POD-YOLO Object Detection Model Based on Bi-directional Dynamic Cross-level Pyramid Network

Yu Zhang, Ming Ma, Zhongxiang Wang, Jing Li, Yan Sun

Abstract—**The existing heavy-backbone object detection models overlook the crucial role of cross-level interactive fusion of feature information in pyramid networks, resulting in the inability to detect occluded objects or small objects in complex scenes. In this thesis, we present a new heavy-neck object detection model called POD-YOLO based on YOLOv5s. Firstly, we propose the POD-RepC3 module to increase the model's capability to obtain the multi-layer feature. Additionally, addressing the issue of large object size span, we propose a bidirectional partial dynamic fusion module (Bi-PDC) as the detection neck of the pyramid network. This module preserves the accurate positioning signals and facilitates cross-level interactive fusion of feature information. Finally, we design Reparameterized Bi-directional Dynamic Feature Pyramid Network (RepBi-DFPN), a deep feature fusion network that integrates contextual information and enhances both feature expression and fusion capabilities of our model. The experiment results suggest that the suggested method is positive on the PASCAL VOC dataset. The mAP@0.5 and mAP@0.5：0.95 performance reached 81.3% and 58.2%, respectively, which increased by 2.4% and 4.1% compared to original algorithm YOLOv5s. Furthermore, experiment results also demonstrate that model's performance can compete with SOTA object detection models. In this paper, the algorithm optimizes the feature fusion capability of the pyramid network to effectively decrease the false detection and missing detection of the model. The model's ability to accurately detect multi-scale targets is significantly improved.**

Index Terms—**Image processing, Object detection, Feature pyramid, Multi-scale，Feature fusion**

Manuscript received November 19, 2023; revised March 19, 2024. This work was supported by Joint Fund Project of the Liaoning Province Nature Fund Project (No.2022-MS-291), and the Scientific research project of Liaoning Province Education Department （ LJKMZ20220781, LJKMZ20220783，LJKQZ20222457）.

Yu Zhang is a Lecturer of School of Computer Science and Technology, Shenyang University of Chemical Technology, Shenyang, China. (e-mail: zhangy@syuct.edu.cn).

Ming Ma is a Postgraduate of School of Computer Science and Technology, Shenyang University of Chemical Technology, Shenyang, China. (Corresponding author to provide phone: +86-155-8499-6930; e-mail: maming_69@163.com).

Zhongxiang Wang is a Postgraduate of School of Computer Science and Technology, Shenyang University of Chemical Technology, Shenyang, China. (e-mail: puo918@163.com).

Jing Li is a Postgraduate of School of Computer Science and Technology, Shenyang University of Chemical Technology, Shenyang, China. (e-mail: 2396796330@qq.com).

Yan Sun is a Postgraduate of School of Computer Science and Technology, Shenyang University of Chemical Technology, Shenyang, China. (e-mail: 1762203252@qq.com).

I. Introduction

COMPUTER vision technology has developed rapidly, and object detection has become one of the hot points in the areas of image processing. It has demonstrated the widespread potential applications in areas such as human-computer interaction, image classification, automatic driving[1], and medical diagnosis. With advancements in computer performance and rapid progress of deep learning and graphics processing technology, the effectiveness of algorithms for object detection has achieved unprecedented levels. In recent years, object detection methods based on deep learning have made remarkable progress. The object detection algorithms that follow a two-stage approach, such as R-CNN [2], Fast R-CNN [3], and Faster R-CNN [4], achieve slightly higher detection accuracy by using RPN networks to extract candidate boxes. Consequently, by continuously optimizing and improving the feature extraction network and anchor box regression work, single-stage detection algorithms like the YOLO [5], SSD [6], YOLO9000[7],YOLOv3 [8], YOLOv4 [9], and YOLOv5 [10] have been developed. These algorithms directly classify and predict objects on the feature map, eliminating the need for additional region classification steps, thereby enhancing detection speed, and the YOLO-series models have gained significant popularity.

Although object detection networks have made significant advancements in architecture design and training strategies, detecting multi-scale objects remains challenging. In a typical object detection framework, the backbone network obtains deep-level latent features, whereas the neck module integrates these features to capture multi-scale information. However, compared to image recognition, object detection requires the higher image resolution, resulting in the computational cost of backbone networks accounting for most of the reasoning cost. This backbone design paradigm is a historical problem left over from the development of image recognition to object detection, instead of a design that is optimized end-to-end for object detection. Consequently, the paradigm has led to the suboptimal performance in object detection models [11]. Therefore, the neck module's feature fusion is crucial for multi-scale object detection. To mitigate the problems associated with the wide range of target size spans, most feature pyramid networks rely on multi-scale features from conventional CNN backbones for fusion. However, as CNNs advance, backbone networks become larger and more computationally demanding. In contrast to conventional backbones, the main emphasis of FPN [12] lies

in the integration of high-level semantic information and low-level space information. Moreover, current research on feature fusion networks heavily depends on optimal backbone design, and the communication of information between different levels of characteristics cannot be fully facilitated. YOLOv3 combines three different scale feature maps using up-sampling and fusion methods, followed by independent detection of these multi-scale feature maps. YOLOv3 combines three different scale feature maps using up-sampling and fusion methods, followed by independent detection of these multi-scale feature maps. [13] uses depthwise separable convolution[14] and SPP[15] in the feature pyramid to decrease the count of parameters and enhance the feature representation; however, this approach increases redundancy and memory usage. YOLOv5 extracts object features using deep residual network with multi-scale prediction accomplished through the FPN-PAN[16] structure. However, current network structure neglects high-frequency information loss during cross-layer feature fusion and does not adequately address large object size spans. Moreover, limited pixel availability in small objects hinders effective feature extraction, making it susceptible to background noise interference and the potential loss of information during convolutional neural network's forward propagation.

