Hindawi
Security and Communication Networks
Volume 2023, Article ID 9868642, 1 page
https://doi.org/10.1155/2023/9868642

Retraction

Retracted: X-Ray Small Target Security Inspection Based on TB-YOLOv5

Security and Communication Networks

Received 5 December 2023; Accepted 5 December 2023; Published 6 December 2023

This article has been retracted by Hindawi, as publisher, following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of systematic manipulation of the publication and peer-review process. We cannot, therefore, vouch for the reliability or integrity of this article.

Please note that this notice is intended solely to alert readers that the peer-review process of this article has been compromised.

Wiley and Hindawi regret that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] M. Wang, Y. Zhu, Y. Liu, and H. Deng, “X-Ray Small Target Security Inspection Based on TB-YOLOv5,” *Security and Communication Networks*, vol. 2022, Article ID 2050793, 16 pages, 2022.

Hindawi
Security and Communication Networks
Volume 2022, Article ID 2050793, 16 pages
https://doi.org/10.1155/2022/2050793

Research Article

X-Ray Small Target Security Inspection Based on TB-YOLOv5

Muchen Wang[1,2] **Yueming Zhu**[1] **Yongkang Liu**[2] **and Huiping Deng**[1]

[1]*School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan, Hubei 430081, China*
[2]*Wuhan University of Science and Technology, School of Materials and Metallurgy, Wuhan, Hubei 430081, China*

Correspondence should be addressed to Huiping Deng; denghuiping@wust.edu.cn

Received 2 June 2022; Revised 24 June 2022; Accepted 28 June 2022; Published 2 August 2022

Academic Editor: Mohammad Ayoub Khan

Security inspection is extremely important for the safety of public places. In this research, we are trying to propose a novel algorithm and investigated theoretically in the X-ray dataset, which can optimize the relative low detection accuracy and the latent omission detection of smaller objects when using You Only Look Once version 5 (YOLOv5). For one side, the transform detection network is selected to be added at the bottom layer of backbone structure to avoid the loss of useful information during sequential calculation. On another side, we attempt to adjust the existing PANet structural elements of the model, including their connections and other related parameters to improve the detection performance. We integrate an efficient BiFPN with the CA mechanism, which can enhance feature extraction, and named it attention-BiFPN. Experimental consequences demonstrate that the detection accuracy of the proposed model, which we name "TB-YOLOv5," has obvious advantages in check performance compared with the mainstream one-stage object detection models. Meanwhile, compared with YOLOv5, the data results display an improvement of up to 14.9%, and the average precision at 0.5 IOU even reached 23.4% higher in the region of small object detection. Our purpose was to explore the potential of changing a popular detection algorithm such as YOLO to address specific tasks and provide insights on how specialized adjustments can influence the detection of small objects. Our work can supply an effective method of enhancing the performance of X-ray security inspection and show promising potential for deep learning in related fields.

1. Introduction

Security inspection plays an indispensable role in safeguarding public occasions from security threats of terrorism. With the development of population density in metropolitan transportation hubs, it is increasingly critical to quickly, automatically, and accurately identify prohibited products in trunks or packages. The disadvantages of traditional manual checking methods caused by subjectivity and visual fatigue after prolonged detection result in poor accuracy judging contraband goods. Consequently, the conventional detection way is no more adapted to the modern high-speed lifestyle and is not satisfying humans' higher demand for safety, so advanced techniques are emerging. Electromagnetic ultrasonic [1] and other detection measures have been used earlier. However, the high requirements of advanced equipment make those technologies impractical to be put into widespread use. Recently, under the circumstances that artificial intelligence technology has a significant breakthrough, especially in convolutional neural networks, the security inspection method based on deep learning to find and recognize target objects in X-ray images [2–4] comes into being.

To effectively limit the prohibited dangerous articles and increase the accuracy of image detection results of different products in luggage considering different conditions, it becomes indispensable for people to optimize the existing related technology. Different from natural images and X-ray detection in other scenes [5], the articles in the suitcase are stacked randomly and overlap with each other seriously, giving rise to the finite resolution ratio and context information available to the model in the process of algorithm design. Meanwhile, detecting small target objects is always a more challenging task.

Although great efforts have been made to improve the detection ability of smaller objects [6], most of them only focus on guiding the processing of specific areas of the picture [7,8], or on improving the resolution ratio of the image, which come at the expense of detecting speed. However, considering the development of computer vision algorithms, many existing object detection networks show outstanding detection effects in different areas. Among them, the detecting networks based on deep learning can be broadly divided into two types—two-stage algorithm and one-stage algorithm. Two-stage algorithms include regional convolutional neural network (R-CNN) [9], faster R-CNN [10], and spatial pyramid pooling net (SPP-NET) [11]. Two-stage object detection algorithms are dependent on reorganization box and classifier. The bounding box is first searched to generate a series of candidate regions. The features are extracted from the original image by convolutional neural network to locate and classify them. Despite the detection accuracy of the two-stage algorithm often showing obvious advantages, the problem of slow detecting speed still exists due to the complex network structure, which means two-stage is not suitable for real-time applications such as security inspection and automatic driving. Therefore, experts further developed a series of one-stage target detection algorithms [12], including single shot multibox detector (SSD) [13], efficient object detection (EfficientDet) [14], and You Only Look Once (YOLO) [15]. Those kinds of algorithms define the process of target detection as a regression problem, and the target objection can be directly located and classified through the regression model. Compared with others, one-stage algorithm has a relatively faster-checking efficiency, which can be better used in a system with high demand for speed. Among all kinds of one-stage algorithms, version 5 of YOLO (YOLOv5) is a popular target detector [16], which earns a reputation for its high performance and running speed. The excellent function of YOLOv5 can be attributed to its flexible structure and can be decomposed, adjusted, or built on many widely accessible platforms. However, many systems attempt to apply the YOLOv5 architecture to optimize, mainly relying on adjusting specific parameters or enlarging training datasets to improve performance [17], rather than changing network structure to modify the model itself when encountering the detection of objection with insufficient feature information. Therefore, those measures cannot effectively ameliorate the performance of smaller object detection. Miao et al. proposed a class-balanced hierarchical thinning (CHR) to solve the problem of information loss caused by overlapping image data in X-ray security check and designed a class-balanced loss function to minimize noise introduced by negative samples [2]. However, this method still fails to solve the problem of insufficient feature information extracted from small objects and needs a huge dataset as support. Benjumea et al. have enhanced the algorithm ability to extract small target objects by modifying the neck structure, which is called YOLO-Z [18], and the results exhibit better detection performance in automatic driving, though there is still much room for the improvement to grantee that this method can be used in X-ray security inspection.

Our study aimed to continue using the networks based on YOLOv5 for X-ray security detection and do some improvements to solve the problem that the existing algorithms' disadvantage is to extract the information of small target features. The proposed algorithm is named "TB-YOLOv5," partly referencing the existing YOLO-Z algorithm. It is improved by adding a transformer module to the bottom layer of backbone and substituting the PANet structure in the neck part with attention-BiFPN, which integrates BiFPN structure with coordinate attention (CA) module to enhance the collection of characteristic information. The transformer model has been widely used in object detection these years [19, 20] and shows advantages in simplicity and efficiency compared with traditional CNN. CA also demonstrates the speed superiority in image processing region [21, 22]. The investigation proves that the TB-YOLOv5 can develop target detection performance in X-ray security inspection. Compared with the traditional YOLOv5 algorithm, the results of TB-YOLOv5 are significantly improved, especially for the small objects, which demonstrate great significance to the development of X-ray security inspection in public places. Furthermore, we believe that this approach could provide new avenues for realizing various intelligent industrial applications. The following part of this article will be divided into three sections: (1) in the second section, we will firstly introduce the conventional YOLOv5 structure and some relevant components in our methods. (2) Then, we will investigate the detection consequences trained by YOLOv5 and TB-YOLOv5 to demonstrate the advantages of its performance. Meanwhile, the third section will give the analysis and comparison of the experimental results and checking inference. (3) In the last section, we will conclude our research and propose the prospect of the algorithm.

2. Materials and Methods

2.1. Materials and Samples. X-ray detection machine is the most common security inspection technology applied by the relevant departments in China at present, widely seen in urban rail transit, railway, airport, key venues, logistics delivery, and other scenes. Using artificial intelligence technology to assist front-line security inspectors in making judgments can effectively reduce the possibility of missing reports caused by fatigue or inattention, but in the actual scene, due to the diversity of objects, imaging angles, occlusion, and other problems, there are certain challenges in the development of the algorithm. The dataset of this study comes from iFLYTEK AI Open Challenge of USTC. The format of the dataset is PASCAL VOC2007 [23], and the labeled files contain 12 categories, including knife, scissors, sharp tools, expandable baton, small glass bottle, electric baton, plastic beverage bottle, plastic bottle with a nozzle, electronic equipment, battery, seal, umbrella, and a total of 4,000 pictures. Figure 1 shows the number of items for each category in the dataset. MSCOCO dataset is generally used for evaluating the size of the targets. For an object with an area less than a specified pixel value, MSCOCO considers it as a small object. Based on this criterion, we defined scissors, knife, small glass bottle, battery, and seal as the small objects in our later research because they have an obviously smaller pixel than others in most images.

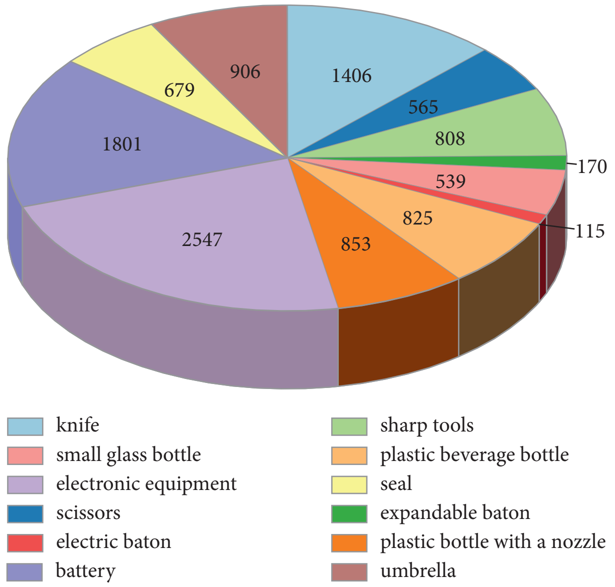


Figure 1: Pie chart of each target objection's number in the presented dataset with twelve categories.

