Multiscale and Direction Target Detecting in Remote Sensing Images via Modified YOLO-v4

Zakria, Jianhua Deng, Rajesh Kumar, Muhammad Saddam Khokhar, Jingye Cai, and Jay Kumar

***Abstract*—Traditional target detection algorithms have difficulty to adapt complex environmental changes and have limited applicable scenarios. However, the deep-learning-based target detection model can automatically learn with strong generalization capability. In this article, we choose a single-stage deep-learning-based target detection model for research based on the model's real-time processing requirements and to improve the accuracy and the robustness of target detection in remote sensing images. In addition, we improve the YOLOv4 network and present a new approach. First, we propose a classification setting of the nonmaximum suppression threshold to increase the accuracy without affecting the speed. Second, we study the anchor frame allocation problem in YOLOv4 and propose two allocation schemes. The proposed anchor frame scheme also improves the detection performance, and experimental results on the DOTA dataset validate their effectiveness.**

***Index Terms*—Image analysis, neural networks, object detection, remote sensing, YOLO.**

I. Introduction

A LARGE number of remote sensing images have been generating regularly, and due to the rapid development of satellite and imaging technology, the task of object detection has gained significant attention of researchers. The objective target detection in remote sensing images is to identify object of interest and, then, predict the type and location. In other words, target detection is the process of detecting instances of semantic objects of a certain class (such as humans, vehicles airplanes, or ships) in digital images and videos. Analyzing such images contributes to social and economic aspects for decision-making as they provide a valuable source of information. For this reason, it has been applied for many applications, such as navigation [1], disaster management [2], road segmentation [3], agriculture survey [4], urban planning [5], geographic information system updating [6], intelligent monitoring [7], and many more. However, target object detection for each application poses some nonoverlapping challenges.

Optical remote sensing images typically employ electromagnetic waves with a visible spectrum ranging from 400 to 760 nm. In the early days of remote sensing technology, the classification and recognition of remote sensing images relied on visual and manual marking methods. It is not only time-consuming and costly but also difficult to process a huge amount of remote sensing data within a specific time. Unlike target detection in landscape images where large objects are considered, dealing with optical remote sensing images poses more complicated challenges especially for multiscale small object detection. First, a small target image is typically of less than 30 pixels in size. Second, weather and environmental changes, such as occlusions due to building, atmosphere, and shadow, as well as other factors, such as different sizes of targets in the same image, different overhead views, similar colors between objects and their surroundings, and so on, can affect small targets performance in remotely sensed images. Furthermore, few more challenges are discussed in the following.

1) *Scale diversity:* Many remote sensing target categories and their size vary even in same categories; size changes, such as ships at ports, may be only tens of meters to more than 300 m in size. Furthermore, shooting height and distance from the target also affect the size.
2) *Shooting angle:* The optical remote sensing images are generally captured vertically from a high altitude. However, images in natural scenes are taken from the ground, so the mode of the same target is usually different. The detector trained on the traditional datasets may have a negative impact on the remote sensing images.
3) *Challenge in a small target:* Most of the time, many small targets are in one aerial image; as a result, we have minimal information related to the target. Therefore, it is easy to be filtered in the pooling layer of the convolutional neural network (CNN), and the feature dimension is also too low, and it is not easy to detect and recognize.
4) *Challenge in multidirection:* The direction of the target is overhead due to the shooting angle of view, while conventional images are horizontally captured on the ground. Therefore, there is greater certainty in the direction of the object, such as moving objects and plants generally in standing upright position on the ground.

Manuscript received September 2, 2021; revised December 26, 2021; accepted December 29, 2021. Date of publication January 6, 2022; date of current version January 20, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFC0821505, in part by Zhongyanggaoxiao under Grant ZYGX2018J075, and in part by the Sichuan Science and Technology Program under Grant 2019YFS0487. *(Corresponding author: Jianhua Deng.)*

Zakria, Jianhua Deng, and Jingye Cai are with the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China (e-mail: zakria.uestc@hotmail.com; jianhua.deng@uestc.edu.cn; jycai@uestc.edu.cn).

Rajesh Kumar and Jay Kumar are with the Yangtze Delta Region Institute (Huzhou), University of Electronic Science and Technology of China, Huzhou 313001, China (e-mail: rajakumarlohano@gmail.com; Jay_tharwani1992@yahoo.com).

Muhammad Saddam Khokhar is with the School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China (e-mail: saddam_khokhar@hotmail.com).

Digital Object Identifier 10.1109/JSTARS.2022.3140776

5) *High complexity of the background:* The optical remote sensing satellite is imaging the ground; it will take indiscriminate shooting and the ground surface information, including bare soil, forests, hills, oceans, etc. Optical remote sensing images generally have a very large field of view, and the image contains a background of different colors, which includes the strong interference in testing.

The fast advancement of remote sensing techniques has substantially increased the quality and quantity of remote sensing images that characterize many objects on the earth's surface, such as buildings, airports, airplanes, and so on. However, the aerial, naval, and land vehicle detection task is among the most challenging and highly focused tasks due to its versatile applicability. A series of works has been proposed for vehicle detection from a landscape view. Among the initial works for vehicle detection in the landscape angle, Hu *et al.* [8] proposed SENet to enhance or suppress the feature information by learning the weight of each channel. Woo *et al.* [9] proposed the hybrid domain attention mechanism, i.e., convolutional block attention module, which combined spatial attention and channel attention. Likewise, Liu *et al.* [10] combined channel attention with YOLOv3, and it enhanced the ability of the network to distinguish the background and the target. In addition to such models, many lightweight networks are also proposed to increase the detection speed, including SqueezeNet [11] and MobileNets [12], to mention a few. Fang *et al.* [13] combined SqueezeNet with YOLOv3-tiny and get a tinier network. However, these techniques cannot be directly applied to optical remote sensing image target detection and identification. To the best of our knowledge, target vehicle detection for the remote sensing image has not been studied in previous works. This naturally brings a significant need of intelligent earth observation through automated analysis and understanding of aerial or satellite images.

