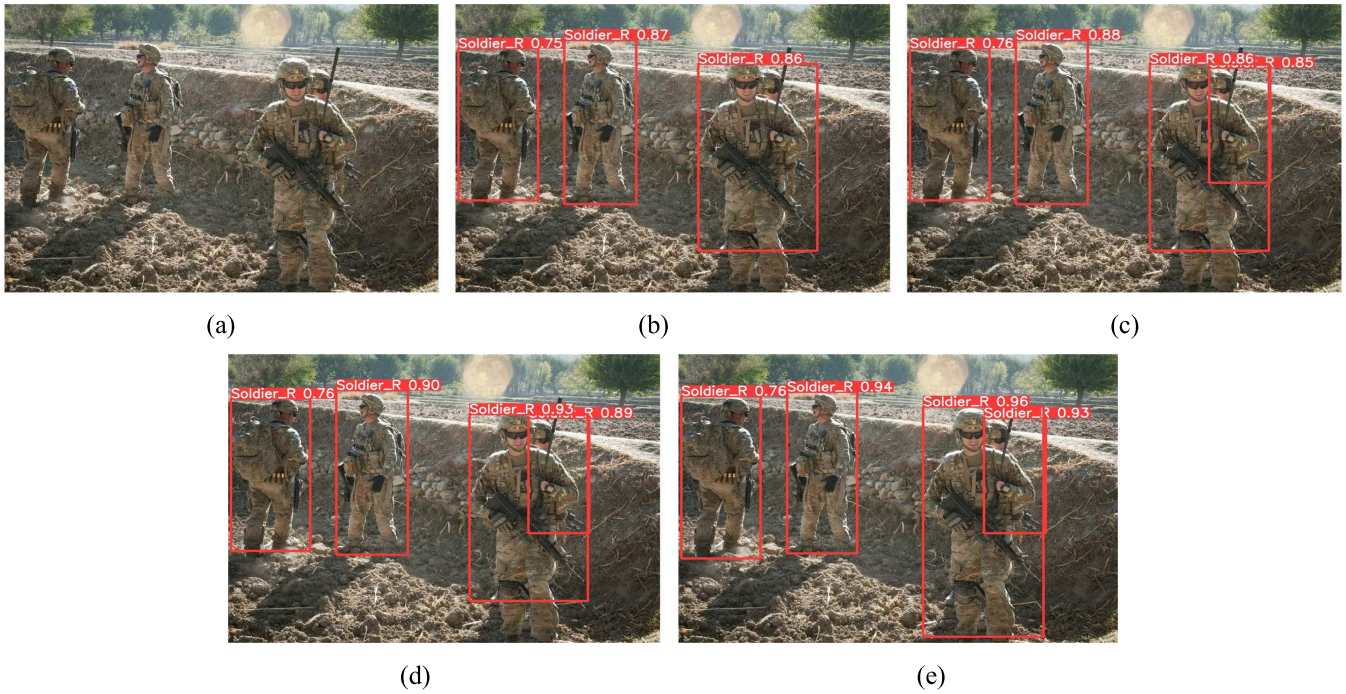


FIGURE 9. The detection results of the improved strategy of the algorithm in scene 2. (a) the original image; (b) baseline; (c) baseline + GhostNet; (d) baseline + GhostNet + coordinate attention; (e) baseline + GhostNet + coordinate attention + improved loss function.

has more advantages in performance, and confirms the correctness of the above P-R curve results.

Table 5 shows the comparison results of five models in the evaluation metric. We can see that YOLO-G has the highest Pr, Re, AP and mAP values among the five models. Compared with YOLOv3, the mAP of YOLO-G is 2.9% higher and FPS is 25.9 higher. YOLO-G has better applicability in the battlefield environment with strict speed requirements.