2.2. Detection Principle

2.2.1. One Stage and YOLO Detection Principle. Two-stage algorithms, such as R–CNN, use the region suggestion method to first generate potential bounding boxes in the image and then run classifiers on these suggested boxes. After classification, the bounding box is refined by post-processing, duplicate detection is eliminated, and the box is rescaled based on other objects in the scene [13]. These complex pipes limit the running speed and are difficult to optimize because each individual component must be trained separately.

YOLOv5 is the highest version of existing YOLO until today. YOLO is the first algorithm to extend the CNN recognition idea to target detection, transforming target detection into a regression problem. Boundary regression in model classification is also called one-stage detection [24]. Each convolution network of YOLO simultaneously predicts multiple bounding boxes and their class probabilities. YOLO trains the whole image and directly optimizes the detection performance. Compared with traditional target detection methods, this unified model is fast in target detection. The reason is that this method treats detection as a regression problem, so it does not need complex pipes and only needs to run the neural network on the new image during the test to predict the detection results. The original YOLOv2 uses BN as regularization to accelerate convergence and avoid overfitting, and the BN layer and ReLU are connected to each convolution layer. To improve the detection performance of YOLO, Darknet-53, Anchor, FPN, and other structures were added to YOLOv3 proposed in 2018. Among them, using the residual component of ResNet for reference, the network can be built deeper, the model capacity is larger, and the feature learning ability is stronger. Darknet-53 makes the network easier to train and faster to merge through residual connections. Conv is used to implement downsampling to reduce the negative gradient effect caused by the pool. Darknet-53 uses a convolution operation with a step size of 2. YOLOv3 uses logistic regression to score the anchor boxes objectively, leaving some anchors with low scores before prediction. This dramatically reduces the amount of calculation.

In a nut, YOLO's average accuracy is higher than other real-time systems. Therefore, we have reason to believe that the YOLO algorithm has a bright application prospect in the field of real-time target detection.

2.2.2. YOLOv5 Detection Principle. In this study, we intend to improve the existing YOLOv5 model to increase the detection capability of small target objects for further application in X-ray security screening recognition. The detection of small target objects has always been a complex field in deep learning due to insufficient relevant information. At the same time, with the increase in people's travel frequency, security identification will increase the requirement of detection speed, which will dramatically increase the complexity of target detection. To optimize the YOLOv5 algorithm, it is indispensable for us to understand its foundation and the current state of development.

The structure of YOLOv5 algorithm is similar to YOLOv4 and further improved on the basis of it. The network structure of traditional YOLOv5 can be divided into four parts: input, backbone, neck network, and prediction (output), and the complete framework of YOLOv5-s, for example, is shown in Figures 1 and 2.

The main functional structures on the input side are mosaic data enhancement, adaptive anchor frame calculation, and adaptive image scaling. The input image resolutions of the YOLO algorithm are generally 416 × pixels × 416pixels, 512 × pixels × 512 pixels, and 608 × pixels × 608pixels. Experiments show that if the input resolution is higher, the model's performance will improve. The data enhancement adopts the mosaic approach, which takes scaling, cropping, and random arrangement to the dataset, thus increasing the complexity of the dataset, making the dataset gets greatly enriched, and adding many small targets, which makes the robustness of trained model much better. Meanwhile, the number of images read from the dataset for training is smaller in each batch, which reduces the memory usage of GPU. In the YOLO series of detection algorithms, the default anchor frame length and width are initially invented for different targets. A prediction frame is an output based on the initially set anchor frame when the dataset is training. The difference between the labeled real frame and the prediction frame is calculated, and then, the parameters in the network structure are iteratively updated in the reverse direction. In the YOLOv5 algorithm, this feature is embedded in the structure, and the best anchor frame values are computed adaptively for different training sets at each learning. The adaptive image scaling in the target detection algorithm is decided by the length and width of images in the respective dataset. The original image is first scaled into a uniform standard size and then sent to the detection network after processing. When YOLOv5 zooms

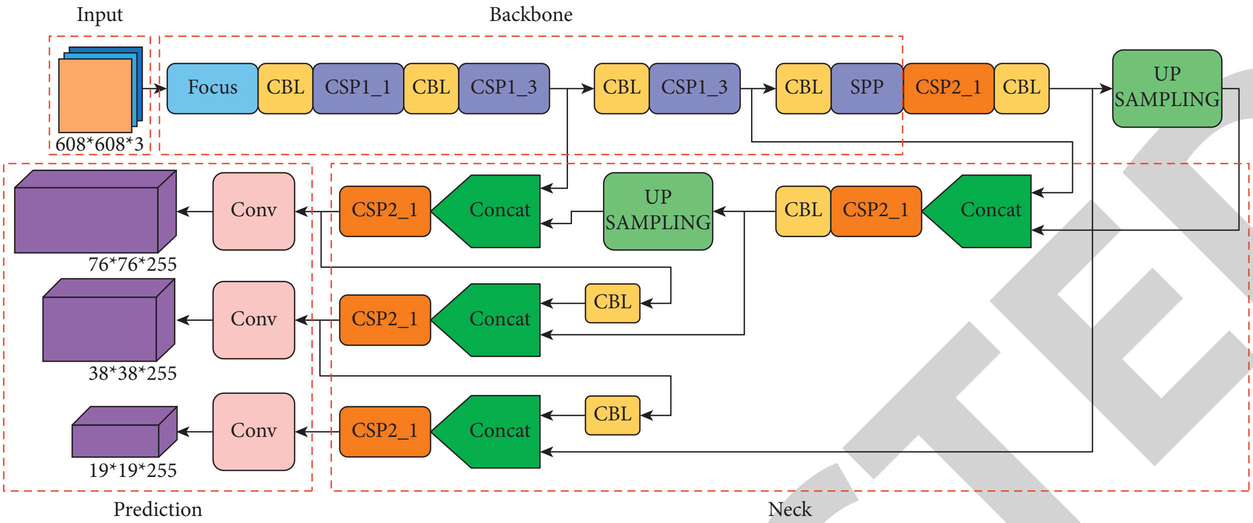


Figure 2: Architectural structure diagram of YOLOv5-s, which consists of input, backbone, neck, and prediction four parts.

the original image, it can adaptively add the least black edge according to the image size. After the black edge of the image is processed, the amount of calculation will be reduced during reasoning, so that the target detection speed of the network will be improved.

The main functional structures of the backbone network include the focus and the CSP modules. The key for the focus structure is the slicing operation, and the convolution operation after the slicing is completed. Different YOLOv5 network structures have exclusive numbers of convolution kernels, and the focus slicing operation is shown in Figure 3(a). There are two types of CSP structures in YOLOv5. The CSP1_X structure in the backbone network consists of CBL module, RES unit module, and convolution layer, and the other CSP2_X structure is in the neck, which includes the convolution layer and X RES unit modules. Using the CSP module can enhance the network's learning ability and make the trained model, which can keep lightweight and have high accuracy at the same time. The addition of the SPP module (spatial pyramid pooling) to the CSP increases the receptive field and extracts the most essential contextual features without causing a reduction in the operation speed. The structure of the two CSP modules is shown in Figure 3(b), and the structure of the SPP module is shown in Figure 3(c).

The neck network has an FPN + PANet structure. FPN is a top-down structure, and the predicted feature maps are computed by fusing the feature information from the higher layers with the lower layer features through an up-sampling operation [25]. The YOLOv5 network structure has a bottom-up feature pyramid added behind the FPN layer, in which there are two PANet structures. This has the advantage of conveying strong semantic features through the top-down FPN layer and strong localization features through the bottom-up feature pyramid. From different backbone layers to different detection layers, the parameter is aggregated. The path aggregation network structure is shown in Figure 4.

The GIOU loss function at the output is an improvement of the traditional IOU loss. Suppose the intersection of the prediction frame and the real frame is A and the concatenation set is B. IOU is defined as the intersection set A divided by the concatenation set B. The loss of IOU can be expressed as follows:

$$\text{IOU\_LOSS} = 1 - \text{IOU} = 1 - \frac{A}{B}. \tag{1}$$

The loss of IOU is relatively simple, but there are also some problems. Firstly, there may be a situation in the prediction frame and the real frame does not intersect when the IOU is 0, which cannot reflect the distance between the prediction frame and the real frame. In addition, when the prediction frame and the real frame have the same size, the IOU may also be the same, and the IOU loss function cannot distinguish between these two situations. Therefore, GIOU loss is proposed for improvement. Defining the minimum outer rectangle of the prediction frame and the real frame be the set C, and the difference set S stands for the difference between the set C and the concatenated set B. Then, the GIOU loss can be expressed as follows:

$$\text{GIOU\_LOSS} = 1 - \text{GIOU} = 1 - \left(\text{IOU} - \frac{|S|}{|C|}\right). \tag{2}$$

The GIOU loss function improves how to measure the intersection scale and reduces the deficiency when it is simply IOU loss.

2.2.3. Comparison of YOLOv5 Network Structures. The four network structures of YOLOv5, YOLOv5-s, YOLOv5-m, YOLOv5-l, and YOLOv5-x, have the same framework, differing only in-depth and width, which are controlled by two parameters, depth multiple and width multiple. There are two CSP structures in the YOLOv5 network structure, CSP1 and CSP2, which have been mentioned above, and the depth of each CSP structure is different in the four networks, respectively.