This article proposes an improved YOLOv4 algorithm for the remote sensing image target detection model. The setting of the nonmaximum suppression (NMS) threshold and anchor allocation schemes are used to improve the YOLO-v4 target detection method. This solves the problems of multidirection target and small target detection in the remote sensing image environment and improves the target detection accuracy. This work provides the following contributions.

1) Target box dimension clustering is introduced, in which the K-means algorithm is used to cluster the target box of the remote sensing image dataset. The optimal width and height value is calculated, and the predefined anchor in YOLOv4 is modified accordingly.
2) We attempt to investigate the target detection in the remote sensing image dataset using different models to show the significance of target detection in remote sensing images.
3) We conduct a deep empirical study on target detection in the remote sensing image dataset to further examine the impact on the performance of the existing methods.

The rest of this article is organized as follows. The related literature is discussed in Section II. Section III provides the details of the proposed approach for target detection in remote sensing images. The experimentation settings, results, and analysis on publicly available datasets are described in Section IV. Finally, Section V concludes this article.

II. Related Work

CNN-based target identification techniques outperform conventional target detection algorithms, such as HOG Cascade [14], deep-spatial-spectral global reasoning [15], graph convolutional networks [16], multimodel deep learning [17], deformable parts model (DPM) [18], and histogram of oriented gradients—support vector machine (HOG-SVM) [19], to mention a few.

CNN-based target detection models are broadly classified into two types: 1) two-stage models and 2) one-stage models.

The region-based convolutional neural network (R-CNN) [20] followed by Fast R-CNN [21], Faster R-CNN [22], and mask R-CNN [23] are two-stage models. The two-stage target detection methods, as the name suggests, split the detection process into two stages. First, the region proposed network [24] is utilized to extract target information, and then, the detection layers anticipate target position and category information. The others are one-stage target detection algorithms, such as single-shot multibox detector (SSD) [25], YOLO [26], YOLOV2 [27], and YOLOV3 [28].

In 2015, YOLO [26] introduced an integrated detection scheme that combines candidate frame extraction, CNN learning features, and NMS optimization in order to simplify the network structure. The detection speed is nearly ten times faster than that of the R-CNN. This enables the deep learning target detection algorithm to meet the requirements of real-time detection tasks while utilizing the computing power available at the time; however, detection performance on small targets is poor. YOLOv2 [27] improves the accuracy of target regression and positioning by adding batch normalization, high-resolution classifier, dimension clusters, convolution with anchor boxes, and other optimization models to the network structure of YOLOv1. YOLOv3 [28] utilizes the residual network based on YOLOv2, and the binary cross loss function is used as the loss function to combine the feature pyramid network (FPN) structure. YOLOv4 [29] and YOLOv3 are very similar in structure, and both use the backbone feature extraction network, then construct a feature pyramid by the extracted image features, and finally output the results of three scales. The process of target classification and bounding box regression prediction is basically the same. Compared to YOLOv3, YOLOv4 significantly improved the detection accuracy while maintaining similar detection speed.

The fast advancement of computer technology allows us to use CNNs for various applications [30], [31], because they need a large amount of processing power. CNN-based target identification techniques outperform conventional target detection algorithms, such as HOG-cascade [14], DPM [18], and HOG-SVM [19], in many ways, including in terms of speed and accuracy. A CNN is a feedforward neural network that uses convolutional computation and the deep structure. It is one of the most crucial aspects of deep learning [32]–[34]. Deep learning

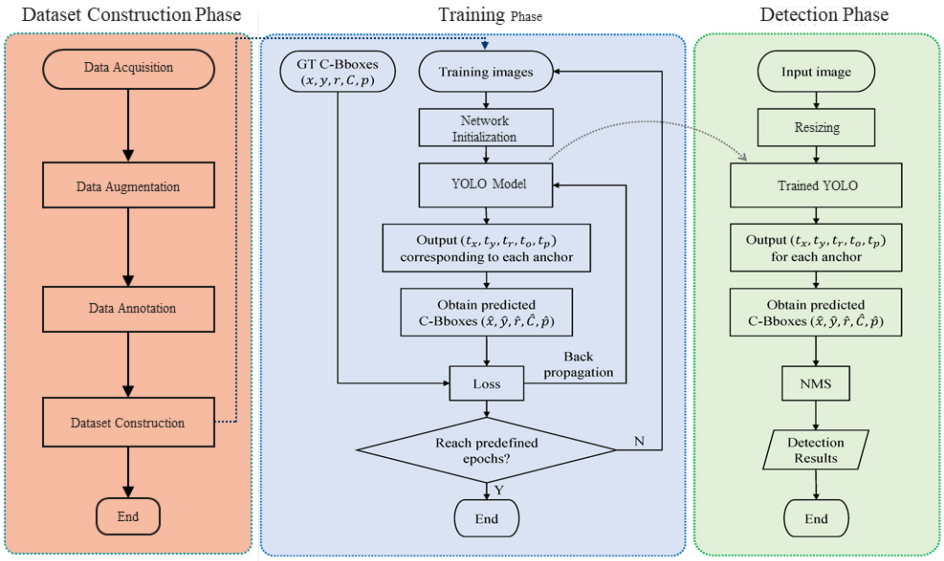

Fig. 1. Flowchart of the proposed methodology.

research in target identification has recently become a popular topic.

III. Overview of the Proposed Method

The overall flow of the proposed methodology for target detection is shown in Fig. 1. The target detection model was developed based on three main stages. First, we selected the remote sensing image dataset from various publicly available datasets and went through data augmentation to form a robust dataset. Then, the YOLO network was optimized and trained on the developed datasets for better feature reuse and representation. Furthermore, the NMS threshold and anchor allocation schemes were proposed for target detection in remote sensing images to make more precise localization and improve detection results. Evaluation metrics were computed to validate the detection performance. Finally, the best model was selected for target detection on remote sensing images.