We propose POD-YOLO, A novel object detection model utilizing YOLOv5s as its foundation, to address above problems. Our approach includes the POD-RepC3 module for capturing multi-layer information and achieving a balance between local and global contexts, the bi-directional partial dynamic fusion module (Bi-PDC) for promoting the cross-level interaction of feature information, the Omni-Dimensional Dynamic Convolution (ODConv) for acquiring the ample semantic information flexibly, meanwhile, Reparameterized Bi-directional Dynamic Feature Pyramid Network (RepBi-DFPN) as a foundation for constructing POD-YOLO by using the YOLOv5s backbone network.

In summary, the key contributions are three-fold:

1) Design a POD-RepC3 module that balances local and global information, enhancing the model's capacity to extract multi-scale feature.
2) Propose the bi-directional partial dynamic fusion module Bi-PDC, which can effectively solve the problem of large object size spanning range and enhance cross-level interactive fusion of features.
3) A flexible, deep and long neck module RepBi-DFPN is proposed for feature fusion, and based on this, we propose a new detector called POD-YOLO, which has better overall feature information fusion capabilities.

II. RELATED WORK

The YOLOv5 architecture is composed of four components: Input, Backbone, Neck, and Head. The input end of model utilizes Mosaic data augmentation to randomly arrange, crop, and adjust the color of the input image.

A. Backbone

Backbone uses the CSPDarknet53 framework[17], which consists of three parts: Conv, C3, and SPPF. In the backbone network, the Conv and C3 layers are connected to strengthen feature extraction, and the C3 module effectively avoids the gradient disappearance. The SPPF structure uses three maximum feature pooling layers to strengthen the network's ability to perceive images and resolve feature information.

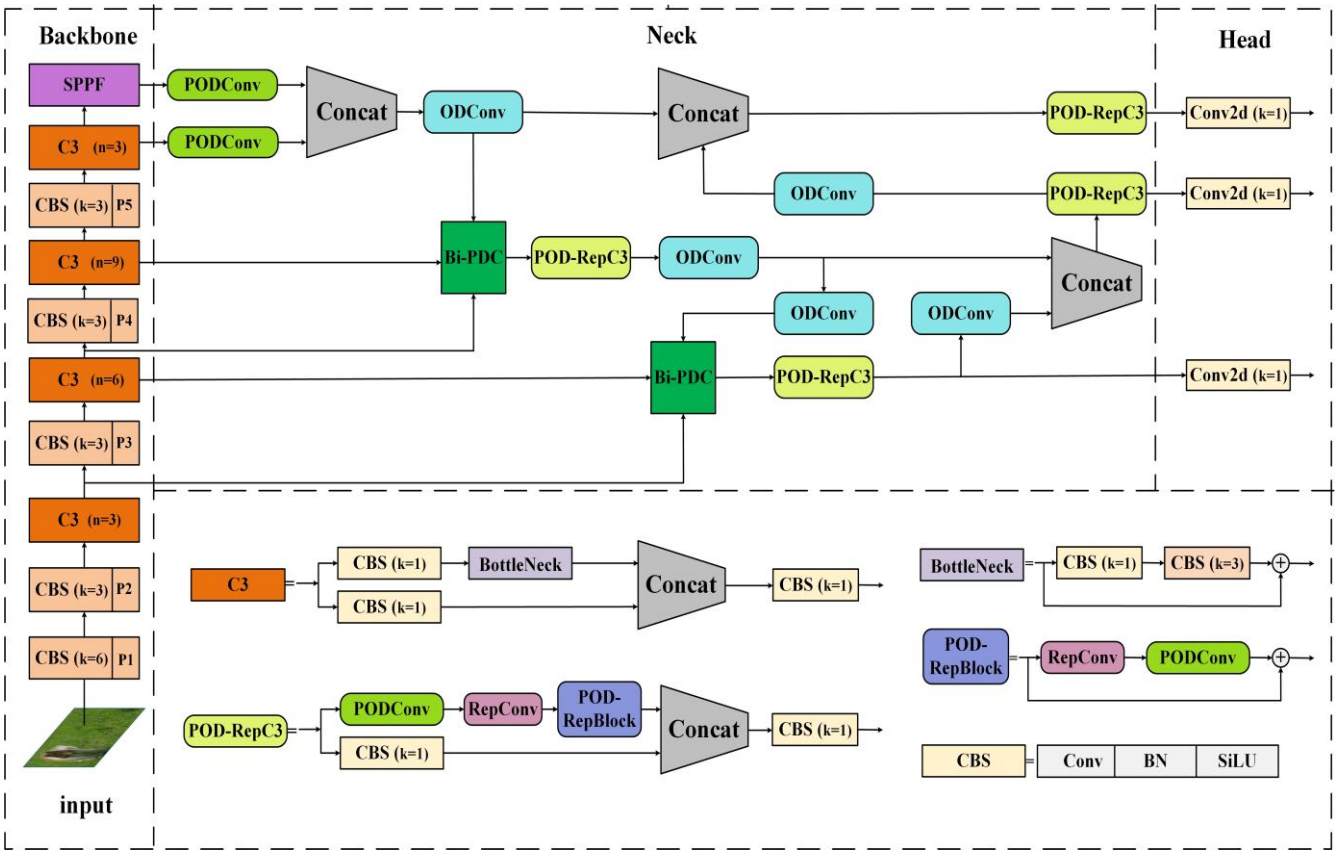


Fig. 1. POD-YOLO Network Model Structure

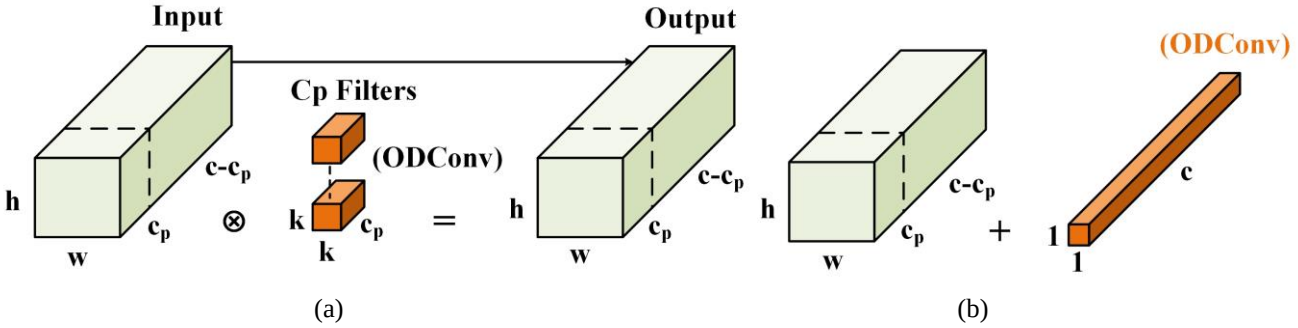


Fig. 2. Structure of the PODConv

B. Neck

The Neck module adopts the FPN and PANet feature pyramid network structures. Specifically, the FPN structure integrates semantic feature information through a top-down pathway, and the PANet structure migrates the strong positioning features from the lower network layer to the higher network layer.