The number of convolutional kernels used in each network structure depends on the specific characteristics

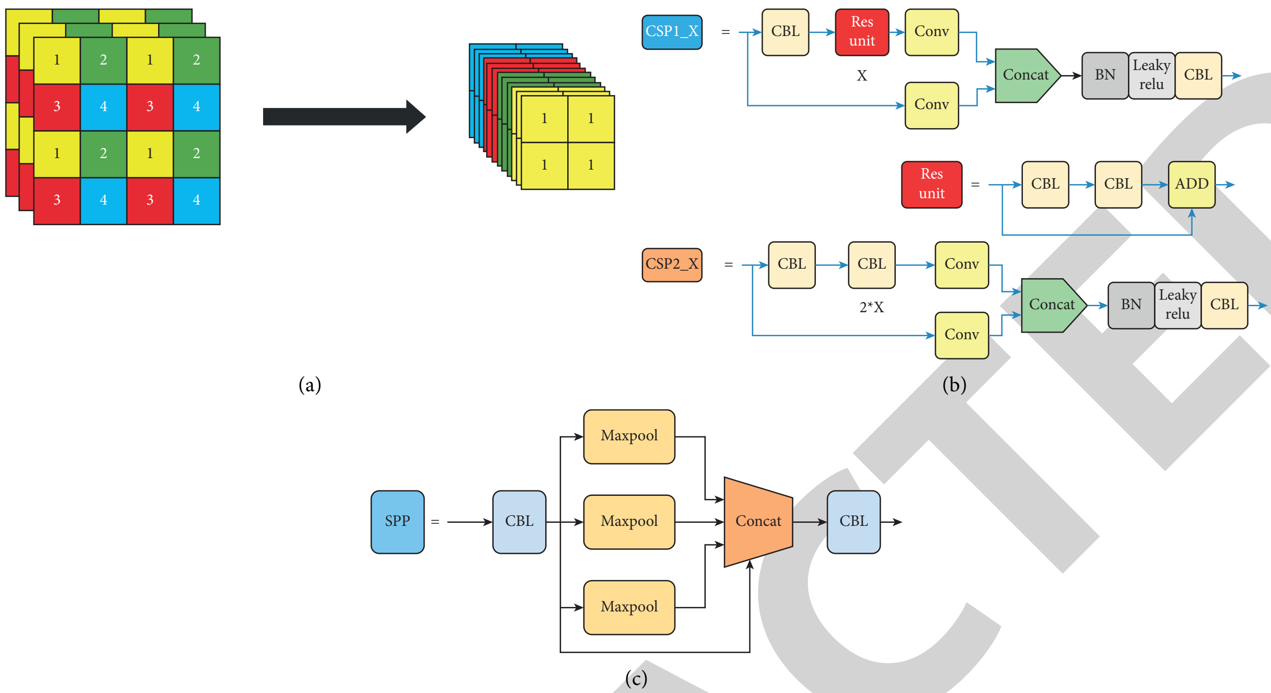


Figure 3: Schematic of the details inside the YOLOv5-s networks. (a) Focus slice operation, (b) CSP module, and (c) and SPP structure.

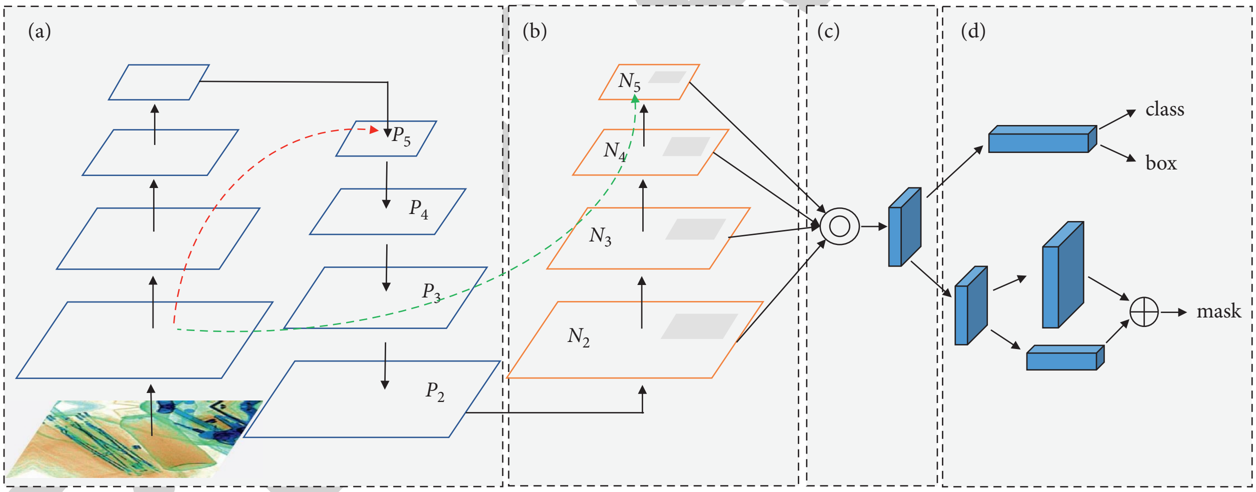


Figure 4: Illustration of PANet framework in YOLOv5. (a) The backbone with FPN structure. (b) Bottom-up augmentation path. (c) Adaptive feature information pooling. (d) The branch of box and fully connection fusion. The channel dimension of feature map does not appear in the schematic.

of locations, which directly affects the third dimension of the convolutional feature map, the width of the network. The more the number of convolutional kernels, the wider the width of the feature map and the better the learning ability of the network to extract features. The four network structures YOLOv5-s, YOLOv5-m, YOLOv5-l, and YOLOv5-x are getting deeper and wider in order, and the detection accuracy is also increasing successively, but the requirement for hardware configuration is also raised. Considering the individuality of different datasets, to ensure the overall detection performance, we need to use the YOLOv5 network structure with the most suitable quantization degree, depth, and feature map width in the series to train the model as much as possible. Table 1 demonstrates the different complexity of different models.

2.3. Our Method's Detection Principle. Although two-stage methods such as Faster-RCNN may yield better detection results, they are not suitable for time-insensitive systems due to their different detection principles. Therefore, our approach needs to improve the one-stage algorithm YOLOv5, which is much faster in detection speed, and improves its checking accuracy for smaller objects. Different types of feature pyramid networks (FPNs) [26–28] are obtained by

Table 1: Four network structures of YOLOv5. The parameters depth multiple and width multiple control the number of BottleneckCSP and the number of convolution kernels of the architectural models, respectively.

	YOLOv5-s	YOLOv5-m	YOLOv5-l	YOLOv5-x
Depth multiple	0.33	0.67	1.0	1.33
Width multiple	0.50	0.75	1.0	1.25
BottleneckCSP BCSPn (true)	1, 3, 3	2, 6, 6	3, 9, 9	4, 12, 12
BottleneckCSP BCSPn (false)	1	2	3	4
Number of convolution kernels	32, 64, 128, 256, 512	48, 96, 192, 384, 768	64, 128, 256, 512, 1024	80, 160, 320, 640, 1280

investigating how to handle feature mappings, rather than just modifying the backbone, which can often be aggregated in different ways to enhance the backbone. Also, adding a transformer to the backbone network is a feasible approach [29], considering that small targets can be extracted with less feature information. In this part, we will introduce each module's principle in detail.

2.3.1. Transformer Principle. Considering that the computation of traditional RNN is restricted to be sequential, that is, the relevant algorithm can only compute sequentially from left to right or from right to left. This mechanism brings two problems: one is the computation of time slice t depends on the computation result at $t-1$ moment, which limits the parallelism capability of the model. Another is that the information may be lost in sequential computation. The proposed Transformer model has successfully solved the two problems mentioned above. It uses the attention mechanism to reduce the distance between any two positions in the sequence to a constant and has better parallelism and conforms to the existing GPU framework. Transformer is essentially an encoder-decoder structure, and the structure of encoder-decoder is shown in Figure 5(a); the specific module of encoder is shown in Figure 5(b). The difference between the encoder and the decoder is that the latter has one more encoder-decoder attention. Two attention is used to calculate the input and output weights, respectively. Self-attention is the relationship between the current translation and the previous text translated. Encoder-decoder attention is the relationship between the current translation and the encoded feature vector.

Then, we analyze the detailed structure in the transformer's encoder. The data first go through a module called "self-attention," which is the transformer's core. In self-attention, each data possess three different vectors, which are query vector $(\mathbf{Q})$, key vector $(\mathbf{K})$, and value vector $(\mathbf{V})$. They are obtained by multiplying the three different weight matrices by the embedding vector $\mathbf{X}$ from three different weight matrices W^Q, W^K, and W^V, where the dimensions of the three matrices are the same, all being 512×64. A weighted eigenvector is obtained by self-attention called $\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$, which can also be denoted as vector $\mathbf{Z}$, and the value is defined as follows:

$$Z = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{soft}\max\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}. \tag{3}$$

After getting the value of $\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$, it will be sent to the next module of encoder—feed-forward neural network. This full connection has two layers, the activation function of the first layer is ReLU, and the second layer is a linear activation function that can be expressed as follows:

$$FNN(Z) = \max(0, ZW_1 + b_1)W_2 + b_2. \tag{4}$$

A whole trainable network structure is the stack of encoder and decoder, and we can get the complete transformer structure in Figure 6. Transformer may not only be applied in the field of NLP machine translation. It has a very promising potential for scientific research.

2.3.2. BiFPN Principle. Since the introduction of FPN [2], it has been widely used for multi-scale feature fusion, and according to the previous introduction of neck part, FPN introduces a top-down channel to fuse features. Recently, PANet, NAS-FPN, and other studies have developed more cross-scale feature fusion network structures such as PANet [30], which adds a bottom-up channel to FPN, and NAS-FPN [31], which uses an irregular topology to search out. While fusing different input features, most previous works simply summarize them indiscriminately. However, since these different input features have different resolutions, we observe that they usually contribute unequally to the fused output features, which means that the feature information is not consistent across scales. At the same time, improved PANet and NAS-FPN bring a great computational effort. Thus, state-of-the-art object detectors are becoming more and more expensive (and some advanced target detectors show excellent performance even at the cost of RAM). For example, NAS-FPN-based detectors require 167M parameters and 3045B FLOPs (30 times more than RetinaNet) to achieve state-of-the-art accuracy. To address these problems, we applied the weighted bi-directional feature pyramid network (BiFPN), proposed by Tan et al., which can perform multi-scale feature fusion easily and quickly. The FPN, PANet, NAS-FPN, and BiFPN structures are, respectively, shown in Figure 7.