A. Dataset Preparation

With the increasing demand for target detection in optical remote sensing images, many aerial remote sensing datasets have been developed in recent years. In this article, we use the DOTA dataset [35], which contains 2000 aerial images of the city and 15 categories of targets in total. There are more than 190 000 fully labeled targets, and each target is composed of the 8-D parameter $(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$. All the other datasets use diagonal coordinates with horizontal boxes $(x_{\min}, y_{\min}, x_{\max}, y_{\max})$. According to the statistics, most of the targets in datasets cannot be detected accurately in the rotation box. However, we perform a target detection task on this dataset by using a script, which is also an important reason for choosing the DOTA dataset. From this dataset, four types of target are selected: plane, ship, large vehicle, and small vehicle. The optical remote sensing images in the DOTA dataset are generally large, which are not

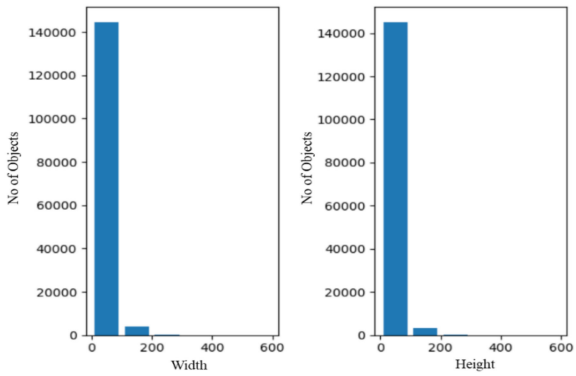

Fig. 2. Histogram of target's width and height distribution.

suitable for direct processing. Therefore, we crop images into 608×608 size. Furthermore, to ensure that the target will not be cropped, we calculate the target's width and height in the dataset, as illustrated in Fig. 2. According to statistics, 97.1% of the target's width is less than 100 pixels, and 97.4% of the target's height is less than 100 pixels. Therefore, setting the width and height overlap area to 100 pixels can ensure that most of the targets will not be cropped. The cropping process of a single image is shown in Fig. 3. The total number of images by cropping is 10 563; the images used for training verification are about 80% (8400), and the test images are 20% (2163).

B. YOLOv4 Algorithm

1) YOLOv4 Structure: Based on the original YOLO object detection architecture, YOLOv4 retains the head of YOLOv3 and uses a more powerful backbone network and CSPDarknet53. Additionally, it uses the idea of spatial pyramid pooling to expand the receptive field and chooses PANet as the neck part for feature fusion, as shown in Fig. 4. Meanwhile, it can be improved

Fig. 3. Horizontal frame target detection algorithm in remote sensing images.

and optimized due to the Mish activation function, Mosaic data enhancement, and DropBlock regularization. Integrating CSP on each large residual block (Resblock) of Darknet53 can enhance the learning ability of the CNN and maintain the accuracy when the weight, computing bottlenecks, and memory costs are reduced. Before each large Resblock, the input is divided into two parts: input to the stacked residual unit, and the other is directly convolved. Then, the results of the two parts are concatenated and finally output through convolution.

2) YOLOv4 Detection Process: YOLOv4 follows the same procedure as the original. It divides the image into several grids, each of which predicts whether or not the target exists. Test results in YOLOv4 contain $(4+1+n)$ $(x, y, w, h, conf, cls_1, cls_2, \ldots cls_n)$ parameters; (x, y, w, h) is the bounding box information, $(conf, cls_2,..., cls_n)$ is the score of each category and $3(4+1+n)$ is the dimension of a cell's output. The input image size is $608 \times 608 \times 3$, and there are n types of detection categories. The YOLOv4 detection network recognizes the image with three output scales. Large-scale targets are detected using the 19×19 grid, medium-scale targets are detected using the 38×38 grid, and small-scale targets are detected by the 76×76 grid. Expand the network output results and arrange them in a total of $3 \times (19 \times 19 + 38 \times 38 + 76 \times 76) = 22743$ prediction results, which are, then, decoded to produce the output.

3) YOLOv4 Bounding Box Regression: In the YOLOv4 algorithm, each cell has three anchor boxes. It chooses the target from the training set, and the largest anchor frame of the intersection over union (IoU) is responsible for detecting the target during training. The prediction method of the specific bounding box is shown in Fig. 5. The bounding box can be predicted as follows:

$$b_x = \sigma(t_x) + c_x \tag{1}$$

$$b_y = \sigma(t_y) + c_y \tag{2}$$

$$b_w = p_w e^{t_w} \tag{3}$$

$$b_h = p_h e^{t_h} \tag{4}$$

where c_x and c_y represent the offset of the grid relative to the upper left. P_w and P_h are the anchor width and height, respectively, b_x takes the center point of the bounding box of the predicted standard, b_w and b_h are the predicted frame width and height, respectively, and $\sigma(.)$ is a sigmoid function. t_x, t_y, t_w, and t_h represent the network's predicted value; by using these parameters, YOLOv4 completes the adjustment of the detection frame.

C. Improvements to the YOLOv4 Model

1) Setting NMS Thresholds by Category: This article initially found that the YOLOv4 model has a low detection accuracy for small targets (see Section IV). This is mainly due to the target's direction and dense arrangement in remote sensing images, as shown in Fig. 6. The YOLO series of algorithms originally detects horizontal frame targets, but the targets in optical remote sensing images are in arbitrary directions and densely arranged. The large vehicle uses a horizontal box as the bounding box; as a result, target overlap is very high. When we apply NMS, it artificially filters out many correctly detected targets. Furthermore, we analyze different categories in the dataset to understand the overlap of bounding boxes in horizontal box labeling, and the statistical results are shown in Table I. Fig. 7 shows the statistical histogram of the overlapped horizontal boxes of each category target.