C. Head

The prediction component in YOLOv5s comprises three prediction layers with varying scales. Among them, the small-scale detection head is used to detect large targets, and the large-scale detection head is used to detect small targets.

III. ALGORITHM DESIGN

A. Overall Architecture of POD-YOLO Model

Although extensive research on effective object detection models, dealing with the challenge of large object size span remains difficult. In order to achieve efficient multi-scale information fusion, we propose the POD-YOLO with a heavy-neck design. This model replaces the conventional convolution modules with the Omni Dimensional Dynamic Convolution[18] to tackle increased parameter redundancy. We introduce a new POD-RepC3 module with efficient fusion characteristics to advance the model's capacity to extract multi-layer feature. Additionally, we propose Bi-PDC module to improve localization accuracy and enable flexible and efficient integration of feature information at various scales in object detection model.

The general structure can be observed in Fig. 1. Firstly, the Bi-PDC connects the backbone network and feature fusion network, preserving object position information in adjacent three-layer feature maps whereas fusing features at different scales. After extracting the deep and complex feature information using the POD-RepC3 module, the ODConv dynamically adjusts convolution kernel size to incorporate contextual information. Finally, RepBi-DFPN maximizes the semantic details contained in output feature map through cross-level connections.

B. Partial Dynamic Convolution

In order to improve neural network efficiency, currently, many research efforts are focused on reducing floating-point operations, such as popular depthwise separable convolution. Although DWConv is effective in reducing the FLOPs, it is usually followed by pointwise convolution, so it is not a straightforward replacement for traditional convolution. Otherwise, it will lead to serious accuracy devastation. Even in practice, it is possible to increase the width of network to offset the decrease in precision, however, this will account for higher memory space. In contrast, FasterNet [19] considers redundancy in feature mapping and proposes PConv to decrease FLOPs and memory access, and improve the effectiveness of spatial feature extraction. Inspired by PConv, we propose a partial dynamic convolution(PODConv) that can strike an improved equilibrium between the precision of the model and its computational effectiveness.

PODConv extracts the spatial features of some input channels and leaves the remaining channels unaltered. In Fig.2(a), we can see for memory access that is consistent or regular, first or final continuous cp channels is used as a representative for feature extraction using ODConv. The remaining (c-cp) channels are unchanged and concatenated with the feature map of cp output channels after feature extraction, preserving feature details of the unoccluded area in object detection and providing more comprehensive context information. To utilize all channel information, ODConv is added after concatenating the cp channel with the (c-cp) channel, as shown in Fig.2(b). ODConv adjusts the convolutional kernel size dynamically to extract features at various scales. It first compresses the input features and then maps them to a lower-dimensional space. Attention weights are calculated for each dimension of kernel space, which are then multiplied with convolutional kernel. Finally, the attention-weighted convolutional kernel is convolved with the input features to fuse information from different scales.

The PODConv feature extraction process adjusts the convolutional kernel size based on object size and scale, obtaining different receptive fields across various feature layers. This effectively improves the expression of edge features, addressing occlusion and scale change problems in object detection. Compared to the traditional convolution, PODConv offers greater flexibility in feature extraction and reduces parameter redundancy.

C. Bi-directional Partial Dynamic Concatenation Module

The previous studies have only fused features from the same or previous layer to aid in the model's detection, as it connects features between adjacent layers. Therefore, we

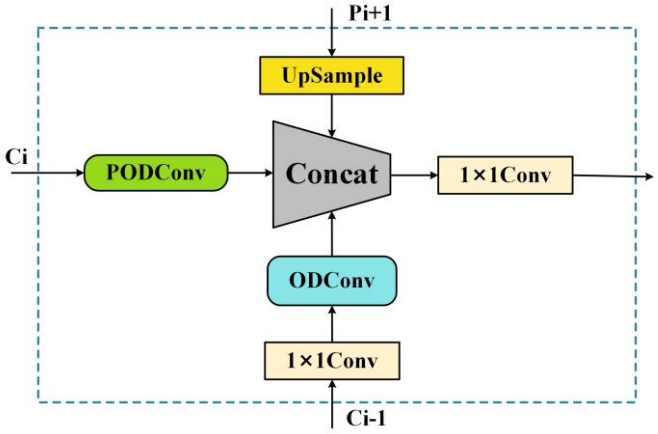


Fig. 3. Structure of the Bi-PDC

propose the bi-directional partial dynamic concatenation module (Bi-PDC) as the detection neck of the feature fusion network to accurately locate objects.

As shown in Fig.3, in the Bi-PDC module, a low-level feature map from backbone Ci-1 is adjusted for the quantity of channels by convolution operations with a convolution kernel size of 1×1. The ODConv dynamically modifies convolution kernel's parameters, weightings the attention of the four dimensions of kernel space, and utilizes feature information in each dimension to achieve convolution operations in different dimensions. Thus, it can capture multi-dimensional features so as to obtain sufficient contextual semantic information. PODConv extracts deep features from the backbone Ci feature map and preserves clear edge information for the accurate localization signals. Finally, the feature information from backbone Ci-1, Ci, and Pi+1 is fused into Pi to fully aggregate adjacent layers' feature maps, combining high-level semantic and low-level spatial information to densify the network structure. As cross-scale feature interaction module, Bi-PDC enhances the expression ability of multi-scale features by improving cross-level interaction and information transmission, ensuring flexible and efficient multi-scale feature fusion in RepBi-DFPN.

