IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 7, 2020, accepted December 17, 2020, date of publication December 22, 2020,
date of current version December 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3046515

legend: Problem, Imp info, Solution

YOLO-ACN: Focusing on Small Target and Occluded Object Detection

YONGJUN LI, SHASHA LI, HAOHAO DU, LIJIA CHEN, DONGMING ZHANG, AND YAO LI

School of Physics and Electronics, Henan University, Kaifeng 475004, China

Corresponding author: Shasha Li (lss1996@henu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant U1704130, and in part by the Key Research and Development and Promotion Projects in Henan Province under Grant 212102210151.

ABSTRACT To further improve the speed and accuracy of object detection, especially small targets and occluded objects, a novel and efficient detector named YOLO-ACN is presented. The detector model is inspired by the high detection accuracy and speed of YOLOv3, and it is improved by the addition of an attention mechanism, a CIoU (complete intersection over union) loss function, Soft-NMS (non-maximum suppression), and depthwise separable convolution. First, the attention mechanism is introduced in the channel and spatial dimensions in each residual block to focus on small targets. Second, CIoU loss is adopted to achieve accurate bounding box (BBox) regression. Besides, to filter out a more accurate BBox and avoid deleting occluded objects in dense images, the CIoU is applied in the Soft-NMS, and the Gaussian model in the Soft-NMS is employed to suppress the surrounding BBox. Third, to significantly reduce the parameters and improve the detection speed, standard convolution is replaced by depthwise separable convolution, and hard-swish activation function is utilized in deeper layers. On the MS COCO dataset and infrared pedestrian dataset KAIST, the quantitative experimental results show that compared with other state-of-the-art models, the proposed YOLO-ACN has high accuracy and speed in detecting small targets and occluded objects. YOLO-ACN reaches a mAP50 (mean average precision) of 53.8% and an APs (average precision for small objects) of 18.2% at a real-time speed of 22 ms on the MS COCO dataset, and the mAP for a single class on the KAIST dataset even reaches over 80% on an NVIDIA Tesla K40.

INDEX TERMS CIoU loss, soft-NMS, attention mechanism, YOLOv3, object detection.

I. INTRODUCTION

Object detection utilizes computers and related algorithms to find objects of certain target classes with precise localization [1]. Real-time and accurate object detection can provide good conditions for object tracking, behavior recognition, scene understanding, and medical detection. In recent years, significant improvements have been made in object detection by using traditional and deep learning methodologies. However, few studies have focused on detecting small targets and occluded objects. The detection accuracy and speed still need to be further improved [2].

Small targets and occluded objects have a few effective pixels, carry only several and incomplete features and are largely submerged in noise and background clutter. After multiple downsample and pooling operations, considerable feature information will be lost. Therefore, the detection of small targets and occluded objects faces significant challenges [3], e.g., long-distance pedestrians and traffic signs are very small or even obstructed. However, the precision and rapid detection of these small targets and occluded objects is a prerequisite for ensuring the safety of unmanned driving. The analysis of remote sensing images requires precise identification of object classes, including vehicles, ships, and buildings. However, the objects are very small or occluded by vegetation. In infrared images, the objects are not only very small but also occluded by strong noise and background, which makes the object features less obvious. Therefore, the detection problem of small targets and occluded objects has become an urgent problem to be solved in the civilian and military fields.

Traditional object detection algorithms are based on sliding windows to select candidate boxes, using Viola-Jones [4], HOG (histogram of gradient) [5] and DPM (deformable part

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

model) [6] to extract features, and using SVM (support vector machine) [7] classifier to classify the features. The traditional algorithms need to design different feature descriptors for detecting different objects. Therefore, they have poor robustness and weak generalization ability. These algorithms [5]–[10] require a large amount of calculation to generate proposals, which leads to low detection precision and slow detection speed [11].

With the emergence of deep learning, breakthrough progress has been made in object detection, in terms of feature expression capability and time efficiency. Current state-of-the-art object detection algorithms are mainly divided into two categories: two-stage and one-stage detectors. The two-stage detectors are represented by R-CNN [12]–[14], and the one-stage detectors are represented by SSD (single shot multibox detector) [15], [16] and YOLO [17]–[19]. These milestone algorithms have achieved good detection results on large datasets, e.g., PASCAL VOC [20], [21] and MS COCO [22], [23]. As a representative one-stage object detection algorithm, YOLOv3 has been widely adopted because of the high speed and accuracy [24], and it directly uses a more powerful network to extract the features and generate regression BBox of the objects. Thus, the computational cost reduces and the design are relatively simple. However, when the background of object detection is complexity and the objects scale and attitude is diversity, YOLOv3 cannot detect small targets and occluded objects well, e.g., false detection, missed detection, and repeated detection [25]. Recently, YOLOv4 [26] algorithm has received widespread attention, which applies a great number of data enhancement techniques. It remains to be analyzed how much data enhancement technology affects the results of detecting small targets and occluded objects.

To focus on small targets and occluded objects, a detection algorithm YOLO-ACN (attention, CIoU loss, and Soft-NMS) based on YOLOv3 is proposed in this article. First, in the network design process, the attention mechanism is introduced in the channel and spatial dimensions in each residual block. Specifically, the efficient channel attention module is utilized to realize the cross-channel interaction without dimensionality reduction. The spatial attention module is used to obtain the complementary feature information. The attention mechanism enables the network to pay more attention to the small targets and occluded objects in an efficient way. In addition, depthwise separable convolution [27], [28] is adopted instead of standard convolution, and leaky ReLU [29] is replaced with hard-swish [30] to reduce the parameters. Then, in the model training process, the degree of overlap, the center point distance, and the aspect ratio of the anchors between the ground truth BBox and predicted BBox are considered as the BBox regression CIoU loss [31], [32]. CIoU loss has faster and more accurate regression during the training process, and it also makes the detection algorithm friendlier to small targets. Finally, when predicting the results, the Gaussian model is employed to suppress the non-maximum value. Combining the CIoU with the Soft-NMS [33] to filter out BBox, deletions of occluded objects are avoided in dense images to some extent.

The contributions of this article are summarized as follows:
1. A novel one-stage object detection algorithm YOLO-ACN is proposed to focus on small targets and occluded objects. The algorithm contains depthwise separable convolution, spatial and channel attention mechanisms, and hard-swish activation function, leading to notable gains of average precision (AP), average recall (AR), and speed.
2. Based on the Darknet in YOLOv3, a lightweight feature extraction network is designed. In the feature extraction network, to significantly reduce the parameters and improve the detection speed, standard convolution is replaced by depthwise separable convolution, the nonlinear operations of batch normalization layer and activation layer are replaced by convolution, and the hard-swish activation function is utilized in deeper layers. At the same time, the attention mechanism is introduced in the channel and spatial dimensions in each residual block of the feature extraction network to focus on small targets.
3. To achieve accurate bounding box (BBox) regression, the GIoU loss of the YOLOv3 is replaced by CIoU loss in the proposed detection model. Then, to filter out a more accurate BBox and avoid deleting occluded objects in dense images, the CIoU is applied in the Soft-NMS, and the Gaussian model in the Soft-NMS is employed to suppress the surrounding BBox.

II. RELATED WORK

In recent years, the object detection algorithm for an optimal trade-off between precision and speed has been a popular research topic [34]. Both the two-stage and one-stage detectors have made great contributions to the improvement of the efficient network and better methodology.

In 2014, R. Girshick *et al.* presented a pioneering two-stage object detector R-CNN [12], which divided object detection into two stages: generate proposals and predict categories. Compared with the traditional techniques, R-CNN significantly improved the performance. However, the computation was not shared, leading to heavily duplicated computation. Therefore, Fast R-CNN [13] was developed, and it reduced the repeated calculation by mapping the relationship between the images and the feature extracted layers. Based on Fast R-CNN, a novel method named Faster R-CNN [14] was designed. In this method, a generator RPN (region proposal network) was used to generate the proposals, and the anchor was introduced to cope with the different sizes of objects, such that detection accuracy and speed were significantly improved.

To improve the performance of detecting small targets, feature pyramid networks (FPNs) to predict in each layer was constructed [35]. Then, in 2018, B. Singh *et al.* proposed scale normalization for image pyramids (SNIP) [36], a training detector based on image pyramids, and solved the problem of extreme changes in the size of the detection dataset. Although it could improve the effectiveness of the model, the increase

of computation was still obvious. To address the problem that detection performance tends to degrade with increasing the IoU thresholds, the multi-stage object detection architecture, Cascade R-CNN [37], is proposed. Although the detection precision was improved, the network became larger than others. K. He *et al.* presented Mask R-CNN [38], adding a parallel branch for predicting the object mask to complete the task of instance segmentation, and this study also introduced ROI (region of interest) Align, using the bilinear difference method, so that the precision of the mask was improved. Mask R-CNN could achieve instance segmentation, but the segmentation cost was high. These two-stage object detection algorithms have higher accuracy, but the time complexity is high. Thus, it is difficult to apply to real-time detection systems. YOLO:

To improve the speed of object detection, J. Redmon *et al.* developed a real-time detector YOLO [17] in 2016, which laid the foundation of the one-stage object detection. YOLO predicted multiple BBox positions and classes at once, and it regarded detection as a regression problem to truly achieve end-to-end detection. However, the detection accuracy of YOLO was low. Using a regression-based idea similar to YOLO and drawing on the method of anchoring in Faster R-CNN, W. Liu *et al.* introduced the SSD [15] algorithm, and it effectively solved the shortcomings of YOLO in the detection of small targets. Inspired by the anchor strategy used in SSD, YOLOv2 [18] was proposed by using k-means clustering to calculate the size of BBox, deleting the fully connected layer and the last pooling layer. YOLOv2 effectively balanced the detection speed and detection precision, which was better than SSD. However, YOLOv2 used the features obtained in the last convolution layer to detect objects, which lost much information. Thus, it was difficult to detect some small objects. Therefore, an improved YOLO version, YOLOv3 [19], was presented. It contained multiple residual blocks, which could reduce the problem of gradient disappearance. Unlike YOLOv2, YOLOv3 divided 9 anchors into 3 different scales. Moreover, it also used feature fusion and upsampled methods to detect more fine-grained features and improve the detection precision of small objects. However, the performance of YOLOv3 decreased with the increase of the intersection over union (IoU) [39], and it did not fit well with the ground truth BBox. In 2020, A. Bochkovskiy *et al.* used CSP-Darknet (cross stage partial) [26] as the backbone network in YOLOv4 to further improve the detection accuracy and speed, and added a SPP block to improve the size of the receptive field. The FPN was replaced by PANet (path aggregation network) for multichannel feature fusion. Although the detection precision was improved for small objects, the problem of detecting occluded objects was not considered. YOLOv4 was much larger than YOLOv3, which increased deployment costs and reduced training speed.