Aiming to improve the detection accuracy of overlapping horizontal boxes, we propose an approach by setting the NMS threshold for each category. In natural scenes, most of the datasets are marked as horizontal boxes. The problem of overlapping horizontal frames is particularly prominent in optical remote sensing images, as shown in Fig. 6. The YOLOv4 algorithm adopts a unified NMS threshold for each category. The threshold is usually set to 0.3 (filtering process) for natural field target detection, which is low for the remote sensing target detection problem. According to the statistical results, the targets in optical remote sensing images are overlapped under the horizontal frame. At the same time, it can be seen that the overlap of each target is different. Thus, setting a uniform NMS threshold has drawbacks for processing of all categories. This article proposes different NMS threshold settings for different categories of objects. In order to better understand the impact of NMS threshold range on the detection accuracy of each category, we modify NMS of the model so that each category can be able to learn its own value. To find the optimal threshold of each category, we set the value with 0.05 difference. The step-by-step procedure of the NMS algorithm is given in Algorithm 1.

2) Improved Target Box Dimension Clustering: The YOLO series model introduced the design of the anchor frame after the YOLOv2 version. In the original algorithm, nine anchor boxes are generated by k-means and are equally distributed to three scales. Each scale corresponds to three anchor boxes. This absolute average may lead to unreasonable anchor frame allocation. For example, some anchor boxes can be processed in the middle scale, but it is divided into the small-scale processing because of the equal division. This article first applied the reclustering technique to extract the anchor box, but the newly generated anchor has poor results compared to the original one. Then, we assigned anchor frames according to scale and allocate the anchor frame by calculating the IoU between the anchor and each scale. The blue box in Fig. 8 is the anchor box, and the other three are positive. The anchor box and the IoU of the middle scale in

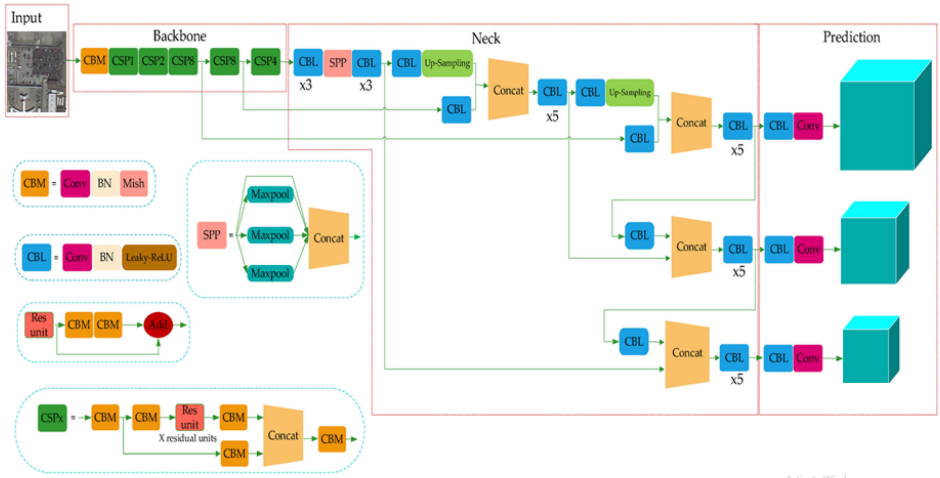

Fig. 4. YOLOv4 structure diagram.

TABLE I
HORIZONTAL FRAME OVERLAP STATISTICS

Category	Number of images	Overlap rate	Average overlap IoU	IoU>0.2	IoU>0.3	IoU> 0.4
Plane	5964	54.3	0.09	262 (4.4%)	23(0.4%)	3
Ship	39457	74.9	0.1437	8650 (21.9%)	3579 (9%)	872
Small vehicle	88271	71.9	0.127	18400 (20.8%)	4661 (5.3%)	143
Large vehicle	15006	75	0.17	4452 (29.7%)	2414 (16.1%)	817

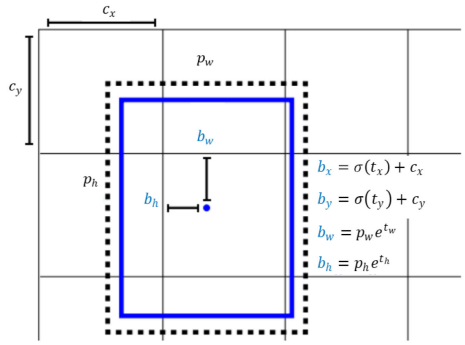

Fig. 5. Bounding boxes with dimension priors and location prediction.

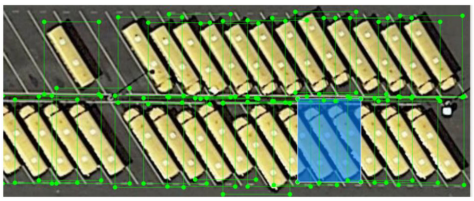

Fig. 6. Demonstrates the horizontal box overlap problem.

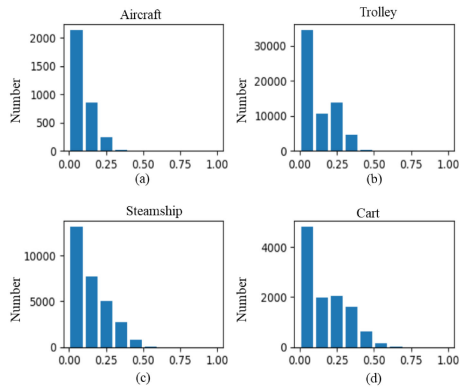

Fig. 7. (a)–(d) Overlapping IoU histograms of different categories of targets.

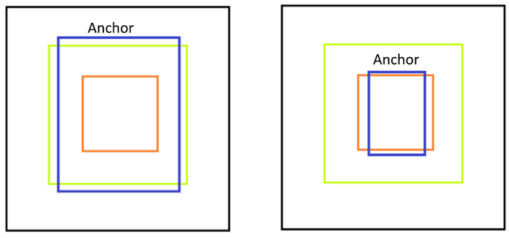

Fig. 8. Schematic diagram of anchor frame allocation.

the left in Fig. 8 are assigned to the middle scale with the largest IoU for processing. The anchor box and the small-scale IoU in the right are the largest and allocated to the small scale for processing. However, still, accuracy is not improved. Finally, to solve the problem in this article, we propose an approach that first divides all targets into three scale targets of the large, medium, and small according to the principle of anchor frame distribution, and then, the K-means algorithm is used to cluster

Algorithm 1: The Pseudocode of Non-maximum Suppression (NMS) for our Approach.

```
Original Bounding Boxes:
B = [b_1.....,b_M], S = [s_1.....,s_M], N_t
B refers to the set of original bounding boxes
S refers to the set of detection scores
N_t is the NMS threshold
Detection result:
D ← []
while B ≠ [] do
    m ←argmax(S) M←b_m
    D ←D ∪M
    B ← B − M
    for b_i ∈ B do
        if IoU(M, b_i) ≥ N_t then
            B ← B − b_i
            S ← s_i
        end
    end
    return D, S
end
```

the targets in the three medium and small scales to obtain three anchor boxes. To calculate the intersection ratio of all target boxes in the dataset with the large-, medium-, and small-scale boxes, the ratio divides the targets into three categories and, then, clusters on each category to obtain anchor boxes of their respective scales.