D.Efficient POD-RepC3 Module

ResNet[20] and DenseNet[21] are widely used in modern CNNs due to their effective solutions for gradient vanishing and the intermediate feature aggregation. CSPNet utilizes cross-stage dense connection to reduce computational burden without losing accuracy. As an important part of YOLOv5, the C3 module is composed of multiple stacks of residual modules, which enhance the model's ability to obtain features by deepening the network and expanding its receptive field. However, the network's multi-branching architecture leads to increase memory consumption during feature fusion, resulting in a noticeable decrease in inference speed. Therefore, this paper introduces POD-RepBlock and POD-RepC3 (Fig.4), which leverage reparameterization [22] to enhance both accuracy and efficiency in object detection.

Specifically, POD-RepBlock comprises of PODConv and RepConv_3×3, reducing the computational redundancy and providing the comprehensive contextual information. The feature map obtained by Bi-PDC module is then input into POD-RepC3, as shown in Fig 4(c). First of all, PODConv

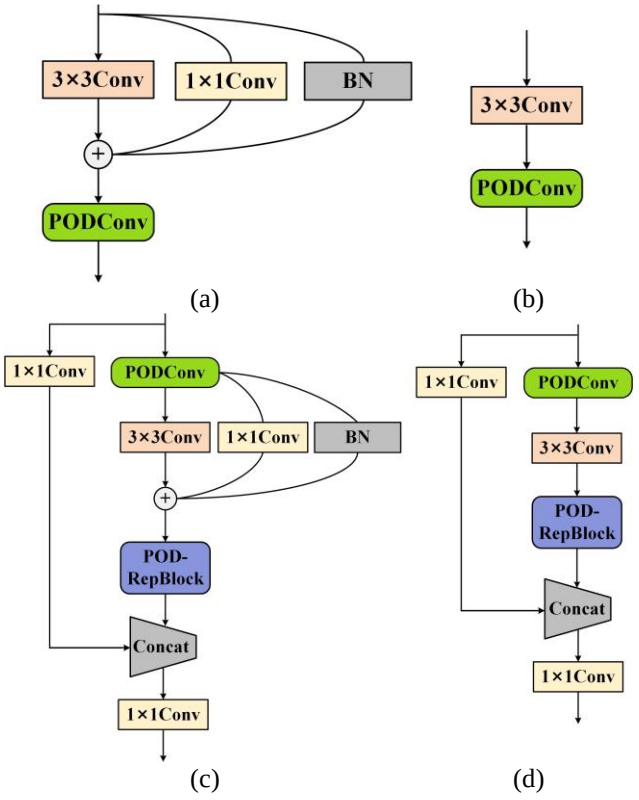


Fig. 4. POD-RepBlock and POD-RepC3 structure diagram

dynamically adjusts the convolution kernel size for certain input channels to obtain varying receptive fields, merges spatial features across different ranges, reduce parameters, and optimizes parameter utilization in feature fusion for enhanced model performance. Then, RepConv learns rich semantic features, captures the shape, texture, and reduces the memory footprint of the model by reducing the parameters of the convolutional layer. Thus, it makes model more lightweight and efficient. Eventually, POD-RepBlock incorporates feature information from various scales to improve the model's comprehension of object context. This allows POD-RepC3 module to propagate different layers of features during fusion, promoting feature information transmission and preservation of spatial details. Consequently, this improves feature expression capability and boosts object detection algorithm performance.

E. Feature Fusion Network RepBi-DFPN

The feature pyramid network aims to integrate features from different scales, which has been proven to be a key and effective part of object detection [23]. However, the single sequential fusion approach used by FPN weakens the correlation between features in each layer as the distance increases, making it difficult to establish an effective mapping between deep and shallow features and ensuring adequate feature fusion. Considering the limitations of one-way information flow, PANet incorporates a bottom-up path aggregation network to reduce the length of information path for low-layer and high-layer features. This approach facilitates accurate signal propagation of low-layer features but comes with higher computational costs. BiFPN

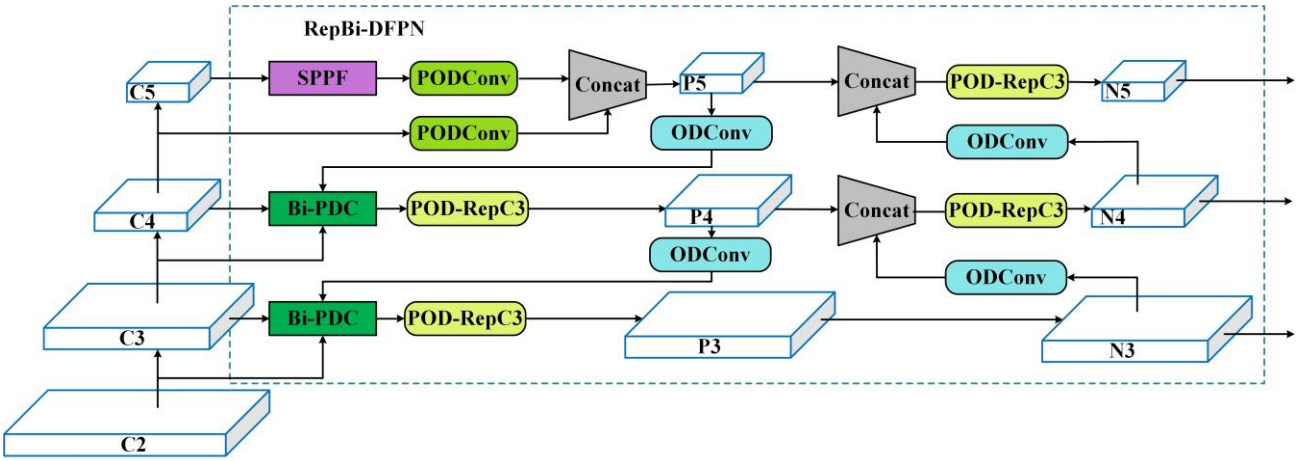


Fig. 5. Structure of the RepBi-DFPN

eliminates nodes with only one input edge and adds skip-layer connections from the original input at the same layer.

