Pedestrian Detection Algorithm Based on Improved YOLO

Pengwei Yu

Abstract— **Pedestrian detection is a crucial aspect of target detection in computer vision. The main objective of this field is to process input video or image data using deep learning network models and label pedestrian target locations using bounding boxes on the original data. Currently, deep learning-based pedestrian detection methods have become dominant in the field due to the rapid development of convolutional neural networks and computer technology. However, existing target detection algorithms face challenges when performing the task on embedded devices due to their complex network structure, large computational and parametric quantities, and limited energy efficiency ratio and power consumption. Furthermore, the current pedestrian detection model still experiences a high number of missed detections due to the uncertainty of the occlusion pattern and the prevalence of occlusion within the dense crowd class. To tackle the aforementioned issues, this paper presents a lightweight network architecture achieved through innovative enhancements to the YOLOv7 algorithm. The primary research focuses on optimizing the network by introducing the RepGhost module. This involves utilizing the original ELAN module and ELAN-H module from YOLOv7 to construct a lighter feature extraction network. Secondly, following the concept of RepPAN, we introduce a new efficient neck network, named RepFPN, with the goal of enhancing both accuracy and efficiency of the model. Finally, we replace the RepBlock module in RepFPN with the RepGhost module to achieve deep fusion of two lightweight and efficient structures. The experimental results on the CityPersons dataset show that the YOLOv7_RepGFPN_RepGhost network model proposed in this paper, compared with the original model, reduces the number of parameters of the network model from the original 36.90M to 20.52M while the detection accuracy is improved by 0.6% to 68.6%, and the number of processed frames per second is also somewhat improved to 32.9 FPS.**

Index Terms—**Pedestrian Detection, Lightweight, Shade, YOLO**

I. Introduction

Pedestrian detection [1] is a popular and long-standing topic in computer vision with applications in real-time surveillance, autonomous driving, and other fields. Pedestrian detection algorithms can be classified into two categories: background modeling-based algorithms and deep learning-based algorithms. While these two types of algorithms may have better detection rates for simple and sparse scenes, they may not meet the accuracy requirements of actual traffic scenes. In the face of strict accuracy requirements, the model size and detection speed of embedded application scenarios may be stretched even further. In recent years, the feature extraction capability of convolutional neural networks has become increasingly powerful, making pedestrian detection algorithms based on deep learning the mainstream. Convolutional neural networks (CNNs) can learn high-level data features from raw pixels in training data, resulting in better feature representations for complex contextual information. While traditional pedestrian detection algorithms can detect pedestrians with distinctive features, they often perform poorly in extreme detection scenarios. As research on pedestrian detection algorithms progresses, the use of convolutional neural networks to train these models is undoubtedly the right choice. Deep learning-based pedestrian detection algorithms can be divided into two categories: two-stage detection algorithms (Two Stages) and one-stage detection algorithms (One Stage). Two-stage detection algorithms include R-CNN [2], Fast-R-CNN [3], Faster-R-CNN [4], and others. This algorithm first locates the target position, generates a candidate box, and extracts feature vectors from it in the first stage. In the second stage, Convolutional neural networks is used to identify the feature information in the target candidate box and classify the target. This method achieves higher accuracy but requires cumbersome training steps and occupies a large amount of space. The two representative single-stage target detection algorithms are YOLO [5] and SSD [6]. These algorithms differ from the two-stage algorithms in that they do not require candidate frames. Instead, they directly classify and regress predictions through convolution operations. They are characterized by their fast speed and small footprint, which makes them more suitable for low power and low energy consumption in embedded application scenarios. However, this class of algorithms also has the problem of poor detection accuracy. Therefore, this paper carries out a series of research work oriented to the imbalance between model size and detection accuracy in the pedestrian detection problem.

The main contributions of this paper are as follows:

- A new lightweight feature extraction network is obtained by introducing the lightweight RepGhost module and replacing the ELAN module in the YOLOv7 network model with the RepGhost module.
- The RepFPN is a new efficient neck network proposed based on the design idea of RepPAN and applied to the detection framework.
- The RepFPN and RepGhost structures were deeply integrated, and the original RepBlock module was replaced with the RepGhost module to achieve a more lightweight effect.

Manuscript received April 02, 2024
Pengwei Yu, School of computer science and technology, Tiangong University, Tianjin, China

- In summary, this paper presents a new lightweight pedestrian detection algorithm based on the YOLOv7 algorithm. The algorithm is improved through a series of innovative modifications and is proven to have significant advantages over other similar algorithms through comparative experiments on the CityPersons [7] dataset.