IV. Experiment and Analysis

A. Experimental Environment

The proposed method was executed on NVIDIA GeForce RTX 2080Ti, 64-GB memory, and Ubuntu 16.04 LTS operating system. We developed the whole framework in Python, which is widely used in deep learning, and mostly, the libraries used are scikit-learn, SciPy, matplotlib, and NumPy.

B. Implementation Details

We use the preprocessed dataset to analyze SSD, RetinaNet, YOLOv3, YOLOv4, and YOLOv4_tiny. With the data enhancement technique, all the network models use an Adam optimizer and other hyperparameters with their respective algorithms' default settings. For fair comparison between the results, all the experimental configurations and hyperparameters of the YOLO-based models were standardized. The initial learning rate for 50 epochs was a little bit higher to freeze the backbone feature extraction network (the backbone part does not perform gradient backpropagation), and for the last 50 epochs, we used lower learning rate and the backbone network was unfrozen. The weight parameters of the model were refined, and the models were trained up to 100 epochs.

C. Evaluation Measures

In order to evaluate the effect of the target detection approach, this article uses accuracy, recall, average precision (AP), mean average precision (mAP), and frames per second (FPS) to evaluate the algorithm.

The accuracy rate is the proportion of the correct samples in the total test, and it is calculated as

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

where true positive (TP) is the number of positive samples accurately predicted, and false positive (FP) is the number of negative samples that are incorrectly detected as positive samples.

The recall rate represents the proportion of positive samples that are accurately predicted, and it is defined as

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

where false negative (FN) is the number of positive samples that are predicted as negative samples. The AP is the area enclosed by recall rate of correctness curve and the x-axis. It is expressed as

$$\text{AP} = \int_0^1 P(y)dy \tag{7}$$

where y is the recall curve under different intersection ratio thresholds.

The mAP refers to the mean value of all categories of the AP and is calculated as

$$\text{mAP} = \frac{\sum_n^N \text{AP}_n}{N}. \tag{8}$$

D. Experimental Results

1) Setting NMS Thresholds by Category: It can be seen from Table I that the overlap rate of each category target is very high with the horizontal bounding box. The plane overlap rate and the average overlap IoU are the lowest. Therefore, it is least affected by the overlap problem with the AP of 87.27% on YOLOv4, as listed in Table II. The large vehicle has the highest overlap rate, and the average overlap IoU is also the highest, which results in the lowest detection accuracy. According to the statistical results of Table II, when the NMS threshold is set to at 0.3, about 16.1% of the correct detection targets of the large vehicles are artificially filtered out.

The experimental surface sets the threshold in the classification [0.5, 0.5, 0.5, 0.55]; the threshold setting of planes, ships, and small vehicles is 0.5; however, for large vehicle, it is set to 0.55; the model performance is good with 77.68% mAP, which is an increase of 2.53% compared to the original model setting. At the same time, the detection frame rate of the model is about 5.5 FPS, which is the same as in the original YOLOv4.

2) Reclustering to Get the Anchor Box: Here, the K-means algorithm is used to cluster the width and height of all the targets in the dataset to obtain a new anchor box, i.e., (8,8), (12,21), (20,22), (21,12), (25,38), (37,17), (43,47), (44,29), and (89,82). With a newly generated anchor box during training and testing, the hyperparameter settings of the environment are consistent

TABLE II
PERFORMANCE OF YOLOV4 AGAINST DIFFERENT SETTINGS OF THE NMS THRESHOLD

Target category/NMS Threshold	0.3	0.4	0.45	0.5	0.55	0.6
Plane	87.27%	87.67%	87.83%	87.88%	87.71%	87.61%
Ship	82.67%	84.52%	84.93%	85.09%	85.00%	84.52%
Small vehicle	72.65%	73.56%	73.81%	73.84%	73.55%	72.89%
Large vehicle	58.02%	62.89%	63.59%	63.90%	63.91%	63.27%

TABLE III
PERFORMANCE OF YOLOV4 WITH THE RECLUSTERING APPROACH TO GET THE ANCHOR BOX

Model	plane	ship	large vehicle	small vehicle	mAP
YOLOv4	87.27%	82.67%	58.02%	72.65%	75.15%
YOLOv4_C1	86.21%	71.37%	53.46%	68.30%	69.83%

TABLE IV
PERFORMANCE OF YOLOV4 WITH ANCHOR FRAME ALLOCATION ACCORDING TO SCALE

Model	plane	ship	large vehicle	small vehicle	mAP	FPS
YOLOv4	87.27%	82.67%	58.02%	72.65%	75.15%	5.3
YOLOv4_C1	86.21%	71.37%	53.46%	68.30%	69.83%	5.3
YOLOv4_C2	85.03%	70.39%	56.47%	63.19%	68.77%	5.9

TABLE V
PERFORMANCE OF YOLOV4 WITH ANCHOR FRAME ALLOCATION ACCORDING TO SUBSCALE CLUSTERING

Model	plane	ship	large vehicle	small vehicle	mAP	FPS
YOLOv4	87.27%	82.67%	58.02%	72.65%	75.15%	5.3
YOLOv4_C1	86.21%	71.37%	53.46%	68.30%	69.83%	5.3
YOLOv4_C2	85.03%	70.39%	56.47%	63.19%	68.77%	5.9
YOLOv4_C3	88.57%	76.60%	60.76%	70.97%	74.22%	5.3

