Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.044284

ARTICLE

Interactive Transformer for Small Object Detection

Jian Wei, Qinzhao Wang* and Zixu Zhao

Department of Weaponry and Control, Army Academy of Armored Forces, Beijing, 100071, China

*Corresponding Author: Qinzhao Wang. Email: airy_snow@outlook.com

Received: 26 July 2023 Accepted: 15 September 2023 Published: 29 November 2023

ABSTRACT

The detection of large-scale objects has achieved high accuracy, but due to the low peak signal to noise ratio (PSNR), fewer distinguishing features, and ease of being occluded by the surroundings, the detection of small objects, however, does not enjoy similar success. Endeavor to solve the problem, this paper proposes an attention mechanism based on cross-Key values. Based on the traditional transformer, this paper first improves the feature processing with the convolution module, effectively maintaining the local semantic context in the middle layer, and significantly reducing the number of parameters of the model. Then, to enhance the effectiveness of the attention mask, two Key values are calculated simultaneously along Query and Value by using the method of dual-branch parallel processing, which is used to strengthen the attention acquisition mode and improve the coupling of key information. Finally, focusing on the feature maps of different channels, the multi-head attention mechanism is applied to the channel attention mask to improve the feature utilization effect of the middle layer. By comparing three small object datasets, the plug-and-play interactive transformer (IT-transformer) module designed by us effectively improves the detection results of the baseline.

KEYWORDS

Small object detection; attention; transformer; plug-and-play

1 Introduction

The object detection model has achieved fruitful research results and has been widely used in production, life, and other fields, significantly improving efficiency. However, these detection models still face challenges from small object detection tasks. As shown in Fig. 1, the model has more false detections and missed detections of small objects. There are three main reasons for this result: first, the small object lacks distinguishable and significant features, the second is that the small object is easy to be obliterated in the surrounding environment, and the third is that in the deep neural network, pooling, normalization, label matching and other modules will gradually attenuate the relevant features of the small objects layer by layer, resulting in the lack of relevant information at the detection head [1,2]. The combined effect of these factors leads to the poor detection results of traditional models on small objects.

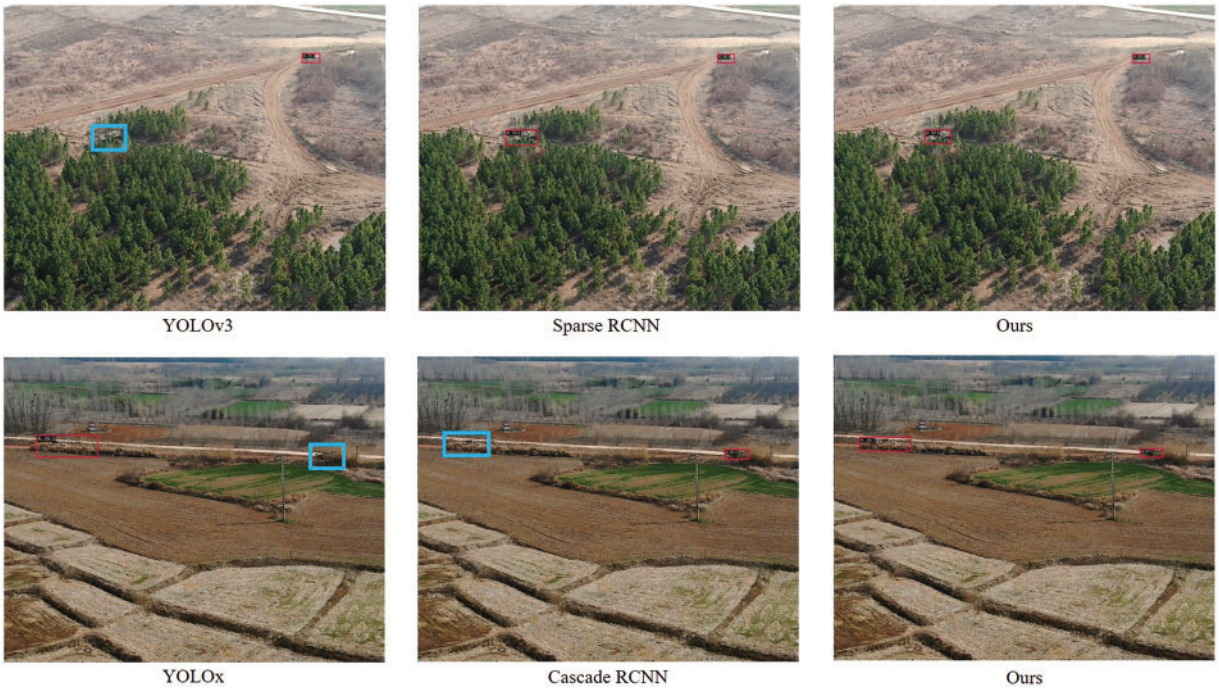

Figure 1: Small object detection. The traditional first-stage and second-stage object detection models cannot effectively deal with unfavorable factors such as object occlusion, environmental interference, and small object size, resulting in easy misdetection and missed detection. The improved model with the addition of the IT-transformer effectively overcomes these challenges

To solve this problem, models such as multi-scale pyramid [3–5] and feature pyramid [6–8] are used to process object features at different scales, that is to improve the detection accuracy of small objects by hierarchical processing and end fusion. Another approach is to use larger feature maps [1,9], such as [1] adding P2 layer features with less loss of feature information to the neck module, which effectively improves the available features of small objects; On the contrary, larger feature maps lead to slower inference speed; Focus [10] proposed a method of slice, which retains as many small object features as possible without compressing the size of the input image; The you only look once (YOLO) [11,12] models add data augmentation strategies, such as mosaic to diversify images in a wider range to improve the contribution of small objects to training loss. In [13,14], the method of deformation convolution is used to change the position of the convolution kernel and guide the convolution kernel to extract the characteristics of a more accurate position. Other studies have proposed the addition of the attention mechanism [15–17], by adding an attention mask representing the importance of each region, to improve the attention of the model to different regions during processing, and effectively suppress the noise of irrelevant regions. At present, the attention mechanism model represented by the transformer [18] shines in many image processing tasks [19–21] and has received more and more attention with its unique feature processing methods.