The FPN, PANet, and BiFPN architectures focus on feature fusion without considering intra-block connectivity of feature fusion networks. Therefore, we propose a new Reparameterized bi-directional dynamic feature pyramid network (RepBi-DFPN) that is both flexible and lightweight. The Bi-PDC in Fig.5 combines feature maps Ci-1, Ci, and Pi+1 from the backbone network to obtain the multi-scale object information and transmit the feature information comprehensively. The POD-RepC3 module enhances the expression ability of underlying features by processing the feature map with sufficient context information and fusing multi-scale features with abundant semantic information. The feature map Pi and Ni-1 are fused again after ODConv feature extraction, effectively integrating high-level semantic information and increasing flow of feature information.

RepBi-DFPN effectively integrates multi-scale feature information by the flexible cross-scale connections and intra-block connection, improves semantic expression ability of features whereas preserving accurate object localization and provides rich feature information. This makes it possible to accurately perceive and localize objects of different sizes, thereby improving object detection performance.

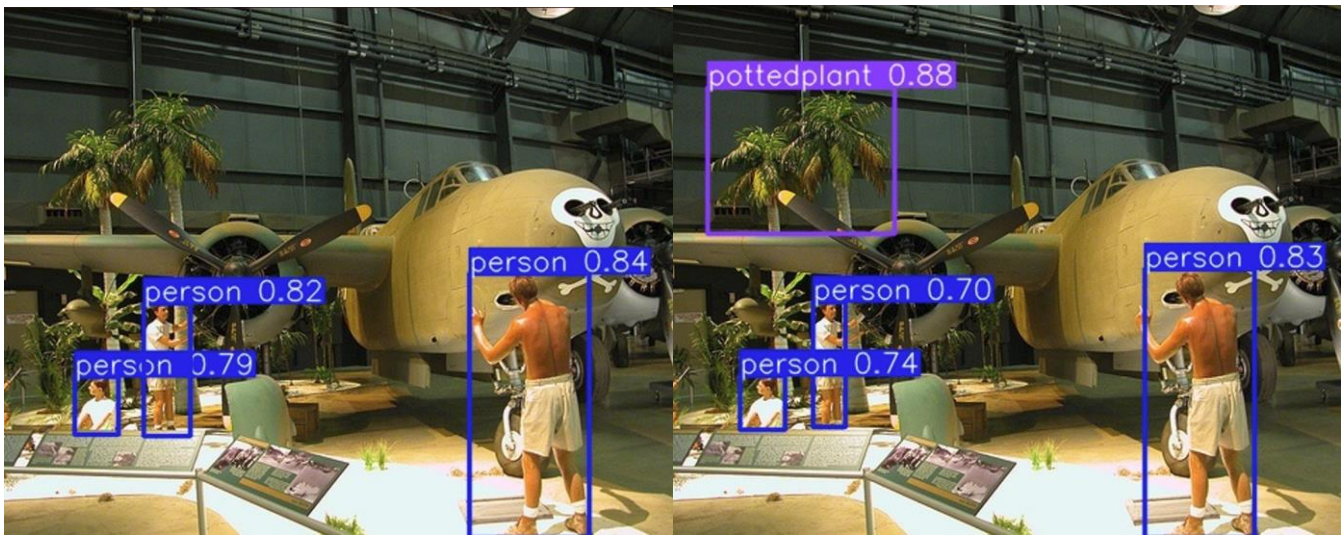
IV. EXPERIMENT

A. Experiment Dataset

We used images from all classes of the PASCAL VOC2007 and 2012 datasets to validate the effectiveness of the model. The dataset encompasses a wide range of 20 distinct object categories, including people, animals, furniture, etc., with a total of 16,551 images in the train+val part of the VOC2012 and 4,952 images in the test part of the test set in the VOC2007.

B. Experimental Environment and Parameter Setting

The experimental environment is Windows 11 operating system, the CPU is AMD Ryzen 7 5800H with Radeon Graphics, the graphics card is NVIDIA GeForce RTX 3060, and the video memory is 6GB. The model is implemented using the Python 1.9.1 deep learning framework and the Python 3.7 programming language, and GPU acceleration using CUDA11.1.



(a) (b)