BiFPN, as shown in Figure 7(d), is based on a simplified version of PANet by adding residual links, removing nodes with only one input edge, and performing weight fusion if the input and output nodes are at the same level. Adding residual links enhances the representation of features by simple residual operations. Removing nodes with a single input edge is because the nodes with a single input edge are not fused, that is why they have less information and do not contribute much to the final fusion. At the same time, removing a single input edge can reduce the computation and speed up the detection. BiFPN fuses more features without

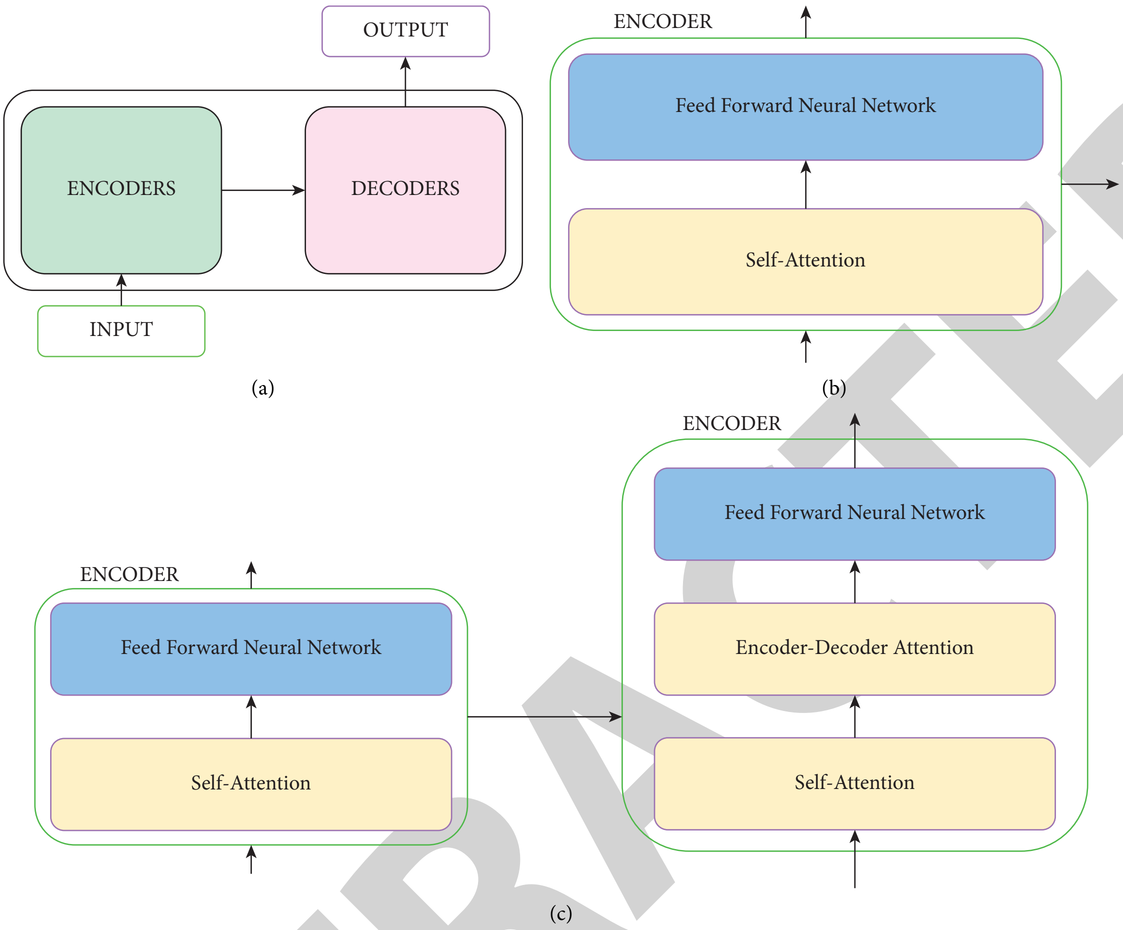

OUTPUT
ENCODERS
DECODERS
INPUT
(a)
ENCODER
Feed Forward Neural Network
Self-Attention
(b)
ENCODER
Feed Forward Neural Network
Self-Attention
ENCODER
Feed Forward Neural Network
Encoder-Decoder Attention
Self-Attention
(c)

Figure 5: Illustration of internal details in transformer structure. (a) The modules of encoder-decoder. (b) The self-attention and the feed-forward neural network constitute one unit in transformer. (c) The completed transformer structure is the composite of self-attention, encoder-decoder attention, and feed-forward neural network.

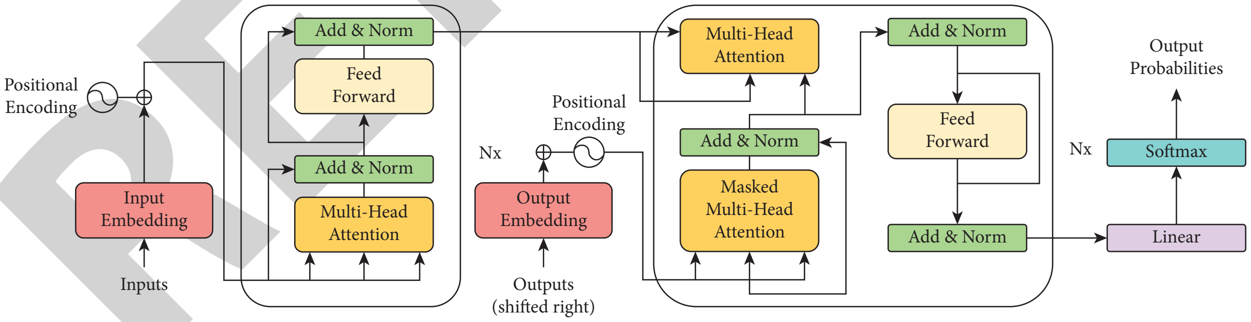


Figure 6: Transformer—model architecture.

increasing the cost and removes the intermediate nodes of P3 and P7 in PANet, resulting in a simplified two-way network.

2.3.3. Coordinate Attention (CA). The CA mechanism [32] has the following advantages. First, it can capture not only cross-channel information but also direction-aware and position-aware information, which can help the model to locate and identify the target of interest more precisely. Secondly, CA is flexible and lightweight, easily inserted into classical modules, such as the inverted residual block proposed by MobileNetV2 [33] and the sandglass block presented by MobileNeXt [34]. Both enhance features utilizing enhanced information representation. Finally, as a pre-trained model, the CA mechanism can significantly benefit downstream tasks on top of lightweight networks, especially those where intensive prediction exists, such as semantic segmentation.

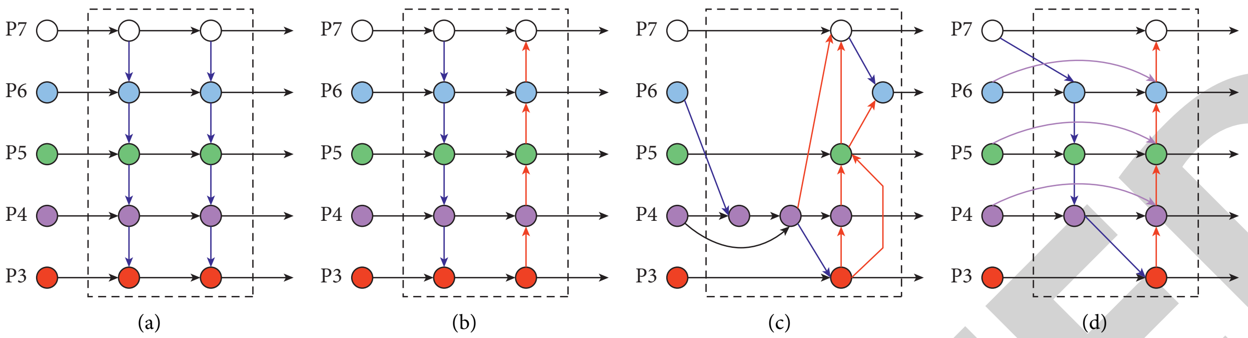


Figure 7: Reticular formation of four structures. (a) FPN adopts a top-down path to mix different features from $P3$ to $P7$. (b) PANet introduces an added down-top path based on FPN. (c) NAS-FPN uses neural networks to exploit unusual information and recycle the same module. (d) BiFPN adds extra branches to attain superior detection accuracy and realize a more efficient trade-off.

The CA module encodes channel relationships and long-range dependencies with precise location information in two steps: coordinate information embedding and coordinate attention generation, structured as Figure 8.

Firstly, we come to part of coordinate information embedding. Global pooling is commonly used in channel attention to globally encode spatial information as channel descriptors, and thus, it is difficult to preserve location information. The authors decompose global pooling into a pair of one-dimensional feature encoding operations to facilitate the attention module to capture spatial long-range dependencies with precise location information. In particular, for the input $\mathbf{X}$, first using the dimensions $(H, 1)$ and $(1, W)$ of the pooling kernel encodes each channel along with the horizontal and vertical coordinate directions, so that the height h and the output of the first c output of the first channel are expressed as follows:

$$z_c^h(h) = \frac{1}{W}\sum_{0\le i\le W} x_c(h, j). \tag{5}$$

Similarly, the width of w of the first c output of the first channel is expressed as follows:

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

These two transformations perform feature aggregation along with two spatial directions, returning a pair of direction-aware attentional maps. This is quite different from the SE(SENet) [35] module that generates a feature vector. Both transformations allow the attention module to capture long-range dependencies along one spatial direction and preserve precise location information along the other spatial direction, which helps the network locate the target of interest more accurately. This coordinate information embedding operation corresponds to the X Avg Pool and Y Avg Pool of Figure 8.

