Impact Factor:					
	ISRA (India) = 6.317	SIS (USA) = 0.912	ICV (Poland) = 6.630
	ISI (Dubai, UAE) = 1.582	РИНЦ (Russia) = 0.126	PIF (India) = 1.940
	GIF (Australia) = 0.564	ESJI (KZ) = 9.035	IBI (India) = 4.260
	JIF = 1.500	SJIF (Morocco) = 7.184	OAJI (USA) = 0.350

SOI: 1.1/TAS DOI: 10.15863/TAS

International Scientific Journal

Theoretical & Applied Science

p-ISSN: 2308-4944 (print) **e-ISSN:** 2409-0085 (online)

Year: 2021 **Issue:** 03 **Volume:** 95

Published: 23.03.2021 **http://T-Science.org**

QR – Issue QR – Article

Samuel Akwasi Danso
Southwest University of Science & Technology, Information &Communication Engineering
PhD Student

Shang Liping
Southwest University of Science & Technology, Information &Communication Engineering
Professor

Hu Deng
Southwest University of Science & Technology, Information &Communication Engineering
Professor

Justice Odoom
Southwest University of Science & Technology, Information &Communication Engineering
PhD Student

Linyu Chen
Southwest University of Science & Technology, Information &Communication Engineering
PhD Student

Zhong-gong Xiong
PhD Student
Southwest University of Science & Technology, Information &Communication Engineering
59 Qinglong Road, 621010 Mianyang-Sichuan, China

OPTIMIZING YOLOv3 DETECTION MODEL USING TERAHERTZ ACTIVE SECURITY SCANNED LOW-RESOLUTION IMAGES

Abstract: *Terahertz technology is nonionizing radiation consequently posing less human risk. However, its spectroscopy-scanned images are characterized by low-resolution images thereby posing significant challenges when object detection is to be performed in such images. Recently, deep learning-based detection has shown much prospects owing to their highly based computer vision approach for its superior efficiency and easy network parameter optimization. In this paper, we perform a comprehensive analysis of prominent object detection models based on terahertz images regarding concealed dangerous and prohibited objects in bags, books, wood etc. as often witnessed in airports, subway stations etc. By way of boosting the performance coupled with detection accuracy of the models, we expand our initial terahertz images via image augmentation. Experimental results reveal that one-way detection method for hidden weapons and non-weapons is far better than two-way detection methods. Moreover, we achieved a 2% increased accuracy and an increased rate of 2.5 due to the optimization fromYOLOv3.*

Key words: *Terahertz image, object detection, deep learning, hidden weapon.*

Language: *English*

Citation: Danso, S. A., et al. (2021). Optimizing YOLOv3 detection model using terahertz active security scanned low-resolution images. *ISJ Theoretical & Applied Science, 03 (95)*, 235-253.

Soi: http://s-o-i.org/1.1/TAS-03-95-39 ***Doi***: crossref https://dx.doi.org/10.15863/TAS.2021.03.95.39

Scopus ASCC: *1700.*

Introduction

To mitigate the increased danger of terrorism activities, illicit items in luggage and personal safety at checkpoints in airports, bus terminal and industry entry gates, etc., the use of diverse detection technologies turn to be highly important [1]. Detection systems such as weapon detectors for travelling passengers and X-ray systems [2] for scanning hand-carried gadgets are effective but additionally have their own shortcomings. X-ray imaging systems can penetrate materials such as paper box, leather bag, clothing, wooden boxes as well as the human body. The hazards of X-ray system are that their radiation is very high and they are very detrimental to the human body. The operators staffing these check points who are performing their lawful businesses turn to be in danger as well as the passengers or the clients being attended to at various checkpoints since these X-rays signals reflect, absorb and transmit contrary to terahertz rays. [3–6]

The terahertz (THz) portion of the electromagnetic spectrum, shown in figure 1, extends from approximately 100 GHz to 10 THz (where 1 THz = 1012 Hz corresponding to 4.14 meV). It lies between the microwave and infrared band; the wavelength in this range is 3 mm to 30 μm. THz waves can penetrate numerous non-metallic materials that may be opaque in the range of visible and infrared light. Moreover, as nonionizing radiation, THz waves present minimal known health risks [7].

Terahertz system can also be referred to as the millimeter wave or submillimeter/far-infrared waves (sometimes also called T-rays). THz waves have attracted increased interest due to their capability to non-destructively penetrate strong objects, including, those made of cloth, paper, wood, plastic, and ceramics, and to produce images of the hidden objects. Sub-THz body safety scanners are also encouraged at airports because of their non-ionization effects.

Higher frequency represents shorter wavelength (1–10 mm) [7], which yields higher resolution terahertz images. Weapon detectors can solely identify similar weapon targets, such as metal handguns, knives, blade, screwdrivers as against non-weapon gadgets such as mobile phone, water bottle, board marker, wireless mouse etc.

This paper focuses on terahertz active imaging for security applications and intends to realize high-speed and high-accuracy detection of weapon and non -weapon objects of terahertz scanned images. Deep learning models have great effects on optical images as well terahertz images.

Optical images basic classification and feature extraction strategies brings boundary path histogram [8], Fourier transform, window Fourier transform [9], wavelet transform [10–12], least squares [13],etc. Additionally, it includes histogram of oriented gradient (HOG) [14] and invariant feature transform (SIFT) [15], the most widely used object-detection-and-recognition model is the deformable parts model (DPM) [16], which uses a support vector machine (SVM) [17] to train an object model and retain the best performance unlike the hand-made features, which lack the self-training processing and visual processing. Today artificial neural network (ANN) and convolutional neural network (CNN) LeCun et al [18] depicts from support virtual Machine (SVM) notwithstanding has gradually attracted peoples' interest [19, 20].

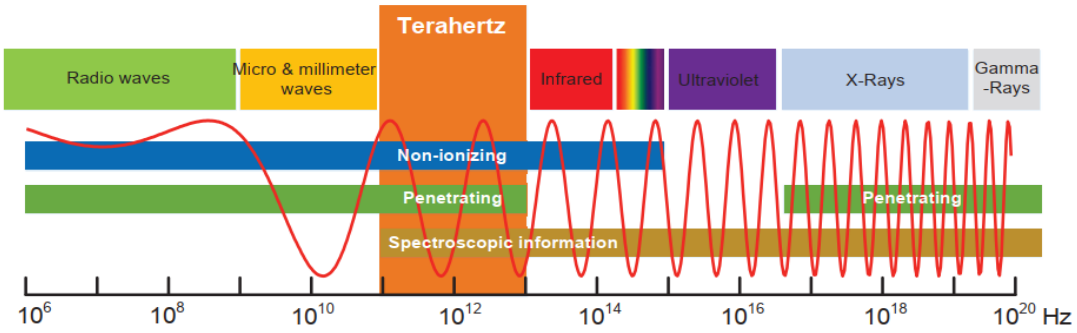

Figure 1 - Terahertz Spectrum region

CNN aims to mimic human perception for intelligent classification, recognition and segmentation. CNNs architectural structure (e.g., AlexNet [21]), the deep structure (e.g., VGG [22], GoogleNet [23], the residual unit embedded structure like ResNet [24], ResNeXt [25], DenseNet [26], DarkNet [27], and the lightweight structure MobileNet [28]. Krizhevsky et al. (2012) with understanding of SVM trained a large, deep CNN (Alexnet) [29]. The activation functions that Rectified linear devices (ReLUs) and others utilized improved nonlinear mapping capacity of this network and lost gradient. Large and improved networks in Alexnet [30, 31] due to more studies in Alexnet [32] Pan et al., (2009). Donahue et al. studied a semi-supervised deep convolution method for multitask mastering of transfer learning. This growing knowledge in CNN turns to caffe best features. Today, caffe is a widely

used open-source framework for deep learning with incredible GPU speed. R-CNN (Fast R-CNN) proposed by using Girshick et al. [33] is a one-stage detection algorithm that improves object proposals and refines their locations and their real-time detection. SNIP [34] discovered the domain-sift and corrects the multi-scale training problem. To solve the problem of multi-scale object detection, STDN [35] introduced a scale-transfer layer into DenseNet without increasing computational complexity. To further boost the detection efficiency, RefineDet [36] combined the RPN and FPN with the fast SSD [37] approach. Fast R-CNN [30], Faster R-CNN [38], R-FCN [39], and YOLO [27] use their numerous features within CNN to predict objects at different scales, as well as SSD [25] and MS-CNN [26]. Combining RPN and Fast R-CNN into a single network, called the Faster R-CNN, the framework of Faster R-CNN and other advanced models, such as the Mask R-CNN [32, 40], adds pixel-level segmentation. PANet [41] was thereafter proposed and has achieved even better segmentation results. The (SPPnet)[42] reduces the training and speed. Spatial pyramid pooling networks had been proposed to speed up R-CNN by way of sharing computation and convolutional features. To solve the micro target detection problem, [23, 29] revised the ResNet by integrating the idea of feature pyramid. PASCALVOC 2007 [43], MS COCO [44] from optical images, x-ray image and terahertz images are slightly different with distinctive imaging mechanisms. Their similarities in frequency spectrum are nearer. As a result, terahertz images of detection have inspection geometry features to optical images and x-ray images, because reflection, absorption, scattering of electromagnetic rays exhibits similar traits like angle, target structure and material penetration factors. In this paper, we attempt to transfer these classification strategies and detection strategies in terahertz images.