In general, since the one-stage object detection models do not require to generate proposals, the positioning proposals task is redefined as a regression task, where the structure is simple, the computational efficiency is high, and end-to-end training can be conveniently carried out, as regression task directly generates the category probability and position coordinate values of the objects. However, the lack of image preprocessing mechanism can easily lead to inaccurate extraction of proposal regions, and the high-level features fail to capture fine-grained descriptions of small targets and occluded objects. Attention blocks:

The lack of an image preprocessing mechanism for one-stage object detection leads to inaccurate extraction of candidate regions, which affects the detection of small targets and occluded objects. Recently, the attention mechanism has been demonstrated to offer great potential in improving the performance of object detection. Jaderberg *et al.* [40] proposed a spatial transformer to realize the spatial attention mechanism, and the spatial information in the images could be transformed accordingly to extract the key information. The channel attention mechanism was presented by Hu *et al.* [41], in which the importance among the channels was calculated through two fully connected layers to filter out the unimportant channel values. F. Wang *et al.* introduced a residual attention network [42] which was designed specifically for detection. The spatial and channel mechanisms were built by superposing residual attention modules. S. Woo *et al.* developed the convolutional block attention module (CBAM) [43] to multiply feature maps along the channel and spatial attention mechanisms. The CBAM also has a wide applicability to other networks. Due to its intuitiveness, versa…echanism has recei…bject detection and s…

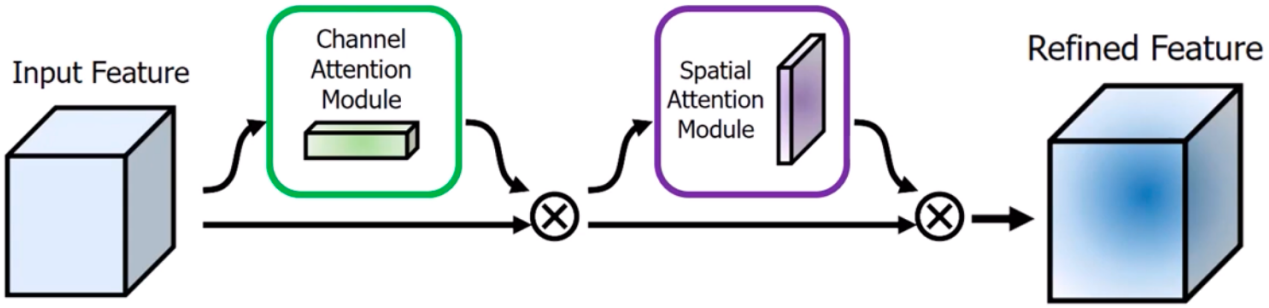
Wi…y of small targets …be beneficial, such …ages, infrared imag…include small target…s article proposes …with an attention …oft-NMS. This algor…ttention mechanism …xtract the key information by superimposing the attention perception features. To overcome the impact of the increasing IoU on the prediction boxes in YOLOv3, the overlap area, the center point distance, and the aspect ratio of the anchor between the ground truth BBox and predicted BBox are adopted in the CIoU loss. Therefore, the prediction boxes and the ground truth BBox are more consistent. The CIoU takes into account the diagonal distance and the center point distance of the smallest BBox, which is composed of two bounding boxes, which is also added to the threshold selection of Soft-NMS. Moreover, the Gaussian model is employed to suppress the surrounding BBox. In addition, in order to significantly reduce the number of weights and computational costs thus incurred so as to improve the speed of the model, the standard convolution is replaced by the depthwise separation convolution and hard-swish is used in the network.

Residual Attention Architecture:
- Combines ResNet & Attention
- Each residual block consists of:
  1. Spatial Attention
  2. Channel Attention

→ stacked multiple Residual Attention modules
↓ Used ECA & SA

Explanation of CBAM:

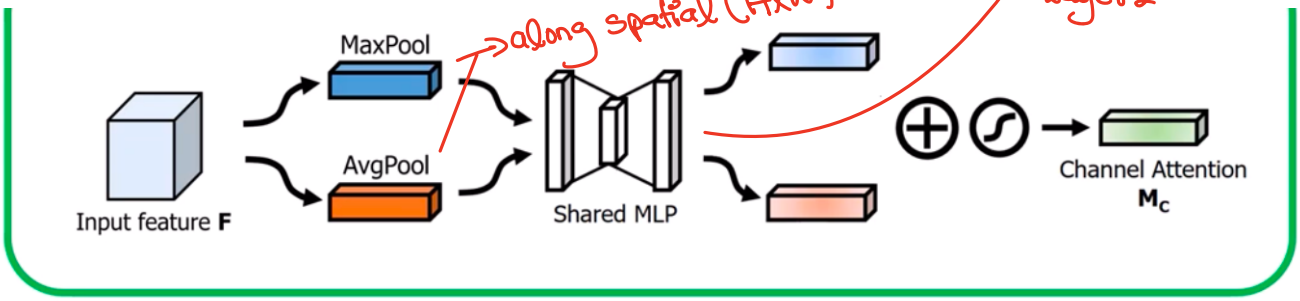
↳ Adds Channel attention and spatial attention modules



$$\mathbf{F}' = \mathbf{M_c}(\mathbf{F}) \otimes \mathbf{F},$$
$$\mathbf{F}'' = \mathbf{M_s}(\mathbf{F}') \otimes \mathbf{F}',$$

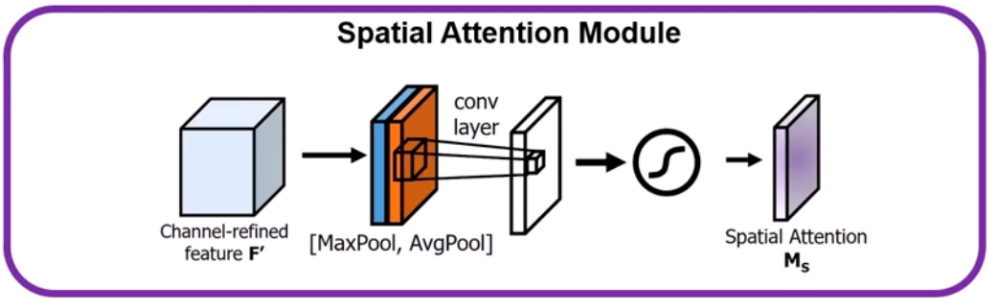
Efficient CA uses conv1D where each channel has its own filters.

Channel attention:



$$\mathbf{M_c}(\mathbf{F}) = \sigma(MLP(AvgPool(\mathbf{F})) + MLP(MaxPool(\mathbf{F})))$$

Spatial attention:



$$\mathbf{M_s}(\mathbf{F}) = \sigma(f^{7\times7}([AvgPool(\mathbf{F}); MaxPool(\mathbf{F})]))$$