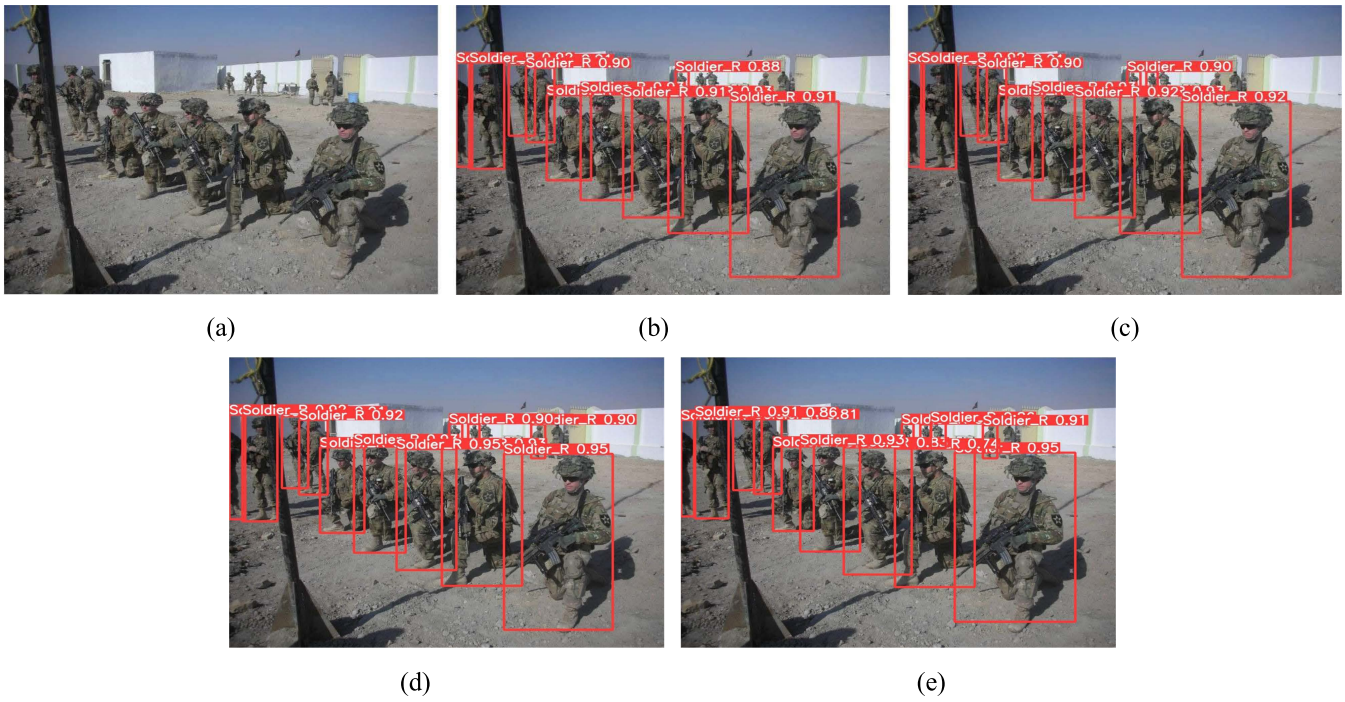


FIGURE 10. The detection results of the improved strategy of the algorithm in scene 3. (a) the original image; (b) baseline; (c) baseline + GhostNet; (d) baseline + GhostNet + coordinate attention; (e) baseline + GhostNet +coordinate attention + improved loss function.

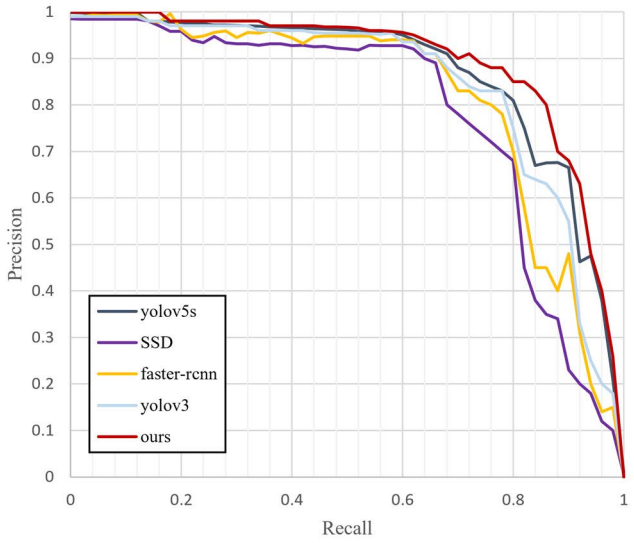


FIGURE 11. P-R curves of each model.

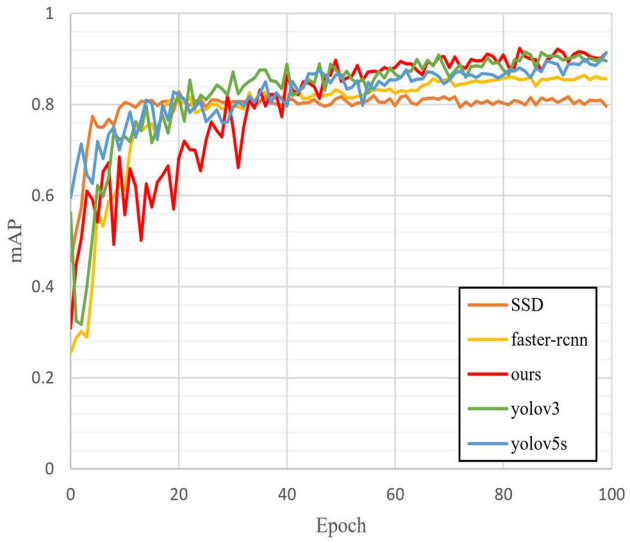


FIGURE 12. mAP curves of each model.

Compared with Faster R-CNN, the mAP of YOLO-G has increased by 7.6%, and the detection speed has increased by 17.8 times. It can be seen that the two-stage detector has shortcomings in detecting military targets, especially the detection speed. Compared with SSD, the mAP of YOLO-G increases by 10.9% and FPS increases by 38.5. Although SSD is a one-stage detector, it is not dominant in detection speed and accuracy, and the detection accuracy is the worst among the five methods. YOLOv5s is the latest one-stage detector at present. From the experimental results, it can be seen that its detection speed is very fast. FPS of YOLOv5s is the largest among the five models, and it is nearly twice faster than YOLO-G with lightweight backbone network. However, in terms of detection accuracy, the method proposed in this paper has more advantages, and the detection speed of YOLO-G is enough to meet the actual needs of the battlefield environment. It can be seen from the model size index that our proposed model is reduced to 1/6 of the original YOLOv3, thus realizing the network lightweight.