In summary, in terms of the actual task requirements, based on the transformer attention mechanism, to fully construct the global and local semantic context of the avatar, we propose an IT-tansformer attention mechanism to solve the detection problem of small objects. Specifically, the traditional transformer adopts the calculation method based on the fully connected layer, resulting in a heavy number of parameters, extremely high requirements for hardware, and insufficient local

semantic characteristics due to the serialized data processing mode. Second, in the multi-head attention mechanism, the query (Q), key (K), and value (V) are obtained separately. That is to say, they do not explore Q and K deeply, nevertheless, the poorly explored relationship between each other weakens the effectiveness of attention masking. To solve these two problems, we design an interactive transformer module that can be plug-and-play. In detail, based on the previous research, we first replace the fully connected layer with a 2D convolution module, use the characteristics of shared weight to provide evidence, reduce the number of overall parameters of the model, realize the lightweight processing of the model, and at the same time, improve the local context between the associated regions with the help of the local field of view of the convolution module. Then, to further enhance the feature representation ability of the middle layer and improve the accuracy of the attention mask, a feature processing method based on cross-fusion K is proposed, and the coupling relationship in the features of the middle layer is highlighted by fusing the K of the Q and V bidirectional branches, to improve the model's attention to detailed information. Finally, unlike the fully connected layer to calculate the interaction effect between each pixel, we focus on the features between different channels, to maintain the consistency of the global spatial position relationship of the features, and effectively improve the feature representation of objects at each scale by applying channel-level multi-head attention to the features of the middle layer.

In summary, our main contributions are:

1. The object detection model based on the IT-transformer is proposed. From the perspective of improving the utilization efficiency of features in the middle layer, the dual-branch model is used to extract the key values of features and provide more effective comparison features for the attention module through cross-fusion. At the same time, to suppress the interference of noise channels, the multi-head attention mechanism is applied to the generation and optimization of channel attention masks, which significantly improves the differentiation of the characteristics of the middle layer.

2. A new small object detection dataset was collected and organized. Given the existing small object detection data set, the types of objects are mostly common objects, and the object acquisition angle and the scene are simple, etc. At the same time, to expand the application of intelligent detection algorithms in the military field, we collect and sort out an Armored Vehicle dataset with diverse viewing angles, variable distances, and complex scenes through network collection and unmanned aerial vehicle (UAV) shooting, and carry out experiments on small object detection models in it.

3. Extensive experimental comparisons and self-ablation experiments were carried out to verify the effectiveness of the module. The results show that the proposed IT-transformer can realize plug-and-play in the first-stage and second-stage detection models, which can effectively improve the detection accuracy of the baseline model. In the three datasets of Armored Vehicle, Guangdong University of Technology-Hardhat Wearing Detection (GDUT-HWD), and Visdrone-2019, the mAP was improved by 2.7, 1.1, and 1.3 compared with the baseline, respectively.

2 Structure

2.1 Object Detection

Object detection models based on deep learning have been fully developed, and they are mainly divided into four branches: first, first-stage detection models, with YOLO [11,12,22,23], single shot multibox detector (SSD) [24], and Retina [25]. They integrate region of interest (ROI) generation and final result prediction, with faster image inference speed; Then there is the second-stage detection model, represented by Faster region-based convolutional network method (Faster RCNN) [26],

Cascade RCNN [27], etc. Their main feature is to set a separate module for more accurate ROI extraction, and the addition of ROI alignment makes the detection accuracy of the object significantly improved; The third is the transformer-based detection model, such as vision transformer (ViT) [19], detection transformer (DETR) [21], DETR with improved denoising anchor box (DINO) [28], etc. Represented by the addition of transformers, they integrate the addition of transformers into object detection tasks, breaking the previous situation of convolutional modules in the image field, and with the unique attention mechanism in transformers, the detection accuracy of such models quickly catches up with a series of traditional state of the art (SOTA) models; The fourth is the detection architecture based on the diffusion model [29–31]. Based on the diffusion model, they regard the positioning problem of the object as the iterative diffusion process from a random noise vector to the true value and complete the detection task of the object through cascade alignment. In this paper, we first take the second-stage detection model Cascade RCNN as the benchmark to make full use of the characteristics of the distributed model structure. At the same time, to further improve the model performance, we also integrate the transformer attention mechanism to achieve the organic integration of the two. Guided by the plug-and-play idea, we have designed an interactive attention module that can adapt to the existing first-stage and second-stage detection models, which can effectively improve the detection performance of the baseline model.

2.2 Small Object Detection

Small object detection is an important part of the computer vision task. According to the definition of the COCO dataset, when the object size is less than 32×32 pixels, the object can provide very limited feature information, resulting in increased detection difficulty. To solve this problem, there are currently four main ideas: first, increase the size of the input image [9,10], so that the feature can remain relatively stable, but too large input size will lead to a significant decrease in inference speed, which is not suitable for scenarios with high real-time requirements; The second is the data augmentation strategy [32,33], represented by the mosaic and generative adversarial network (GAN). In the data preprocessing stage equipped with mosaic, through controllable parameter adjustment, the proportion of small objects in all training instances in the training process is increased, and the parameter update process dominated by large-size objects in the past is improved. In [34], GAN synthesis and diversification of small objects are used to increase the number of positive samples in the training process; Third, the multi-scale training and testing strategy [6,35,36] is adopted to improve the consistency detection ability of the model for objects at each scale by changing the input image size within a large range. The fourth is to add an attention mechanism [2,17,37], which improves the attention of the model to specific regions and objects by additional calculation of attention masks that indicate the importance of pixels. Starting from the perspective of improving the attention of the model, this paper proposes an interactive attention mechanism. With the help of the IT-transformer, the model can effectively represent the importance of the feature under the single-scale training strategy, to improve the accuracy of the small object.

2.3 Transformer

Transformer [18] was originally used to process serialized text data and made its mark in the natural language processing (NLP) field. ViT [19] converts image data into serialized data composed of multiple pixel blocks for the first time, and then performs image classification and detection tasks in the way of transformers, opening the way for transformers to expand into the image field. Based on the transformer architecture, many excellent models have emerged, such as DETR [21], and Swin transformer [20]. The main feature of the transformer is the feature processing method based on

mutual attention between tokens, which covers the global semantic information in a single position, which greatly improves the accuracy of the model inference results. However, under the single scale setting, the transformer controls the number of model parameters by dividing the specified number of tokens, but it still produces significantly higher parameters than the convolution module. Because of the serialized image, the semantic relationship between adjacent tokens is broken. Experiments show that when the dataset is small, the transformer-based model is difficult to effectively learn the effective interrelationship matrix, resulting in low performance. This paper uses the attention mechanism in the transformer to improve the cross-K value by integrating the middle-layer features. Furthermore, by integrating the convolution module, we strengthen the semantic correlation between tokens, to improve the performance of the model in the smaller dataset.

3 Method

In this part, first, we briefly introduce the relevant content of traditional transformers and then introduce the structure and optimization indicators of IT-transformers in detail.