Fig. 6. Comparison of YOLOv5s and POD-YOLO detection results

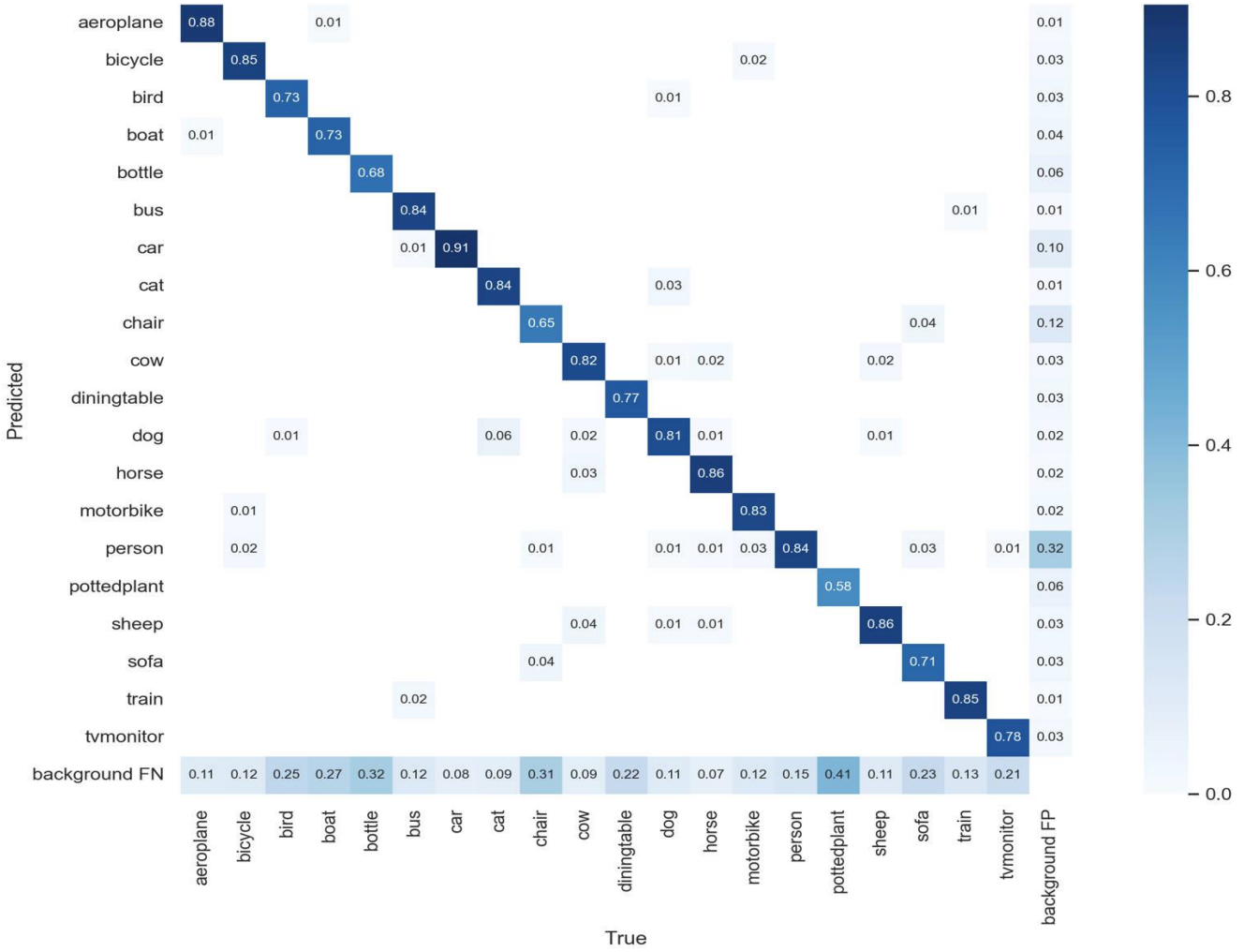


Fig. 7. Confusion matrix for POD-YOLO object detection experiment

The algorithm requires an input image size of 640 × 640, the Warmup method is used to warm up the model and the network is trained using the cosine learning rate decay strategy, the initial rate of learning is set at 0.01, and the last round of learning rate decay ratio is 0.01. The size of the training batch is 16, and the total number of epochs is 300. The network parameters are iterated using the stochastic gradient descent optimizer SGD, incorporating a weight decay rate of 0.0005 and a momentum factor of 0.937.

C. Evaluation Measures

To evaluate the efficacy of the object detection model proposed in this paper, mAP, Params and FLOPs are used as evaluation indexes.

Create a graph displaying the precision-recall curve, where recall is represented on the horizontal axis and precision on the vertical axis and integrate it to find the area AP under the curve. AP indicates average precision. The relevant expressions are as stated below:

$$P = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$AP = \int_0^1 P(R)dR \tag{3}$$

P and R in the expression are precision and recall respectively. P(R) represents the P-R curve. TP denotes the true positive, which is the count of samples accurately classified as positive by the model. FP indicates the false positive, which is the count of samples that the model inaccurately classifies as positive. FN indicates false negatives, which is the count of negatives that the model predicts incorrectly.

The IoU threshold used to determine whether the object is detected correctly is 0.5 to 0.95, and a total of 10 values are taken at medium intervals. The ultimate assessment metric is derived by computing the mean of 10 precision values, denoted as mAP@0.5:0.95. And the mean average precision obtained at an IoU threshold of 0.5 is referred to as mAP@0.5. The mAP is calculated as follows, with 1 being the number of categories:

$$mAP = \frac{1}{c}\sum_{i=1}^{c} AP_i = \frac{1}{c}\sum_{i=1}^{c}\int_0^1 P(R)dR \tag{4}$$

D. Visualization Analysis

Fig.6 illustrates the experimental outcomes of our algorithm in comparison with YOLOv5s, aiming to visually validate the efficacy of our approach. Based on the results of the experiments, it becomes apparent that the object detection using YOLOv5s in Fig.6(a) appear to miss the objects, whereas

TABLE I
RESULT OF ABLATION EXPERIMENTS

Model	Size	mAP@0.5(%)	mAP@0.5:0.95(%)	Params(M)	FLOPs(G)
Baseline	640×640	78.9	54.1	7.074	16.1
+ POD-RepC3	640×640	80.1(+1.2)	55.6(+1.5)	7.082	14.3
+ODConv	640×640	79.6(+0.7)	54.9(+0.8)	7.422	14.8
+Bi-PDC	640×640	80.6(+1.7)	56.2(+2.1)	6.958	16.3
POD-YOLO	640×640	**81.3**(+2.4)	**58.2**(+4.1)	8.177	14.1

TABLE II
PERFORMANCE COMPARISON OF VARIOUS ALGORITHMS

Model	Size	mAP@0.5(%)	mAP@0.5:0.95(%)	Params(M)	FLOPs(G)
YOLOv5s	640×640	78.9	54.1	7.1	16.1
YOLOv5m	640×640	**81.5**	57.5	20.9	48.3
YOLOv4	640×640	79.0	57.3	52.9	120
MobileNetv3-YOLOv5s	640×640	55.3	32.6	**3.6**	**6.4**
ShuffleNetv2-YOLOv5s	640×640	56.1	35.4	5.6	11.6
YOLOv7-tiny	640×640	79.0	53.3	6.1	13.3
YOLO-SK	640×640	79.1	55.0	6.9	16.1
Ours	640×640	**81.3**	**58.2**	8.2	14.1

the scene is effectively scanned by the POD-YOLO algorithm to identify all objects (Fig.6(b)) and exhibits a more significant enhancement in the precision of detection when compared to the YOLOv5s algorithm. Therefore, our proposed algorithm effectively addresses the problem of inadequate fusion of global multi-scale information caused by the loss of high-frequency information in feature fusion process.