In order to detect objects of different scales, a basic strategy is to use **featurized** image pyramids [23] to obtain features at different scales.Yolov3 backbone is also known as Darknet-53 [27] .In this paper, we also adopt this training framework to instruct an effective model network on terahertz weapon and non-weapon security scanned image.

This paper is arranged as follows. In Section 2, the introduction of Terahertz scanned images and augmented dataset arrangements. In Section 3, the methodology of object detection grouped into one-stage, two- stages detector models, and their concealed diagrams are explained. In Section 4, we present the experimental outcomes and corresponding evaluation of the model. Section 5 discusses the optimized best results based on analyzed models and Section 6 concludes this paper.

Dataset Description

In this section, we introduce the acquisition steps of terahertz image and the expansion methods for the image data set, including rotation, translation, affine transformation, transmission transformation and so on. Finally, the corresponding statistical analysis of the expanded data set is carried out.

Data Acquisition

Due to acquisition rate up to 5000 lines per second teraFAST-256 device can accommodate scan speed up to 15 m/s. The sensor has single sensitivity band at 100±10GHz but experimental power source is between 100GHz. The conveyer belt speed of 10.1m/s is for image capture.

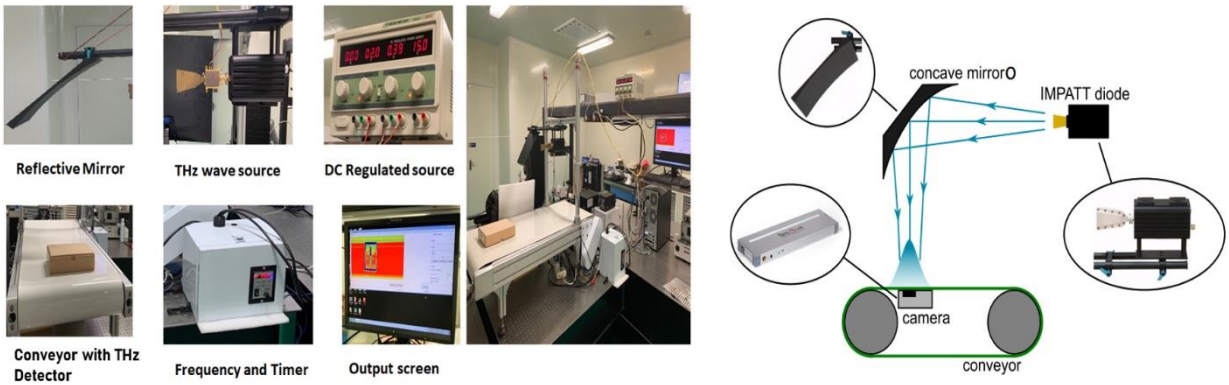

Figure 2 - Terahertz image acquisition

The size of the image data collected by the device is 512px*256px. For our research, we collected a total of 8 kinds of terahertz images of objects, including 4 types of weapon images and 4 types of non-weapon images (in total, 369 images, because there might be more than one instance of a single image).

The raw data information is shown in table 1 and Figure 3.

Table 1. Original terahertz image data

Class	Screwdriver	Blade	Knife	Scissors	Board marker	Mobile phone	Wireless mouse	Water bottle
Number of images	65	19	66	59	40	40	40	40

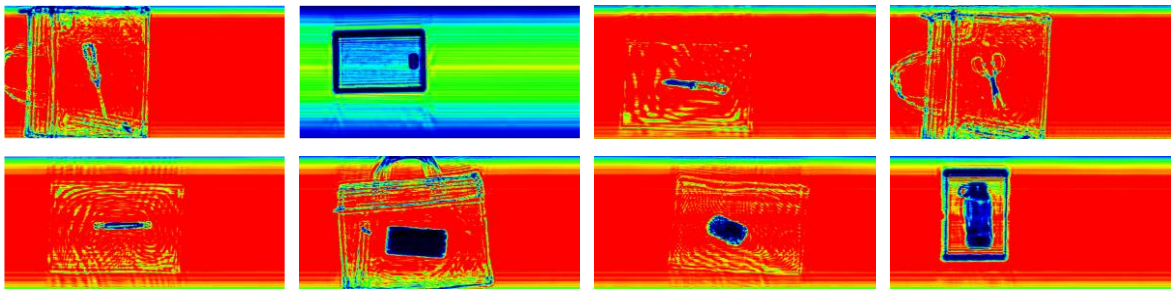

Figure 3 - Scanned THz images

Data Augmentation

It is indisputable fact that terahertz technology is fairly new hence associated images are scanty. It is therefore not bizarre that from the previous steps the number of terahertz data sets collected was too small.

Consequently, the target detection algorithm may be under-fitted in the case of so little data since terahertz image database is uncommon, and the performance of the model cannot meet the actual detection accuracy requirements. For this reason, it is necessary to expand the original image data. The methods of data argumentation used in this study are shown in figure 4.

These image augmentation methods can be used alone or combined with a variety of transformations, that solves the problem of little training data to a certain extent. After augmentation, we get a total of 1884 images, and mark the location of the object. The next section will make a statistical analysis of the augmented data set.

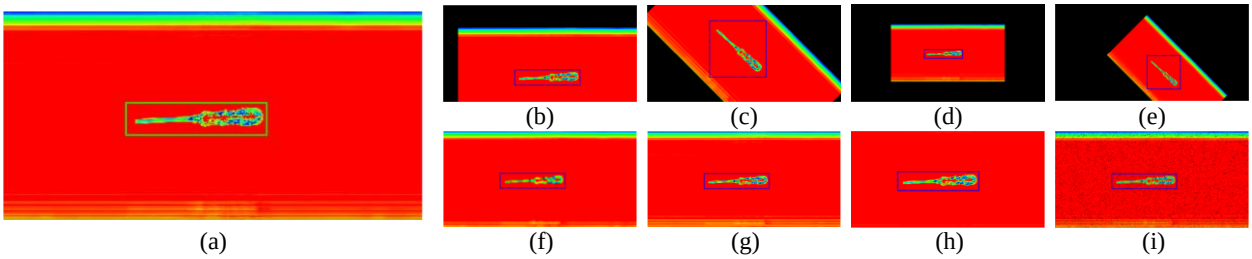

Figure 4 - Methods for augmenting terahertz images. (a) Original image. (b) Translation. (c) Rotation. (d) Scaling. (e) Affine. (f) Blurring. (g) Sharpen. (h) Cropping. (i) Dropout.

Statistical Information of Dataset

The statistical analysis of the data set is helpful for us to understand the characteristics of the data and to optimize the subsequent model. First, we make statistics on the number of instances and the average bounding box size of eight (8) categories, and get the following results:

Table 2. Dataset Statistics Analysis

Class	Number of instances	Average bounding box size
Screw drive	390	108px*84px
Blade	200	36px*35px
Knife	396	89px*75px
Scissors	354	104px*91px
Board marker	240	78px*68px
Mobile phone	240	110px*87px
Wireless mouse	240	70px*75px
Water bottle	240	118px*91px

As can be seen from table 2, the number of blade categories is the fewest, and the average bounding box is also the smallest. Screwdriver and knife are the largest in number and, bounding box is relatively large. In addition, we also counted the ratio of all bounding box areas to the whole picture, and obtained the histogram shown in figure 5.

From the results of figure 5, we can see that most of the bbox area ratio is about 3% to 10%. A small part is concentrated in the 1% area ratio, and the maximum proportion is no more than 25%. Further, we can analyze the size distribution of different types of bbox, as shown in figure 6.

It can be seen from figure 6 that the bbox size of the blade category is relatively small, which may also cause the target detection algorithm to become worse under this category. The sizes of other categories are widely distributed and evenly distributed.

Methodology

In this section, we introduce five target detection algorithms for terahertz images, namely, YOLOv3 and SSD, (one-stage detection) and Faster RCNN as well as Cascade RCNN (two-stage detection). These algorithms will be used to detect weapon class objects and non-weapon class objects in terahertz images.

One-stage detector

The single-stage detector realizes a series of end-to-end processes such as image input, feature extraction, regression location and object classification, and no other processes are introduced as shown in figure 6. This kind of target detection algorithm ensures a certain accuracy under high detection speed, and is also widely used in industrial detection.