3.1 Revisiting Transformer

Transformer is a deep neural network model based on an encoder and decoder, and its core content is the construction of the attention mechanism. Thanks to the globally encoded token, the transformer uses fully connected modules to ensure that each token has a broad field of view and a full range of connection relationships, which ensures better performance in advanced visual tasks such as object detection and segmentation. The transformer attention mechanism is based on the calculation process of the matrix, specifically, the calculation of Q, K and V based on the characteristics of the middle layer, and then transpose and multiply the three. The interrelationship matrix reflecting the importance of each token is obtained, that is, the attention mask. The structure of a traditional transformer is shown in Fig. 2.

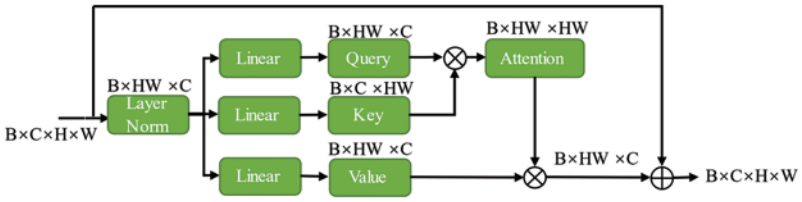

Figure 2: The traditional transformer

Suppose the input characteristics are $X \in R^{C\times H\times W}$ that in the traditional transformer calculation process, it is necessary to first normalize the flattened two-dimensional matrix of X ($X' \in R^{C\times HW}$), and then multiply it with three weight matrices (W_Q, $W_K \cdot W_V$), representing fully connected operations to obtain the representation of Q, K, and V, where Q is calculated by:

$$Q = X' \times W_Q \tag{1}$$

K and V are calculated similarly. In particular, to ensure the unbiased nature of the extracted features, the bias coefficient needs to be set to zero when processing with a fully connected matrix.

Then, by transposing each multiplication of Q and K, the correlation matrix between the two is obtained. Finally, the softmax activation function is used to normalize it to (0–1), that is, the spatial

attention mask reflecting the importance of each token is obtained. The calculation process is:

$$Attention = \text{Softmax}(Q \times K') \tag{2}$$

Under multi-head attention, several such attention matrices can be calculated at the same time. Next, these matrices are integrated by stitching and merging. At last, the hop connection method is used to weighted fusion with the input features to obtain the feature map optimized by attention masking, and send it to the subsequent detection module. Its calculation formula is:

$$X_{out} = X + (X' \cdot Attention).reshape(C, H, W) \tag{3}$$

In this process, since the calculation of Q, K, and V uses a fully connected layer module, its parameter quantity is $(CHW)^2$. The increased parameters will lead to a decrease in training efficiency, increased energy consumption, and other problems. Therefore, many jobs are faced with the problem of controlling the number of overall parameters of the model when designing and deploying transformer models.

In addition, it is worth noting that the calculation and processing of Q, K, and V are the core content of the transformer and directly determine the effectiveness of attention masking. However, traditional transformers are only processed through 3 separate fully connected layers. Q, K, and V are the basis for calculating attention masks, so it is necessary to explore their processing methods in more depth to improve the accuracy of attention masks.

3.2 IT-Transformer

The overall structure of the IT-transformer is shown in Fig. 3. The research shows that the transformer structure is different from the traditional convolution-based model. In feature processing, due to the lack of the local field of view, the transformer-based architecture cannot complete the acquisition of local semantic context, which will significantly affect the detection performance of the model when the training dataset is small. In addition, as we introduced earlier, transformers widely use the fully connected layer to calculate the characteristics of the middle layer, resulting in a large number of parameters. In this regard, referring to the research results of many existing structural convolutions and transformers, to balance the number of parameters and the demand for attention mechanisms, we design Q, K, and V calculation methods based on convolutional modules. First of all, through weight sharing, the convolution module can effectively use the local correlation semantic context between adjacent pixels, that is, the local field of view of the convolution kernel. On the other hand, it can significantly reduce the parameters.

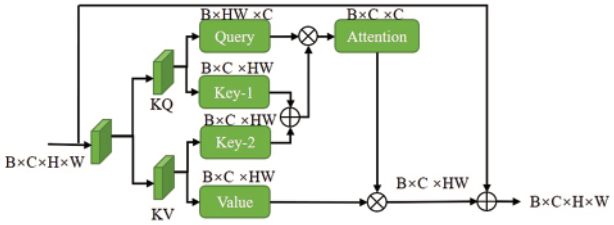

Figure 3: The IT-transformer

Taking the calculation of Q as an example, the traditional transformer middle layer features donated as X, and C, H, and W is 1024, 64, 64, 64, respectively, so its parameter quantity is: $Param(Q_{transformer}) = (1024 \times 64 \times 64)^2$. In IT-transformer, when the 3×3 convolution kernel module is

used to obtain Q, K, and V, the parameters are: $Param(Q_{IT-transformer}) = (1024 \times 9)^2$. It can be seen that through this lightweight design, the number of parameters of the IT-transformer module has nothing to do with the size of the middle layer features, and the number of parameters is reduced by a factor of $(64 \times 64/9)^2$ compared with the fully connected method in Fig. 2. The number of parameters is greatly compressed, which helps to improve the efficiency of model training and reduce the hardware requirements of the model.

At the same time, to further strengthen the connection between Q, K, and V, we use synchronous calculation. As can be seen from Fig. 3, Q and K_1, K_2 and V are calculated by the same convolution module, and then through channel splitting, we get Q, K, and V with more close coupling effects.

$$Q, K_1 = chunk(Conv_{KQ}(X), \dim = 1) \tag{4}$$

$$K_2, V = chunk(Conv_{KV}(X), \dim = 1) \tag{5}$$

By setting up the dual branch, we obtain K rooted in Q and V, and it can be said that the extracted features $K_1 \cdot K_2$ contain more explicit cross-coupling features, which provide richer sampling information for attention calculations. When calculating attention, according to the unified requirements of the transformer architecture, we get the key feature expression after crossover, namely:

$$K_{IT} = K_1 + K_2 \tag{6}$$

We also use multi-head attention to complete the analysis from multiple different dimensions, and fully use the characteristics of the middle layer, to achieve the purpose of improving the effectiveness of attention masking. First of all, we know that the contribution of different channel feature maps is different, some feature maps are accurately extracted to the decisive features, while other channels may introduce noise. If the characteristics of each channel are set to the same weight, it will inevitably affect the final judgment of the model. Therefore, under the premise that the convolution module has been used to extract the local semantic context of the feature map, we pay more attention to which channel features have a more important position in the multi-head attention. Therefore, unlike the way the transformer module focuses more on spatial attention, IT-transformer focuses on different channels. Under the bullish attention mechanism, we divide Q, K, and V into subsets according to the number of heads N, where $Q_{sub} \in R^{\frac{C}{N} \times H \times W}$, $K_{IT-sub} \in R^{\frac{C}{N} \times H \times W}$, $V_{sub} \in R^{\frac{C}{N} \times H \times W}$.