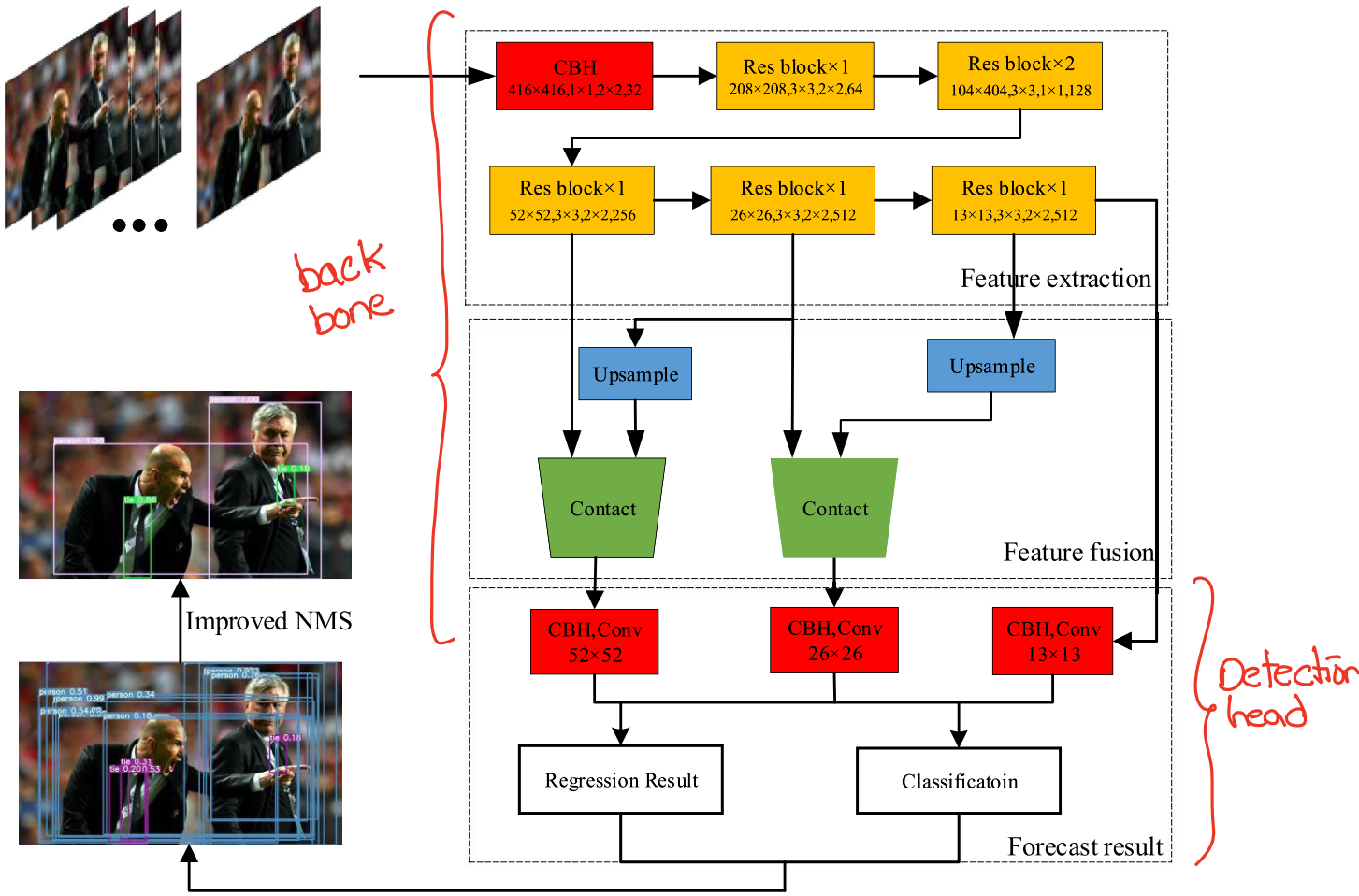


FIGURE 1. The proposed YOLO-ACN network. YOLO-ACN uses a lightweight feature extraction network with attention mechanism in each residual block to focus on small targets and occluded objects. Besides, in the CBH module, the hard-swish is applied as the activation function to decrease the calculation load. In the post-processing stage, the Soft-NMS (non-maximum suppression) is combined with CIoU to obtain accurate bounding box.

III. PROPOSED METHOD

A. YOLO-ACN NETWORK

The entire detection architecture proposed in this article is shown in Figure 1. The network mainly consists of three parts: feature extraction, feature fusion, and forecast result. First, the images are input through multiresidual blocks to extract features. The CBH represents the convolution, batch normalization [44], and hard-swish layer. Hard-swish can reduce the parameters and speed up the detection process. The attention mechanism is introduced in the residual blocks to extract the features and semantic information of small objects. Then, in the stage of feature fusion, feature maps with different sizes are obtained in the residual blocks, and the feature maps obtained by upsampling are concatenated to obtain feature maps with different sizes of receptive fields. After the concatenation layer, three feature maps of different sizes are obtained: 52×52, 26×26, and 13×13. Finally, the prediction results on feature maps are carried out to obtain the information of predicted BBox of different sizes, object categories, and confidence. The improved Soft-NMS algorithm retains the predicted BBox of other objects and simultaneously removes the overlapped BBox with the same object to obtain the final prediction. In addition, the performance of loss function is improved to increase the convergence speed during model training.

In the extracting path of residual blocks with attention mechanism, each residual block has repeated convolutional blocks consisting of 1×1 conv, 3×3 depthwise separable conv, 1×1 conv. Within the residual block, the input feature map feeds into the sequence of operations mentioned above, which produce the output feature maps. After each residual block, the dimension of the input is reduced by half and the number of the feature maps is doubled. Then, the feature maps with sizes of 52×52, 26×26, and 13×13 can be obtained. The feature extraction network utilizes a lightweight convolutional neural network which has fewer network parameters and better real-time performance than Darknet and ResNet. In the feature fusion, the feature maps obtained by downsampling and upsampling are combined through the concatenate operation to obtain the feature maps of different sizes. Finally, a 1×1 convolutional layer is adopted to predict three different sizes of feature maps, and the final prediction results are obtained by the improved Soft-NMS.

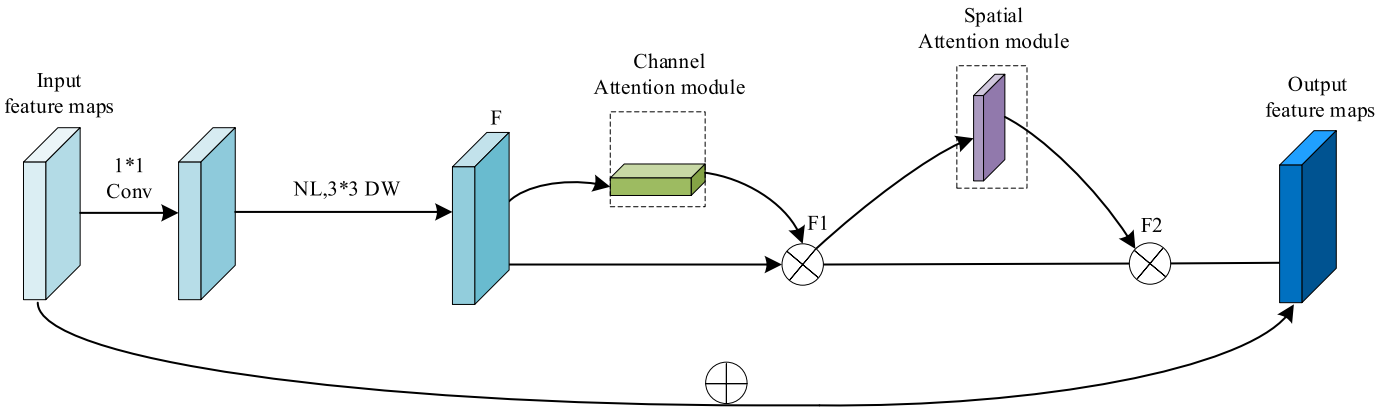


FIGURE 2. The overview of the residual block with the attention mechanism. The module has two sequential sub-modules: channel and spatial. The attention modules are added after the depthwise (DW) separable convolution, and the intermediate feature map is adaptively refined through every residual block of deep networks.

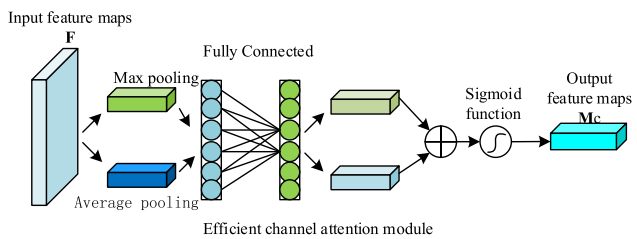


FIGURE 3. The efficient channel attention module. Given the aggregated features obtained by max pooling and average pooling with fully connected layer, the efficient channel attention generates channel weights by sigmoid function.

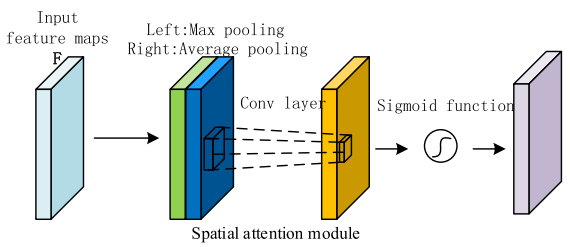


FIGURE 4. The effective spatial attention module. The input feature maps utilize the max pooling operation and the average pooling operation, then forward them to a convolution layer to aggregate the spatial information.

B. RESIDUAL BLOCK WITH THE ATTENTION MECHANISM

The attention mechanism can make the neural network focus on the shallow layers feature maps, and allocate computing resources to more important features. Unlike YOLOv3, the attention mechanism is introduced in the residual blocks in the feature extraction stage. In [40]–[42], different attention mechanisms are used to extract the features. In the residual blocks, the channel and the spatial attention mechanisms are combined to introduce the attention mechanism in two dimensions to guide where the network should pay attention, and then a higher weight is assigned to improve the ability of expressing small objects. The structure of the residual block extended by developing the attention mechanism is shown in Figure 2.

In Figure 2, the input feature maps go through a convolution layer with a kernel size of 1×1 and a convolution layer with a kernel size of 3×3 to obtain the feature maps F. First, the channel attention mechanism is introduced on the feature maps F. The channels relationship among the features is employed to generate the channel attention feature maps, and a weighted operation is performed on the feature maps F to obtain the channel feature maps $F1$. Then, the relationship among the spatial features is used to complement the feature information and obtain the spatial feature maps $F2$. Finally, a weighted operation is performed on the input feature maps and spatial feature maps $F2$ to obtain the output feature maps. The efficient channel attention and spatial attention modules are shown in Figure 3 and 4 respectively.

In Figure 3, to effectively calculate the attention feature maps among the channels, the channel information of the feature maps are obtained by the maximum pooling operation $MaxPool(F)$ and the average pooling operation $AvgPool(F)$ for the input feature maps F, $F \in R^{C\times H\times W}$. C represents the number of channels, and H and W represent the size of the feature map. Then, the two obtained feature maps are put into the fully connected layer to generate the channel weights and capture the nonlinear cross-channel interaction information without dimensionality reduction. In the improved channel module based on [43] and [45], the two fully connected layers do not perform dimension reduction by ratio r but keep the dimensions to better capture all inter-channel measurement dependencies. The calculation of the channel attention module can be given by Eq.:

$$\begin{aligned} M_C(F) &= \delta\left(C1D_k\left(AvgPool(F) + MaxPool(F)\right)\right) \\ &= \delta\left(C1D_k\left(F^c_{avg} + F^c_{max}\right)\right) \end{aligned} \tag{1}$$

where δ is the activation function, $C1D_k$ represents the 1-dimensional convolution, k represents adjacent channels of F, and k is set to 5, which means in fully connected layers, the second Fully Connected (FC) layer can perceive 5 channels in the first FC layer. F^c_{avg} and F^c_{max} represent the feature maps obtained after the average pooling and max pooling operation in the channel attention module respectively, and c means the channel dimension.