TABLE VI
PERFORMANCE (%) COMPARISON OF OUR APPROACH WITH OTHER APPROACHES

Model	plane	ship	small vehicle	large vehicle	mAP	FPS	Parameter amount (M)
SSD [25]	70.97%	65.36%	23.58%	40.41%	50.08%	7.5	24.15
Mask R-CNN [38]	88.41%	83.68%	73.64%	53.7%	70.70%	907	24.15
Cascade Mask R-CNN [39]	88.17%	70.36%	73.64%	60.41%	70.71%	7.2	24.15
Faster R-CNN [40]	88.82%	78.01%	72.02%	60.56%	70.76%	14.3	24.15
RetinaNet [41]	29.86%	11.55%	2.16%	1.07%	11.16%	6.8	36.4
YOLOv3 [28]	82.72%	79.00%	71.35%	53.82%	71.73%	5.5	61.54
YOLOv4_tiny [42]	79.16%	65.50%	41.89%	46.89%	58.36%	19.5	5.89
YOLOv4 [29]	87.27%	82.67%	72.65%	58.02%	75.15%	5.3	63.95
YOLOv4_threshold =0.5	87.88%	85.09%	73.84 %	63.90%	_	_	_
YOLOv4_C1	86.21%	71.37%	68.30 %	53.46%	69.83%	5.3	_
YOLOv4_C2	85.03%	70.39%	63.19 %	56.47%	68.77%	5.9	_
YOLOv4_C3	88.57%	76.60%	70.97 %	60.76 %	74.22%	5.3	_

with those in the original model. The performance comparison of YOLOv4 with the newly generated anchor box and the original anchor box is shown in Table III. It is seen from Table IV that the performance of the model is poor with the newly generated anchor frame. It is due to the three scales in the model, and the degree does not match the assignment of the anchor frame.

3) Assign Anchor Frame According to Scale: In order to allocate anchor frames reasonably, we calculate the IoU between the anchor frame and each scale and, then, assign the anchor frame to the largest scale of the IoU for processing. After the anchor frame and the IoU calculation of each scale, one anchor box (8,8) is allocated to the small-scale processing, three anchor boxes (12,21), (20,22), and (21,12) are allocated to medium-scale processing, and five anchor boxes (25,38), (37,17), (43,47), (44,29), and (89,82) are allocated to large-scale processing. Modifying the model based on the allocation result, against 608×608 image size, the *a priori* boxes are reduced from 22 743 to 11 913, which is 52.4% of the original. Moreover, model results are illustrated in Table V.

4) Assign Anchor Frame According to Subscale Clustering: In order to solve the problem of the reduction of scale anchor frames, it also takes into account the problem of reasonable allocation of anchor frames. We propose an approach that first divides all targets into three scale targets of large, medium, and small according to the principle of anchor frame distribution, and then, the K-means algorithm is used to cluster the targets in

TABLE VII
SPEED COMPARISON OF OUR APPROACH WITH OTHER APPROACHES

Model	Speed (fps)
SSD [25]	7.5
Mask R-CNN [38]	907
Cascade Mask R-CNN [39]	7.2
Faster R-CNN [40]	14.3
RetinaNet [41]	6.8
YOLOv3 [28]	5.5
YOLOv4_tiny [42]	19.5
YOLOv4 [29]	5.3
YOLOv4_threshold =0.5	-
YOLOv4_C1	5.3
YOLOv4_C2	5.9
YOLOv4_C3	5.3

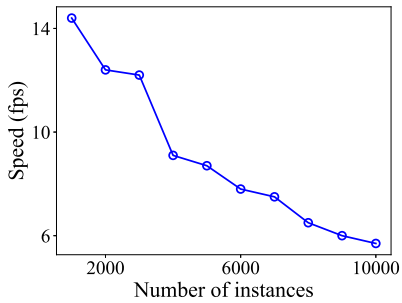

Fig. 9. Different number of instance frames per seconds.

TABLE VIII
ABLATION STUDY OF DATA AUGMENTATION

Model	Base.	Data augmentation				
High overlap		✓	✓	✓	✓	✓
Multi-scale Train			✓	✓	✓	✓
Multi-scale Test				✓	✓	✓
Rotation Train					✓	✓
Rotation Test						✓
mAP	62.05	63.28	65.97	69.97	72.62	74.22

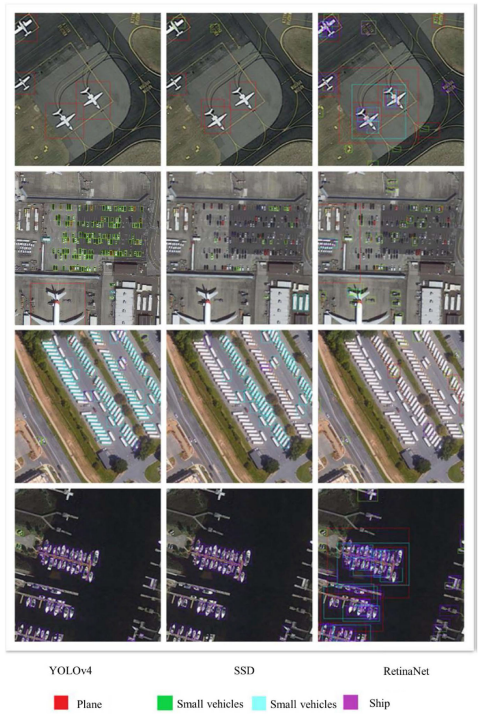

Fig. 10. Visual results of different models.

the three large, medium and small scales to obtain three anchor boxes. This operation calculates the intersection ratio of all target boxes in the dataset with the large, medium, and small scale boxes. The ratio divides the targets into three categories and, then, clusters on each category to obtain anchor boxes of their respective scales. After calculation, the small-scale anchor frames are (6,10), (9,7), and (13,7); medium-scale anchor frames are (12,22), (20,20), and (22,12); and large-scale anchor frames are (30,25), (39,42), and (86,77). The test results are shown in Table VI, and the speed comparison is shown in Table VII. Fig. 9 depicts the processing speed performance over the number of instances. As shown in Fig. 9, the number of instances with the highest performance for YOLOv4 is 8,000. Additionally, from 2000 to 8000 instances, the improvement in YOLOv4 is 1.40 points in mAP. Therefore, the increased number of instances required more compositional.