The computational focus of attention also becomes the acquisition channel-level attention mask, which is calculated as follows:

$$Attention_{channel} = Cat(V_{sub} \cdot Soft\max(K_{IT-sub} \cdot Q')) \tag{7}$$

We obtain the mask that reflects the attention of each group of channels by parallel computing, and then we also use the splicing method to obtain the attention mask that reflects the features of all the intermediate layers, among them $Attention_{channel} \in R^{C \times C}$. Finally, by adding the input of the module by jumping the connection, the feature representation of the middle layer is further strengthened, and the enhanced feature map is obtained.

$$X_{out} = X + Attention_{channel} \tag{8}$$

3.3 Loss Function

We detail the structure and working process of IT-transformers. In fact, in the detection task of small objects, to improve the overall detection accuracy of the model, we add the P2 level feature map refer to [1,9] and detect small objects in the large-size feature map. Here, using Cascade RCNN as the

baseline, we design an IT-transformer-enhanced small object detection model, the overall structure of which is shown in Fig. 4.

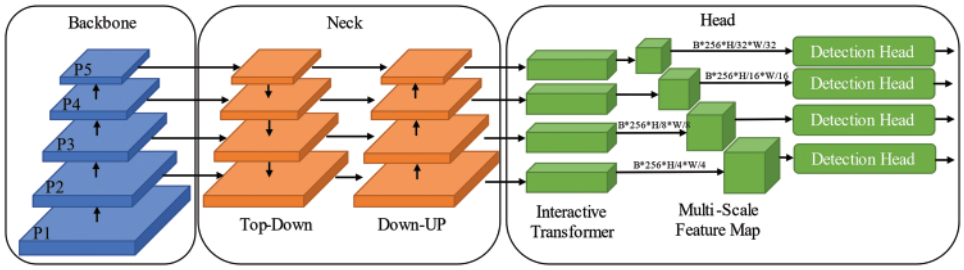

Figure 4: The improved Cascade RCNN with IT-transformer

It can be seen that the IT-transformer can be plugged directly into the back end of the feature pyramid network (FPN), which also means that the IT-transformer can achieve a plug-and-play effect. In this regard, we conducted experimental verification in Section 4.6, showing the wide utility and effectiveness of the IT-transformer.

As shown in Fig. 4, this paper selects Cascade RCNN [27] as the baseline model, and builds the object detection model by inserting IT-transformer into it, so its loss function is mainly composed of 2 parts, and its calculation formula is:

$$\mathcal{L} = \mathcal{L}_{\text{RPN}} + \mathcal{L}_{\text{ROI}} \tag{9}$$

Among them, the $\mathcal{L}_{\text{RPN}}$ model makes the initial judgment of the object presence and position of the feature map, which is composed of binary classification $\mathcal{L}_{object}$ loss and location regression loss $\mathcal{L}_{\text{loc}}$, specifically:

$$\mathcal{L}_{object} = -\log\left[p_i p_i^{'} + (1-p_i)(1-p_i^{'})\right] \tag{10}$$

$$\mathcal{L}_{loc} = \lambda \cdot \frac{1}{N_{reg}} p_i^{'} \cdot \mathcal{L}_{reg}\left(b_i, b_i^{'}\right) \tag{11}$$

where i represents the serial number of the anchor, p_i is the probability that the i-th anchor has an object, $p_i^{'}$ is the label assigned by the first anchor (1 when containing the object, otherwise 0), N_{reg} is the total number of valid object boxes currently predicted by the model, b_i is the number the coordinates of the object position predicted by the i-th anchor, similarly, $b_i^{'}$ are the real coordinates assigned by the i-th anchor containing the object, which is λ the adjustment coefficient of loss, which is set to 1.0 by default according to mmdetetion[1].

So far, we get a series of ROIs. Then, the $\mathcal{L}1$ loss fine-tuning object location box is used, which is calculated as:

$$\mathcal{L}_{\text{loc}}[f] = \sum_{t=1}^{N} \mathcal{L}_{reg}\left(f\left(x_i, b_i\right), g_i\right) \tag{12}$$

where $f(x_i, b_i)$ is the positional regression function, which is used to regress the candidate bounding box b_i to the object bounding box g_i. In fact, due to the fine-tuned regression method using cascading

[1] https://github.com/open-mmlab/mmdetection

f, it consists of phased progressive functions, specifically:

$$f(x_i, b_i) = f_T \circ f_{T-1} \circ \cdots f_1(x, b) \tag{13}$$

in this paper $T = 3$, the regression position is fine-tuned under three conditions.

Further, in the t first stage, the position $\mathcal{L}_1$ loss function based on the calculation formula is as follows:

$$\mathcal{L}_{\text{reg}} = \sum_{i=1}^{N_{reg}} \begin{cases} c_i, & IOU(x, g) > u \\ 0, & otherwise \end{cases} \tag{14}$$

where c_i represents the object class vector predicted by the anchor when the intersection over union (IOU) exceeds the threshold.

Finally, we use the cross-entropy loss function $\mathcal{L}_{\text{class}} = -\sum_{i=1}^{M} \left(c_i \log c_i' + (1 - c_i) \log \left(1 - c_i'\right)\right)$ to calculate the category loss of the object, and then the total loss function is as follows:

$$\mathcal{L} = \mathcal{L}_{object} + \mathcal{L}_{loc} + \mathcal{L}_{loc}[f] + \mathcal{L}_{\text{class}} \tag{15}$$

4 Experiments

For small object detection tasks, GDUT-HWD[2], Visdrone-2019[3], etc., are available public benchmark datasets. To fully verify the effectiveness of the IT-transformer, we compare 8 typical algorithms in the above two datasets. In addition, we have built our own dataset of ground objects in the military field and conducted comparative experiments in it. The distribution of objects of each scale in the three datasets is shown in Fig. 5, and it can be seen that the Armored Vehicle dataset we collected and sorted out has similar instance distribution characteristics to the other two, which are composed of small and medium-sized objects, which has great detection difficulty.