Spatial Attention module:

In Figure 4, to calculate the attention feature maps between spatial $AvgPool(F)$ and $MaxPool(F)$ that are adopted to aggregate the channel information for the input feature maps, two features of 2D $F^s_{avg} \in R^{1\times H\times W}$ and $F^s_{\max} \in R^{1\times H\times W}$ are generated respectively. After the convolution layer with a 7×7 convolution kernel is used to generate the spatial attention feature maps, the places that need to be emphasized or suppressed are encoded. The specific calculation of the spatial attention module can be expressed as Eq.:

$$\begin{aligned} M_s(F) &= \delta\left(f^{7\times7}\left(AvgPool(F); MaxPool(F)\right)\right) \\ &= \delta\left(f^{7\times7}\left(F^s_{avg}; F^s_{max}\right)\right) \end{aligned} \tag{2}$$

where δ is the activation function and $f^{7\times7}$ represents the convolution operation with a kernel size of 7×7. F^s_{avg} and F^s_{max} represent the feature maps obtained after the average pooling and max pooling operation in the spatial attention module respectively, and s represents the spatial dimension.

In addition, the standard convolution operation is applied in the residual blocks in YOLOv3, whereas in YOLO-ACN the depthwise separable convolution is adopted to separate one kernel into two. The depthwise separable convolution operation can map the correlation between spatial and channel dimensions to obtain a multichannel dimension, which can significantly reduce the computation cost of the convolution layer and improve the operation speed of the convolution layer. In the previous attention module, a 3×3 or 5×5 convolution kernel is adopted, but in the improved attention module, the 3×3 or 5×5 convolution is separated as 1×1 and 3×3 or 1×1 and 5×5 to realize the different channels using the different convolution kernels. With the attention mechanism and the depthwise separable convolution introduced in the residual blocks, the network can enhance the ability of feature expression in a specific region without increasing the computation. Thus, the performance of the object detection process is further improved.

C. HARD SWISH ACTIVATION FUNCTION

The leaky ReLU is applied as the activation function in YOLOv3 and [7], [24]–[26]. However, it is a monotonic and linear function, and its difference is zero. Since the leaky ReLU cannot maintain a negative value, most neurons are not updated. To avoid this problem, Google Brain performs swish [46] to directly replace the leaky ReLU. It has been experimentally demonstrated that swish works better than leaky ReLU, and does not need to modify the network architecture and the initialization. The swish function can be defined as Eq.:

$$swish = x \cdot \sigma(\beta x) \tag{3}$$

where $\sigma(x)$ represents the sigmoid activation function and β can be either a constant or a trainable hyperparameter. However, the largest problem of the swish is that it is computationally intensive.

The hard-swish is first adopted in MobileNetV3 [30], which is based on the swish. The hard-swish function is nonmonotonic and smooth. The nonmonotonic property helps to keep a small negative value, so that the gradient of the network is stabilized. The smooth function has a good generalization ability and effective optimization ability of the experiment results, which can improve the quality of the results. Compared with the swish function, the amount of calculation is relatively small. The hard-swish can be represented as Eq.:

$$h-swish = \frac{x\left(ReLU6\left(x+3\right)\right)}{6} \tag{4}$$

where ReLU6 imposes an upper limit of 6 on the basis of ReLU; then ReLU6 is shifted three units to the left and is finally divided by 6 to obtain a curve similar to the sigmoid function. This function is used to replace the sigmoid function in the swish function, then, the hard-swish function is obtained.

The hard-swish can make the boundary value harder, which can also vate the small targets and occlude e objects can be detected. In the p -swish is selected as the activati still kept unchanged as leaky R nly adopted in the deeper layers a ameters are kept and the activati ACN model is modified as the h on, the speed is faster by avoiding the exponential operation and the accuracy for small objects is higher.

> The hard swish has a property where it sharpens the activations near the decision boundary.
> For example, if an image region contains basely an object, you want the network to increase it's prediction from a small to bigger value

D. LOSS FUNCTIONN

In [5], [11]–[14], the IoU is applied to calculate the intersection ratio of the BBox. However, the IoU has nothing to do with the location of the objects, and it cannot reflect how the two objects overlap. To address the problem, YOLOv3 applies the GIoU [47] to replace the IoU.

There are three ways to overlap the BBox in Figure 5. The IoU values are same in three overlap ways, $IoU(a) = IoU(b) = IoU(c) = 0.33$, whereas GIoU values are different, Figure 5 explains how the object BBox may overlap. $GIoU(a) = 0.3$, $GIoU(b) = 0.24$, and $GIoU(c) = -0.1$. If the direction of alignment between the predicted BBox and the ground truth BBox is better, the GIoU value is higher. If the BBox A and B are not properly aligned, the area of BBox C will increase and the GIoU value will decrease. The expression of GIoU can be written as Eq.:

$$GIoU(A,B) = \frac{A\cap B}{A\cup B} - \frac{C-(A\cup B)}{C} \tag{5}$$

where C is the area of the largest rectangle contained by the two boxes, and A and B represent the areas of any two overlapping BBox.

YOLOv3 adopts GIoU loss as the regression of the BBox, which considers both the overlap area and the scale to make the detection model have a higher detection accuracy. However, when the predicted and ground truth BBox have a good

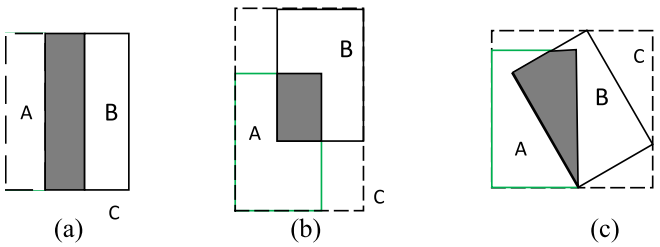


FIGURE 5. Three different ways of overlap between two bounding boxes with the exactly same IoU values, but different GIoU values.

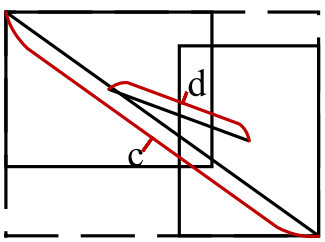


FIGURE 6. CIoU loss for bounding box regression, where c is the diagonal length of the smallest enclosing box covering two boxes, $d = \rho\left(b, b^{gt}\right)$ is the distance of central points of two boxes.

alignment direction, the GIoU also has the problem that the IoU will diverge during the calculation process. The CIoU [31], [32] loss considers the overlap area, the central point distance, and the aspect ratio of the BBox. Compared with GIoU, CIoU makes the prediction boxes converge more quickly. Therefore, this article uses the CIoU loss as the regression of the BBox. The formulation of CIoU loss can be given as Eq.:

$$Loss_{CIoU} = 1 - IoU + R_{DIoU} + \alpha v \tag{6}$$

$$R_{DIoU} = \frac{\rho^2\left(b, b^{gt}\right)}{c^2} \tag{7}$$

where IoU means the intersection over union of the BBox, R_{DIoU} represents the distance between the center points of the two bounding boxes b and b^{gt}, c represents the diagonal distance of the smallest rectangle formed by the two bounding boxes, α is a weight function, and v is used to measure the similarity of aspect ratios. The intuitive diagram is shown in Figure 6.

Based on the CIoU loss, the loss function in this work consists of regression BBox loss, confidence loss and class loss. Therefore, the total loss function can be computed as Eq.:

Loss function:

$$\begin{aligned} Loss &= Loss_{CIoU} + Loss_{obj} + Loss_{cls} \\ &= 1 - IoU + \frac{\rho^2\left(b, b^{gt}\right)}{c^2} + \alpha v \\ &\quad - \sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{ij}^{obj}\left[\hat{c}_i \log\left(c_i\right) + \left(1-\hat{c}_i\right)\log\left(1-c_i\right)\right] \\ &\quad - \gamma_{noobj}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{ij}^{noobj}\left[\hat{c}_i \log\left(c_i\right) + \left(1-\hat{c}_i\right)\log\left(1-c_i\right)\right] \\ &\quad - \sum_{i=0}^{S^2} I_{ij}^{obj}\sum_{c\in classes}\left[\hat{p}_i\left(c\right)\log\left(p_i\left(c\right)\right) \right. \\ &\quad \left. + \left(1-\hat{p}_i\left(c\right)\right)\log\left(1-p_i\left(c\right)\right)\right] \end{aligned} \tag{8}$$

1 if object is there, 0 otherwise

punishment for no obj when there is obj.