TABLE 5. Comparison of evaluation indexes among different models.

Model	Pr	Re	AP(Soldier_R)	AP(Scoldier_RPG)	mAP@0.5	FPS	Size(MB)
Faster R-CNN	85.9%	45.6%	79.6%	83.8%	81.7%	5.5	338.0
SSD	80.4%	20.3%	73.5%	83.3%	78.4%	58.3	91.1
YOLOv3	87.8%	80.4%	80.9%	91.8%	86.4%	71.9	235.1
YOLOv5s	88.7%	85.4%	83.3%	92.9%	88.1%	192.3	14.4
YOLO-G	88.9%	86.3%	85.6%	93.0%	89.3%	97.8	42.7

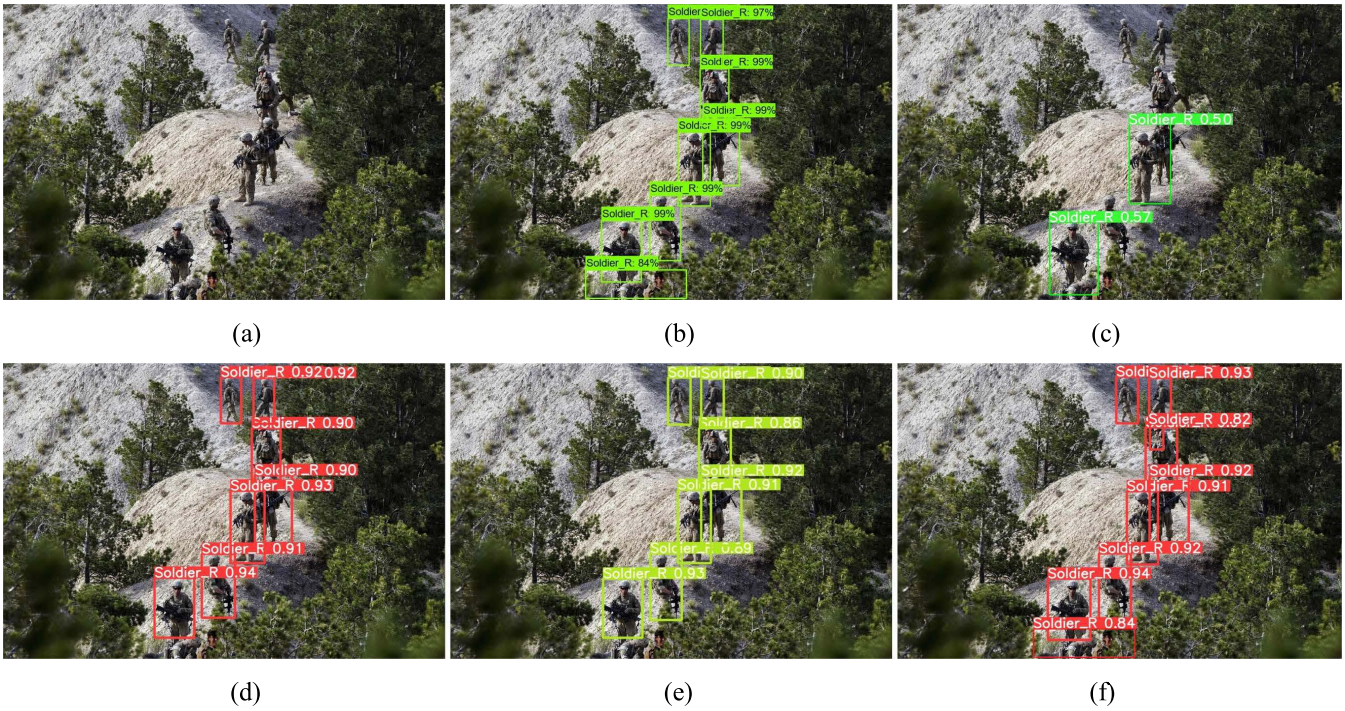


FIGURE 13. Comparison of detection performance of different models in scene 1. (a) the original image; (b) Faster R-CNN; (c) SSD; (d) YOLOv3; (e) YOLOv5s; (f) YOLO-G.

To fully illustrate the applicability of our model to different image scenes, we give the target detection results of several typical scene conditions in Figs. 13-17. (a), (b), (c), (d), (e) and (f) respectively correspond to the detection results of original images, Faster R-CNN, SSD, YOLOv3, YOLOv5s and our algorithm.