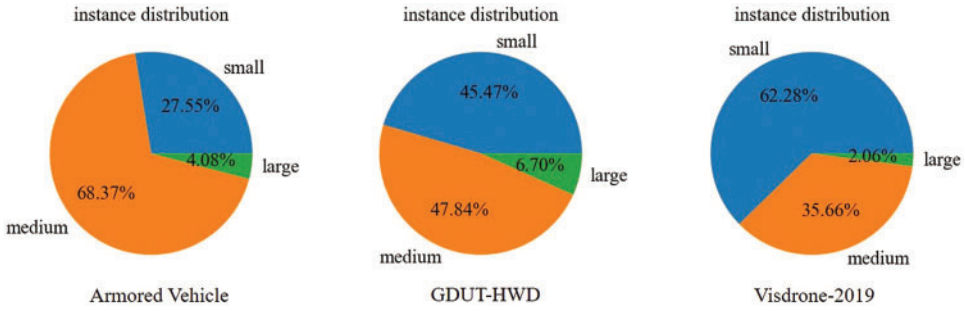

Figure 5: The distribution of three used datasets

4.1 Datasets

Armored Vehicle: We collected, organized, and annotated 4975 images through online searches and local shooting. In the dataset, there are 10250 labeled boxes, and we use 3920 as the training set, which contains 8022 instances, and the remaining 1057 as the validation set, containing 2210 instances. There is only one type of object in the dataset, and its size distribution is shown in Fig. 5. We label

[2] https://github.com/wujixiu/helmet-detection/tree/master/hardhatwearing-detection
[3] https://github.com/VisDrone/VisDrone-Dataset

the data in a coco format to ensure that it can be used directly for multiple training architectures. The difference is that in the Armored Vehicle dataset, the object's viewing angle, distance, environment, scale, weather, and other characteristics are more complex, making it more difficult to detect small objects.

GDUT-HWD [38]: This is a very common hard hat detection dataset in industrial scenarios, containing 3174 training images, consisting of 5 types of labeled boxes, which is a lightweight benchmark for small object detection.

Visdrone-2019 [9]: This is a small object dataset of large scenes from an aerial perspective, consisting of 10209 images and 2.6 million annotation boxes, which can be used to test the detection performance of the model on small objects, and at the same time can test the efficiency of model reasoning. Due to its large image size, we divide each image into 4 non-overlapping subplots concerning [39].

4.2 Metrics

We select mean average precision (mAP), APs, APm, and APl commonly used in object detection tasks as evaluation indicators and precision and recall are the basis for calculating each value. AP is the area around the precision-recall (P-R) curve and the coordinate axis, and its calculation formula is:

$$AP = \int_0^1 P(x)dx \tag{16}$$

For datasets with multiple class objects, mAP is the average of APs across all classes, expressed as:

$$mAP = \frac{1}{c}\sum_{i=1}^{c} AP_i \tag{17}$$

APs refer to objects with a size of less than 32×32, and in the same way, APm and APl correspond to 96×96, and 128×128, respectively. In the course of the experiment, we also calculate the evaluation results of mAP50 concerning the practice of [1,9], which means the mAP is calculated when IOU = 0.5.

4.3 Settings

All experiments in this paper are based on the mmdetection architecture, which ensures the fairness and reproducibility of the test. In the experimental process, we adopt a single-scale training strategy, and the input image size is uniformly limited to 640×640 (the Visdrone-2019 dataset is set to 1280×960), and only random flipping is used for data augmentation in the data preprocessing stage. For the learning rate and the number of detection heads, we determine through a grid search, which is described in Section 4.6. In the following experiment, learning rate (lr) is 4E-2 and N is 8 in the following experiment. Other parameters refer to the default settings of mmdetection.

4.4 Results in the Armored Vehicle Dataset

The experimental results are shown in Table 1, from which it can be seen that the improved IT-Cascade-RCNN model achieves higher detection accuracy. The longitudinal comparison shows that IT-Cascade-RCNN improves 14.8 mAP compared with the typical first-order detection model YOLOx and 12.8 mAP compared with the typical second-order model Sparse. In particular, IT-Cascade-RCNN also achieved better results than DINO and DiffusionDET which based on diffusion

models, exceeded 4.5 and 4.4 mAP. Furthermore, the IT-transformer also surpassed another attention-based model, named adaptive training sample selection (ATSS) [40], 6 mAP in a word. It is worth noting that under the AP50, although the DINO and DiffusionDET model achieved higher detection results, the performances have not been well extended to other threshold conditions, and they failed to balance between various early warning restrictions, object detection accuracy, and false alarm rate. In contrast, IT-Cascade-RCNN provides better results at various IOU thresholds. Further, we find that the accuracy of IT-transformer for large objects has decreased, because we have integrated global features and local features in the middle features of IT-transformer, resulting in the introduction of interference in some environmental information brought by local semantic features, resulting in a smaller APl.

Table 1: The metrics in armored vehicle dataset

Model	mAP	AP50	APs	APm	APl
Retina	44.5	84.1	24.2	51.2	54.2
Sparse RCNN	44.6	81.6	30.1	50.4	54.8
YOLOv3	42.7	82.4	38	49.9	59
YOLOx	42.6	80.1	29.5	49.6	47.9
ATSS	51.4	87.3	34.4	57.1	69.1
DINO	52.9	89	36.1	58.8	69.7
DiffusionDET	53	89	40.2	57	62.1
Cascade RCNN	54.7	87.3	38.5	60.1	70.6
+IT-transformer	57.4	88	41	63.5	68.4

Fig. 6 shows the visualization of the detection results of each model in the Armored Vehicle dataset. Fig. 6 can more clearly show the detection effect of the five models, from which we can see that in the first line of images, Retina, YOLOx and DINO have serious false alarm problems, identifying non-existent areas as objects, while Faster RCNN fails to detect objects at all, and the improved model with cross-attention mechanism correctly detects objects; In the second line, Retina, Faster RCNN, and YOLOx also have the problem of missing detection, although DINO detects all objects, but the precision measurement accuracy is not as high as the improved model; Similarly, when the object in the third row is partially occluded, although the first three models are correctly positioned to the object, the detection accuracy does not reach a higher level, but unfortunately, DINO missed an object at this time; The fourth line shows the level difference between the models more vividly, when the object is obscured by smoke and dust, resulting in the object feature being disturbed, Retina, YOLOx and DINO fail to detect the object, and the Faster RCNN obtains less accurate detection results, compared to the improved Cascade RCNN model showing accurate results.

4.5 Results in the GDUT-HWD Dataset

We also perform experiments in lightweight GDUT-HWD datasets to test the ability of the IT-transformer to deal with small object detection in industrial scenarios, and the experimental results are shown in Table 2. From this, we found that IT-Cascade-RCNN also showed good performance advantages, improving by 14.1 mAP compared with the typical first-order detection model YOLOx, 16.9 mAP compared with the second-order detection model represented by sparse, and 13.3 mAP higher than the DINO based diffusion model. Among the more challenging small-scale object

detection results, IT-Cascade-RCNN also achieved the highest detection accuracy of 34.3, which is 2.1 higher than the benchmark Cascade-RCNN. In summary, the results show that IT-transformer can effectively improve the detection performance of the model.