1, if obj is NOT there, 0, otherwise

punishment when prediction is obj when no object

punishment for wrong classification

where $Loss_{CIoU}$ is the improvement in the loss function and we have defined it in (6). The inclusion problem of the ground truth BBox and the predicted BBox is solved by calculating the Euclidean distance between the boxes, and the detection of occluded objects between the overlapping frames is more accurate. $Loss_{obj}$ is the confidence loss, which is represented by cross-entropy. Regardless of whether the anchor box contains the objects, the confidence loss will be calculated. Therefore, the confidence loss consists of two parts: I_{ij}^{obj} and I_{ij}^{noobj} represent whether the j-th box in the i-th grid contains objects or not. $Loss_{cls}$ means the loss of the object category, and the loss is also calculated by cross-entropy. When the j-th anchor box of the i-th grid is responsible for touching a real object, the resulting BBox will calculate the class loss. γ_{noobj} means that the confidence of no object in the grid is also weighted, and there will be a lower prediction confidence penalty. Similar to the loss function in YOLOv3, the value γ_{noobj} is still 0.5. The other loss function parameter values are shown in Table 1. By adding the loss functions to the constructed network, the convergence speed of the BBox during model training is effectively improved, and the accuracy of model detection is improved.

TABLE 1. The value of each parameter of the loss function.

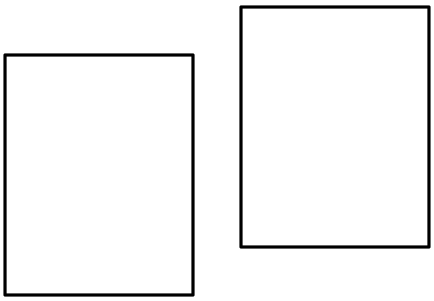
Parameter	Value
IoU threshold	0.15
cls loss gain	37.4
cls BCELoss positive weight	1.0
obj loss gain	64.3
obj BCELoss positive weight	1.0

E. IMPROVEMENT OF NON MAXIMUM SUPRESSIONE

In the prediction stage, NMS is widely adopted in [31]–[36] to solve the problem of multiple repeated prediction boxes around the object. The YOLOv3 also applies the IoU value as the main idea of NMS to choose the BBox. Based on the manually set threshold, the candidate boxes with the highest confidence are kept, but those boxes with the low confidence are deleted. However, the uncertainty of manually setting the threshold and deleting the low-confidence candidate box will ignore the occluded objects. The Soft-NMS [48] algorithm solves the shortcomings of the NMS. When multiple predicted bounding boxes appear around the detection object, the confidence of the predicted BBox is reduced to keep the occlusion objects with low confidence. The Gaussian penalty function is utilized to reduce the confidence. In addition, the center point distance and the aspect ratios are added to the NMS threshold selection to further improve the Soft-NMS algorithm, which has a better suppression effect on the predicted boxes and more focus on the occluded objects.

IoU as a Loss function:

↳ Normal loss funcs like L1 & L2 are NOT scale invariant.
↳ So, we use IoU but
  ↳ IoU is 0, when there is NO overlap



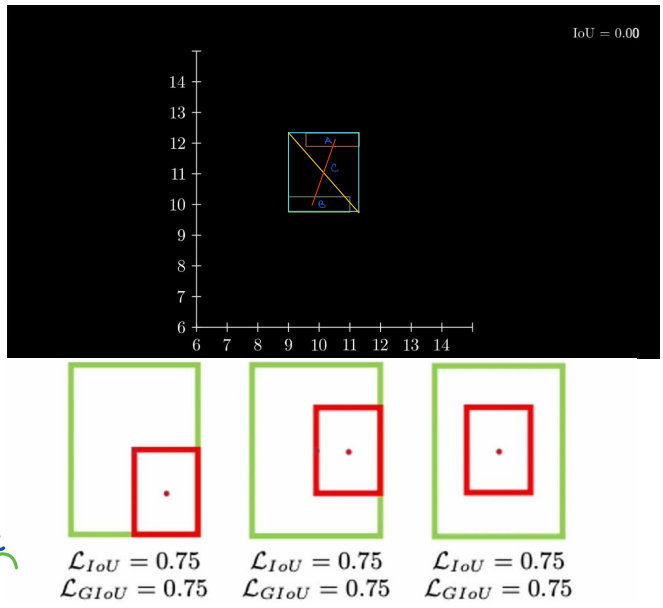
GIoU:

↳ Create the smallest box that encloses both predicted and gnd truth box.

$$GIoU = IoU - \frac{C - (A \cup B)}{C}$$

$$= IoU - \frac{C - [A + B - (A \cap B)]}{C}$$

$$Loss = 1 - GIoU$$



↳ Degrades to IoU when A & B overlap.

DIoU:

↳ Minimize the norm distance b/w centre point of the bboxes.

$$L_{DIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2}$$

$\rho^2(b, b^{gt})$ → Euclidean distance

c^2 → diagonal length of C.

↳ This doesn't take into account aspect ratio of the boxes.

CIoU:

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v,$$

α ↓ hyperparam

v → Aspect ratio

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2$$

Soft NMS:

The calculation can be described as the Eq.:

$$S_i = \begin{cases} S_i, & IoU - R_{CIoU}(M, b_i) > N_t \\ S_i \dfrac{1}{2\pi\sigma^2} e^{-\frac{CIoU(M,b_i)^2}{2\sigma^2}}, & IoU - R_{CIoU}(M, b_i) > N_t \end{cases} \quad (9)$$

↳ Take into account the aspect ratio & centre distance

where S_i represents the score of the current box, R_{CIoU} means CIoU loss, which considers the overlap degree and the distance of the center point, b_i represents the predicted BBox of each category, M is the BBox with the largest score, and N_t represents the threshold for screening the two overlapping boxes, which is set to 0.3. The greater the overlap degree of the predicted BBox b_i and the se, the stronger the suppression effect, and t ated confidence S_i. The smaller the overlap ame bi and the selected frame M is, the tion, and the greater the updated confidenc pre-dicted frames of other objects are ret cted frames overlapped by the same objec ner-ally speaking, this method not only effectively improves the detection accuracy of the model but also solves the occlusion problem.

1. Overlap Suppression:
↳ For b_i,
↳ if IoU with M is large, S_i is reduced
↳ if IoU with M is less, S_i is reduced less
2. Retaining Non-Overlapping bbox:
↳ Boxes with minimal overlap with M, are retained.

IV. EXPERIMENTAL ANALYSIS

The experiments using the MS COCO [22], [23], [49], KAIST [50], [51], and Campus Video Datasets demonstrate the YOLO-ACN's ability to improve detection accuracy and speed of small targets and occluded objects over conventional and related models. Training and deployment of models are performed using a server equipped with Intel XeonE5-26031.8GH CPU and NVIDIA Tesla K40 12GB GPU card with a 2880 CUDA parallel processing core. All models are trained on 2 GPU cards Cross Fire.

A. MS COCO DATASET

The MS COCO dataset [49] is the most widely applied public object detection dataset, which contains 80 categories and more than 330k images, including 200k labeled images and 500k labeled objects. Because the MS COCO dataset contains many categories and images, which is widely adopted in the object detection task, and because it also contains many small targets and occluded objects, the MS COCO dataset is chosen to train the model. The split method of the object size is the same as that of the detection evaluation of the COCO dataset. Specifically, when the area of an object is less than 322, it is considered as a small object, about 41%, when the area of the object is greater than 322 and less than 962, it is considered as a medium-sized object, about 43%, and 24% objects with an area greater than 962 are considered as large objects. Adam optimizer is adopted to train the proposed network. Generally, for the MS COCO dataset, the size of the images is randomly cropped to 416 × 416, and the IoU threshold is set to 0.15. Data augmentation is used to overcome the over-fitting by artificially increasing the training samples with class-preserving transformations, such that each image is rotated by 1.98 and the saturation is increased by 1.5%. The initial learning rate is 0.00579 and learning rate decay of 0.01 every 5 epochs with a mini-batch size of 16. A weight decay of 0.000484 and a momentum of 0.937 are used.

Figure 7 shows the change in the common performance evaluation indicators during the training process. The first three columns in Figure 7 are the BBox loss (measured by GIoU), confidence loss and class loss in the training dataset and validation dataset. Loss plays an important role in the training process, as it reflects the relation between the true value and the predicted value. The smaller the loss is, the closer the predicted value is to the true value and the better the performance of the model. The loss curves show that the three types of loss values gradually decrease and become stable with the increasing number of epochs, which mean that the proposed method produces better model parameters when optimizing the neural network. The last two columns are precision, recall, mean average precision (mAP), and $F1$. These four indicators can measure the performance of the model in the classification problem. The higher the value is, the higher the detection accuracy of the model. With the indicators stabilized, one batch in the dataset is tested with the training model, and the results are shown in Figure 8. Except for the long-distance "airplane" in the file COCO_val2014_0000000543203.jpg, which is missed because of insufficient training epochs, the proposed model detects almost all labeled objects (large or small) in the original image. The large objects include microwaves, couches, dogs, and motorcycles, while the small objects include ties, airplanes, clocks, and cups. The test results intuitively show the overall precision and the practicability of the object detection network designed in this article.

COCO API can extract the file information labeled on the MS COCO dataset, e.g., BBox parameters, the area size of objects, object number and other information in the images. The COCO API is employed to evaluate the performance of the training model. Thus, COCO API is applied to test 5000 images in Test 2017 under the MS COCO dataset. The test results are shown in Table 2. The three aspects of the IoU, area, and maximum number of objects are analyzed to calculate the average precision and average recall. First, the AP by 10 IoU thresholds of 0.50:0.05:0.95 is averaged, and separate IoU calculations are performed when IoU=0.50 and IoU=0.75. The AP value is higher when IoU=0.50. Then, the AP and the AR are calculated by different detection areas (small or medium or large) of objects. With the increase of the area, the AP and the AR are more accurate. Finally, the AP is calculated by different maximum numbers (1, 10, and 100) of objects detected in each image. The greater the number of object is, the higher the precision. These evaluation indicators validate that the performance of the proposed model is better.