Looking specifically at the operation of the CA mechanism, the two feature maps generated by the previous module are first cascaded and then transformed using a shared 1×1 convolution F_1 that is expressed in the following equation (7). The downsampling ration $\mathbf{f} \in R^{(C/r)\times(H+W)}$ is an intermediate feature map of the spatial information in the horizontal and vertical directions, and is used to control the size of the module as in the SE module.

$$\mathbf{f} = \delta\left(F_1\left[\mathbf{z}^h, \mathbf{z}^w\right]\right). \tag{7}$$

Then, along the spatial dimension, $\mathbf{f}$ is decomposed into two separate tensors $\mathbf{f} \in R^{(C/r)\times(H+W)}$ and $\mathbf{f}^w \in R^{C/r\times W}$, and then, two 1×1 convolution F_h and F_w is operated on feature maps $\mathbf{f}^h$ and $\mathbf{f}^w$, respectively, and transform them into the same channels as the same as the input X. The result of the following equations is obtained:

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

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

The $\mathbf{g}^h$ and $\mathbf{g}^w$ are defined as attention weights. Then, the final output of the CA module can be expressed in the following equation:

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

This part of the coordinate attention generation corresponds to the remaining part of Figure 8, so that the CA module has completed both horizontal attention and vertical attention, and it is also a kind of channel attention.

2.4. TB-YOLOv5 Principle. For small object target detection, the transformer module can enhance the extracted information to make up for the small volume object contextual information, and the BiFPN structure with three inputs can better integrate the input features. These methods are introduced in detail in Sections 2.3.1 and 2.3.2. Both are better means to enhance small target detection. The addition of CA mechanism makes up for the problem that the features extracted by the convolution operation in the general algorithm are more limited. The information obtained is more abbreviated, which makes it challenging to integrate the corresponding features, and allows the neck part to better focus on the detection object we want when performing feature extraction, thus improving the detection accuracy of small target objects.

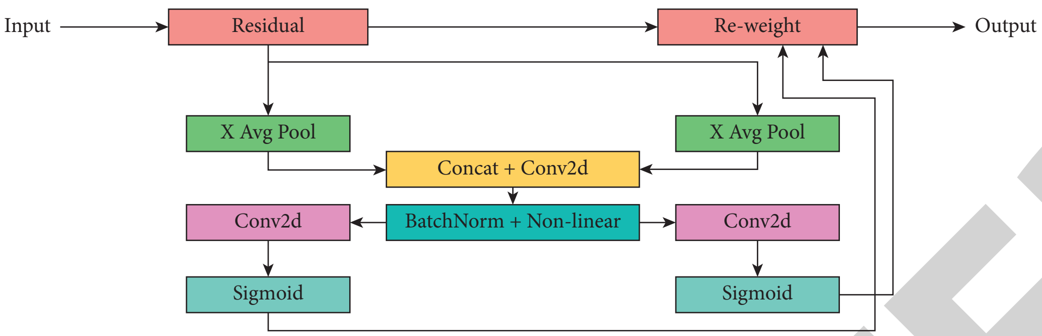


Figure 8: Schematic diagram of the proposed coordinate attention block, while the X Avg Pool and Y Avg Pool stand for $1D$ horizontal and vertical global pooling, respectively.

To increase the feature information extraction capability, we added transformer capability to the bottom layer of backbone to increase, extract information, and optimize detection. The PANet of the traditional neck part is replaced with a BiFPN structure, and a CA mechanism is added to form the new neck structure named attention-BiFPN. A CA module can be regarded as a computational unit to enhance the feature. A CA module can be considered as a computational unit that enhances the expressiveness of features in a mobile network. It can take any intermediate feature tensor as input and output the same size as the tensor with enhanced representations by transformation [32]. The original YOLOv5 algorithm is optimized with the added transformer structure and the designed attention-BiFPN mechanism, and the improved algorithm is named TB-YOLOv5.

In addition, after the above analysis, we know that a significant advantage of the one-stage series algorithm for X-ray security detection is that it has a faster detection speed than the traditional two-stage algorithm, which can be better applied to real-time systems. This is an important reason why the CA mechanism is used, which has the advantage of fast detection speed, compared with some attention mechanisms such as CBAM (convolutional block attention module), an attention mechanism compared with [36], which needs to detect target features in both time and space dimensions of serial detection of target features, so CBAM attention mechanism detection increases the detection time, which contradicts our intention of trying to apply the algorithm to X-ray cases. On the contrary, CA is used to extract information features by coordinates, modeling the location information faster, and the detection time is controlled and suitable for detection needs. In addition, compared with SE block, another attention mechanism has been widely used in recent years, and it only considers the importance of each channel by modeling the channel relationships and ignores the location information. However, the location information is important for generating spatially selective attention maps, which cannot improve the detection results for small targets. The network structure of TB-YOLOv5 is shown in Figure 9.

2.5. Evaluating Indicators. As the number of iterations increases during the training process, various relevant parameters change. In our study, we adopt some usual indicators to measure the performance of results and the exact definitions of them are as follows [37–39]:

Loss Indexes. It is defined by the GIOU loss function; the closer the value is to 0, the more accurate the target frame, detection, and classification are.

Precision. It is defined by the number of correct targets marked divided by the total number of targets marked; the closer to 1, the higher the accuracy rate. TP and FP mean true positive and false positive, respectively.

$$\text{Precision} = \frac{TP}{TP + FP} \times 100\%. \tag{11}$$

Recall. It is defined by the number of correct targets marked divided by the total number of targets to be marked; the closer to 1, the higher the accuracy rate, and FN means false negative.

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{12}$$

mAP.5. AP is the area enclosed after plotting with precision and recall as the two axes; when IOU is set to 0.5, the closer to 1, the higher the accuracy. The mAP.5-small especially expresses the small object detection results in this article.

$$mAP = \frac{\sum_{i=1}^{c} AP_i}{c}, \tag{13}$$

where C stands for the total number of categories and AP_i indicates the ith category value of AP.

3. Results and Discussion

3.1. Parameter Setting and Experimental Environment. When a complete dataset passes through the neural network once and returns once, the process is called an epoch. Passing the complete dataset once in the neural network is not enough, and we need to pass the complete dataset multiple times in the same neural network. We use a finite number of datasets, and we optimize the learning process using an iterative process called gradient descent. Therefore, when an epoch is too large for the computer, it needs to be divided into smaller pieces. However, as the number of epochs increases, the number of updates of the weights in the

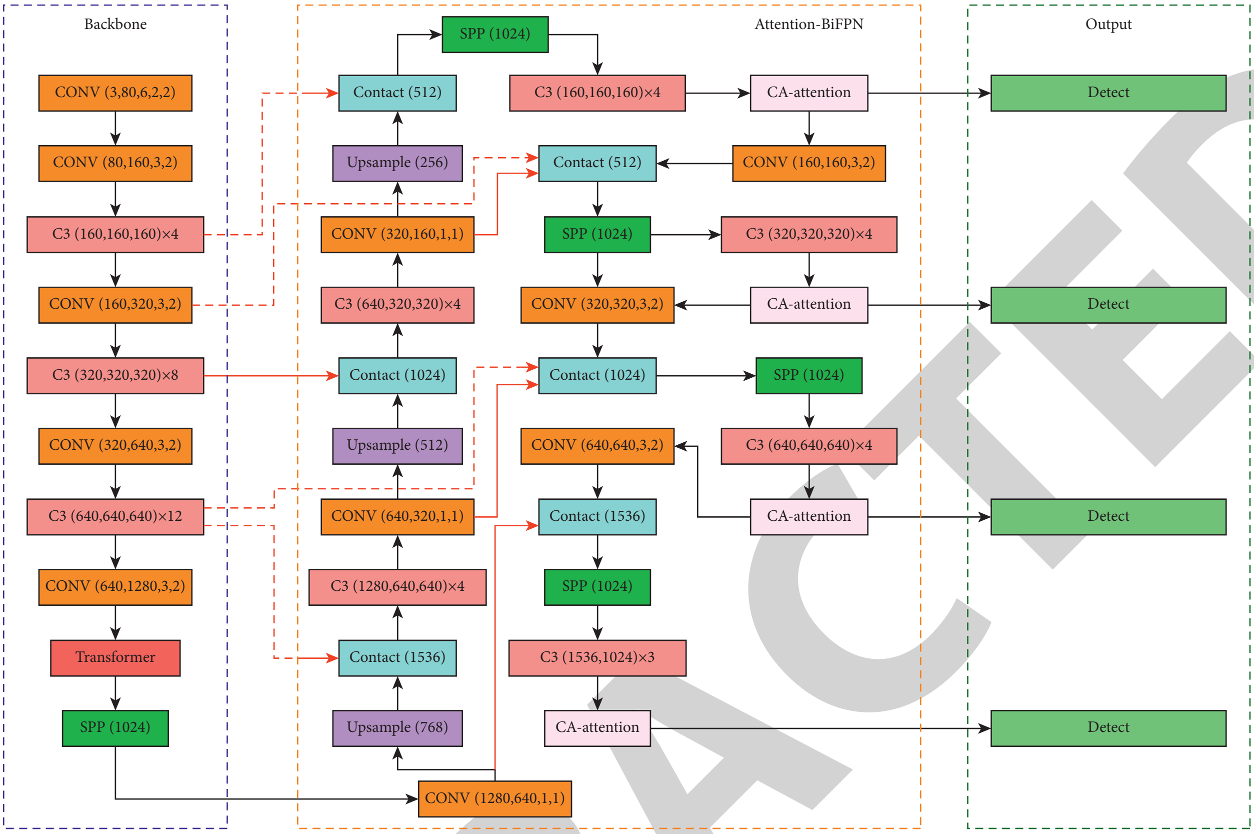


FIGURE 9: Network structure model and parameters of TB-YOLOv5, which consisted of input, backbone, attention-BiFPN, and output four parts while the input section is omitted.

neural network increases, and the curve may go from underfitting to overfitting. In this experiment, we choose all epochs = 500.

The batch size will determine the number of samples we train at a time, and it will also affect the degree of optimization and speed of the model. Batch size is chosen correctly to find the best balance between memory efficiency and memory capacity. A proper increase in batch size can improve memory utilization by parallelization, and the number of iterations in a single epoch is reduced, increasing the running speed in this experiment, and batch size = 8.