II. Related Works

The field of pedestrian detection has undergone significant development after years of research, particularly in improving detection accuracy. However, based on current research at home and abroad, pedestrian detection algorithms using deep learning still have several limitations. These include difficulty in resource efficiency, low accuracy in pedestrian occlusion scenes, slow detection speed, and other issues.

Tian et al. [8] proposed the DeepParts model, which is based on the RCNN algorithm and utilizes partial detectors to address the occlusion problem in deep learning-based pedestrian detection methods. The model consists of numerous partial detectors that are not predefined but rather determined based on the input image, making it more robust. Zhou et al. [9] proposed a multi-label learning method based on a generalized partial pedestrian detection framework. The method jointly learns multiple partial detectors and mines the correlation between the partials. Zhang et al. [10] proposed a partial model that combines an attention mechanism with the Faster RCNN model to address the occlusion problem. The attention mechanism assigns greater weights to key features, directing the detector to focus on unoccluded features. Pang et al. [11] retrieved pixel information of small pedestrians by zooming in on them, using the relationship between small and large pedestrians. This approach partially solves the issue of accurately dealing with occluded pedestrians.

The algorithms mentioned above utilize an anchor-based mechanism, which is a two-stage approach to pedestrian detection. However, the prediction of each anchor generates a large number of background frames that do not contain the pedestrian target. This substantially increases the computational volume of the model and the effect is not satisfactory in real-time application scenarios with high requirements. To address the aforementioned issues, literature [12] proposed the CornerNet model. This model utilizes corner pooling for vertex localization and leverages the position information of the upper-left and lower-right corners to locate pedestrians. As a result, the detection efficiency of pedestrian targets is significantly improved.

However, unfortunately, all the methods mentioned above cannot achieve a win-win situation in terms of efficiency and accuracy, in order to solve the problems of the existing models, this paper is based on the state-of-the-art target detection algorithm YOLOv7, which was first released in July 2022. using CityPersons as the pedestrian detection dataset, and through a series of means, such as optimizing the network structure, we propose YOLOv7_RepGFPN_ RepGhost, a new pedestrian detection model. According to the final experimental results, the model shows significant improvement in the indicators of detection accuracy, detection speed and model size.

III. Method

The overall framework of the Network Modeling of YOLOv7_RepGFPN_RepGhost proposed in this paper is shown in Fig.1.

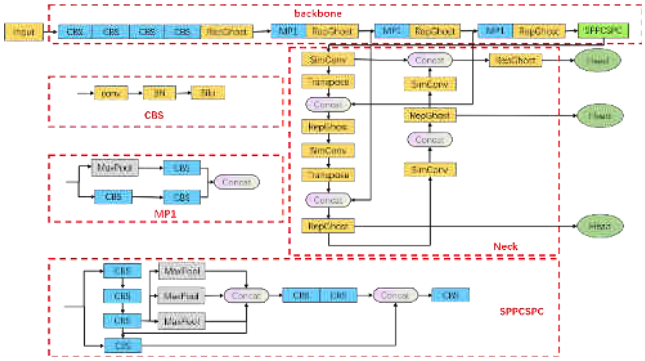

Fig. 1. YOLOv7_RepGFPN_RepGhost structure diagram

The method first introduces the RepGhost module and replaces all ELAN modules in the original YOLOv7 network with RepGhost modules, resulting in a lightweight feature extraction network. Next, a new efficient neck network called RepFPN is designed based on the idea of RepPAN. Finally, RepFPN and RepGhost are integrated more deeply by replacing the RepBlock module in RepFPN with the RepGhost module, resulting in a new lightweight algorithm for detecting pedestrian targets.