From the detection result of the first scene, it can be seen that the detection result of SSD is the worst, with only two targets detected. The detection results of YOLOv3 and YOLOv5s are similar, but they are missed for the occluded targets. As a two-stage detector, the detection effect of Faster R-CNN is better than the above-mentioned one-stage detectors, but there is also a missed detection target. However, YOLO-G proposed in this paper detects all targets, including those missed by other models due to occlusion. In the second scene, there are two kinds of targets. Only SSD identifies the Soldier_RPG as Soldier_R, and the detection result is the worst among all the models. Other detection models can correctly identify it. Faster R-CNN fails to detect the farthest small target. YOLOv3 has a false detection, that is, the target and the object held by the target are identified as Soldier_R. YOLOv5s and our method can perfectly detect the targets in the picture, but the confidence of each target detected by YOLO-G is higher than that of YOLOv5s. In the third scene, the detection results of Faster R-CNN, YOLOv3 and YOLOv5s are similar. Except that the farthest occluded target is not identified, other targets can be correctly detected. SSD has poor detection effect for small targets in the distance, and has 3 missed small targets. YOLO-G can fully identify all targets. The fourth scene also contains two kinds of targets. Faster R-CNN, YOLOv3 and YOLOv5s all can detect most of the targets, but all the small targets in the distance are missed to varying degrees. SSD hardly detects all small targets. However, our method can effectively detect all kinds of targets with high confidence. Although the fifth scene is very simple, it contains the situation of occlusion. From the detection results, it can be seen that other detection models can correctly detect the two kinds of targets in the figure, except SSD which identifies Soldier_RPG as Soldier_R. Faster R-CNN has a better detection effect on the occluded target, second only to our YOLO-G model, but superior to other one-stage detectors. The detection

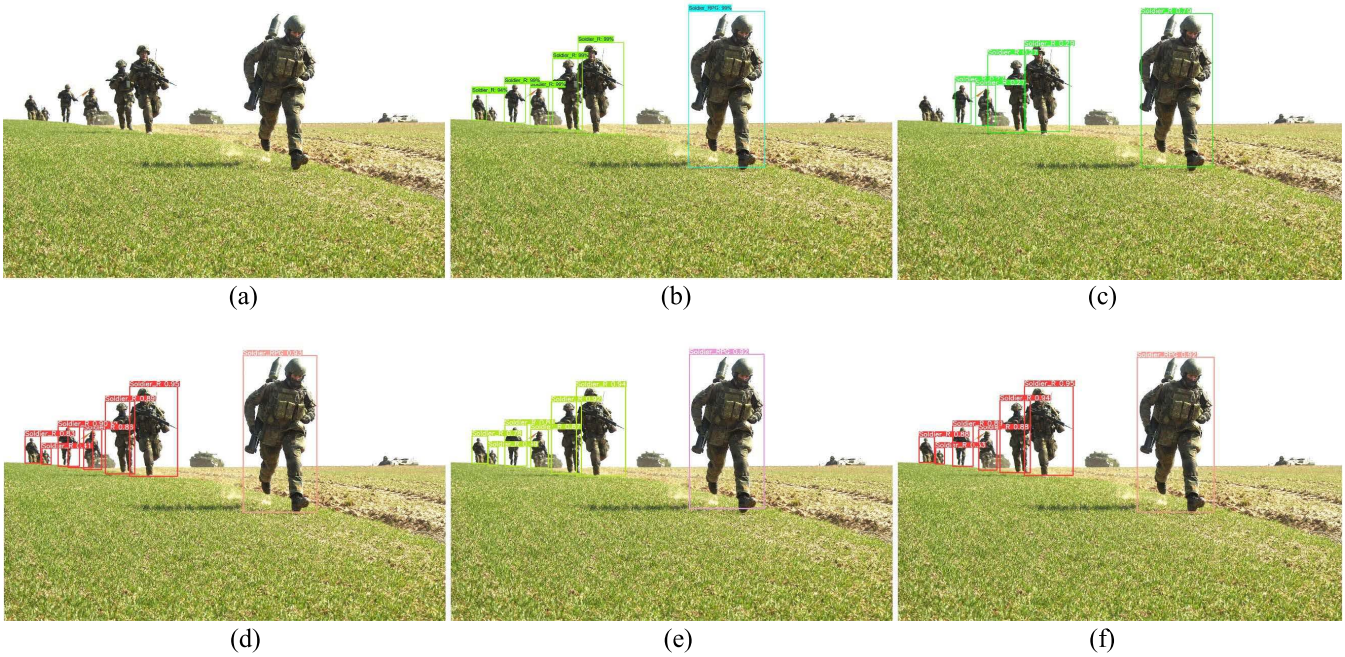


FIGURE 14. Comparison of detection performance of different models in scene 2. (a) the original image; (b) Faster R-CNN; (c) SSD; (d) YOLOv3; (e) YOLOv5s; (f) YOLO-G.