Moreover, the detection results of YOLO-ACN are also compared with those of YOLOv3. The specific comparison results are shown in Figure 9. In this figure four test images are selected from nine representative images in the test set of MS COCO 2017. The image (a) shows that the detection accuracy of the proposed YOLO-ACN for detecting a large

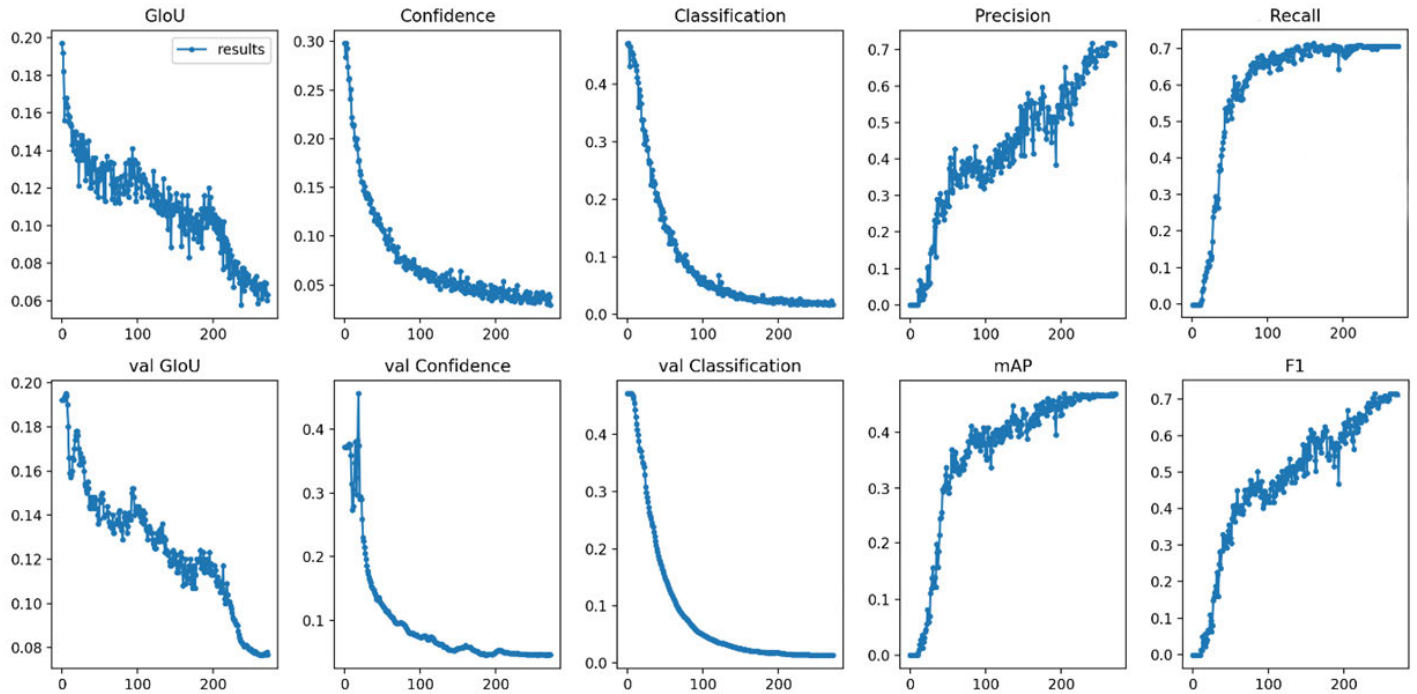


FIGURE 7. Graphs of training and testing results on the MS COCO dataset. The first three columns are the bounding boxes loss (measured by GIoU), confidence loss and class loss in the training dataset and validation dataset. The remaining four curves represent common evaluation indicators for object detection task, and they are P (precision), R (recall), mAP (mean average precision), and F1 (balanced score).

FIGURE 8. YOLO-ACN test results on the validation set of MS COCO 2014. When the first batch training is completed, these images are randomly selected for testing. From the test results, although only one batch is completed, most objects can be detected, especially small targets and occluded objects.

single object in the image is similar to that of YOLOv3. In image (b) and (c), YOLOv3 detects three small persons, but the proposed model detects four small persons, and the proposed model can detect kites in (c). In image (d) when there are multiple objects in dense images, the proposed model can detect small object laptops. The results prove that

(a) (b) (c) (d)

FIGURE 9. Comparison of the detection results of YOLOv3 (the top row) and YOLO-ACN (the bottom row) on typical images on the test set of MS COCO 2017. The detection result of the large size object in the (a) are similar, and the small objects can also be detected well with the proposed YOLO-ACN network, e.g., the long-distance person in (b); the kites and cars in (c); the cups and laptops in (d).

(a) YOLOv3 (b) YOLO-ACN

FIGURE 10. Comparison results of object occlusion problem. (a) the detection result of the YOLOv3, (b) the detection result of the YOLO-ACN. The tie which occluded by the arm can be detected with the proposed YOLO-ACN network, whereas the YOLOv3 cannot detect it.

TABLE 2. Test results on COCO API. By computing AP (average precision) and AR (average recall) based on different IoU values, area sizes, and the number of objects contained in the images to test the performance of the YOLO-ACN.

IoU	Area	maxDets	AP	AR
0.50:0.95	all	100	0.313	-
0.50	all	100	0.498	-
0.75	all	100	0.303	-
0.50:0.95	small	100	0.182	-
0.50:0.95	medium	100	0.321	-
0.50:0.95	large	100	0.372	-
0.50:0.95	all	1	-	0.243
0.50:0.95	all	10	-	0.475
0.50:0.95	all	100	-	0.541
0.50:0.95	small	100	-	0.363
0.50:0.95	medium	100	-	0.601
0.50:0.95	large	100	-	0.783

YOLO-ACN obtains better detection results for small objects through a more accurate design.

In addition, for the occlusion problem, the Soft-NMS [33] algorithm is improved in forecasting the final results, and the Gaussian model is introduced to suppress the surrounding BBox instead of deleting them, and the overlap area, the center point distance, and the aspect ratio between the ground truth BBox and the predicted BBox are added to the Soft-NMS. As the metric of the BBox improved, a more suitable BBox is selected. It is proved in the experiments that the improvement of the Soft-NMS algorithm effectively solves the deficiencies of the original YOLOv3 detection model for the detection of occluded objects.

Figure 10 shows the comparison results with YOLOv3. Figure 10 (a) shows the detection results of YOLOv3. The left person and the tie can be detected, and the right person can be detected, but the tie is not detected by the NMS algorithm, which deletes the BBox with lower confidence. Figure 10 (b) shows the detection results of the YOLO-ACN. The two persons and their ties can be detected accurately by the improved Soft-NMS that retains the occluded objects.

For further quantitative evaluation of the YOLO-ACN performance, the comparisons with some state-of-the-art models are performed. Table 3 shows that the one-stage object

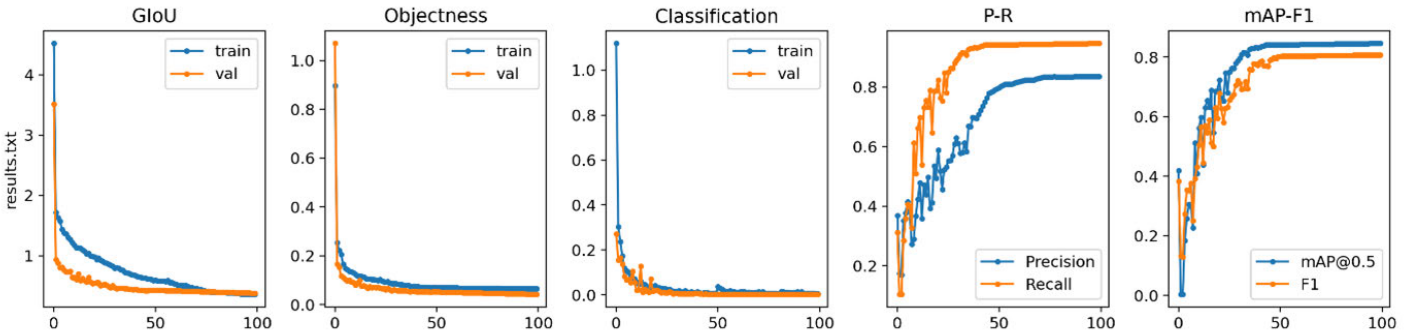


FIGURE 11. The training results of multispectral images. These curves represent GIoU loss, Object loss, Classification loss, P-R (Precision and Recall), and mAP-F1 (mean Average Precision and F1-score) respectively.

TABLE 3. Quantitative comparison of YOLO-ACN and the other state-of-the-art object detectors. The results are reported in terms of mAP percentage and times on the test set of MS COCO 2014.