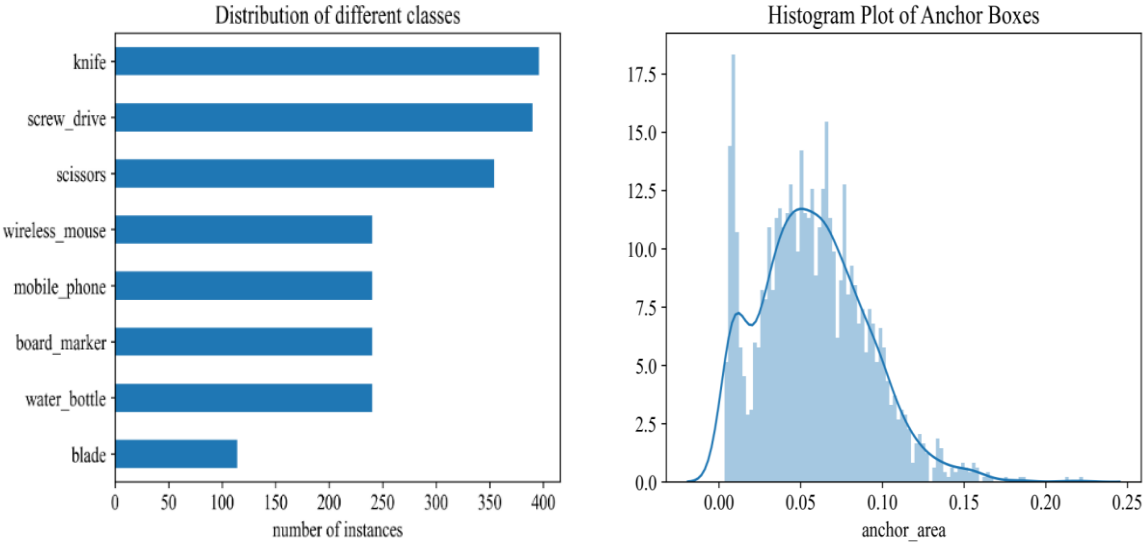

Figure 5 - Graph of Classes distribution & Histogram of Bounding boxes

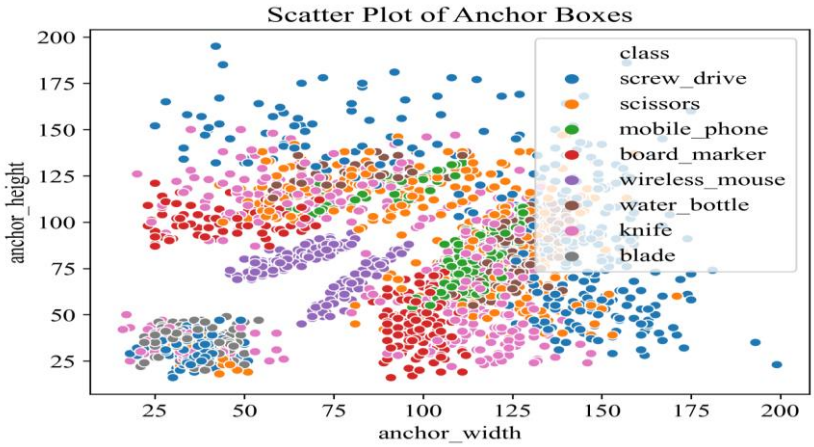

Figure 6 - Scatter diagram of bounding boxes

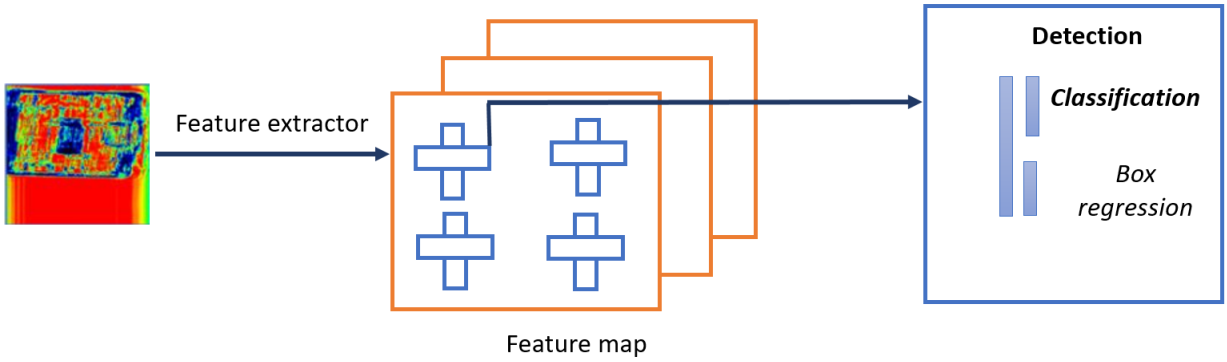

Figure 7 - One-stage detector Architecture

YOLOv3

YOLOv3 is the improved algorithm for YOLOv2 [27] and uses the darknet-53 network as the feature extraction network and the residual module (residual block).

In order to better detect objects of different scales, YOLOv3 draws lessons from feature pyramid network[45].We refer readers to [45–47] for further information on YOLOv3 regarding the ideas of outputting the three feature graph maps of different sizes. The overall structure of YOLOv3 as shown in figure 8.

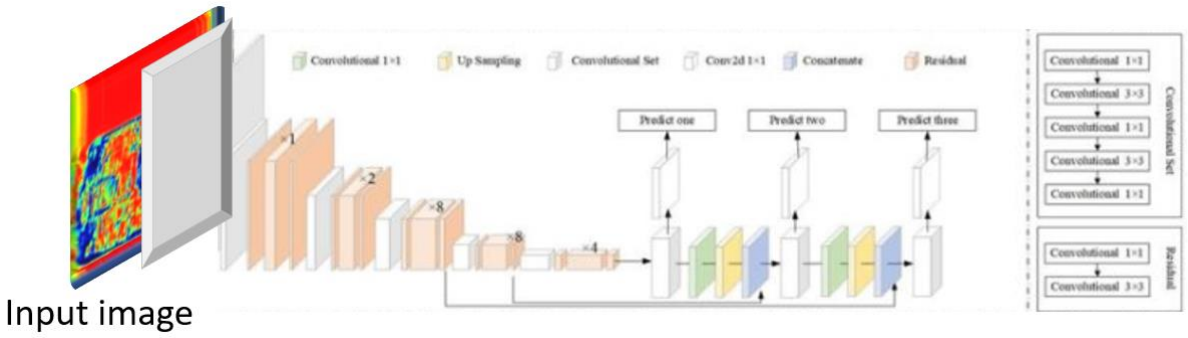

Figure 8 - YOLOv3 Model Structure [48]

The loss function of YOLOv3 is composed of the loss sum of three output characteristic graphs. The loss of each feature graph is composed of bounding box loss, object confidence loss and non-object confidence loss (non-object confidence). The expression is as follows:

$$
\begin{aligned}
loss = \lambda_{box} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_{i,j}^{obj} \left[(t_x - t_x)^2 + (t_x - t_x)^2 + (t_x - t_x)^2 + (t_x - t_x)^2 \right] \\
+ \lambda_{box} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_{i,j}^{obj} \left[-\log(p_c) + \sum_{c \in classes} BCE(\hat{c}_i, c_i) \right] \\
+ \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_{i,j}^{obj} \left[-\log(1 - p_c) \right]
\end{aligned} \tag{1}
$$

In the above formula, the weight control parameter, *tx*, *ty*, *tw*, *th* is the offset, 1_i^{obj} denotes if object appears in cell *i* and 1_{ij}^{obj} denotes that the *j*-th bounding box predictor in cell *i* is "responsible" for that prediction that needs to be learned.

Single Shot MultiBox Detector (SSD)

The Single Shot Detector (SSD) [48] is one of the first attempts to use the pyramidal characteristic hierarchy of the convolutional neural network to detect items of different sizes effectively.

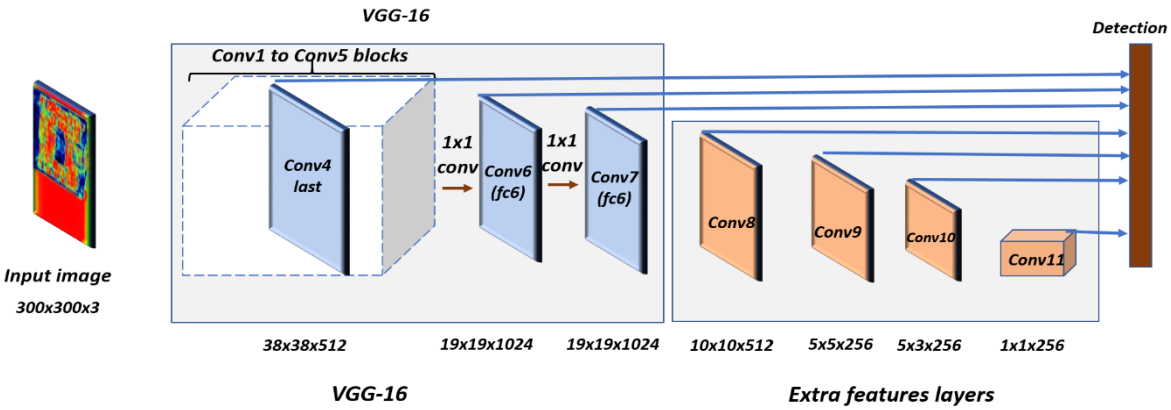

Figure 9 - SDD Model Structure

SDD loss function is the sum of a localization loss and a classification loss.

$$L = \frac{1}{N}(L_{cls} + \alpha L_{loc}) \qquad (2)$$

Where N is the number of bounding boxes matched and where α balances the weights between two losses, chosen by cross validation. The loss of localization is a smooth loss of L1 between the expected correction of the bounding box and the true values. The transformation of the correction of coordinates is the same as what R-CNN does in the regression of bounding boxes.