To evaluate the detection performance of POD-YOLO across various categories in the dataset, an analysis is conducted. Fig.7 shows the confusion matrix after model training, which is composed of 20 categories. The matrix's diagonal elements reveal accurate prediction and classification of most small target classes in the dataset. POD-YOLO algorithm not only comprehensively considers the target information but also effectively utilizes highly relevant feature information. This solves the problem that small or obscured targets are prone to lose important information during object detection.

E. Ablation Studies

The proposed method's effectiveness is verified through ablation experiments on PASCAL VOC dataset, maintaining consistent environment and parameter settings. Table I displays the results of the ablation experiments (bold indicates the highest precision). In the table, POD-RepC3 indicates that the POD-RepC3 module is applied to feature fusion network, ODConv indicates that the conventional convolution in RepBi-DFPN is replaced by ODConv. Bi-PDC means that the feature fusion network uses the new Bi-PDC module at its neck. The ablation experiment compares the parameters Params, FLOPs and mAP of the algorithm, where Baseline represents the backbone network of YOLOv5s.

To augment the capacity for expressing features, the POD-YOLO uses the POD-RepC3 module to promote the transmission and share of feature information. As shown in Table.1, with the introduction of this module, the FLOP is reduced by 1.8G, and the mAP@0.5 increased by 1.2%, the mAP@0.5:0.95 increased by 1.5%. The use of POD-RepC3 in the feature fusion process not only significantly decreases computational redundancy but also effectively enhances the model's understanding of the object context, thereby improving recognition accuracy. Using ODConv instead of traditional convolution effectively integrates semantic information and greatly improves accuracy while reducing FLOP by 1.3G. The mAP@0.5 improved by 0.7% and the mAP@0.5:0.95 improved by 0.8%. More comprehensive implementation of multi-scale feature fusion is particularly important for feature fusion networks. Therefore, The Bi-PDC module serves as the intermediary component connecting the backbone and the feature fusion network in the RepBi-DFPN to accurately preserve the object to accurately preserve the object localization information. Improved the mAP@0.5 and the mAP@0.5:0.95 of the object detection model by 1.7% and 2.1%, respectively. Compared with the original model, the mAP@0.5 and the mAP@0.5:0.95 of the proposed model are significantly improved, reaching 81.3% and 58.2%, respectively.

F. Comparison With Other State-of-the-art Detectors

The detection capability of POD-YOLO is further validated through comparative experiments conducted on the baseline model, YOLO lightweight models, and YOLO complex models. Experimental conditions for each detection algorithm are consistently maintained, with all algorithms utilizing the same training and testing datasets. The results of these experiments are displayed in Table II.

1) Comparison with the baseline model

In terms of model complexity, although POD-YOLO has more parameters than YOLOv5s, POD-YOLO has less computation than the baseline model. And its performance surpasses that of the YOLOv5s by a considerable margin. The mAP@0.5 and the mAP@0.5:0.95 increased by 2.4% and 4.1%, respectively. Meanwhile, with the purpose of compare the effectiveness of model more intuitively, the mAP curves of the POD-YOLO and YOLOv5 models are compared as displayed in Fig.8. It is evident that when

compared to the baseline model, the proposed algorithm can achieve more comprehensive multi-scale feature fusion by using flexible cross-scale connection and intra-block connection in the feature fusion network, which can reduce complexity of model and efficiently improve the detection accuracy of the model.

2) Comparison with YOLO lightweight models

The POD-YOLO maintains similar model complexity to YOLOv7-tiny[24] and YOLO-SK[25], however, achieves the highest precision. And it is 2.3% and 2.2% higher than YOLOv7-tiny and YOLO-SK in the mAP@0.5, respectively. The proposed algorithm's effectiveness is further validated by replacing backbone network of the baseline model YOLOv5s with two mainstream lightweight neural network models, MobileNetv3 and ShuffleNetv2. The two models are named the MobileNetv3-YOLOv5s and ShuffleNetv2-YOLOv5s, respectively. Experimental results show that due to the influence of specific design strategies, the target details and rich context information in complex scenes cannot be captured, resulting in the low detection accuracy of MobileNetv3-YOLOv5s and ShuffleNetv2-YOLOv5s.

3) Comparison with YOLO complex models

Compared to YOLOv5m and YOLOv4, POD-YOLO has significant advantages in terms of the computation and parameters. POD-YOLO has 60.8% fewer parameters than YOLOv5m and less than a quarter of its computation, but the mAP@0.5 has same performance, and the mAP@0.5:0.95 performance is better. It is evident that POD-YOLO makes the model have a better ability of overall feature information fusion through the effective use of parameters, and finally achieves the effect of improving detection performance.

Comprehensively evaluating these different detection algorithms, the proposed algorithm is relatively better, which meets the detection requirements of multi-scale targets and maintains the characteristics of efficient network model.

V.CONCLUSION

To address the problem of insufficient cross-level feature information fusion in current object detection model, we present a novel heavy neck object detection model based on YOLOv5s. Firstly, the POD-RepC3 module is proposed to enhance the model's capacity in capturing multi-scale feature information. Meanwhile, the Bi-PDC module functions as the neck of the feature fusion network to enhance cross-level fusion of feature and solve the problem of large object size spans. Secondly, PODConv is used to the flexible adaptation of receptive field sizes, thereby to enhance the model's ability to identify and classify accurately and optimize the utilization of computing resources. Finally, a flexible, deep and long neck module RepBi-DFPN is designed for feature fusion, which fully fused the local features and global context information of the object, enable the POD-YOLO algorithm to have better overall feature information fusion capabilities.