The model training process uses the original 4000 generated datasets with X-ray security images, divided into a training set and a validation set in the ratio of 8 : 2.

The configuration of the training device used in our research is Intel Xeon E5-2678 processor, NVIDIA 2080 Ti graphics card, 256 GB RAM, software running environment is Windows 10 operating system, deep learning framework version number Torch 1.7, and Python 3.7 used libraries including CV2, Matplotlib, and NumPy.

3.2. Detection Based on YOLOv5. The initial YOLOv5 target detection algorithm network structure (YOLOv5-s, YOLOv5-m, YOLOv5-l, and YOLOv5-x) was compared for detection accuracy on the X-ray security dataset. The detection accuracies for different categories of objects and the overall average detection accuracy are shown in Figure 10. We specifically define the set of small target objects to further observe the improvement effect of small targets specifically. Section 2.1 has stated that the small target objects in our dataset include seal, knife, scissors, battery, and small bottle glass. We have additionally marked the small objects in red in the figure.

By comparing the four network structures, we can find that for the same dataset of X-ray security screening, the versions *s*, m, *l*, and x of YOLOv5 algorithms have a large difference in detection accuracy for 12 different objects. The average detection accuracy between the four networks is mainly in this accuracy interval of 0.54–0.58. Among them, the detection results of electric equipment are as follows: umbrella reaches 0.8 or even 0.9, most of the object detection accuracy is around 60, for small target objects detection accuracy is lower, only 0.3–0.4, and for scissors detection accuracy is even lower to only about 0.2.

Although there is a difference in the number of pictures of different objects in the dataset, the expandable baton, which has a smaller number of pictures in the dataset, has a higher detection accuracy than some objects with a much larger number of pictures in the dataset compared with itself. On the contrary, some of the small objects, such as scissor, shows poor detection results even when it has a certain number of image data to ensure the training effect. Therefore, although it is impossible to control the same

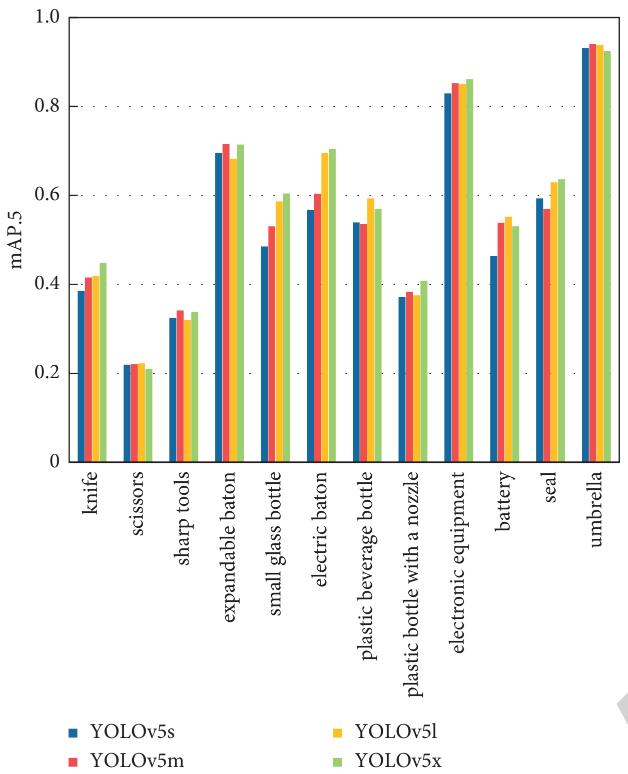


Figure 10: Detection results of YOLOv5 algorithm for four network structures. Small target objects are marked in red.

number of materials for each image in the dataset, we can see that the algorithm of the YOLOv5 series will have a decrease in detection accuracy for small targets due to less information feature extraction. Therefore, there is a great need to improve this problem to ensure that the YOLOv5 algorithm can be better used in real target detection conditions.

3.3. Detection Based on TB-YOLOv5

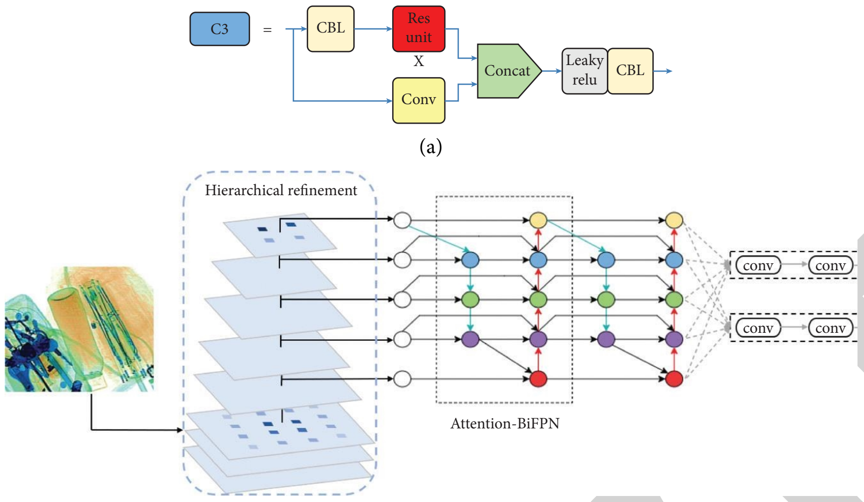
3.3.1. Improvement Method. In the backbone structure of TB-YOLOv5, the focus structure is replaced by the Conv module. In addition, a new detection branch has been added to better detect the lower-level feature information. As we can see in Figure 9, the first C3 structure in the backbone structure is also input to the attention-BiFPN to detect the feature map, and the feature information is extracted during the optimization process of the CA attention module, and then, the optimized information is input to the detection layer of TB-YOLOv5 to obtain the results and output. The C3 structure is suitable for YOLOv5 5.0 and higher, as shown in Figure 11(a). Because of its low number of convolutional layers, the newly added detection branch contains lower-level feature information. The detection module is used to fuse the lower-level visual feature information in the backbone structure with the higher-level visual feature information that our YOLOv5-BT algorithm can obtain a more robust output. The attention-BiFPN-based aggregation path is shown in Figure 11(b).

Using the same no pretrained versions *s*, *m*, l, and *x* network structure model as the regular YOLOv5 for the improved TB-YOLOv5 algorithm, the detection results obtained for each of the 12 objects are shown in Figure 12. Compared with the data in Figure 10, the mAP.5 values under the detection of TB-YOLOv5 algorithm are significantly improved. Some objects such as small glass bottles and plastic bottles have improved from less than 0.6 to about 0.65–0.7. The detection accuracy of the electric baton is somewhat reduced, which may be related to its special morphological structure and the division of the dataset. The indicators are shown in Figure 13. Overall, TB-YOLOv5 improves the detection of this dataset very well.

Figure 14 depicts the difference between the modified mAP.5 value of TB-YOLOv5 and the YOLOv5 training results. This bar chart intuitively makes us feel the improvement of TB-YOLOv5's object detection efficiency. Most of the products have an increased detection accuracy under different networks, and version *x* of YOLOv5 shows the greatest performance in general. The electric baton only gets inspection development under the YOLOv5-m structure.

3.3.2. Comparison of Model Detection Effect and Small Object Detection. The detection accuracy of the mainstream one-stage target detection algorithms (YOLOv3, YOLOv4, and YOLOv5) is compared with the proposed TB-YOLOv5 in this study on the X-ray security dataset. Also, the average detection accuracy of mAP.5-small for five small target objects (knife, scissors, small glass bottle, battery, and seal) and the overall average detection accuracy of mAP.5 are shown in Table 2. From the data in the table, it can be seen that the YOLOv5 algorithm has significantly improved the performance of detection results compared with YOLOv3 and YOLOv4 algorithms, and our modified TB-YOLOv5 algorithm has additional development results compared with YOLOv5. Furthermore, the four network versions s, *m*, l, and *x* of YOLOv5 become better in detection results as the depth and width of the network structure increase.

The improvement of detection results by the TB-YOLOv5 algorithm is different in different network structures. The comparison of the detection results of TB-YOLOv5 and YOLOv5 algorithms with different structures, including the detection results of all objects mAP.5 and small target objects mAP.5-small, is shown to us in Table 3. In comparing the detection results of all objects, the improvement of the *s* network structure is poor, and with only 6.8% improvement, the improvement of the *m* and *l* network structures is similar, both reaching 11.6%. In addition, the network model of TB-YOLOv5-x has the best detection accuracy of 66.4% for the X-ray security inspection dataset, which is a 14.9% improvement compared with YOLOv5. For small target detection objects, the m-network structure is slightly better than the l-network structure. x-Model has the best small target detection, and s-model has the worst results. Compared with the total mAP.5 value, we can discover that the mAP.5-small detection value is improved more significantly. This shows that our algorithm improvement



(a)

(b)

Figure 11: (a) Schematic diagram of the C3 module structure removes a Conv and a BN layer compared with CSP mentioned in Section 2.2.2. (b) Aggregation path based on the attention-BiFPN structure.

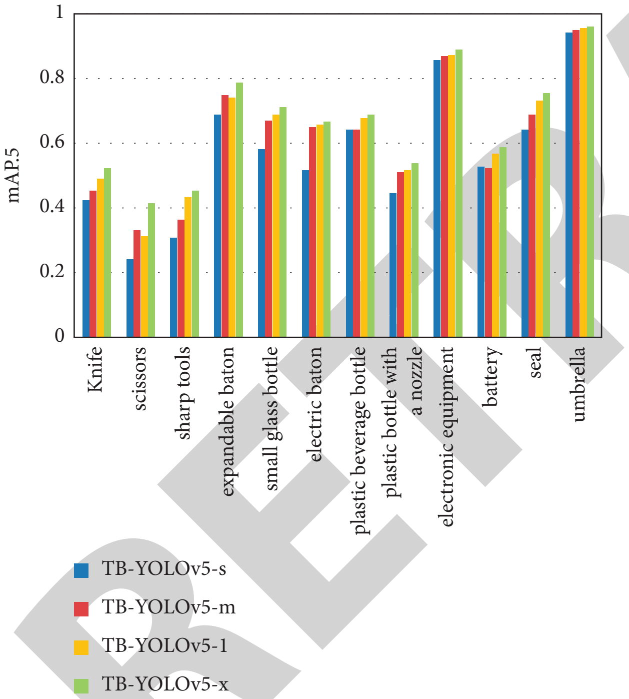


Figure 12: Detection results of TB-YOLOv5 algorithm for four network structures. Small target objects are marked in red.

for smaller objects in the dataset achieves the expected results numerically.