It can be seen from Table V that the new method has improved the recognition accuracy of plane and large vehicles. Still, compared with the original model, the recognition accuracy of ships and small vehicles has dropped a lot, especially the detection accuracy of ships has dropped by more than 6%. The AP drops by nearly 1%.

From Table VIII, it can be seen that the NMS procedure is used to integrate the results of images taken at diverse angles and scales into a single image. Table VIII shows that both scale and rotation data augmentations improve item detection performance by a substantial margin, which is consistent with the large-scale and orientation variability in DOTA. The region of interest (RoI) transformer and an FPN were previously included in this baseline (Basc) model. This shows that the FPN and the RoI transformer do not completely solve the issue of scale and rotation fluctuations, and geometric modeling using CNNs remains an open challenge.

E. Performance Comparison With Other Approaches

In a comprehensive comparison of various models, YOLOv4 has the best target detection performance for optical remote sensing images, the mAP reached 75.15%, and the AP value of each category is the highest. YOLOv4 and YOLOv3 have almost the same detecting speed, but the accuracy of the YOLOv4 model is higher. The YOLOv4 tiny model has the fewest parameters; the detection speed is faster when the model is smaller, but the accuracy is lower. In comparison to the RetinaNet and SSD algorithms, the YOLOv4 algorithm has high detection accuracy. RetinaNet inspection results are a little bit unexpected, and the reason may be that the algorithm itself is effective in target detection of optical remote sensing images. If the results are not good, it is also possible that the network is not well trained. Visual detection results for YOLOv4 can be seen in Fig. 10. Based on the above experimental results, we choose the YOLOv4 algorithm to detect targets in optical remote sensing images.

We also compare our work with some recent techniques such as Fourier-based rotation-invariant feature boosting [36] and spatial frequency [37]. Still, these techniques use frequency and rotation to extract the high-level features. Compared to extracting features in rotation coordinates, this article uses the YOLOv4 deep learning models to improve the frame detection performance and detect the object when targets have similar shape.

V. Conclusion

This article briefly introduced traditional single-stage target detection algorithms, such as SSD, RetinaNet, and YOLO series, and used the optical remote sensing target detection dataset to train and test these models. Comparative results showed that YOLOv4 has the best detection performance for optical remote sensing targets. Furthermore, we explained the structure of the YOLOv4 algorithm, the detection process, the frame regression method and the improvement relative to YOLOv3, and the analysis of detection results of the YOLOv4 algorithm. Aiming at the serious overlap of horizontal frames in optical remote sensing images, this article proposed that a classification setting of the NMS threshold method increases, which increases the mAP by 2.53% without affecting the speed of the model. In addition, we also studied the anchor frame allocation problem in YOLOv4 and proposed two allocation schemes. The class anchor frame scheme further improved the detection performance of some categories.

References

[1] J. Gao and Y. Liu, "Applications of remote sensing, GIS and GPS in glaciology: A review," *Prog. Phys. Geography*, vol. 25, no. 4, pp. 520–540, 2001.

[2] K. Kaku, "Satellite remote sensing for disaster management support: A holistic and staged approach based on case studies in Sentinel Asia," *Int. J. Disaster Risk Reduction*, vol. 33, pp. 417–432, 2019.

[3] W. Wang, N. Yang, Y. Zhang, F. Wang, T. Cao, and P. Eklund, "A review of road extraction from remote sensing images," *J. Traffic Transp. Eng.*, vol. 3, no. 3, pp. 271–282, 2016.

[4] P.-C. Chen, Y.-C. Chiang, and P.-Y. Weng, "Imaging using unmanned aerial vehicles for agriculture land use classification," *Agriculture*, vol. 10, no. 9, 2020, Art. no. 416.

[5] T. Wellmann *et al.*, "Remote sensing in urban planning: Contributions towards ecologically sound policies," *Landscape Urban Planning*, vol. 204, 2020, Art. no. 103921.

[6] A. M. F. Al-Quraishi and A. M. Negm, "Updates, conclusions, and recommendations for environmental remote sensing and GIS in Iraq," in *Environmental Remote Sensing and GIS in Iraq*. New York, NY, USA: Springer, 2020, pp. 517–529.

[7] E. K. Wang, F. Wang, S. Kumari, J.-H. Yeh, and C.-M. Chen, "Intelligent monitor for typhoon in IOT system of smart city," *J. Supercomput.*, vol. 77, no. 3, pp. 3024–3043, 2021.

[8] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.

[9] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[10] D. Liu, Y. Wu, N. Luo, and B. Zheng, "Gaussian-YOLOv3 target detection with embedded attention and feature interleaving module," *J. Comput. Appl.*, vol. 40, no. 8, pp. 2225–2230, 2020.

[11] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and $<$ 0.5MB model size," 2016, *arXiv:1602.07360*.

[12] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[13] W. Fang, L. Wang, and P. Ren, "Tinier-YOLO: A real-time object detection method for constrained environments," *IEEE Access*, vol. 8, pp. 1935–1944, 2019.

[14] J. Chen, T. Takiguchi, and Y. Ariki, "Rotation-reversal invariant HOG cascade for facial expression recognition," *Signal Image Video Process.*, vol. 11, no. 8, pp. 1485–1492, 2017.

[15] X. Cao, X. Fu, C. Xu, and D. Meng, "Deep spatial-spectral global reasoning network for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5504714.

[16] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2020.

[17] D. Hong *et al.*, "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, May 2020.

[18] D. Zhang, "Vehicle target detection methods based on color fusion deformable part model," *EURASIP J. Wireless Commun. Netw.*, vol. 2018, 2018, Art. no. 94.