A. *RepGhost*

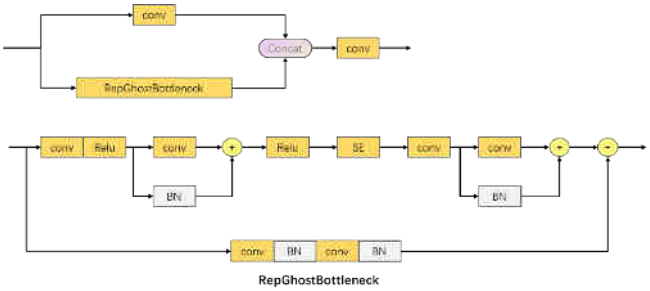

Fig. 2. RepGhost structure diagram

RepGhost [13] is a hardware-efficient Ghost module that utilizes reparameterization techniques. It is a lightweight CNN module that outperforms other lightweight modules, such as GhostNet [14] and MobileNetV3 [15], with fewer parameters and less latency on mobile devices. The feature reuse technique employed in this module has been a key factor in the design of lightweight convolutional neural networks. The technique known as feature reuse involves obtaining additional features by connecting existing feature maps from different layers. For instance, in DenseNet [16], the feature maps from previous layers are reused and fed to their successor layers within a phase, resulting in an increase in feature maps. In GhostNet, more feature maps are generated by concatenating them with the original feature maps using inexpensive operations. Both models use feature multiplexing techniques through Concat operations to increase the number of channels and network capacity while keeping the FLOPs relatively low. The Concat operation has become the standard for feature reuse. Although it has 0

Params and 0 FLOPs, its computational cost on hardware devices is not negligible. The number of parameters and FLOPs are not direct cost indicators of the actual runtime performance of a machine learning model. On hardware devices, the addition operation is more efficient than the Concat operation due to simpler memory mechanisms. Therefore, the RepGhost module considers introducing structural reparameterization methods. The effectiveness of this family of methods is validated in the design of CNN architectures. The model is trained as a complex structure to take advantage of its performance benefits, but is then converted to a simpler inference structure after training without any additional time cost. The structure of the RepGhost module is shown in Figure.2.

This paper references the RepGhost module to the YOLOv7 network model. Specifically, the ELAN module and ELAN-H module in YOLOv7, as shown in Fig. 3, are replaced by the RepGhost module, resulting in a new lightweight network model.

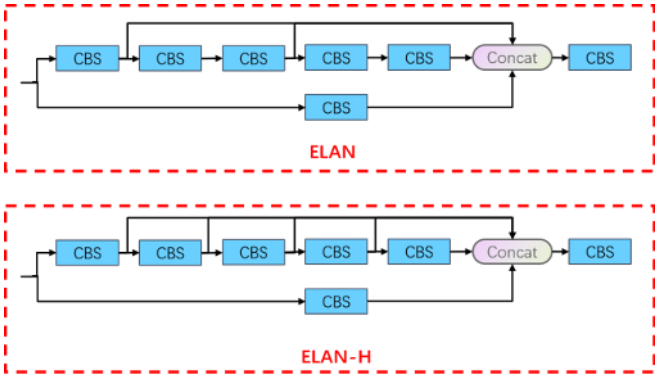

Fig. 3. ELAN+ELAN-H model structure diagram

B. *RepFPN*

This paper presents a new neck structure for YOLOv7, which is similar in structure to RepVgg [17]. The design of convolutional neural networks has received significant attention in academia and industry due to the successful results achieved by VGG in image classification and detection tasks. Several classical networks have been proposed, including Inception [18] and ResNet [19]. These well-designed architectures have improved image recognition accuracy, but struggle to balance accuracy and speed in hardware.

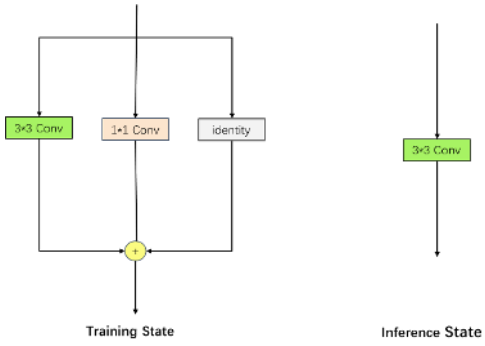

Fig. 4. Rep Conv

Fig.4 illustrates the transition of rep conv between the training and inference states. This transition involves adjusting the model structure and parameter tuning. During the initial training phase, the rep conv ensures accuracy by incorporating additional 1x1 convolution and identity mapping. These adjustments optimize the model performance and improve its accuracy. During the inference stage, the rep conv transitions should focus on increasing the speed of inference while maintaining or improving the model's accuracy. One way to achieve this is by fusing the parameters of some of the structures. This transformation typically involves reparameterizing the rep conv module by fusing the model's parameters at the end of training with the pre-trained parts for the inference phase. Through reparameterization, the multi-branch structure is converted into a single-branch 3x3 convolution in the inference state.