Where 1_{ij}^{match} indicates whether the i-th bounding box with coordinates $(p_x^i, p_y^i, p_w^i, p_h^i)$ is matched to the j-th ground truth box with coordinates $(g_x^i, g_y^i, g_w^i, g_h^i)$ for any object $d_m^i, m \in \{x, y, w, h\}$.

$$\mathcal{L}_{\text{cls}} = -\sum_{i \in \text{pos}} 1_{ij}^{k} \log(\hat{c}_i^k) - \sum_{i \in \text{neg}} \log(\hat{c}_i^0), \text{ where } \hat{c}_i^k = \text{softmax}(c_i^k) \qquad (3)$$

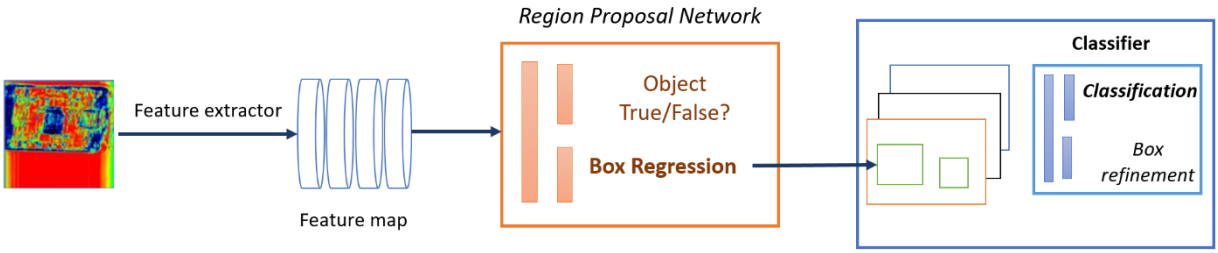

Figure 10 - Two-Stage detector architecture

Where 1_{ij}^{k} indicates whether the *i-th* bounding box and the *j-th* ground truth box are matched for an object in class k [48].

The R-CNN clusters of models are all regional focused from figure 8. The detection takes place in two stages: (1) first, by selecting search or regional proposal network the model proposes a collection of regions of interest. The regions proposal network are sparse as possible candidates for the bounding box may be infinite. (2) Secondly, only the region candidates are processed by a classifiers [49].

Faster R-CNN

To incorporate the area proposal algorithm into the CNN model, an intuitive speed-up solution is Faster R-CNN [50] which does exactly this: create a single, unified model consisting of RPN (region proposal network) and fast R-CNN with shared convolution layers.

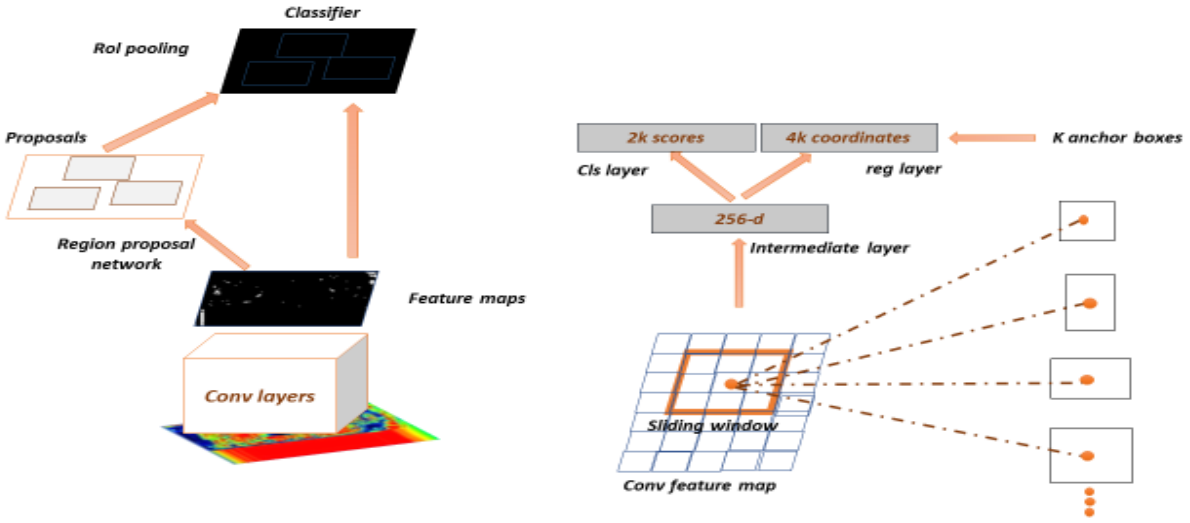

Figure 10 - Faster R-CNN Model Structure

Faster R-CNN is optimized for a multi-task loss function [49]

Symbol Explanation

p_i Predicted probability of anchor i being an object.

p_i^* Ground truth label (binary) of whether anchor i is an object.

t_i Predicted four parameterized coordinates.

t_i^* Ground truth coordinates.

N_{cls} Normalization term, set to be mini-batch size (~256) in the paper.

N_{box} Normalization term, set to the number of anchor locations (~2400) in the paper

λ A balancing parameter, set to be ~10 in the paper (so that both L_{cls} and L_{box} term are roughly equally weighted.

The multi-task loss function combines the losses of classification and bounding box regression:

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{box}}$$
$$\mathcal{L}(\{p_i\},\{t_i\}) = \frac{1}{N_{\text{cls}}}\sum_i \mathcal{L}_{\text{cls}}(p_i, p_i^*) + \frac{\lambda}{N_{\text{box}}}\sum_i p_i^* \cdot L_1^{\text{smooth}}(t_i - t_i^*) \quad (4)$$

Where L_{cls} is the log loss function over two classes, as we can easily translate a multi-class classification into a binary classification by predicting a sample as being a target object otherwise L_1^{smooth} is the smooth L_1 loss.

$$\mathcal{L}_{\text{cls}}(p_i, p_i^*) = -p_i^* \log p_i - (1 - p_i^*)\log(1 - p_i \quad (5)$$

Mask/Cascade R-CNN

Mask R-CNN [32] extends Faster R-CNN to segmentation of images at pixel level. The key point is to decouple prediction activities from the classification and the pixel-level mask.

It introduced a third branch, based on the Faster R-CNN architecture, to predict an object mask in parallel with existing branches for classification and localization. The mask branch is a small completely linked network added to each RoI predicting a pixel-to - pixel segmentation mask.

Since segmentation at the pixel level involves much more fine-grained alignment than bounding boxes, the R-CNN mask enhances the RoI pooling layer (named " PsRoI Pooling layer") so that RoI can be better and more accurately mapped to the regions of the original image. The PsRoI Pooling layer is designed to fix the location misalignment caused by quantization in the RoI pooling. Bilinear interpolation is used to measure the input values of floating-point positions [51].

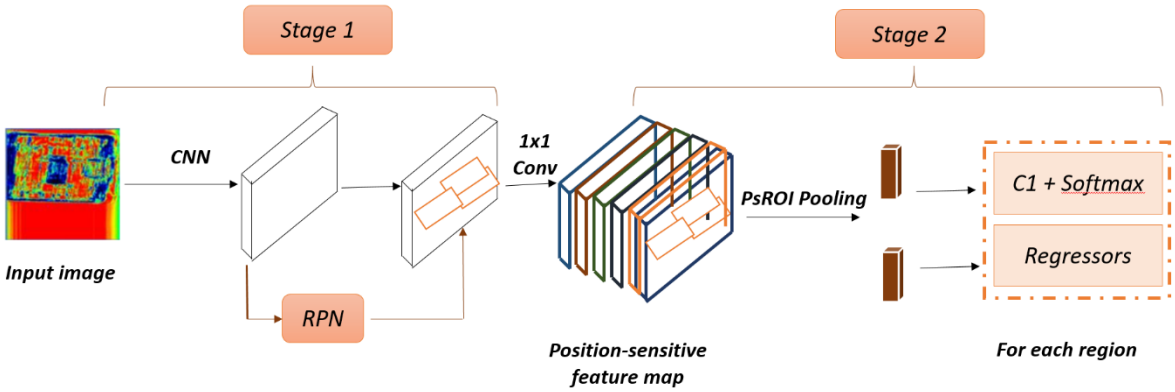

Figure 11 - Mask R-CNN Model Structure

The multi-task loss function of Mask R-CNN combines the loss of classification; localization and segmentation mask $\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{box} + \mathcal{L}_{mask}$ where

$\mathcal{L}_{cls}$ and $\mathcal{L}_{box}$ are same as in Faster R-CNN. The mask branch generates a mask of dimension m x m for each RoI and each class; K classes in total. Thus, the total output is of size K*m^2.

$\mathcal{L}_{mask}$ is defined as the average binary cross-entropy loss, only including k-th mask if the region is associated with the ground truth class k.

$$L_{mask} = -\frac{1}{m^2}\sum_{1\le i,j\le m}[y_{ij}\log\hat{y}_{ij}^{k} + (1-y_{ij})\log(1-\hat{y}_{ij}^{k}) \quad (6)$$

where y_{ij} is the label of a cell (i, j) in the true mask for the region of size m x m; $\hat{y}_{ij}^{k}$ is the predicted value of the same cell in the mask learned for the ground-truth class k [49].

Experimental Results and Discussion

In this section, we first introduce the indicators used to evaluate the accuracy of the detection model. Secondly, we compare and analyze the detection results of the five detection models mentioned above under the terahertz image data set, and discuss the differences between them.

Finally, we select a model with the best detection speed and accuracy as our final detection algorithm for security detection of terahertz images. The hardware and software configuration of experiments is shown in Table 11.

Table 3. Hardware and software configuration of experiment

Hardware/Software	Parameters
Operating System	Ubuntu18.04 LTS 64bit (Linux 4.15)
Central Processing Unit	Intel(R) Core (TM) i7-7800X CPU @ 3.50GHz
Graphical Processing Unit	NVIDIA RTX 2080(8G)
RAM	DDR4 32G
CUDA	CUDA 10.1
cuDNN	cuDNN 7.6.1
Deep Learning Framework	PyTorch 1.4

Evaluation for Metric Detection

In this paper, we adopted the detection metrics introduced in [22] applied to table 5, which includes average precision (AP) over multiple Intersection over Union (IoU) values. The IoU can be calculated in equation 7.

$$IoU(Box_{pred}, Box_{gt}) = \frac{Box_{pred} \cap Box_{gt}}{Box_{pred} \cup Box_{gt}} \quad (7)$$

The calculations of precision and recall are shown in Figure 12. Traditionally, The average precision (AP) is a detection measure which combines the classification accuracy and location accuracy for each object. *mAP* is the mean AP for all objects. (likewise AR and mAR).