Method	Backbone	AP	AP50	AP75	APS	APM	APL	Time/ms
RFCN	ResNet-50	32.1	51.9	33.1	14.2	33.3	50.7	170
Faster R-CNN	ResNet-101	34.7	55.5	36.7	13.5	38.1	52.0	420
D-FCN	In-ResNet	37.5	58.0	-	19.4	40.1	52.5	85
Mask R-CNN	ResNeXt	39.8	62.3	43.3	22.1	43.2	51.2	400
YOLOv2	DarkNet-19	21.6	44.0	19.2	5.0	22.4	35.5	25
SSD513	ResNet101	31.2	50.4	33.3	10.2	31.5	49.8	125
RetinaNet	ResNet-101	39.1	59.1	42.3	21.8	42.7	50.2	90
YOLOv3	DarkNet-53	31.6	55.3	32.1	17.2	33.8	40.1	29
YOLOv4	CSPDarknet53	36.1	54.7	38.9	17.3	40.6	50.4	43
YOLO-ACN	Our Method	31.8	53.8	30.3	18.2	32.1	37.2	22

detection detectors have a lower detection accuracy and faster detection speed compared with the two-stage object detection detectors, consistent with [15]–[17]. The detection accuracy of YOLO-ACN is similar to that of the two-stage detection detector Faster R-CNN, but its speed is nearly 20 times faster. Compared with the classic one-stage object detection detector SSD513, the precision of YOLO-ACN is 2.9 higher and the speed is 5 times faster. Compared with YOLOv2, the overall AP increased by 10.2, and the APs increased by 13.2. Compared with YOLOv3, the proposed model has a detection accuracy of 18.2% on small targets due to the introduction of the attention mechanism and CIoU loss, which is 1% higher than that of YOLOv3. Since the proposed model combines the Soft-NMS with CIoU and uses the depthwise separate convolution and the hard-swish activation functions, the speed is also increased by 7 ms. Compared with YOLO-ACN, YOLOv4 achieves higher AP, but it dramatically lowers the inference speed, making it infeasible for real-time application. Specially, YOLO-ACN has a 1% improvement in the detection accuracy of small targets, and nearly twice as fast in speed. The purpose of this article is to pay more attention to small targets and occluded targets while maintaining the real-time detection speed of one-stage object detection. In the experimental results of our platform, we obtain relatively good experimental results in terms of accuracy and speed.

B. KAIST DATASET

Pedestrian detection, as an active research field in computer vision, plays an important role in surveillance, tracking systems [52], and pedestrian safety [38]. However, most of the existing pedestrian detectors are based on colorful images [6], [7] and are unable to obtain useful information at night when the light intensity and contrast are poor; thus the precision of pedestrian detection is limited. KAIST [50], [51] is a commonly adopted pedestrian detection dataset, and it consists of visible light-infrared image pairs. In the infrared images, because the background occludes the pedestrian object, the network extracts only a few features; it is difficult for the model to detect the objects as well as small targets. Therefore, The KAIST dataset is selected to verify the detection performance of the model.

First, the infrared images containing the pedestrian and the corresponding visible light images in the KAIST dataset are selected. Then, the selected KAIST dataset is input into the object detection model to train. Before training, the initial performance of the network is reset; the GIoU loss gain is 3.54; the class loss gain is 37.4; the confidence loss gain is 64.5; the IoU threshold value is 0.15; and the learning rate is 0.00579, which gradually decreases with the training batch size of 16.

To obtain better detection results, the model is further trained on the KAIST dataset based on the detection model trained under the MS COCO dataset, and the training results of each batch are visually analyzed, as shown in Figure 11. When the training epochs reach 100, the GIoU loss, confidence loss, and class loss in the multispectral images training and testing gradually decrease with the training epoch and finally tend to stabilize, indicating that the predicted results

(a) (b) (c) (d) (e)

FIGURE 12. Pedestrian detection results of YOLOv3, YOLOv4, and YOLO-ACN on the KAIST dataset. The first two columns (a) and (b) are the test result of YOLOv3 and YOLO-ACN on the visible light images (Since the test results of YOLOv4 on the three visible light images are the same as those of YOLO-ACN, only the results of YOLO-ACN on them are given). The last three columns (c), (d), and (e) is the test result of YOLOv3, YOLOv4, and YOLO-ACN on the corresponding infrared images.

TABLE 4. Comparison of the accuracy of different object detectors on the KAIST dataset.

Method	Backbone	AP	AR	mAP	F1	Time/ms	Parameters/M	Weight Size/MB
YOLOv3	DarkNet-53	73.5	76.9	79.6	74.8	25	61.5	246.4
YOLOv4	CSPDarkNet	76.9	75.8	81.0	76.3	37	63.9	256.3
YOLO-ACN	Our Method	76.2	87.9	82.3	81.6	20	47.4	177.6

of the proposed training model gradually approach the true results. At the same time, the precision rate P and the recall rate R of the model are being improved with the increase of the training batches, and the values of F1-score and the accuracy mAP are also increasing. The final accuracy mAP even reaches over 80%.

Table 4 lists the comparative results of three methods on the KAIST dataset. Compared with YOLOv3, the proposed algorithm YOLO-ACN improves the performance with gains of 3.67% AP, 14.3% AR, 3.39% mAP, and 9.1% F1. Contrasted with YOLOv4, the average precision (AP) is similar, but the other evaluation metrics bring amazing performance gains, e.g., 15.96% AR, 1.6% mAP, and 6.94% F1. Moreover, the processing time of YOLO-ACN is an average of 22 ms with batch=1, about 80% of YOLOv3, and about 60% of YOLOv4. As seen, on the evaluation indicators of mAP and F1, the YOLO-ACN method achieves the best results among the three methods. The standard YOLOv3 and YOLOv4 parameters are more than 61 million and 64 million, and the weight size both have reached over 200MB; whereas YOLO-ACN reduces the parameters by 22%, and the weight size is also reduced by nearly 30%. The fewer parameters have greatly reduced the model volume, and the smaller weight size is suitable for deployment on devices with limited computing power. Given the feature that the objects contained in the KAIST infrared image dataset are blocked by the background, YOLO-ACN can better pay attention to the small pedestrians that are occluded in the image. The main contributions for these encouraging results are as follows: the attention mechanism extracts accurate features of objects; the CIoU loss achieves precise regression of BBox. Also the CIoU is applied in the Soft-NMS; the Gaussian model in the Soft-NMS is employed to suppress the surrounding BBox.

After training, the test results of YOLO-ACN are compared with those of YOLOv3 and YOLOv4. Figure 12 shows three visible light images and the corresponding infrared images that are selected to test YOLOv3, YOLOv4, and YOLO-ACN. From column (a), we can see that YOLOv3 can basically detect the existence of pedestrian targets with the exception of the long-distance pedestrian object in the second image of column (a). However, the object is covered by strong noise and low-contrast background in the infrared images from column (c). The missed detection is serious for this reason. For example, one pedestrian is not detected in the second image of column (c), and the two pedestrians in the third image are missed. From column (d), we find that YOLOv4 can detect the existence of pedestrian targets in the infrared images, except the two pedestrians that are too close in the third image. The detection results of the YOLO-ACN in column (b) of the visible light images and column (e) of the infrared images represent that the detection accuracy of both visible light and infrared images are relatively high, especially the occluded pedestrian in the third images of column (b) and column (e), and the long-distance small

(a) (b) (c)

FIGURE 13. Video detection results of some typical frames. (a) Results of YOLOv3. (b) Results of YOLOv4. (c) Results of YOLO-ACN. Compared with YOLOv3 and YOLOv4, YOLO-ACN has an amazing performance on small targets and occluded objects detection, for example, the cell phone in the first image of the column (c), the person beside the tree in the second image, and the skateboard in the third image.

targets in the second image of column (b) can be detected correctly. Comparing the results, due to the occlusion of the background in the infrared images, the features carried by the pedestrian are just a few, so it is hard to extract the effective features of pedestrians. However, by improving the attention mechanism, loss function, and Soft-NMS algorithm, the proposed model has good detection results in small targets and occluded objects which all carry a few pixels and features. Thus, the proposed model has not only a good detection effect on pedestrians in visible light but also a higher accuracy than YOLOv3 in the detection of pedestrians in infrared images. Contrasted with YOLOv4, the test results of the visible light images are similar, but the performances of YOLO-CAN in the infrared images are amazing.

C. VIDEO DATASET

To test the real-time performance and generalization ability of the proposed model, a Campus Video Dataset is made by ourselves. The 5 videos on campus are randomly collected and the effective images of the collected videos are intercepted, and their resolution are 1920×1080. Then the images are labeled with LabelImg. Finally, an object detection dataset is prepared.

The Campus Video Dataset is employed to train the detection model YOLO-ACN, and then the training results are obtained. Compared with YOLOv3, the results are analyzed from the aspect of detection speed and detection accuracy. In terms of detection speed, for a video of 155 s, the YOLOv3 takes 320.255 s about 14 fps on the device utilized in the experiment, and YOLO-ACN takes 280.047 s, approximately 16 fps. For the detection accuracy, each frame of the detected video is obtained. Then, the detection results of the same frame are compared. The following frames of images are selected from the obtained 4491 frame images. The comparison results are shown in Figure 13. (a) and (b) show that the test results of YOLOv3 and YOLOv4. Compared with the test results of YOLO-ACN in (c), the small object cell phone in the first line and the skateboard in the third line can be detected only by YOLO-ACN; the person who is occluded by the trees in the second line can be detected by YOLOv4 and YOLO-ACN. The results demonstrate that the performance of the proposed model in detecting small targets and occluded objects in the video is also better than that of YOLOv3 and YOLOv4.

D. ABLATION STUDY

The quantitative ablation experiments on the PASCAL VOC [20], [21] dataset are conducted to study the impact of the proposed method for the experimental results. PASCAL VOC containing 20 classes, is also a widely used dataset

TABLE 5. Ablation study of detection precision on the test set of PASCAL VOC.

Channel attention	Spatial attention	CIoU loss	Soft-NMS	AP	AR	mAP50	F1
	√	√	√	51	54	50.2	51.8
√		√	√	41.8	59.6	52.1	49.1
		√	√	51.6	53.8	51.3	52.1
√	√		√	55.6	56.3	54.2	55.9
√	√	√		55.5	55.8	54.5	55.5
√	√	√	√	55.8	56.7	55.7	56.2

TABLE 6. Ablation study of detection speed.

hard-swish	depth separable convolution	Speed(ms)	NMS	total
	√	26.1	2.0	28.2
√		25.8	1.8	27.6
√	√	22.2	1.9	24.1

in the object detection. The train and validation datasets of VOC2007 and VOC2012 are utilized for training and the test of VOC2007 is apply for testing. In the ablation study, the four factors of the channel attention, special attention, CIoU loss, and Soft NMS are considered. The impact of the detection accuracy is compared, and the experimental results are shown in Table 5. The two aspects of hard-swish and depthwise separable convolution are used to compare the impact of the detection speed, and experimental results are shown in Table 6.