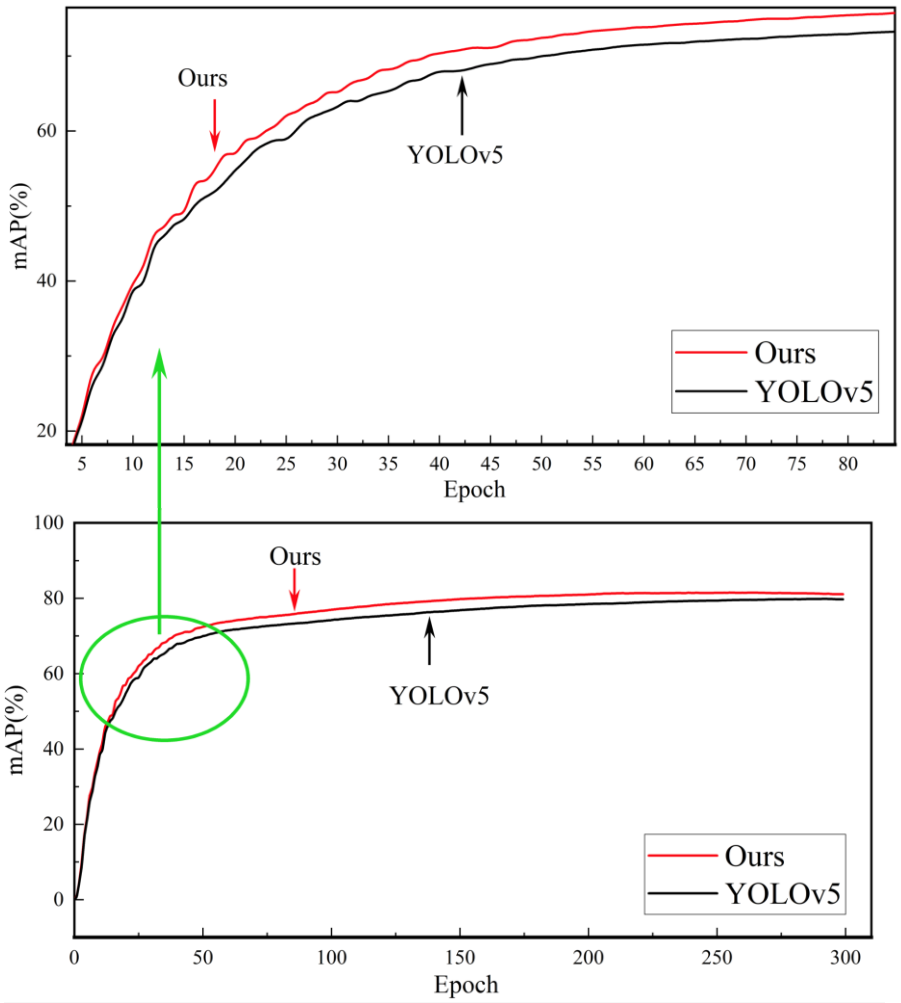


Fig. 8. mAP comparison curves of POD-YOLO and YOLOv5 models

REFERENCES

[1] M. Y. Zhang, "Vehicle Detection Method of Automatic Driving based on Deep Learning," *IAENG International Journal of Computer Science*, vol. 50, no.1, pp.86-93, 2023.

[2] R. Girshick , J. Donahue, T. Darrell, "Rich feature hierarchies for accu rate object detection and semantic segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014: 580-587.

[3] R. Girshick, "Fast R-CNN," *2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015*, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.

[4] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 1 June 2017, doi: 10.1109/TPAMI.2016.2577031.

[5] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016*, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Computer VisionECCV2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pp. 21–37, 2016.

[7] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *201 7 IEEE Conference on Computer Vision and Pattern Recognition (CV PR)*, Honolulu, HI, USA, 2017, pp. 6517-6525, doi: 10.1109/CVPR.20 17.690.

[8] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," p. arXiv:1804.02767Accessed on: April 01, 2018. doi: 10.48550/arXi -v.1804.02767.

[9] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," p. arXiv:2004.10934Acces -sed on: April 01, 2020. doi: 10.48550/arXiv.2004.10934.

[10] Ultralytics.YOLOv5[EB/OL].(2020-06-3)[2021-4-15]. https://github.com/ultralytics/yolov5.

[11] Y. Jiang, Z. Tan, J. Wang, X. Sun, M. Lin, and H. Li, "GiraffeDet: A Heavy-Neck Paradigm for Object Detection," ArXiv, vol. abs/2202.04 256, 2022.

[12] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 21-26 July 2017, pp. 936-944, doi: 10.1109/CVPR.2017.106.

[13] J Cao, P. H. Li, H. Zhang, and G. Su, "An Improved YOLOv4 Lightweight Traffic Sign Detection Algorithm", *IAENG International Journal of Computer Science*, vol. 50, no.3, pp.825-831, 2023.

[14] L. Sifre and S. Mallat, "Rigid-Motion Scattering for Texture Classification," ArXiv, vol. abs/1403.1687, 2014.

[15] K. He, X. Zhang, S. Ren and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904-1916, 1 Sept. 2015.

[16] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation," 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8759-8768, 2018.

[17] C. Y. Wang, H. Y. M. Liao, Y. H. Wu, P. Y. Chen, J. W. Hsieh, and I. H. Yeh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 14-19 June 2020, pp. 1571-1580, doi: 10.1109/CVPRW50498.2020.00203.

[18] C. Li, A. Zhou, and A. Yao, "Omni-Dimensional Dynamic Convolution," ArXiv, vol. abs/2209.07947, 2022.

[19] J. Chen, S. h. Kao, H. He, W. Zhuo, S. Wen, C. H. Lee and S. H. G. Chan, "Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 17-24 June 2023, pp. 12021-12031, doi: 10.1109/CVPR52729.2023.01157.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 27-30 June 2016 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[21] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 21-26 July 2017 2017, pp. 2261-2269, doi: 10.1109/CVPR.2017.243.

[22] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding and J. Sun, "RepVGG: Ma king VGG-style ConvNets Great Again," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville*, TN, USA, 2021, pp. 13728-13737, doi: 10.1109/CVPR46437.2021.01352.

[23] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and Efficient Object Detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13-19 June 2020 2020, pp. 10778-10787, doi: 10.1109/CVPR42600.2020.01079.

[24] C. -Y. Wang, A. Bochkovskiy and H. -Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," 2023 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 2023*, pp. 7464-7475, doi: 10.1109/CVPR52729.2023.00721.

[25] S. Wang and X. Hao, "YOLO-SK: A lightweight multiscale object detection algorithm," *Heliyon*, vol. 10, no. 2, 2024.