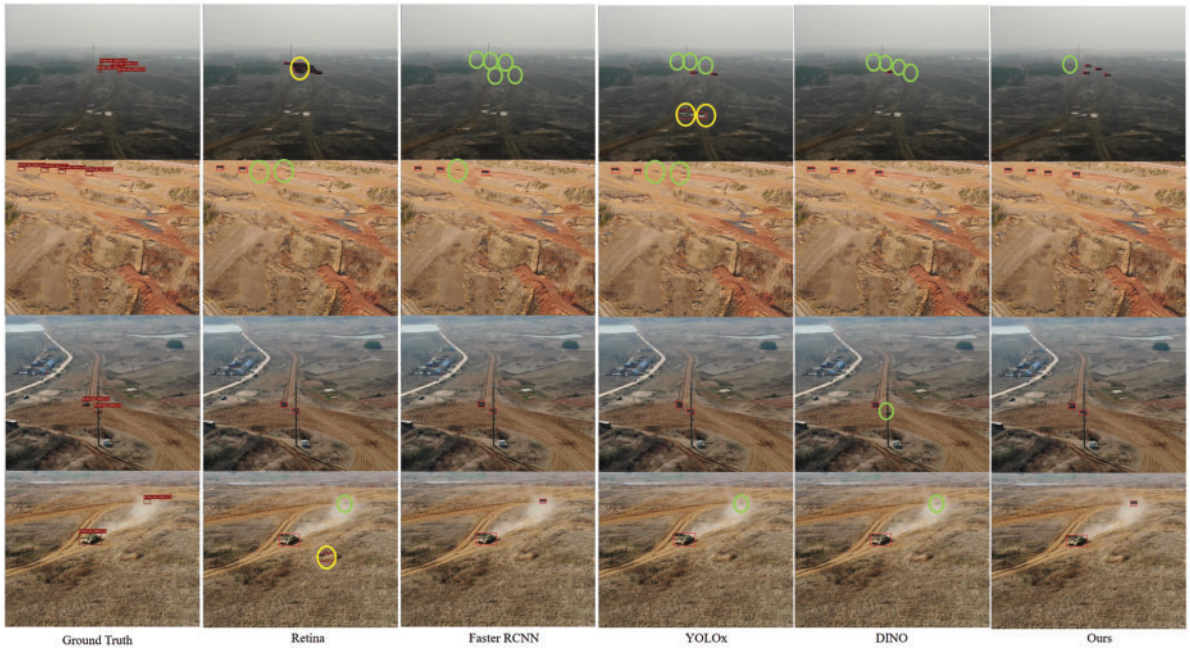

Figure 6: Detection results (green circles indicate missed detections and yellow circles indicate false detections)

Table 2: The metrics in GDUT-HWD dataset

Model	mAP	AP50	APs	APm	APl
Retina	34.9	59.1	18.2	48.5	58
Sparse RCNN	33.4	56.5	20.3	44.5	53.4
YOLOx	36.2	67.9	22.2	47.9	52.8
YOLOv5	40.5	75.6	26.8	52	57.1
YOLOv7	34.5	70.8	24.1	47.1	37.2
DINO	37	69.8	19.3	50.1	65.6
Cascade RCNN	49.2	79.7	32.2	62.5	70.8
+IT-transformer	50.3	80.8	34.3	63.1	70.6

Fig. 7 is the visualization of some model detection results, and it is found that Retina, Faster RCNN, YOLOx, and DINO have serious missed detection problems, and none of them can detect the object marked by the green circle in the Fig. 7. At the same time, Retina and Faster RCNN also have the problem of false detection, and they misjudge the object category marked by the yellow circle; Finally, Faster RCNN also has the problem of duplicate detection, and the object marked by the blue circle in the duplicate detection figure is repeated; Among the detected objects, the improved Cascade RCNN model has a higher degree of confidence. On the whole, the model improved by the cross-transformer

shows better performance, which effectively improves the detection accuracy of the model for small objects.

Figure 7: Detection results (green circles indicate missed detections, yellow circles indicate false detections and blue circles indicate retests)

4.6 Ablation Experiments

The IT-transformers we design are mainly affected by factors such as learning rate, normalization layer, number of detection heads, etc., to test their impact on precision measurement accuracy more reliably, we carry out ablation experiments on them separately in this part.

4.6.1 The Impact of lr

We use the grid search method to test the influence of different learning rates on the detection accuracy of the model. During the experiment, we sampled 15 learning rates from 1E-3 to 5E-2 and experimented in the GDUT-HWD dataset, and the relevant results are shown in Fig. 8.

Observing the experimental results in Fig. 8, it is first confirmed that the difference in learning rate does have a great impact on the detection accuracy of the model, for example, with the increase of the learning rate, the detection accuracy of the model shows an upward trend. Furthermore, it can be seen that when lr is set to 4E-2, the model achieves the highest results, reaching 48.9 mAP. Therefore, in the full text, we set the lr to 4E-2.

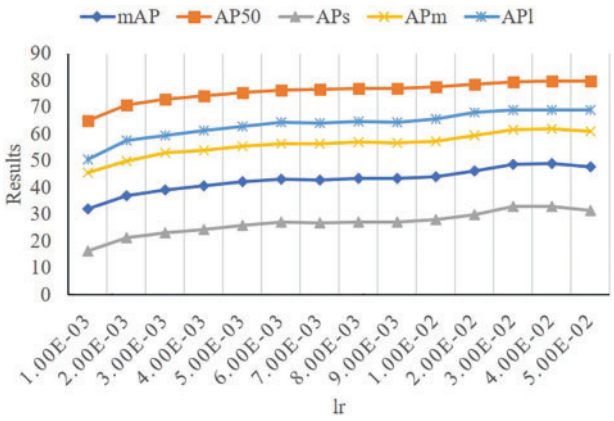

Figure 8: The impact of lr

4.6.2 The Impact of Head Number

The bull attention mechanism determines how many angles the interrelationships between features need to be extracted, and we know that the number of attention heads is not as many as possible, and vice versa. Our ablation experiments confirmed this as well. As shown in Table 3, it can be seen that when the number of detection heads is small, an effective attention mask cannot be generated, resulting in an interaction between feature maps, which cannot provide more effective feature information for the detection head, and when the number of attention heads is too large, too much redundant information will be introduced, which will also weaken the expression ability of features. From the experimental results, when the head number is 8, the model performs better.

Table 3: The results of different numbers of head

Head	mAP	AP50	APs	APm	APl
2	48.1	79.2	32.1	61.2	68.3
4	48.1	79.2	31.8	61.2	69.3
6	48	79.5	31.4	61	69.2
8	48.9	79.8	33	62	69.1
10	48.3	79.3	32	61.6	69
12	48.6	79.6	33.2	61	68.8

As shown in Table 4, we also experiment with the normalization layer in the transformer. The results show that the model performs better when the normalization layer is not used. We believe that the possible reason is that the normalization operation affects the representation of the middle-layer features, and when the normalization operation is carried out, the features are compulsorily concentrated on some prior knowledge, which weakens the ability of the model to rely on its ability to induct effective bias, drowns the middle-layer features that have a direct impact on the detection results, and causes the model detection accuracy to decline. On the contrary, by reducing the constraints of prior knowledge on the model learning process, and more through self-learning guidance, the model can more effectively learn the universal characteristics of different object features, to achieve more accurate detection in the detection process.