On our experimental platform, 100 epochs of training are conducted and other parameters are unchanged. For the detection accuracy, when the channel attention mechanism is not added, other improved methods are retained. Similarly, the spatial attention mechanism, the CIoU loss, or Soft-NMS are not added to study the influence of these different methods. From the average detection accuracy, the attention mechanism affects the detection accuracy of the model, which increases from 51.3% to 55.7%, an increase of 4.4%, so compared with the CIoU and Soft-NMS algorithm which improves the model by about 1%, the attention mechanism has a major impact on the precision improvement of model detection. These four algorithms are indispensable for the improvement of overall accuracy.

For the detection speed of the model, the improvement method of the model used for precision only focusing on the activation function hard-swish and the depthwise separable convolution are maintained, respectively. Table 6 shows the detection speed of the model, and the speed of the Soft-NMS postprocessing. Both factors have improved the overall detection speed of the model.

Although the epochs in training sets as 100, there is still a little increase in training results, but the overall trend is stable. From the training results of these 100 epochs, the attention mechanism, CIoU loss, and improved Soft-NMS algorithm increase the detection accuracy of the entire model, and the depthwise separable convolution and hard-swish algorithm boost the detection speed of the entire model. Therefore, the ablation experiment shows that the overall performance of the model has been improved on the basis of the improvement.

V. CONCLUSION

Inspired by the YOLOv3 and convolution block attention module. In this article, a one-stage detection model YOLO-ACN is proposed by developing a lightweight network with the attention mechanism, improving the measurement of BBox, introducing the CIoU loss function, and optimizing the Soft-NMS. The detection accuracy and speed of small targets and occluded objects are further increased in this method. The MS COCO dataset is used to train, validate, and test the model YOLO-ACN. Experiment results show that the precision of the YOLO-ACN is similar to Faster R-CNN which is a two-stage detection algorithm, but the detection speed has a significant improvement. The detection accuracy is 2.9 times higher than that of the classic one-stage object detection algorithm SSD513, and the speed is 5 times faster. Compared with YOLOv3, the detection accuracy is similar, and the detection speed is slightly faster, but the proposed model achieves promising performance in detecting small targets and occluded objects. The mAP for small targets reaches 18.2%, so the accuracy is better than that of YOLOv3. To further verify the detection performance and robustness of the proposed YOLO-ACN, visible light and infrared images of the KAIST dataset and a self-built Campus Video Datasets are adopted. The detection results are compared with YOLOv3, further verifying the universality and efficiency of the YOLO-ACN in detecting small targets and occluded objects.

REFERENCES

[1] X. Wu, D. Sahoo, and S. C. H. Hoi, "Recent advances in deep learning for object detection," *Neurocomputing*, vol. 396, pp. 39–64, Jul. 2020, doi: 10.1016/j.neucom.2020.01.085.

[2] X. Ying, Q. Wang, X. Li, M. Yu, H. Jiang, J. Gao, Z. Liu, and R. Yu, "Multi-attention object detection model in remote sensing images based on multi-scale," *IEEE Access*, vol. 7, pp. 94508–94519, 2019.

[3] Y. Liu, H. Liu, J. Fan, Y. Gong, Y. Li, F. Wang, and J. Lu, "A review of research and application of small target detection based on deep learning," *Chin. J. Electron.*, vol. 48, no. 3, pp. 590–601, 2020.

[4] P. Viola and M. Jones, "Robust real-time face detection," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 3. Washington, DC, USA: IEEE Computer Society, 2001, p. 747, doi: 10.1109/iccv.2001.937709.

[5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 886–893.

[6] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* Washington, DC, USA: IEEE Computer Society, Jun. 2010, pp. 2241–2248, doi: 10.1109/cvpr.2010.5539906.

[7] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[8] 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.

[9] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[10] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 1491–1498.

[11] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732.
[12] 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.
[13] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.
[16] C. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," *CoRR*, vol. abs/1701.06659, pp. 1–11, Jan. 2017. [Online]. Available: http://arxiv.org/abs/1701.06659
[17] 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.
[18] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6517–6525.
[19] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *CoRR*, vol. abs/1804.02767, pp. 1–6, Apr. 2018. [Online]. Available: http://arxiv.org/abs/1804.02767
[20] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 734–750.
[21] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6568–6577.
[22] Y. Zhao, F. Shi, M. Zhao, W. Zhang, and S. Chen, "Detecting small scale pedestrians and anthropomorphic negative samples based on light-field imaging," *IEEE Access*, vol. 8, pp. 105082–105093, 2020.
[23] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 652–663, Apr. 2017.
[24] H. Zhu, X. Yan, H. Tang, Y. Chang, B. Li, and X. Yuan, "Moving object detection with deep CNNs," *IEEE Access*, vol. 8, pp. 29729–29741, 2020.
[25] X. Zhang, F. Guo, Y. Liang, and X. Chen, "Summary of small target detection algorithms based on deep learning," *Softw. Guide*, vol. 19, no. 05, pp. 276–280, 2020.
[26] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," Apr. 2020, *arXiv:2004.10934*. [Online]. Available: http://arxiv.org/abs/2004.10934
[27] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.
[28] L. Sifre and S. Mallat, "Rigid-motion scattering for texture classification," *CoRR*, vol. abs/1403.1687, pp. 1–19, Mar. 2014. [Online]. Available: http://arxiv.org/abs/1403.1687
[29] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, 2013, vol. 30, no. 1, p. 3.
[30] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.
[31] 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*, 2020, pp. 12993–13000.
[32] Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, and W. Zuo, "Enhancing geometric factors in model learning and inference for object detection and instance segmentation," May 2020, *arXiv:2005.03572*. [Online]. Available: http://arxiv.org/abs/2005.03572
[33] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, vol. 3, 2006, pp. 850–855.
[34] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.
[35] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[36] B. Singh and L. S. Davis, "An analysis of scale invariance in object detection–SNIP," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3578–3587.
[37] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.
[38] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.
[39] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang, "Acquisition of localization confidence for accurate object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 784–799.
[40] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.
[41] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.
[42] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6450–6458.
[43] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.
[44] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, pp. 1–11, Feb. 2015. [Online]. Available: http://arxiv.org/abs/1502.03167
[45] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11534–11542.
[46] P. Ramachandran, B. Zoph, and Q. V. Le, "Swish: A self-gated activation function," 2017, *arXiv:1710.05941*. [Online]. Available: https://arxiv.org/abs/1710.05941v1
[47] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 658–666.
[48] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS—Improving object detection with one line of code," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5562–5570, doi: 10.1109/iccv.2017.593.
[49] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
[50] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1037–1045.
[51] Y. Choi, N. Kim, S. Hwang, K. Park, J. S. Yoon, K. An, and I. S. Kweon, "KAIST multi-spectral Day/Night data set for autonomous and assisted driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 934–948, Mar. 2018.
[52] J. W. Davis and V. Sharma, "Background-subtraction using contour-based fusion of thermal and visible imagery," *Comput. Vis. Image Understand.*, vol. 106, nos. 2–3, pp. 162–182, May 2007.

YONGJUN LI was born in Kaifeng, Henan, China, in 1977. He received the M.S. degree in circuit and system from Guangxi Normal University, in 2005, and the Ph.D. degree in communication and information system from Xidian University, in 2017.

From 2005 to 2008, he was a Lecturer with the School of Physics and Electronics, Henan University. Since 2012, he has been an Assistant Professor. He has presided over and participated in a number of National Natural Science and Technology Funds, and took part in the Shaanxi Province Key Science and Technology Innovation Team Project. He has applied for six national invention patents and published more than 30 articles in *Multimedia Tools and Applications*, *Optical Engineering*, *Journal of Electronic Imaging*, *Journal of Zhengzhou University*, and *Journal of Henan University*. His research interests include image processing and artificial intelligence.

SHASHA LI was born in Luoyang, Henan, China, in 1996. She received the B.S. degree in electronic science and technology from the Henan Institute of Engineering, Zhengzhou, China, in 2019. She is currently pursuing the M.D. degree in optical engineering with Henan University, Kaifeng, China. Her research interests include computer vision, image processing, and object detection.

HAOHAO DU received the B.S. degree in electronic science and technology from the Henan Institute of Engineering, Zhengzhou, China, in 2018, and the M.S. degree in optical engineering from Henan University, Kaifeng, China, in 2020. His research interests include deep learning, image processing, and object detection.

LIJIA CHEN is currently an Associate Professor of communication engineering with the School of Physics and Electronics, Henan University, Henan, China. He has been serving as the Director of graduate studies in detection technology and automation, the Head of communication engineering, and a Member of the Research Group with the Laboratory of Advanced Computation Methods and Intelligent Applications. He is also a Council Member of the Henan Communication Academy. He has published more than ten journal articles, more than ten conference papers, and three granted patents. His research interests include evolutionary computation, communication networks, digital signal processing, and evolutionary design methods on electronic circuits and systems.

DONGMING ZHANG received the B.S. degree in mechanical design manufacture and automation from Central South Forestry University, China, in 2001, and the M.S. degree in precision instrument and machinery from Wuhan University, China, in 2005. From July 2005 to August 2008, he was an Assistant with Henan University, China, where he has been a Lecturer since September 2008. His current research interests include image processing, intelligent video surveillance systems, object detection, depth prediction, and machine learning.

YAO LI was born in Henan, China, in 1998. He received the B.S. degree from Hainan University, China, in 2020. He is currently pursuing the M.S. degree with Henan University. His current research interests include computer vision, image processing, and deep learning.