[19] M. Bilal and S. M. Hanif, "Benchmark revision for HOG-SVM pedestrian detector through reinvigorated training and evaluation methodologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1277–1287, Mar. 2020.

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

[21] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[22] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[23] Z. Huang, Z. Zhong, L. Sun, and Q. Huo, "Mask R-CNN with pyramid attention network for scene text detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Waikoloa Village, HI, USA, 2019, pp. 764–772.

[24] K. Shih, C. Chiu, and Y. Pu, "Real-time object detection via pruning and a concatenated multi-feature assisted region proposal network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Brighton, U.K., 2019, pp. 1398–1402.

[25] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[26] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.

[27] X. Zhang, Z. Qiu, P. Huang, J. Hu, and J. Luo, "Application research of YOLO v2 combined with color identification," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discov.*, Zhengzhou, China, 2018, pp. 138–1383.

[28] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018, *arXiv:1804.02767*.

[29] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[30] S. Kujawa, J. Mazurkiewicz, and W. Czekala, "Using convolutional neural networks to classify the maturity of compost based on sewage sludge and rapeseed straw," *J. Cleaner Prod.*, vol. 258, 2020, Art. no. 120814.

[31] R. Hashimoto *et al.*, "Artificial intelligence using convolutional neural networks for real-time detection of early esophageal neoplasia in Barrett's esophagus (with video)," *Gastrointestinal Endoscopy*, vol. 91, no. 6, pp. 1264–1271.e1, 2020.

[32] J. Zakria, J. Cai, M. U. Deng, M. S. A. Khokhar, and R. Kumar, "Efficient and deep vehicle re-identification using multi-level feature extraction," *Appl. Sci.*, vol. 9, no. 7, 2019, Art. no. 1291.

[33] J. Zakria, J. Cai, J. Deng, M. S. Khokhar, and M. U. Aftab, "Vehicle classification based on deep convolutional neural networks model for traffic surveillance systems," in *Proc. IEEE 15th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process.*, 2018, pp. 224–227.

[34] Zakria *et al.*, "Trends in vehicle re-identification past, present, and future: A comprehensive review," *Mathematics*, vol. 9, 2021, Art. no. 3162.

[35] G. Xia *et al.*, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3974–3983.

[36] X. Wu, D. Hong, J. Tian, J. Chanussot, W. Li, and R. Tao, "ORSIm detector: A novel object detection framework in optical remote sensing imagery using spatial-frequency channel features," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5146–5158, Jul. 2019.

[37] X. Wu, D. Hong, J. Chanussot, Y. Xu, R. Tao, and Y. Wang, "Fourier-based rotation-invariant feature boosting: An efficient framework for geospatial object detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 302–306, Feb. 2020.

[38] M. Buric, M. Pobar, and M. Ivasic-Kos, "Ball detection using YOLO and Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Sci. Comput. Intell.*, 2018, pp. 319–323.
[39] K. Chen *et al.*, "Hybrid task cascade for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4974–4983.
[40] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.
[41] D. Zhu, G. Xu, J. Zhou, E. Di, and M. Li, "Object detection in complex road scenarios: Improved YOLOV4-tiny algorithm," in *Proc. IEEE 2nd Inf. Commun. Technol. Conf.*, 2021, pp. 75–80.

Zakria received the M.S. degree in computer science and information technology from the NED University of Engineering and Technology, Karachi, Pakistan, in 2017, and the Ph.D. degree in computer vision from the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China, in 2020.

He is currently a Postdoctoral Researcher with the School of Information and Software Engineering, University of Electronic Science and Technology of China. He has a vast academic, technical, and professional experience in Pakistan. His research interests include artificial intelligence and computer vision, particularly vehicle reidentification.

Jianhua Deng received the graduate degree in information security from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2006, and the Ph.D. degree in electrical engineering from the Dublin Institute of Technology, Dublin, Ireland, in 2014.

After graduation, he joined the School of Computer Science and Engineering, UESTC, as a Staff, where he is currently a Vice-Professor. He is the reviewer of some SCI journals (e.g., *Wireless Personal Communications*). His research interests include wireless communication, statistical machine learning, artificial intelligence, and deep learning.

Rajesh Kumar received the Ph.D. degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2019.

He is currently an Associate Professor with the Yangtze Delta River Institute, University of Electronic Science and Technology of China, Huzhou, China. He has authored or coauthored more than 35 articles in various International journals and conferences. His research interests include machine learning, deep leaning, malware detection, Internet of Things, and blockchain technology.

Muhammad Saddam Khokhar received the M.S. degree in computer science and information technology from the NED University of Engineering and Technology, Karachi, Pakistan, in 2016, and the Ph.D. degree in computer vision from the School of Computer Science and Telecommunications Engineering, Jiangsu University, Zhenjiang, China, in 2020.

He is currently with the School of Computer Science and Telecommunications Engineering, Jiangsu University. He has a vast academic, technical, and professional experience. He supervised more than 30 academic final-year research projects in Pakistan. His current research interests include computational intelligence, pattern recognition, and computer vision.

Jingye Cai received the B.S. degree in radio electronics from Sichuan University, Chengdu, China, in 1983, and the M.S. degree in signal and information processing from the University of Electronic Science and Technology of China (UESTC), Chengdu, in 1990.

He is currently a Professor with the School of Information and Software Engineering, UESTC. He has presided over or participated in more than 30 scientific research projects, and 11 of them have passed the technical appraisal of the ministry. His research interests include communication and radar signal processing, intelligent computing, digital information processing, and spectra estimation.

Jay Kumar received the master's degree in computer science from Quaid-i-Azam University, Islamabad, Pakistan, in 2018 and the Ph.D. degree in computer science and technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2021.

He is currently working as Postdoctoral Fellow in Dalhousie University, halifax, Canada. His research has currently published ten SCI indexed high class journal papers including ACL, IEEE TRANSACTIONS ON CYBERNETICS, IEEE Sensors Journal and Information Sciences. His main research interests include data stream mining and natural language processing.