Unless otherwise specified, AP and mAP used in this paper. The detection metrics are listed in Table 4.

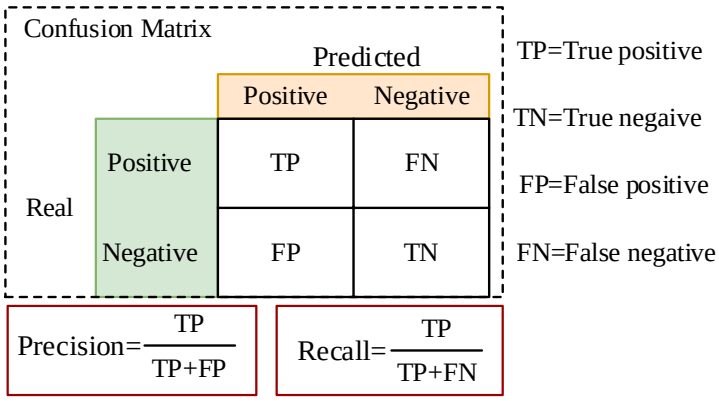

Figure 12 - Calculation of precision and recall.

Datasets and Training Configuration

The target detection task not only needs to determine which object it belongs to, but also needs to determine the location of the object. This paper uses the evaluation indicators used in the COCO data set and the labelImg software tool. We included the average precision (AP) and average recall (AR) indicators, with results further subdivided into: under small (area smaller than 32px*32px), medium (area between 32px*32px and 96px*96px) and large (area larger than 96px*96px) detection areas. Finally, the augmented data total of 1884 images, and divide the training, and test sets at the proportion of 4:1:1. Finally, we randomly divided the data samples get 1205 training sets, 302 validation sets and 377 images of the test set. In order to ensure the rationality of the algorithm comparison, we uniformly use the parameters shown in table 5 during the model-training phase.

AP (averaged across all 10 IoU thresholds and all categories) should be considered the most important metric when considering model performance in our research. For metric AR, the larger the value the lesser the false negative rate which is important for defect inspection along the overhead transmission line. Finally, in order to measure the image detection speed of different models, we also use the detection speed index: frame per second (FPS).

To train the defect detection model introduced in previous section 2, it is essential to setup the training configurations properly. In our research, we use the configurations listed in Table 5.

Table 4. Dataset information for training and testing

Metric	Meaning
AP	Average precision for [IoU = 0.50:0.95 | area = all | maxDets = 100]
$AP^{@0.5}$	Average precision for [IoU = 0.50 | area = all | maxDets = 100]
$AP^{@0.75}$	Average precision for [IoU = 0.75 | area = all | maxDets = 100]
AP^{small}	Average precision for [IoU = 0.50:0.95 | area = small | maxDets = 100]
AP^{medium}	Average precision for [IoU = 0.50:0.95 | area = medium | maxDets = 100]
AP^{large}	Average precision for [IoU = 0.50:0.95 | area = large | maxDets = 100]
AR^{1}	Average recall for [IoU = 0.50:0.95 | area = all | maxDets = 1]
AR^{10}	Average recall for [IoU = 0.50:0.95 | area = all | maxDets = 10]
AR^{100}	Average recall for [IoU = 0.50:0.95 | area = all | maxDets = 100]
AR^{small}	Average recall for [IoU = 0.50:0.95 | area = samll | maxDets = 100]
AR^{medium}	Average recall for [IoU = 0.50:0.95 | area = medium | maxDets = 100]
AR^{large}	Average recall for [IoU = 0.50:0.95 | area = large | maxDets = 100]

Table 5. Models Training Parameters

Parameter	Value
Training epochs	50f
Optimizer	Adam
Learning rate	1e-4
Batch size	8

Evaluation results

After completing the model training, we evaluate the performance of the five models on the test set. Since there are eight categories to be detected, we first present the overall results in table 6.

As can be seen from table 6, the yolov3 model has the best results in evaluating the overall performance of the dataset.

Table 6. Models Evaluation Performance

Metric	yolov3	SSD300	SSD512	Faster RCNN	Cascade RCNN
AP	0.739	0.713	0.716	0.562	0.632
$AP^{0.5}$	0.992	0.963	0.981	0.884	0.878
$AP^{0.75}$	0.854	0.774	0.828	0.596	0.669
AP^{small}	0.51	0.339	0.345	0.26	0.221
AP^{medium}	0.705	0.681	0.695	0.524	0.599
AP^{large}	0.816	0.781	0.776	0.643	0.682
AR^{1}	0.763	0.747	0.744	0.624	0.678
AR^{10}	0.791	0.759	0.756	0.652	0.697
AR^{100}	0.791	0.759	0.756	0.652	0.697
AR^{small}	0.699	0.366	0.472	0.358	0.282
AR^{medium}	0.761	0.727	0.733	0.617	0.658
AR^{large}	0.85	0.823	0.808	0.709	0.74

The higher the AP, the higher the true positive rate, the average recall, and the smaller the false negative rate, which is very important in the field of security. To ensure high recognition accuracy, it is also necessary to ensure a lower missed detection rate.

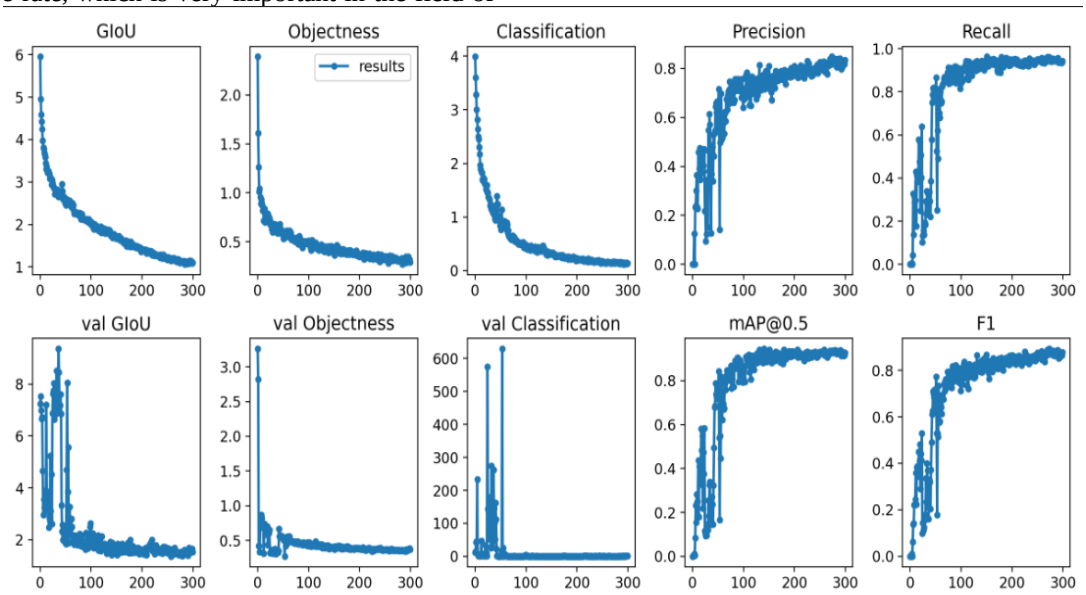

Figure 13 - Yolov3 training Model

The more surprising result is that the performance of two-stage detector Faster RCNN and Cascade RCNN on terahertz data is not satisfactory, and its AP index decreases by 23.9% and 14.5% respectively compared with yolov3. The detection effect of the model is the worst. For SSD300 and SSD512 models, there is little difference in AP indicators, but in AP0.5 and AP0.75 indicators, SSD512 model is 1.9% and 6.5% higher than SSD300 respectively. The reason is that SSD300 zooms to 300px*300px size during image input, while the original image is 512px256px. A lot of feature information is lost after image compression, resulting in a decline in SSD300 performance. However, compared with the yolov3 model, the SSD500 model still decreased by 3.1% under the AP index. The calculations of precision and recall are shown in Figure 6.

In table 7, we analyze the recognition effect of six (6) categories and select AP as the analysis index, which is stricter to the detection accuracy.

Table 7. Model Detection Accuracy for hiding weapons & non-weapons

Category	yolov3	SSD300	SSD512	Faster RCNN	Cascade RCNN
Screw drive	0.659	0.642	0.63	0.409	0.449
blade	0.539	0.479	0.557	0.26	0.385
knife	0.642	0.563	0.594	0.326	0.33
scissors	0.695	0.605	0.608	0.433	0.455
Board marker	0.78	0.803	0.771	0.663	0.773
Mobile phone	0.899	0.878	0.875	0.837	0.886
Wireless mouse	0.832	0.868	0.826	0.748	0.907
Water bottle	0.87	0.866	0.865	0.816	0.872
mAP	0.740	0.713	0.716	0.562	0.632

Combined with the detection accuracy results obtained above, we use AP as abscissa and AR as ordinate to draw figure 13 to show the results, in which the circle size represents the speed of model reasoning.