Then, we come to the inference of the model. As a real-time system, the X-ray security inspection should instantly judge the baggage passing the machine. We utilize the training outcome to conduct detection operations on the validation set and come to the detecting time of 800 images. Though the total inference of TB-YOLOv5 indeed has an extension compared with YOLOv5, which can be attributed to extra models that are added in it, the costing time on each image only has a few tens of microseconds. The microsecond difference can be overlooked because it takes several seconds for the luggage compartment to pass through the detector. Meanwhile, a better performance in inspecting accuracy plays a more important role in our research, so the tiny increase in detection time can be tolerated. Table 4 demonstrates different inferences under different network scales for the whole validation set and per image, respectively.

To get a more intuitive feeling of the improvement of TB-YOLOv5 in small target object detection, we select some images in the validation set for comparison. As shown in Figure 15, from the yellow dashed box in the figure, we can notice that these small target objects are easily missed in the YOLOv5 algorithm because of the missing information of small volume features. However, in our improved TB-YOLOv5, these small objects are successfully detected. The improvement of the algorithm can be seen more intuitively on the detection effect of the X-ray security dataset.

Other images in the validation set are shown in Figure 16. Though the missing inspection is still existed, from the yellow dashed box in the figure, we can notice that some relevant "big scale" target objects do not have obvious performance improvement in the TB-YOLOv5 algorithm because the characteristic features are enough. Consequently, the "umbrella" and "electronic equipment" are both detected, and the "electronic equipment" show better detection performance under TB-YOLOv5 structure.

3.4. Discussion. In our study of popular object detectors such as YOLOv5 to better detect smaller objects, we are able to identify architectural modifications that offer significant performance improvements compared with the original model. The context in which we apply the proposed technique, X-ray security detection, is an environment that can benefit significantly from such improvements. As we have seen, this change does have a quantifiable impact on

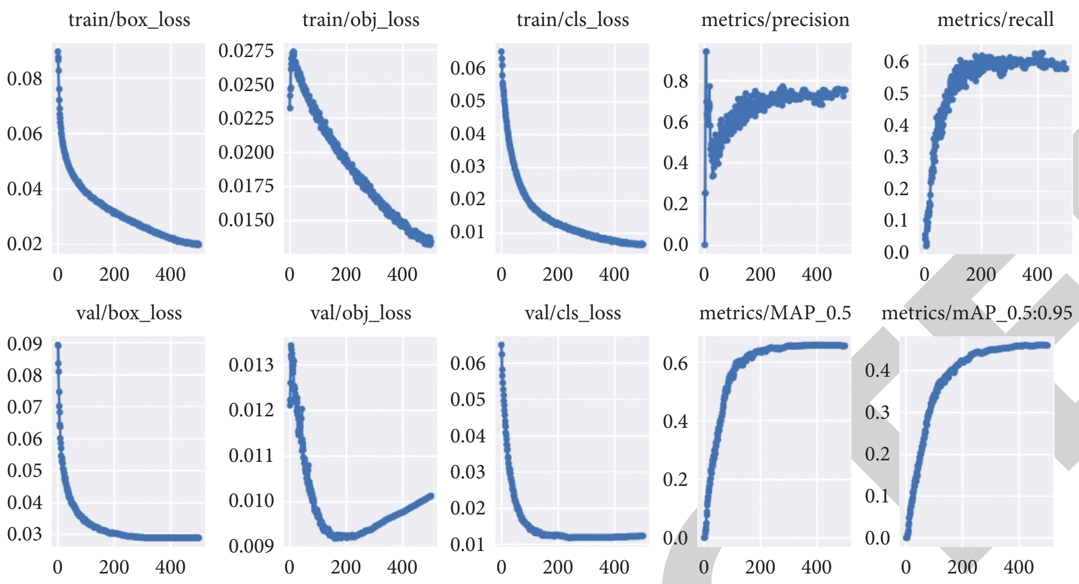


Figure 13: Changes in relevant parameters of training model. Most of the evaluating indicator changes tend to be stable after 500 training iterations.

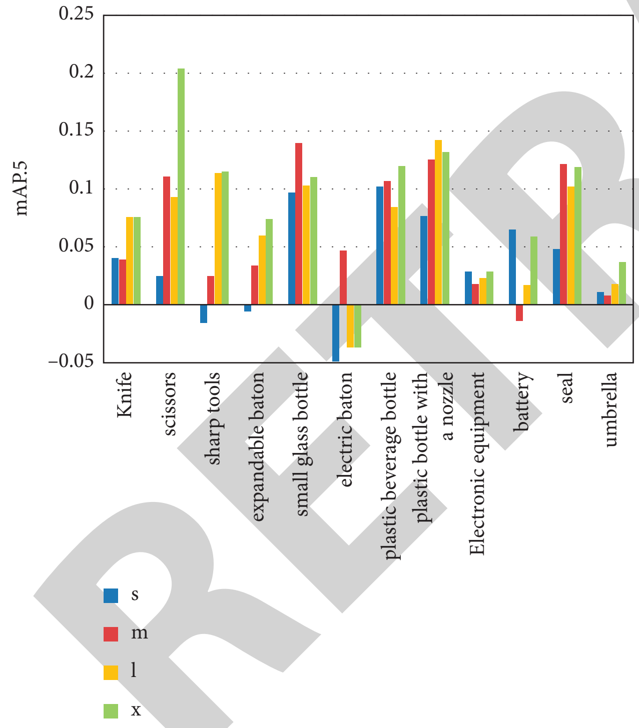


Figure 14: Changes in the detection accuracy of 12 objects between TB-YOLOv5 and YOLOv5. The data above the abscissa mean the improvement achieved by TB-YOLOv5, while the data below the abscissa stand for the negative effect trained by TP-YOLOv5.

Table 2: Performance comparison of different object detection algorithms. With the progression of YOLO detector, the performance becomes much better for newer version, and the different network structures show different detecting results for YOLOv5.

Methods	mAP.5	mAP.5-small
YOLOv3	0.479	0.392
YOLOv4	0.492	0.4022
YOLOv5-s	0.532	0.428
YOLOv5-m	0.552	0.4534
YOLOv5-l	0.571	0.4804
YOLOv5-x	0.578	0.4846
TB-YOLOv5-s	0.568	0.483
TB-YOLOv5-m	0.616	0.5312
TB-YOLOv5-l	0.637	0.5586
TB-YOLOv5-x	0.664	0.5982

detection. In this work, we have not only significantly improved detection performance but also identified specific techniques that can be applied to any other application involving the detection of small- or long-range targets.

The result is that the TB-YOLOv5 family of models outperforms the YOLOv5 class of models in X-ray security screening, especially for smaller objects, which has been the focus of this study. At the same time, detection performance is improved and enhanced for medium-sized objects. Although our focus here is on modifying the popular YOLOv5 model, the methods and techniques we explore can potentially evolve into a wholly original model structure.

Finally, while this study suggests significant empirical gains from the proposed architectural changes, the consistency and generalizability of the results can and should be further investigated. For example, further testing using different datasets and possible challenges such as security detection would greatly aid the analysis. While we have demonstrated the usefulness of the technique our study presented, these techniques can only be refined and better understood when applied to different environments and settings. Doing so would be an important step toward a more robust solution for small target detection. In addition, there are more directions and techniques that would fit well into

Table 3: Performance comparison of YOLOv5 and TB-YOLOv5. With the increase in depth and width, the detection results of both YOLOv5 and TB-YOLOv5 become much better.

Scales	mAP.5			mAP.5-small		
	YOLOv5	TB-YOLOv5	Difference (%)	YOLOv5	TB-YOLOv5	Difference (%)
S	0.532	**0.568**	6.8	0.428	**0.483**	12.9
M	0.552	**0.616**	11.6	0.4534	**0.5312**	17.2
L	0.571	**0.637**	11.6	0.4804	**0.5586**	16.3
X	0.578	**0.664**	14.9	0.4846	**0.5982**	23.4

Table 4: Inference comparison of YOLOv5 and TB-YOLOv5. With the increase in depth and width, the detection speed of both YOLOv5 and TB-YOLOv5 becomes much slower.

Scales	Inference (ms) (800 images)			Inference (μs) (per image)		
	YOLOv5	TB-YOLOv5	Difference (%)	YOLOv5	TB-YOLOv5	Difference (%)
s	40.3	**54.6**	35.5	50.38	**68.25**	35.5
m	44.8	**70.5**	57.4	56.00	**88.13**	57.4
l	46.2	**71.3**	54.3	57.75	**89.13**	54.3
x	51.4	**83.4**	62.3	64.25	**104.25**	62.3

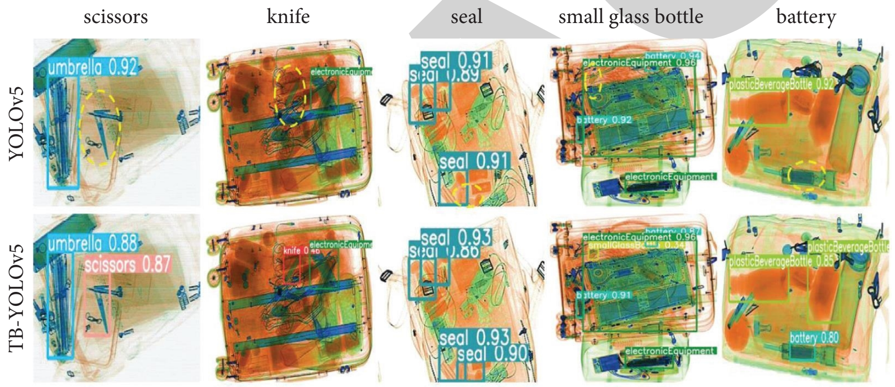


Figure 15: Visual demonstration of the improved detection comparisons of TB-YOLOv5 compared with YOLOv5 over some X-ray dataset images covering small-scale objects under x network. Yellow dotted lines circle the missed inspection of small targets investigated by YOLOv5, while they are all detected by TB-YOLOv5 algorithms.