Table 4: The impact of normalized layer

Norm	mAP	AP50	APs	APm	APl
LN	48.8	79.9	32.9	61.9	69.1
BN	48.4	79.8	32.2	61.5	68.8
None	48.9	79.8	33	62	69.1

4.6.3 The Impact of Kernel Size

IT-transformer better integrates the ability of convolutional modules to obtain local semantic features. In fact, local semantic features can provide more environment and reference information for the identification of small objects, and help achieve accurate classification and positioning. In order to determine a more suitable field of view, in this part, we conducted a comparative experiment on the size of the convolution kernel in the Armored Vehicle dataset, and the results are shown in Table 5.

Table 5: The impact of kernel size (ensure that the size of the output feature map remains unchanged)

Kernel size			mAP	AP50	APs	APm	APl	Parameters (G)	TFlOPS
Size	Stride	Padding							
1	1	0	55.5	87.3	39.1	61.8	71.9	0.0774	0.114
3	1	1	56.9	87.1	40.1	63.1	70	0.0879	0.171
5	1	2	56.9	87.1	40.7	62.9	69.4	0.109	0.285
7	1	3	57.4	88	41	63.5	68.4	0.14	0.457
9	1	4	56.6	87	40.3	62.7	65.7	0.182	0.686

It can be seen from Table 5 that the change in convolution kernel size has a significant impact on IT-transformer, in which with the increase of convolution kernel size, the receptive field of intermediate feature fusion also increases, providing the validity of intermediate layer features, which is reflected in the detection results is the steady improvement of various indicators, such as kernel size is 7, reaching a maximum value of 57.4, in which the APs reaches 41; However, with the further increase of the convolution kernel size, such as kernel size 9, too many environmental features are integrated into the middle layer features, which interferes with the utilization effect of the middle layer features, resulting in a downward trend in object detection accuracy. At the same time, it is obvious that as the size of the convolution kernel increases, the number of parameters of the model will increase simultaneously, and the computing power expenditure will increase, but it is worth the effort to improve the accuracy of object detection.

4.6.4 The Result of Plug-and-Play

As we mentioned earlier, IT-transformer has plug-and-play features and can significantly improve accuracy. In this regard, we selected typical first-stage and second-stage detection models such as Retina, Faster RCNN, and Cascade RCNN in the GDUT-HWD, Armored Vehicle, and Visdrone-2019 for experiments. The results are shown in Table 6. After adding IT-transformer, the baseline model has achieved significant performance improvements, such as in GDUT-HWD, with the IT-transformer, the mAP of Faster RCNN and Cascade RCNN increased by 8.8 and 1.1, respectively; meanwhile, in the Armored Vehicle dataset the accuracy of Retina is improved by 4.1 mAP and the accuracy of small objects by 20.56%, compared with 6.18% of APm and 0.95% of APl. IT-transformer's effect on model performance improvement can also be reflected in the Visdrone-2019.

Table 6: The results of plug-and-play

Dataset	Model	mAP	AP50	APs	APm	APl
GDUT-HWD	Faster RCNN	40.1	73	23.2	53.6	62.4
	+IT-transformer	48.9	79.4	33	62	69.1
	Cascade RCNN	49.2	79.7	32.2	62.5	70.8
	+IT-transformer	50.3	80.8	34.3	63.1	70.6
Armored vehicle	Faster RCNN	51.7	86.4	35.6	57.8	67.2
	+IT-transformer	52.8	86.3	36	59.1	68.1
	Retina	44.7	82.8	24.8	51.8	63.1
	+IT-transformer	48.8	85.4	29.9	55	63.7
	Cascade RCNN	54.7	87.3	38.5	60.1	70.6
	+IT-transformer	57.4	88	41	63.5	68.4
Visdrone-2019	Faster RCNN	18.7	32.8	9.7	30.7	48.6
	+IT-transformer	19.8	33.6	10.6	32.4	48.5
	Cascade RCNN	20.7	34	10.6	33.5	49
	+IT-transformer	22	35.5	11.9	35.1	49.8

Experimental results show that the IT-transformer designed in this paper does exhibit good plug-and-play and can be directly used in many types of benchmark models. Fig. 9 is the test results in the Visdrone-2019 dataset, and we test the effect of cross-transformer addition on the detection effect of the Cascade RCNN model before and after the addition of the cross-transformer. It can be seen that the addition of cross-transformers significantly improves the false detection and missed detection of Cascade RCNN.

Figure 9: The result in Visdrone-2019 (yellow circle indicates false detection, green circle represents missed detection)

5 Conclusion

For the challenging small object detection task, we first analyze and sort out the existing solution ideas, summarize them into four basic methods, and then, combined them with the current mainstream attention mechanism, based on the traditional transformer model, from the perspective of compressing the number of model parameters and strengthening the coupling of middle-layer features, we design a cross-K-value transformer model with a double-branch structure, and at the same time, we apply the idea of multi-head attention to the processing process of channel attention masking. By experimenting with the self-built Armored Vehicle dataset and 2 additional benchmarks, the improved Cascade RCNN model based on cross-transformer was verified and a higher detection level was achieved. Finally, by combining the cross-transformer with the existing first-order and second-order detection models, the ablation experiment confirms that the cross-transformer has good plug-and-play performance and can effectively improve the detection results of each baseline. In addition, we also collected and collated an Armored Vehicle dataset containing a class of military ground objects to provide data support for related research.

Acknowledgement: None.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Qinzhao Wang; data collection, analysis, and interpretation of results: Jian Wei; draft

manuscript preparation: Zixu Zhao. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data will be available on request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Xu, J. Gu, Y. Hua and Y. Liu, "DKTNet: Dual-key transformer network for small object detection," *Neurocomputing*, vol. 525, pp. 29–41, 2023.

[2] C. Xu, J. Ding, J. Wang, W. Yang, H. Yu *et al.,* "Dynamic coarse-to-fine learning for oriented tiny object detection," in *Proc. of CVPR*, Vancouver, Canada, pp. 7318–7328, 2023.

[3] J. Fan and X. Liu, "Multi background island bird detection based on faster R-CNN," *Cybernetics and Systems*, vol. 52, pp. 26–35, 2021.