Table 8. Inference Models with time

Model	yolov3	SSD300	SSD512	Faster RCNN	Cascade RCNN
Inference time (ms per image)	21.2	50.4	61	79.6	61

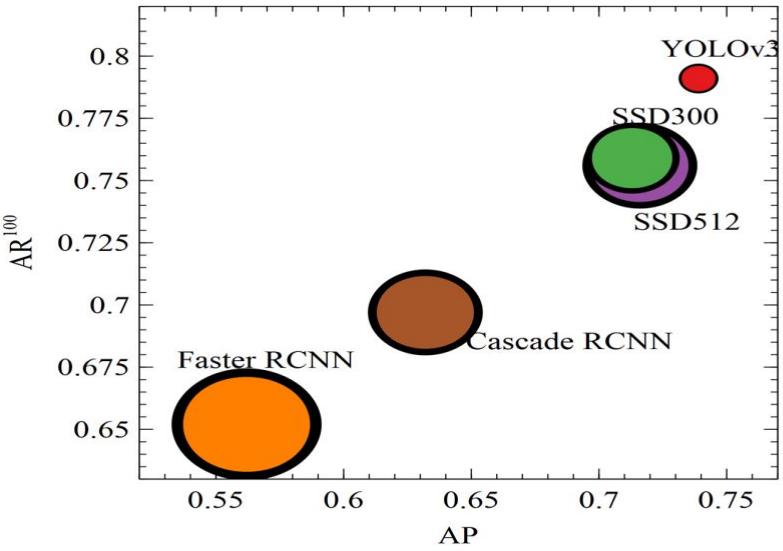

Figure 14 - Comparison of models' detection performance

From figure 14, we can see that while ensuring the detection performance, yolov3 also ensures the detection speed. Compared with other models, yolov3 is more superior. Figure 15 shows the recognition results of yolov3 in different categories.

Figure 15 - True-Recognition of hidden weapons and non- weapons

It can be seen that it is good to identify the type and the specific location of objects. However, when we check the model test results, we also find that the probability of false recognition of category blade is relatively high, and it is easy to be wrongly identified as scissors, knife, and screen drive as shown in figure 15.

It is necessary to further improve the recognition effect of this category.

Optimizing YOLOv3

From the previous experimental results, we can see that among the single-stage and two-stage algorithms proposed in this paper, the YOLOv3 algorithm is currently the best in terms of detection accuracy and speed, and the detection speed can achieve 21.2ms image, which is close to the acquisition speed of terahertz images.

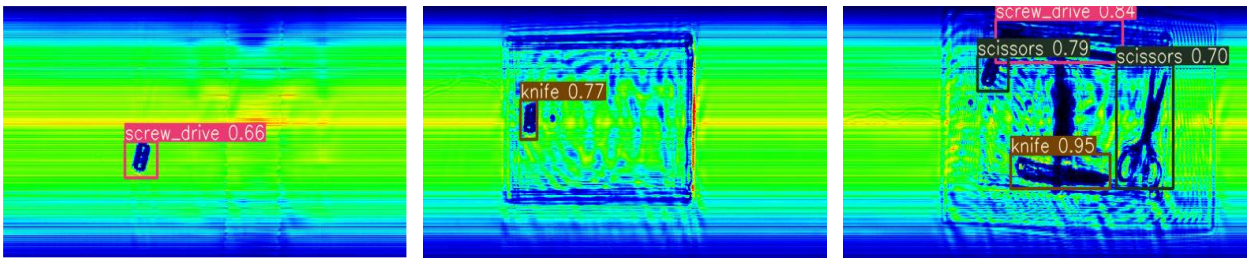

Figure 15 - Falsehood –Recognition of hidden weapons

In order to further improve the detection speed and ensure sufficient accuracy, we will improve the backbone and neck parts of YOLOv3 in this section.

Optimizing Backbone

The overall structure of YOLOv3 includes data input, backbone, neck and head, for detection. The schematic diagram of the structure is shown in figure 16.

In the previous experiment, the backbone of the YOLOv3 detection model is the darknet53 network proposed by [27] which draws lessons from the resnet network, so there are more convolution layers to extract image features.

Here we draw lessons from the cross stage partial structure proposed in the [27] to transform the darknet53, and its structure is shown in figure 16.

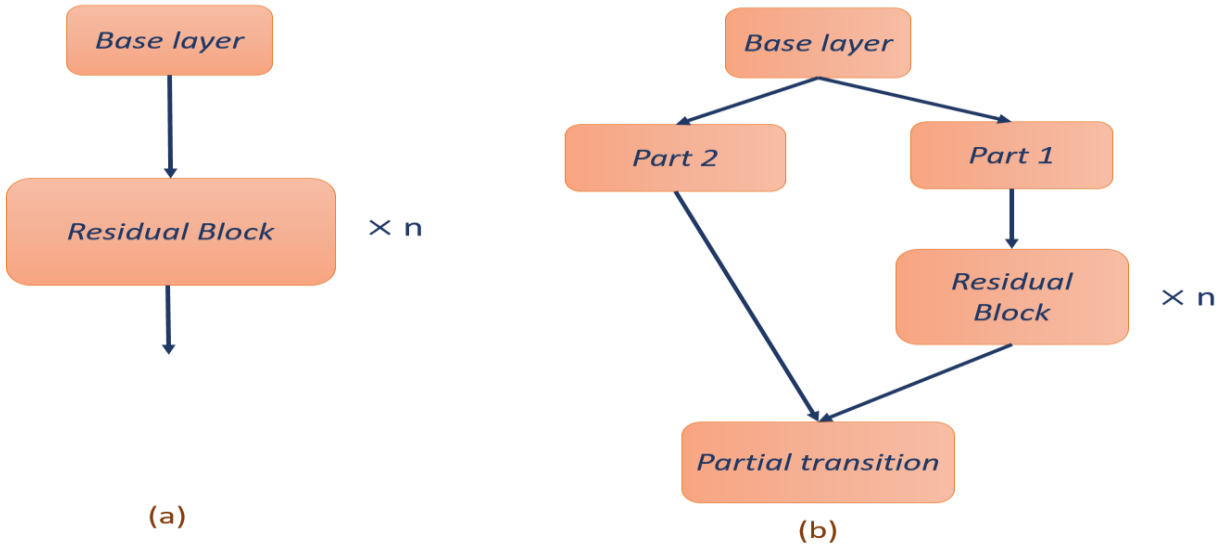

Figure 16 - Our model structure

The graph (a) is the original darknet structure, and the graph (b) is the improved structure where Partial transition represents the convolution operation and pooling operation. Using this structure can not only ensure accuracy, but also effectively reduce the calculation of the model, so as to obtain faster reasoning speed.

Optimizing Neck

In the neck section, our improvement is to add PANet and SPP structures. The structure of PANet is as shown in figure 17.

The function of adding bottom-up path augmentation network to the FPN network is to make full use of the object position information in the shallow features, which will help to improve the target position detection accuracy of the network.

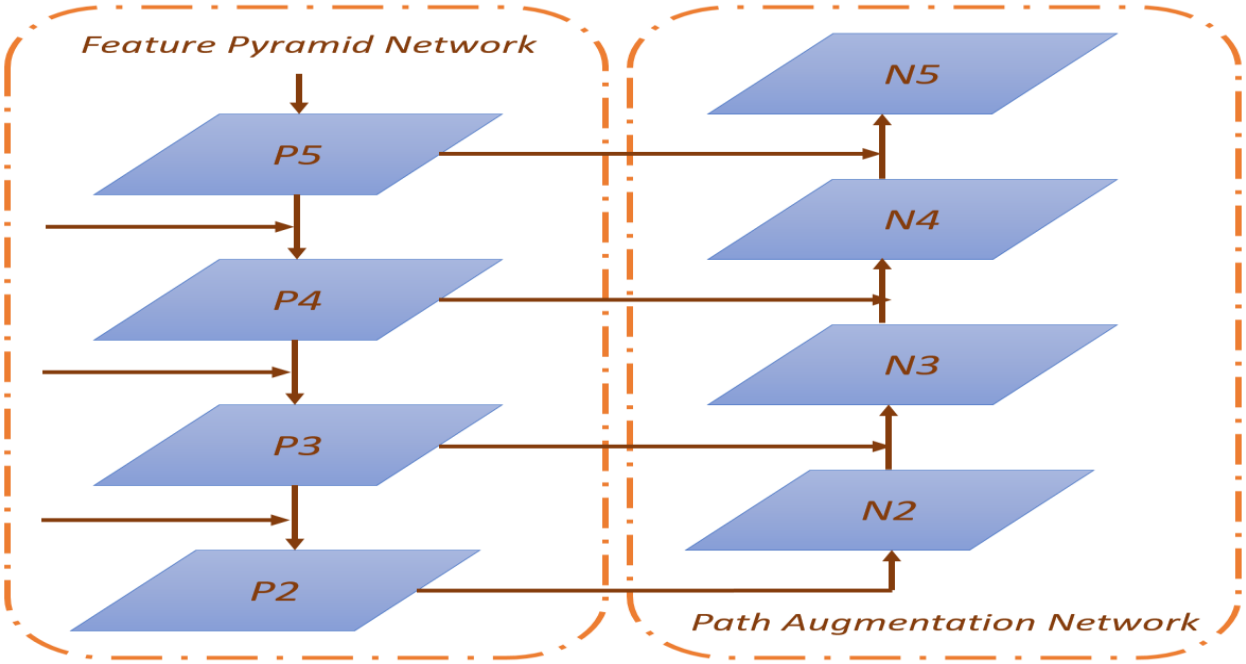

Figure 17 - The structure of PANet

SPP network solves the problem of fusion of different size features, and can ensure that images of different sizes are input during training and testing.