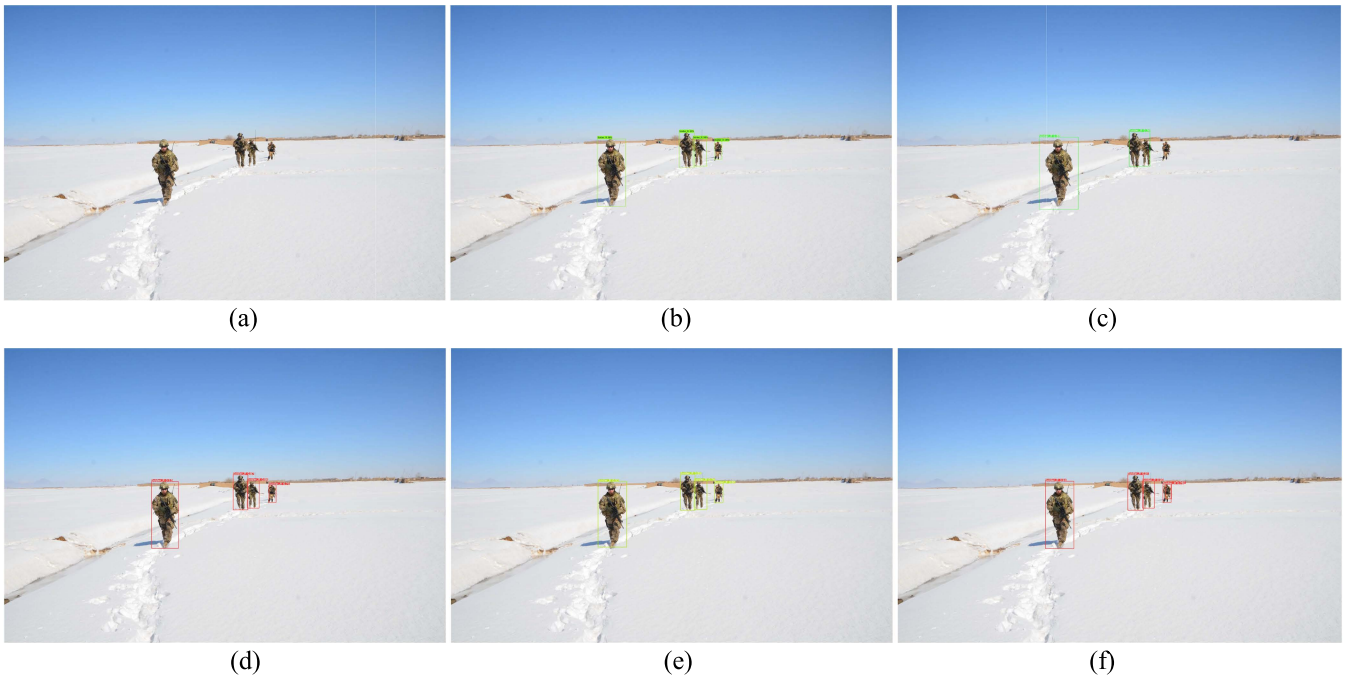


FIGURE 15. Comparison of detection performance of different models in scene 3. (a) the original image; (b) Faster R-CNN; (c) SSD; (d) YOLOv3; (e) YOLOv5s; (f) YOLO-G.

results of YOLOv3 and YOLOv5s are similar, and they can correctly identify the two types of targets. YOLO-G is the best among several models, which can not only effectively identify two kinds of targets, but also has high detection confidence.

Therefore, through the above qualitative and quantitative results, we can conclude that the detection performance of our proposed method is the best. Compared with other methods, our method improves the detection accuracy, and the detection speed can meet the requirements of normal use.

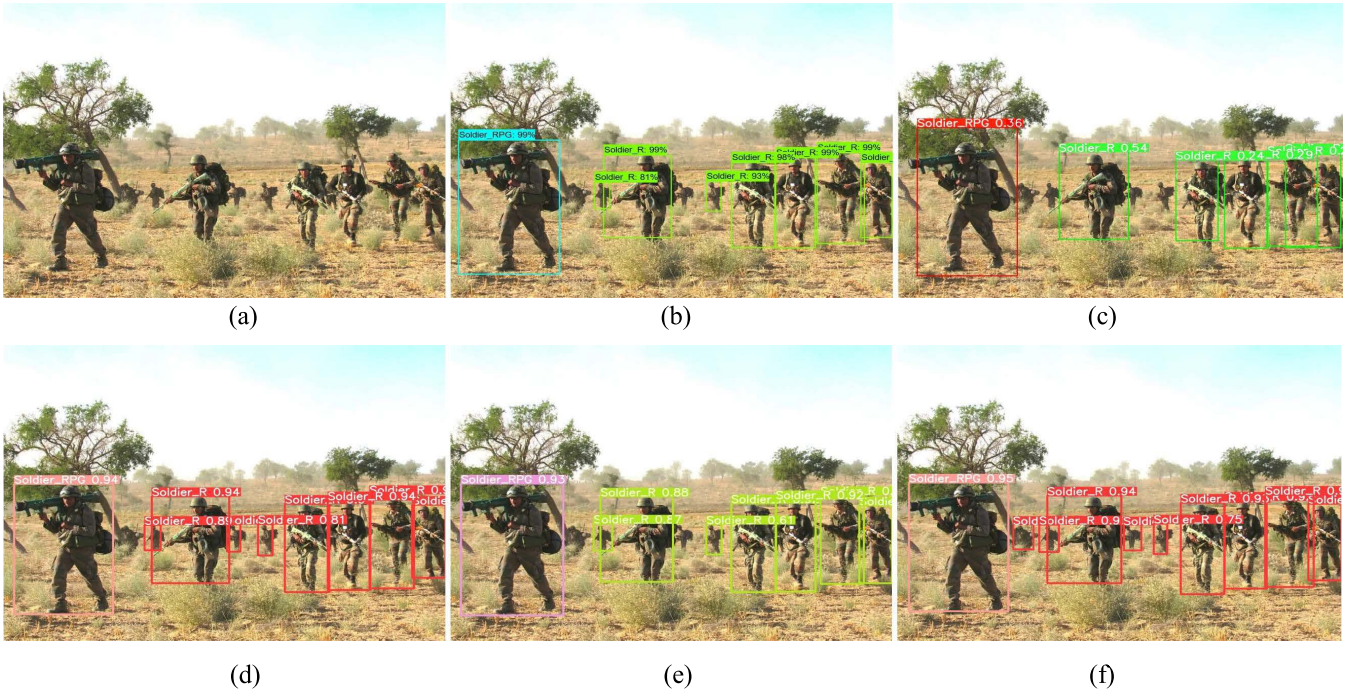


FIGURE 16. Comparison of detection performance of different models in scene 4. (a) the original image; (b) Faster R-CNN; (c) SSD; (d) YOLOv3; (e) YOLOv5s; (f) YOLO-G.