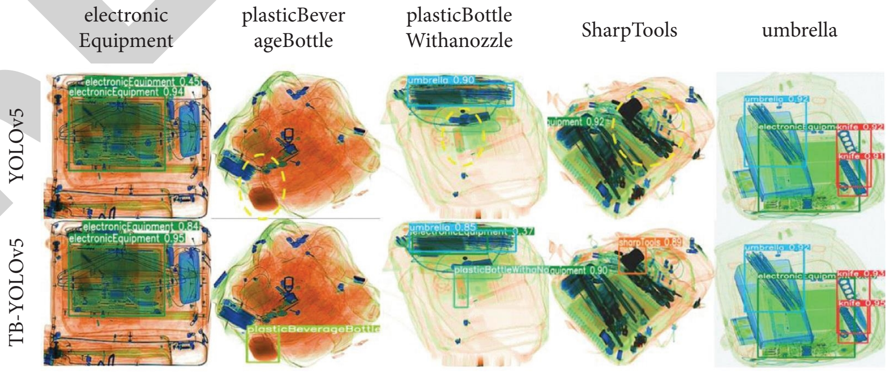


Figure 16: Visual demonstration of the improved detection comparisons of TB-YOLOv5 compared with YOLOv5 over some X-ray dataset images of other objects under x network. Yellow dotted lines circle the missed inspection of small targets investigated by YOLOv5, while they are detected by TB-YOLOv5 algorithms. Other things are both detected by the two network structure.

this topic and were not considered in this study, but these will remain the subject of future research.

4. Conclusion

The security inspection based on X-ray is a very useful way in each country to guarantee safety in public places such as transportation hubs. The traditional detection methods have many defects to be improved. In this research, we investigate the performance of different architectural models applied to the YOLOv5 objection detector and propose the novel concept of TB-YOLOv5, proving that the algorithm improves the disadvantages of insufficient feature information of target detection in the trunks scanned by X-ray images. Based on the previous studies, the transformer module is added in the backbone, and the attention-BiFPN constituted CA mechanism, and BiFPN principle is applied. The average value of twelve objections' detection accuracy trained by TB-YOLOv5 shows the highest 14.9% increase compared with YOLOv5. Furthermore, the small-object detection ability of TB-YOLOv5 acquires a more significant improvement in the article, in which the maximum is reached 23.4%. We validate the proposed technique in X-ray security inspection, underlining the specific requirements and limitations, and expect further research. The proposed TB-YOLOv5 structure system can be updated to detect better smaller targets in a situation where existing methods are unable to achieve. Through the experimentation results still have disadvantages that some objection detection precisions are relatively low and the model is unable to be applied in actual project now, and we have reasons to believe that the TB-YOLOv5 algorithm based on attention-BiFPNs has a large potential prospect in other relevant fields such as realizing factory intelligence and automatic driving.

Data Availability

The datasets used and/or analyzed during this study are available from the corresponding author on reasonable request.

Disclosure

Muchen Wang and Yueming Zhu are co-first authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Muchen Wang and Yueming Zhu contributed equally to this work.

Acknowledgments

This study was supported by National Natural Science Foundation of China (NSFC61702384) and Major State Basic Research Development Program of China (202110488008).

References

[1] K. C. Wang, *Research of Rail Detection System Based on Electromagnetic Acoustic Technique [D]*, pp. 7-8, Harbin Institute of Technology, Harbin, 2010.

[2] C. Miao, L. Xie, F. Wan et al., "Sixray: a large-scale security inspection x-ray benchmark for prohibited item discovery in overlapping images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2119–2128, Long Beach, CA, USA, June 2019.

[3] D. Mery and C. Arteta, "Automatic defect recognition in x-ray testing using computer vision," in *Proceedings of the WCCV*, pp. 1026–1035, pages,Santa Rosa, CA, USA, March 2017.

[4] D. Mery, E. Svec, M. Arias, V. Riffo, J. M Saavedra, and S Banerjee, "Modern computer vision techniques for x-ray testing in baggage inspection," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 4, pp. 682–692, 2017.

[5] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3462–3471, Honolulu, HI, USA, 2017.

[6] N. D. Nguyen, T. Do, T. D. Ngo, and D. L. Duy, "An evaluation of deep learning methods for small object detection," *Journal of Electrical and Computer Engineering*, vol. 2020, Article ID 3189691, 2020.

[7] B. Singh, M. Najibi, S. Larry, and S. N. I. P. E. R. Davis, "Efficient multi-scale training," *Advances in Neural Information Processing Systems*, vol. 31, pp. 9310–9320, 2018.

[8] B. Singh, M. Najibi, A. Sharma, and L. S. Davis, "Scale Normalized Image Pyramids with AutoFocus for Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, 2022.

[9] R. Girshick, J. Donahu, T. Darrell, and M. Jitendra, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, IEEE, Columbus, OH, USA, June 2014.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2015.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2014.

[12] B. Wu, F. Iandola, H. J. Peter, and K. Keutzer, "SqueezeDet: Unified, Small, Low Power Fully Convolutional Neural Networks for Real-Time Object Detection for Autonomous Driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Honolulu, HI, USA, July 2017.

[13] W. Liu, D. Anguelov, D. Erhan, and R. Scott, "SSD: single shot multibox detector," *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 9905, pp. 779–788, IEEE, Piscataway, 2016.

[14] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: scalable and efficient object detection," *Proceedings of the 2020 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10778–10787, IEEE, Piscataway, 2020.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: unified, real-time object detection,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, IEEE, Piscataway, 2016.

[16] J. Glenn, A. Stoken, J. Borovec et al., *ultralytics/yolov5: v5.0 - YOLOv5-P6 1280 Models, AWS, Supervise.Ly and YouTube Integrations*, 2021.

[17] G. Yang, W. Feng, J. Jin et al., “Face mask recognition system with YOLOV5 based on image recognition,” in *Proceedings of the 2020 IEEE 6th International Conference on Computer and Communications, ICCC 2020*, pp. 1398–1404, Chengdu, China, December 2020.

[18] A. Benjumea, I. Teeti, F. Cuzzolin, and A Bradley, “YOLO-Z: Improving Small Object Detection in YOLOv5 for Autonomous vehicles,” 2021, https://arxiv.org/abs/2112.11798.

[19] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S Zagoruyko, “End-to-end object detection with transformers,” *European Conference on Computer Vision*, Springer, Cham, 2020.

[20] X. Zhu, X. Zhu, W. Su et al., “Deformable detr: deformable transformers for end-to-end object detection,” 2020, https://arxiv.org/abs/2010.04159.

[21] Z. Qin, P. Zhang, F. Wu, and X. Li, “Fcanet: frequency channel attention networks,” in *Proceedings of the IEEE/CVF international conference on computer vision*, Montreal, QC, Canada, October 2021.

[22] E. Pérez-Pellitero, C. C. Sibi, L. Aleš et al., “NTIRE 2021 challenge on high dynamic range imaging: dataset, methods and results,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN, USA, June 2021.

[23] Dataset: http://challenge.xfyun.cn/topic/info?type=Xray-2021.

[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, IEEE, June 2014.

[25] T.-Yi Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, *Feature Pyramid Networks for Object Detection*, CVPR, Honolulu, Hawaii, 2017.

[26] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, “A unified multi-scale deep convolutional neural network for fast object detection,” *ECCV*, pp. 354–370, Springer, Cham, 2016.

[27] W. Liu, D. Anguelov, D. Erhan et al., “SSD: Single Shot Multibox Detector,” *ECCV*, Vol. 9905, Springer, Cham, 2016.

[28] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, “Overfeat: Integrated Recognition, Localization and Detection Using Convolutional Networks,” 2014, https://arxiv.org/abs/1312.6229.

[29] A. Vaswani, N. Shazeer, N. Parmar et al., “Attention is all you need[J],” *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[30] S. Liu, Qi Lu, H. Qin, J. Shi, and J. Jia, “Path Aggregation Network for Instance Segmentation,” in *Proceedings of theCVPR*, June 2018.

[31] G. Ghiasi, T. Y. Lin, and Q. V. Le, “Nas-fpn: learning scalable feature pyramid architecture for object detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7036–7045, Long Beach, CA, USA, June 2019.

[32] Q. Hou, D. Zhou, and J. Feng, “Coordinate attention for efficient mobile network design,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13713–13722, Nashville, TN, USA, June 2021.

[33] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: inverted residuals and linear bottlenecks,” in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.

[34] Z. Daquan, Q. Hou, Y. Chen, J Feng, and S Yan, “Rethinking Bottleneck Structure for Efficient Mobile Network Design,” in *Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science()*Vol. 12348, Springer, Cham, 2007.

[35] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, June 2018.

[36] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, “Cbam: Convolutional Block Attention module[C]//Proceedings of the European Conference on Computer Vision (ECCV),” *CBAM: convolutional block Attention module, Computer Vision - ECCV 2018*, Springer, vol. 11211, pp. 3–19, Cham, 2018.

[37] D. Bahdanau, K. Cho, and Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and translate[J],” 2014, https://arxiv.org/abs/1409.0473.

[38] M. Tan, R. Pang, and Q. V. Le, “Efficientdet: scalable and efficient object detection,” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10781–10790, Seattle, WA, USA, 2020.

[39] Y. Sha, Y. Zhang, X. Ji, and H. Lei, “Transformer-Unet: Raw Image Processing with Unet[J],” 2021, https://arxiv.org/abs/2109.08417.