Experimental Results and Discussion

After improving the structure of YOLOv3, we use the previous experimental conditions and data sets to retrain, and evaluate the results of the improved model on the test set. The training graph and results are shown in figure 18.

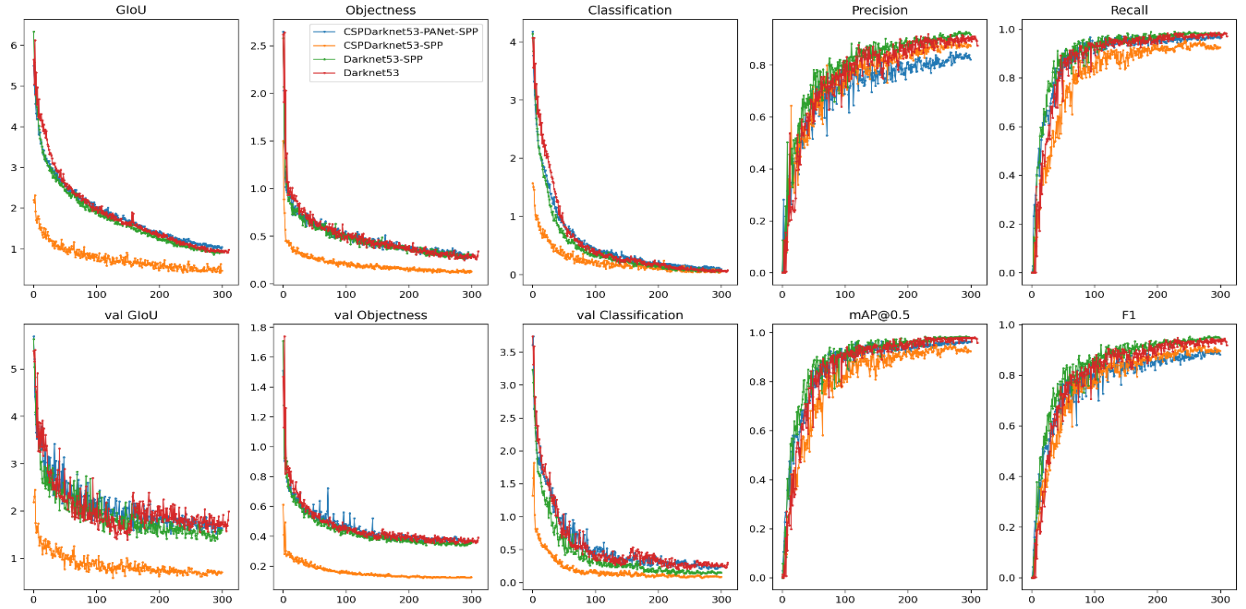

Figure 17 - Our Optimization Training Model

Table 8. Optimized model

Model	AP	AP0.5	AP0.75	AP_{small}	AP_{medium}	AP_{large}	Inference time(ms)
Darknet53	0.739	0.992	0.854	0.51	0.705	0.816	21.2
Darknet53-SPP	0.735	0.991	0.855	0.591	0.703	0.8	21.5
CSPDarknet53-SPP	0.733	0.986	0.864	0.534	0.716	0.788	6.9
CSPDarknet53-PANet-SPP	0.748	0.99	0.873	0.585	0.725	0.804	8.4

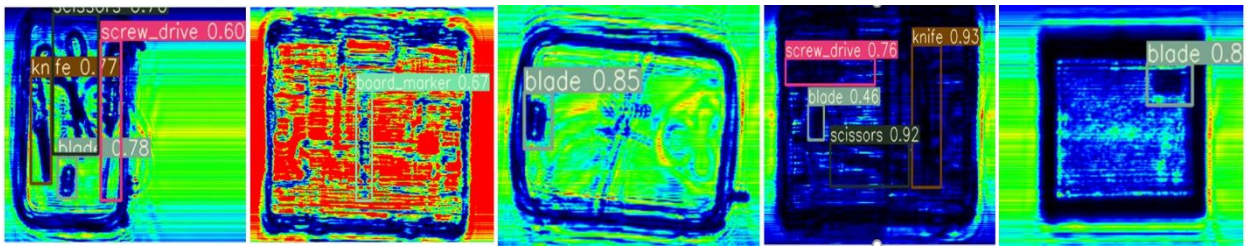

Figure 18 – Improved Yolov3 model correctly identifies blade

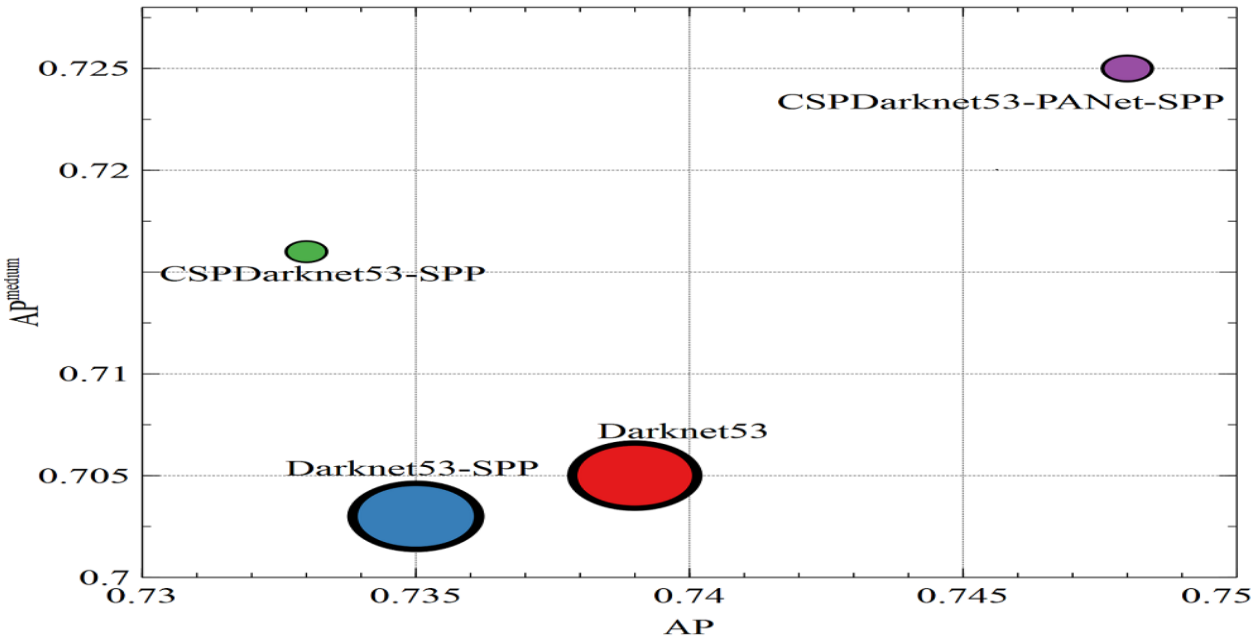

Figure 17 - Comparison of different optimized models

After the backbone structure of YOLOv3 blade detection has been signiificantly improved better than the former at Figure 18.

From the comparison results of the above table, it can be seen that the detection speed of the modified backbone model CSPDarknet53-SPP and CSPDarknet53-PANet-SPP is 3.1 and 2.5 times faster than that of the yolov3 model with Darknet as backbone, respectively.

Among them, the reasoning speed of CSPDarknet53-PAN-SPP model is 17% slower than that of CSPDarknet53-SPP without PAN (path aggregation network), and the increased time loss mainly occurs in the process of characteristic information propagation in PANet.

However, the adoption of the PANet structure also brings a 2% performance improvement to the AP index of the model.

From the optimization results of yolov3, the reasoning speed of the model is greatly improved by using the new CSPDarknet53 network, but the improvement in prediction accuracy is not very

obvious. Considering the detection speed and accuracy, CSPDarknet53-PANet-SPP network will be the best choice for detecting terahertz images.

Conclusions

In this paper, we perform a comprehensive comparative study of five deep learning detection algorithms for detecting terahertz hidden weapons and non-weapons objects. We also performed data argumentation to increase the database. In other terms, the greater the Dataset, the greater the improved detection method efficiency achieved.

On the successful distribution of hidden weapons and non-weapons in terahertz image, we implement large average recall and average precision of intersection of union (IOU) parameters. In addition, we pooled this approach and Yolov3 into a uniform terahertz detection system and it performed the best as compared to other deep learning models proving that one-way detection method for hidden weapons and non-weapons is far better than two-way detection methods.

It can also be deduced from experimental results that after optimization of *Yolov3*, the detection speed and accuracy of our new model out performs five elected existing models with respect to the detection of hidden weapons and non-weapons in terahertz images.