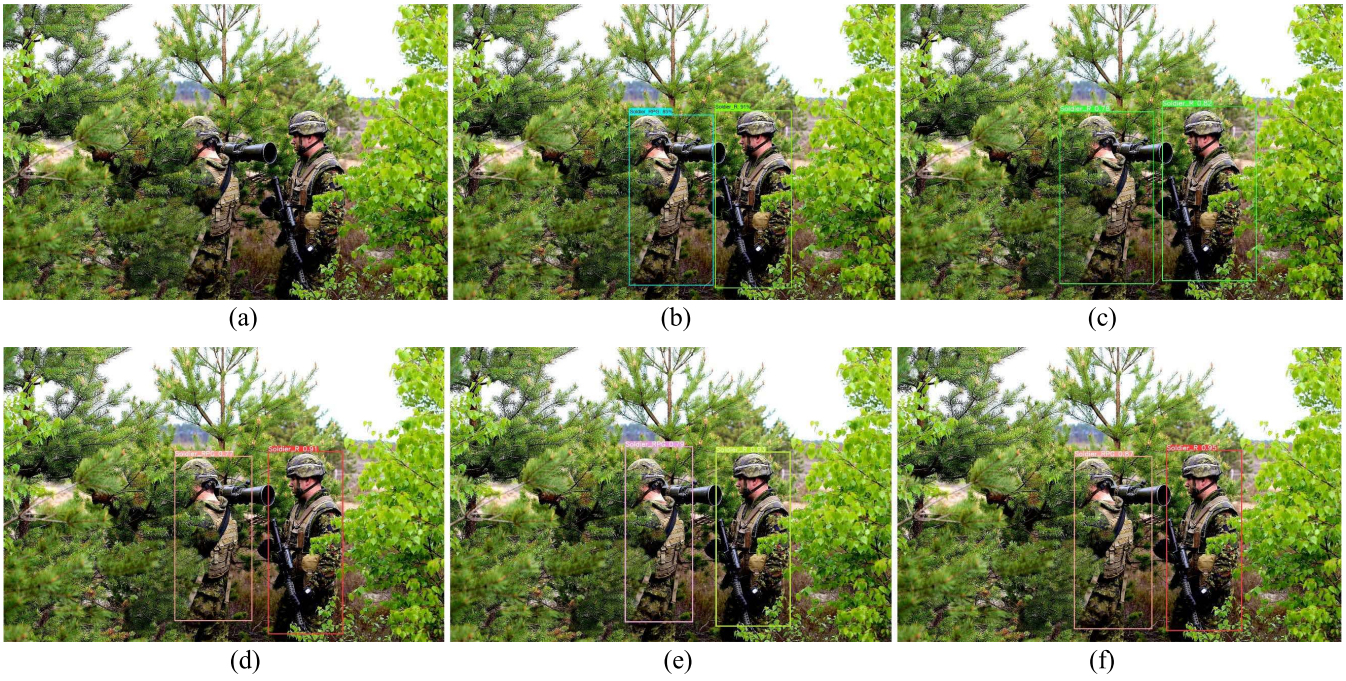


FIGURE 17. Comparison of detection performance of different models in scene 5. (a) the original image; (b) Faster R-CNN; (c) SSD; (d) YOLOv3; (e) YOLOv5s; (f) YOLO-G.

VI. DISCUSSION

In order to effectively evaluate the method proposed in this paper, we have compared several indexes. We completed the ablation experiment and the comparison experiment among several algorithms by using the self-built dataset. According to the experimental results, YOLO-G greatly improves the performance of the detection model by improving the feature extraction network, attention mechanism and loss function. The P-R curves generated by the experiment further verify the correctness of the index values we obtained. They also

show that YOLO-G model has better detection accuracy and speed when detecting military targets, and can be used for military target detection in actual battlefield environment.

In the ablation experiment, the detected mAP increased by 0.9%, 1% and 1% respectively, that is, the detection accuracy of YOLO-G has improved by about 3% compared with YOLOv3. In terms of detection speed, YOLO-G increases by 25.9 compared with YOLOv3. The substantial improvement of FPS makes YOLO-G keep high-precision detection of military targets, and at the same time, it can greatly improve the detection speed. This will help to find the targets faster and more accurately, judge the corresponding threat degree according to the weapons equipped with the targets, and lay the foundation for the next step of making fire distribution plan.