[4] C. Guo, B. Fan, Q. Zhang, S. Xiang and C. Pan, "AugFPN: Improving multi-scale feature learning for object detection," in *Proc. of CVPR*, Seattle, WA, USA, pp. 12595–12604, 2020.

[5] G. Ghiasi, T. Y. Lin and Q. V. Le, "NASFPN: Learning scalable feature pyramid architecture for object detection," in *Proc. of CVPR*, Long Beach, CA, USA, pp. 7036–7045, 2019.

[6] Y. Zhu, Q. Zhou, N. Liu, Z. Xu, Z. Ou *et al.,* "ScaleKD: Distilling scale-aware knowledge in small object detector," in *Proc. of CVPR*, Vancouver, Canada, pp. 19723–19732, 2023.

[7] G. Zhao, W. Ge and Y. Yu, "GraphFPN: Graph feature pyramid network for object detection," in *Proc. of ICCV*, Montreal, Canada, pp. 2763–2772, 2021.

[8] M. Hu, Y. Li, L. Fang and S. Wang, "A^2-FPN: Attention aggregation-based feature pyramid network for instance segmentation," in *Proc. of CVPR*, Nashville, TN, USA, pp. 15343–15352, 2021.

[9] C. Yang, Z. Huang and N. Wang, "QueryDet: Cascaded sparse query for accelerating high-resolution small object detection," in *Proc. of CVPR*, New Orleans, LA, USA, pp. 13668–13677, 2022.

[10] T. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal loss for dense object detection," in *Proc. of ICCV*, Venice, Italy, pp. 2980–2988, 2017.

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

[12] Z. Ge, S. Liu, F. Wang, Z. Li and J. Sun, "YOLOX: Exceeding yolo series in 2021," arXiv preprint arXiv:2107.08430, 2021.

[13] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang *et al.,* "Deformable convolutional networks," in *Proc. of ICCV*, Venice, Italy, pp. 764–773, 2017.

[14] X. Zhu, H. Hu, S. Lin and J. Dai, "Deformable ConvNets V2: More deformable better results," in *Proc. of CVPR*, Long Beach, CA, USA, pp. 9308–9316, 2019.

[15] K. Zhang, T. Li, S. Shen, B. Liu, J. Chen *et al.,* "Adaptive graph convolutional network with attention graph clustering for co-saliency detection," in *Proc. of CVPR*, Seattle, WA, USA, pp. 9050–9059, 2020.

[16] G. Peng, J. Zhang, H. Li and D. Wang, "Attentional pyramid pooling of salient visual residuals for place recognition," in *Proc. of ICCV*, Montreal, Canada, pp. 865–874, 2021.

[17] X. Zhu, S. Lyu, X. Wang and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. of ICCV*, Montreal, Canada, pp. 2778–2788, 2021.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.,* "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 34–45, 2017.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai *et al.,* "An image is worth 16×16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.

[20] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei *et al.,* "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. of ICCV*, Montreal, Canada, pp. 10012–10022, 2021.
[21] X. Zhu, W. Su, L. Lu, B. Li, X. Wang *et al.,* "Deformable DETR: Deformable transformers for end-to-end object detection," arXiv preprint arXiv:2010.04159, 2020.
[22] A. Bochkovskiy, C. Wang and H. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, 2020.
[23] J. Redmon and A. Farhadi, "YOLO9000: Better faster stronger," in *Proc. of CVPR*, Honolulu, HI, USA, pp. 7263–7271, 2017.
[24] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed *et al.,* "SSD: Single shot multibox detector," in *Proc. of ECCV*, Amsterdam, Netherlands, pp. 21–37, 2016.
[25] T. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal loss for dense object detection," in *Proc. of ICVV*, Venice, Italy, pp. 2980–2988, 2017.
[26] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, pp. 91–99, 2015.
[27] Z. Cai and N. Vasconcelos, "Cascade R-CNN: High quality object detection and instance segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 1483–1498, 2019.
[28] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su *et al.,* "DINO: Detr with improved denoising anchor boxes for end-to-end object detection," arXiv Preprint arXiv:2203.03605, 2022.
[29] S. Chen, P. Sun, Y. Song and P. Luo, "DiffusionDet: Diffusion model for object detection," arXiv preprint arXiv:2211.09788, 2022.
[30] I. Ethem Hamamci, S. Er, E. Simsar, A. Sekuboyina, M. Gundogar *et al.,* "Diffusion-based hierarchical multi-label object detection to analyze panoramic dental X-rays," arXiv preprint arXiv: 2303.06500, 2023.
[31] S. Nag, X. Zhu, J. Deng, Y. Song and T. Xiang, "DiffTAD: Temporal action detection with proposal denoising diffusion," arXiv preprint arXiv:2303.14863, 2023.
[32] S. Y. Liu, H. Y. Guo, J. G. Hu, X. Zhao, C. T. Zhao *et al.,* "A novel data augmentation scheme for pedestrian detection with attribute preserving GAN," *Neurocomputing*, vol. 401, pp. 123–132, 2020.
[33] S. Liang, H. Wu, L. Zhen, Q. Hua, S. Garg *et al.,* "Edge YOLO: Real-time intelligent object detection system based on edge-cloud cooperation in autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, pp. 25345–25360, 2022.
[34] B. Bosquet, D. Cores, L. Seidenari, V. M. Brea, M. Mucientes *et al.,* "A full data augmentation pipeline for small object detection based on generative adversarial networks," *Pattern Recognition*, vol. 133, pp. 108998–109007, 2023.
[35] S. Ji, Q. Ling and F. Han, "An improved algorithm for small object detection based on YOLO v4 and multi-scale contextual information," *Computers and Electrical Engineering*, vol. 105, pp. 108490–108499, 2023.
[36] X. Yang, J. R. Yang, J. C. Yan, Y. Zhang, T. F. Zhang *et al.,* "SCRDet: Towards more robust detection for small, cluttered and rotated objects," in *Proc. of ICCV*, Seoul, Korea (South), pp. 8232–8241, 2019.
[37] V. Vidit and M. Salzmann, "Attention-based domain adaptation for single-stage detectors," *Machine Vision and Applications*, vol. 33, pp. 65–74, 2022.
[38] J. Wu, N. Cai, W. Chen, H. Wang and G. Wang, "Automatic detection of hardhats worn by construction personnel: A deep learning approach and benchmark dataset," *Automation in Construction*, vol. 106, pp. 102894–102915, 2019.
[39] Z. Liu, G. Gao, L. Sun and Z. Fang, "HRDNet: High-resolution detection network for small objects," in *Proc. of ICME*, Montreal, Canada, pp. 1–6, 2021.
[40] S. Zhang, C. Chi, Y. Yao, Z. Lei and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. of CVPR*, Seattle, WA, USA, pp. 9759–9768, 2020.