References:

1. Yang, X., Wu, T., Zhang, L., Yang, D., Wang, N., Song, B., & Gao, X. (2019). CNN with spatio-temporal information for fast suspicious object detection and recognition in THz security images. *Signal Processing*. https://doi.org/10.1016/j.sigpro.2019.02.029
2. Strecker, H. (1983). A local feature method for the detection of flaws in automated X-ray inspection of castings. *Signal Processing*. https://doi.org/10.1016/0165-1684(83)90005-1
3. (n.d.). Terahertz science & technology. *The international journal of THz*. Scinco Inc, Williamsburg VA.
4. (2007). *Assessment of millimeter-wave and terahertz technology for detection and identification of concealed explosive and weapons.* National Academies Press, Washington D.C.
5. (2018). *Initial Assessment of the Measurement and Retrieval Performance of the Upper-Atmospheric Terahertz Limb-Sounder LOCUS.* Science and Technology Facilities Council, Didcot.
6. (2011). *Terahertz and mid infrared radiation. Generation, detection and applications*. NATO science for peace and security series. B. Springer, Dordrecht The Netherlands.
7. Zhang, X.-C., & Xu, J. (2010). In*troduction to THz wave photonics*. Springer, New York.
8. Haralick, R.M., Shanmugam, K., & Dinstein, I.'H. (1973). *Textural Features for Image Classification.* IEEE Trans. Syst., Man, Cybern. https://doi.org/10.1109/TSMC.1973.4309314
9. Fogel, I., & Sagi, D. (1989). *Gabor filters as texture discriminator*. Biol. Cybern. https://doi.org/10.1007/BF00204594
10. Guo, T., Seyed Mousavi, H., Huu Vu, T., & Monga, V. (2017). *Deep Wavelet Prediction for Image Super-Resolution*. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 104–113.
11. Yu, S.-N., Li, K.-Y., & Huang, Y.-K. (2006). *Detection of microcalcifications in digital mammograms using wavelet filter and Markov random field model.* Computerized Medical Imaging and Graphics. https://doi.org/10.1016/j.compmedimag.2006.03.002
12. Guo, M.-F., Zeng, X.-D., Chen, D.-Y., & Yang, N.-C. (2018). *Deep-Learning-Based Earth Fault Detection Using Continuous Wavelet Transform and Convolutional Neural Network in Resonant Grounding Distribution Systems.* IEEE Sensors J. https://doi.org/10.1109/JSEN.2017.2776238
13. Charnes, A., Frome, E.L., & Yu, P.L. (1976). The Equivalence of Generalized Least Squares and Maximum Likelihood Estimates in the Exponential Family. Jour*nal of the American Statistical Association.* https://doi.org/10.1080/01621459.1976.10481508
14. Lowe, D.G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*. https://doi.org/10.1023/b:visi.0000029664.99615.94
15. Lindeberg, T. (2012). *Scale Invariant Feature Transform.* Scholarpedia. https://doi.org/10.4249/scholarpedia.10491

16. Felzenszwalb, P., McAllester, D., & Ramanan, D. (2008). *A discriminatively trained, multiscale, deformable part model*. In: Staff, I. (ed.) 2008 IEEE Conference on Computer Vision and Pattern Recognition. 2008 IEEE Conference on Computer Vision and Pattern Recognition, 2008/06. I E E E, [Place of publication not identified] (2008). https://doi.org/10.1109/cvpr.2008.4587597
17. El-Naqa, I., Yongyi Yang, Wernick, M.N., Galatsanos, N.P., & Nishikawa, R.M. (2002). *A support vector machine approach for detection of microcalcifications*. IEEE Transactions on Medical Imaging. https://doi.org/10.1109/tmi.2002.806569
18. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., & Jackel, L.D. (1990). Handwritten digit recognition with a back-propagation network.Advances in Neural Information Processing Systems, 2, 396?404, *Morgan Kaufman. Advances in Neural Information Processing Systems* 2, 396.
19. Ordóñez, F., & Roggen, D. (2016). *Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition.* Sensors. https://doi.org/10.3390/s16010115
20. Fukushima, K. (1980). Neocognitron. A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*. https://doi.org/10.1007/bf00344251
21. Krizhevsky, A., Sutskever, I., & Hinton, G.E. (2012). *Imagenet classification with deep convolutional neural networks*. Paper presented at advances in neural information processing systems.
22. (2014). *Simonyan: Very deep convolutional networks for large-scale image recognition.* arXiv preprint arXiv:1409.1556.
23. He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition.* IEEE Trans. Pattern Anal. Mach. Intell. https://doi.org/10.1109/TPAMI.2015.2389824
24. (2016). He: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. *Deep residual learning for image recognition*, 770.
25. Xie, S., Girshick, R.B., Dollár, P., Tu, Z., & He, K. (2017). Aggreg*ated Residual Transformations for Deep Neural Networks.* In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, pp. 5987–5995 (2017). https://doi.org/10.1109/CVPR.2017.634
26. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q. (eds.) (2017). *Densely Connected Convolutional Networks.* 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
27. Redmon, J., & Farhadi, A. (2018). YOLOv3. An Incremental Improvement. CoRR abs/1804.02767
28. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). *MobileNets. Efficient Convolutional Neural Networks for Mobile Vision Applications*. CoRR abs/1704.04861.
29. Lazebnik, S., Schmid, C., & Ponce, J. (2006). *Beyond Bags of Features. Spatial Pyramid Matching for Recognizing Natural Scene Categories*. In: CVPRW '06. 2006 Conference on Computer Vision and Pattern Recognition Workshop : 17-22 June 2006. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06). IEEE, New York (2006). https://doi.org/10.1109/cvpr.2006.68
30. Girshick, R. (2015). *Fast R-CNN.* In: 2015 IEEE International Conference on Computer Vision. 11-18 December 2015, Santiago, Chile : proceedings. 2015 IEEE International Conference on Computer Vision (ICCV), 2015/12. IEEE, Piscataway, NJ (2015). https://doi.org/10.1109/iccv.2015.169
31. (2015). Ren: Advances in Neural Information Processing Systems. *Faster r-cnn: towards real-time object detection with region proposal networks*, 91.
32. He, K., Gkioxari, G., Dollár, P., & Girshick, R.B. (2017). *Mask R-CNN*. CoRR abs/1703.06870
33. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). *Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: Computer Vision and Pattern Recognition (CVPR)*, 2014 IEEE Conference on. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014/06. [publisher not identified], [Place of publication not identified] (2014). https://doi.org/10.1109/cvpr.2014.81
34. Singh, B., & Davis, L.S. (2017). *An Analysis of Scale Invariance in Object Detection* - SNIP.
35. Zhou, P., Ni, B., Geng, C., Hu, J., & Xu, Y. (eds.) (2018). *Scale-Transferrable Object Detection. 2018* IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.
36. Xie, H., & Wu, Z. (2020). *A Robust Fabric Defect Detection Method Based on Improved RefineDet*. Sensors. https://doi.org/10.3390/s20154260
37. Womg, A., Shafiee, M.J., Li, F., Chwyl, B. (eds.) (2018). *Tiny SSD.* A Tiny Single-Shot Detection

Deep Convolutional Neural Network for Real-Time Embedded Object Detection. 2018 15th Conference on Computer and Robot Vision (CRV). 2018 15th Conference on Computer and Robot Vision (CRV).

38. Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN. Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. https://doi.org/10.1109/tpami.2016.2577031
39. Li, J., Liang, X., Shen, S., Xu, T., Feng, J., & Yan, S. (2017). Sc*ale-aware Fast R-CNN for Pedestrian Detection.* IEEE Transactions on Multimedia. https://doi.org/10.1109/tmm.2017.2759508
40. He, K., Gkioxari, G., Dollár, P., & Girshick, R.B. (2020). *Mask R-CNN*. IEEE Trans. Pattern Anal. Mach. Intell. (2020). https://doi.org/10.1109/TPAMI.2018.2844175
41. Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). *Path Aggregation Network for Instance Segmentation*.
42. Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., & Sun, J. (2018). *DetNet*. A Backbone network for Object Detection.
43. Everingham, M., van Gool, L., Williams, C.K.I., Winn, J., & Zisserman, A. (2009). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*. https://doi.org/10.1007/s11263-009-0275-4
44. (2014). Lin: Microsoft COCO. *Common Objects in Context* Volume 8693.
45. Li, X., Lai, T., Wang, S., Chen, Q., Yang, C., Chen, R., Lin, J., & Zheng, F. (2019). *Weighted Feature Pyramid Networks for Object Detection.* In: 2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking, ISPA/BDCloud/SocialCom/SustainCom 2019, Xiamen, China, December 16-18, 2019, pp. 1500–1504. https://doi.org/10.1109/ISPA-BDCloud-SustainCom-SocialCom48970.2019.00217
46. Lin, T.-Y., Dollár, P., Girshick, R.B., He, K., Hariharan, B., & Belongie, S.J. (2017). *Feature Pyramid Networks for Object Detection*. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, pp. 936–944. https://doi.org/10.1109/CVPR.2017.106
47. Liang, Y., Wang, C., Li, F., Peng, Y., Lv, Q., Yuan, Y., & Huang, Z. (2019). TFPN. Twin Feature Pyramid Networks for Object Detection. In: 31st IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2019, Portland, OR, USA, November 4-6, 2019, pp. 1702–1707. https://doi.org/10.1109/ICTAI.2019.00251
48. Liao, M., Shi, B., Bai, X., Wang, X., & Liu, W. (2016). TextBoxes. A Fast Text Detector with a Single Deep Neural Network. CoRR abs/1611.06779
49. Weng, L. (2018). *Object Detection Part 4. Fast Detection Models.* lilianweng.github.io/lil-log
50. Ren, Y., Zhu, C., Xiao, S., & Marotti de Sciarra, F. (2018). Object Detection Based on Fast/Faster RCNN Employing Fully Convolutional Architectures. *Mathematical Problems in Engineering*. https://doi.org/10.1155/2018/3598316
51. Zhu, Y., Zhao, C., Guo, H., Wang, J., Zhao, X., & Lu, H. (2019). *Attention CoupleNet. Fully Convolutional Attention Coupling Network for Object Detection.* IEEE transactions on image processing : a publication of the IEEE Signal Processing Society. https://doi.org/10.1109/TIP.2018.2865280