In the comparative experiment of several algorithms, YOLO-G is ahead of the other four methods in the detection accuracy index, and the detection speed is second only to the latest one-stage detector YOLOv5s. Compared with Faster R-CNN, SSD, YOLOv3 and YOLOv5s, the mAP values of targets detection have increased by 7.6%, 10.9%, 2.9% and 1.2% respectively. On the whole, YOLO-G still has the best performance.

Although our method has many excellent performances, it is not without defects in the process of model training. Because the research background of this paper is to detect armed men equipped with different weapons, this dataset has no publicly available resources. After the targeted augmentation of the dataset and many experiments, we got the expected results. Secondly, aiming at the dataset we made, we debugged the parameters of YOLO-G detection algorithm for many times to get the appropriate values.

VII. CONCLUSION AND FUTURE WORKS

In view of the task of military target detection in complex scenes, we built a military target dataset with multiple scenes, and proposed an improved target detection method based on YOLOv3. In this method, GhostNet is used as the feature extraction network. The use of lightweight feature extraction network reduces the parameters and computation of the model, thus improving the detection performance. By introducing the coordinate attention mechanism, the ability of our model to extract useful features is improved, and the detection performance of the detection model is improved. Based on DIOU and Focal loss function, the loss function of YOLOv3 network is modified. The use of DIOU-NMS further improves the detection accuracy of military targets in complex environment. According to the ablation experiment and the comparison experiment among algorithms, YOLO-G has excellent detection performance for military targets such as armed men equipped with weapons, and can meet the detection requirements in complex battlefield scenes.

There are only two kinds of detection targets in this paper, which is determined by our subject background. In the future, we will add more categories of military targets, such as tanks, helicopters, armored vehicles. The detection performance of our proposed model is further verified by improving the dataset. Moreover, we will improve YOLOv4 and YOLOv5 by using the optimization method in this paper, and verify the detection effect of our improved method in our self-built dataset. At the same time, we will further optimize the proposed detection model and deploy it to the embedded platform to verify the applicability of the detection model in the equipment with limited hardware resources. Finally, we will test our method specifically for small targets to verify the detection performance of the model.

REFERENCES

[1] A. Kline, D. Ahner, and R. Hill, "The weapon-target assignment problem," *Comput. Oper. Res.*, vol. 105, pp. 226–336, May 2019.

[2] H. Peng, Y. Zhang, S. Yang, and B. Song, "Battlefield image situational awareness application based on deep learning," *IEEE Intell. Syst.*, vol. 35, no. 1, pp. 36–43, Jan. 2020.

[3] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa, "A system for video surveillance and monitoring," Robotics Inst., Carnegie-Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-RI-TR-00-12, 2000.

[4] A. Miller, "Situational awareness—From the battlefield to the corporation," *Comput. Fraud Secur.*, vol. 2006, no. 9, pp. 13–16, Sep. 2006.

[5] A. K. Jain, N. K. Ratha, and S. Lakshmanan, "Object detection using Gabor filters," *Pattern Recognit.*, vol. 30, no. 2, pp. 295–309, 1997.

[6] B. N. Nelson, "Automatic vehicle detection in infrared imagery using a fuzzy inference-based classification system," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 1, pp. 53–61, Feb. 2001.

[7] S.-G. Sun, "Automatic target detection using binary template matching," *Opt. Eng.*, vol. 44, no. 3, Mar. 2005, Art. no. 036401.

[8] V.-E. Neagoe, S.-V. Carata, and A.-D. Ciotec, "An advanced neural network-based approach for military ground vehicle recognition in SAR aerial imagery," *Int. Sci. Committee*, vol. 18, p. 41, Jan. 2016.

[9] W. Budiharto, V. Andreas, J. S. Suroso, A. A. S. Gunawan, and E. Irwansyah, "Development of tank-based military robot and object tracker," in *Proc. 4th Asia–Pacific Conf. Intell. Robot Syst. (ACIRS)*, Jul. 2019, pp. 221–224.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2013, pp. 1097–1105.

[11] T. P. Hanratty, R. J. Hammell II, B. A. Bodt, E. G. Heilman, and J. C. Dumer, "Enhancing battlefield situational awareness through fuzzy-based value of information," in *Proc. 46th Hawaii Int. Conf. Syst. Sci.*, Jan. 2013, pp. 1530–1605.

[12] S. Liu and Z. Liu, "Multi-channel CNN-based object detection for enhanced situation awareness," 2017, *arXiv:1712.00075*.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[14] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2001, pp. 511–518.

[15] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 346–361.

[17] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards realtime object detection with region proposal networks," in *Proc. NIPS*, 2015, pp. 91–99.

[19] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. NIPS*, 2016, pp. 379–387.

[20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single shot MultiBox detector," in *Proc. ECCV*, 2016, pp. 21–37.

[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
[22] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6517–6525.
[23] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[24] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[25] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-YOLOv4: Scaling cross stage partial network," 2020, *arXiv:2011.08036*.
[26] C. Guo, B. Fan, Q. Zhang, S. Xiang, and C. Pan, "AugFPN: Improving multi-scale feature learning for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12595–12604.
[27] F. Wong and H. Hu, "Adaptive learning feature pyramid for object detection," *IET Comput. Vis.*, vol. 13, no. 8, pp. 742–748, Dec. 2019.
[28] Y. Zeng, C. Ritz, J. Zhao, and J. Lan, "Attention-based residual network with scattering transform features for hyperspectral unmixing with limited training samples," *Remote Sens.*, vol. 12, no. 3, p. 400, Jan. 2020.
[29] J. Li, J. Gu, Z. Huang, and J. Wen, "Application research of improved YOLO v3 algorithm in PCB electronic component detection," *Appl. Sci.*, vol. 9, pp. 3738–3750, Jan. 2019.
[30] J. Moran, L. Haibo, Z. Wang, H. Bin, and C. Zheng, "The application of improved YOLO V3 in multi-scale target detection," *Appl. Sci.*, vol. 9, no. 18, pp. 3775–3788, Sep. 2019.
[31] G. Liu, J. C. Nouaze, P. L. T. Mbouembe, and J. H. Kim, "YOLO-tomato: A robust algorithm for tomato detection based on YOLOV3," *Sensors*, vol. 20, no. 7, pp. 1–20, Apr. 2020.
[32] M. Liu, X. Wang, A. Zhou, X. Fu, Y. Ma, and C. Piao, "UAV-YOLO: Small object detection on unmanned aerial vehicle perspective," *Sensors*, vol. 20, no. 8, p. 2238, Apr. 2020.
[33] W. He, Z. Huang, Z. Wei, C. Li, and B. Guo, "TF-YOLO: An improved incremental network for real-time object detection," *Appl. Sci.*, vol. 9, no. 16, pp. 3225–3240, Aug. 2019.
[34] H. Zhu, Y. Miao, and X. Zhang, "Semantic image segmentation with improved position attention and feature fusion," *Neural Process. Lett.*, vol. 52, pp. 329–351, May 2020.
[35] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1580–1589.
[36] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," 2021, *arXiv:2103.02907*.
[37] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 12993–13000.
[38] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.
[39] X. Song, S. Gao, and C. Chen, "A multispectral feature fusion network for robust pedestrian detection," *Alexandria Eng. J.*, vol. 60, no. 1, pp. 73–85, Feb. 2021.
[40] J. Ma, H. Wan, J. Wang, H. Xia, and C. Bai, "An improved one-stage pedestrian detection method based on multi-scale attention feature extraction," *J. Real-Time Image Process.*, vol. 18, no. 4, pp. 1–14, 2021.
[41] X. Chen, Y. Zhang, L. Lin, J. Wang, and J. Ni, "Efficient anti-glare ceramic decals defect detection by incorporating homomorphic filtering," *Comput. Syst. Sci. Eng.*, vol. 36, no. 3, pp. 551–564, 2021.
[42] L. Xie, X. Xiang, H. Xu, L. Wang, L. Lin, and G. Yin, "FFCNN: A deep neural network for surface defect detection of magnetic tile," *IEEE Trans. Ind. Electron.*, vol. 68, no. 4, pp. 3506–3516, Apr. 2021.
[43] Z. Sun, M. Dai, X. Leng, Y. Lei, B. Xiong, K. Ji, and G. Kuang, "An anchor-free detection method for ship targets in high-resolution SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 7799–7816, 2021.
[44] Z. Sun, X. Leng, Y. Lei, B. Xiong, K. Ji, and G. Kuang, "BiFA-YOLO: A novel YOLO-based method for arbitrary-oriented ship detection in high-resolution SAR images," *Remote Sens.*, vol. 13, no. 4209, pp. 1–28, Oct. 2021.
[45] X. Ni, G. Dong, L. Li, Q. Yang, and Z. Wu, "Kinetic study of electron transport behaviors used for ion sensing technology in air/EGR diluted methane flames," *Fuel*, vol. 288, Mar. 2021, Art. no. 119825.
[46] H. I. H. Alsaadi, R. M. ALmuttari, O. N. Ucan, and O. Bayat, "An adapting soft computing model for intrusion detection system," *Comput. Intell.*, vol. 2021, pp. 1–21, Jan. 2021.
[47] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR05)*, Sep. 2005, pp. 886–893.
[48] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[49] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 6848–6856.
[50] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[51] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, German, Sep. 2018, pp. 8–14.

LINGREN KONG received the B.E. and M.E. degrees from the China University of Mining and Technology, Beijing, China. He is currently pursuing the Ph.D. degree in objective detection and weapon-target assignment with the Beijing Institute of Technology, Beijing.

His current research interests include multiobjecive optimization, computer vision, and machine learning.

JIANZHONG WANG received the B.E., M.E., and Ph.D. degrees from the Nanjing University of Science and Technology, Nanjing, China.

From 1990 to 2002, he was with the Wuhan University of Technology, Wuhan, China, where he is currently a Professor of mechanical and electrical engineering. Since 2002, he has been working with the Beijing Institute of Technology, Beijing, China, where he is a Professor with the School of Mechatronical Engineering and the State Key Laboratory of Explosion Science and Technology. His current research interests include intelligent systems, unmanned ground vehicle (UGV), and multi robot cooperative technology.

PENG ZHAO received the Ph.D. degree from the Beijing Institute of Technology, Beijing, China, in 2020.

He is currently working with the China North Vehicle Research Institute, Beijing. His current research interests include multiobjecive optimization, computer vision, and machine learning.