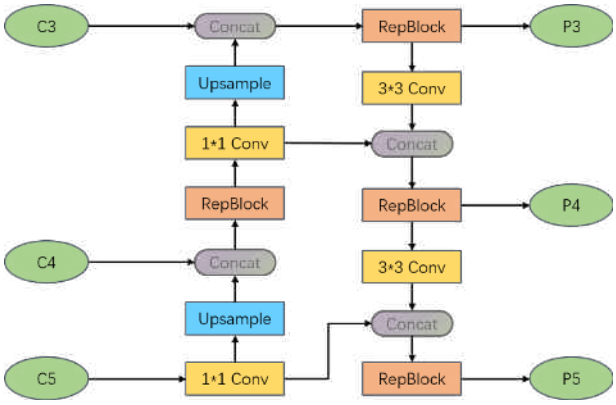

Fig. 5. RepPAN model structure diagram

Based on the Rep-PAN neck design pattern, this paper proposes a new efficient neck structure in the style of RepVGG for YOLOv7 and implements it in the detection framework. This is referred to as the Rep-PAN neck, as shown in Fig 5. Fig 6 shows the designed efficient neck network structure called RepFPN.

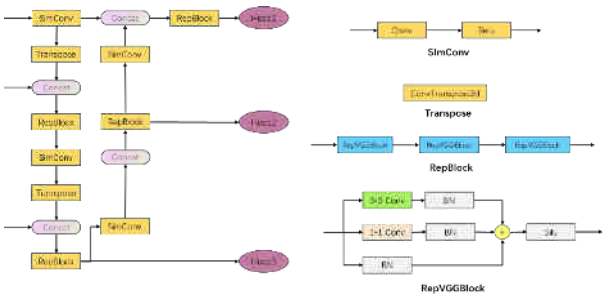

Fig. 6. RepFPN model structure diagram

C. *RepGFPN-RepGhost*

On the basis of the two improvements introduced above, the neck of the network model is further optimized and improved, taking into account the characteristics of the RepFPN and RepGhost modules, and further replacing the original Repblock module in the RepFPN with the RepGhost module, so that we get a brand new backbone network and feature pyramid, both of which together constitute the The detection framework of YOLOv7-RepGFPN-RepGhost.

IV. EXPERIMENTATION

A. *Datasets*

The CityPersons pedestrian dataset is based on the

Cityscapes [20] dataset, with the original images being re-labeled. Compared to the Caltech dataset released in 2009, the CityPersons dataset has higher image resolution, making it more suitable for deep learning models to learn the feature details of pedestrian target instances in the images. In addition, the images contain more complex scenes, diverse pedestrian instances, and higher crowd density. This allows the pedestrian detection model trained on this dataset to learn richer pedestrian poses, making the improved network model more robust to complex scenes.

The CityPersons pedestrian dataset comprises six categories, including a dummy category called 'Ignore Regions.' This category is irrelevant to the problem addressed in this paper, which focuses on improving the detection accuracy of pedestrian instances and reducing the model's weight. Since the focus of this paper is on improving the accuracy of pedestrian instance detection and reducing model complexity, the categories in the original dataset have been consolidated into a single 'human' label. Table I shows the details of the processed dataset.

Table I
Statistics of the datasets

Datasets	Image	Person	Ignore Regions	Person/image
CityPersons	2975	19654	6768	6.61

B. Experimental Setup and evaluation indicators

The experiments in this paper use the Pytorch framework for network structure modification, four NVIDIA Tesla V100 GPU boards on a docker deployment cluster for training, and an embedded device, the Jetson Xavier NX, for inference and testing operations. The environment configured in the Jetson Xavier NX development board is Pytorch 1.6, CUDA 10.4 with OpenCV 4.1.1.

In addition, in this paper, the size of the input image is set to 640*640, and CIOU Loss is used as the loss function and Adam is used as the optimizer for optimization in the network model in this chapter.The Total Batch size is set to 16, the Batch size of each GPU is set to 4, the Epoch is set to 300, and the Learning Rate is set to 0.001, in order to simulate the cosine annealing strategy, the learning rate of the network framework is automatically adjusted according to the depth of the network during the training process.

In pedestrian classification tasks, it is common to treat pedestrian instances as Positive (Positive) and the background as Negative (Negative). Based on this division, we can define the following four scenarios based on the predicted and true labeled values of the network model.1. TP (True Positive) refers to the number of targets that are correctly identified as pedestrians by the model out of all the actual pedestrian targets; 2. FP (False Positive) denotes targets that are incorrectly predicted as pedestrians by the model, but are not actually pedestrians quantity; 3, FN (False Negative) denotes the number of those that are actually pedestrians but are not identified by the model; 4, TN (True Negative) then refers to the number of non-pedestrian targets other than pedestrian targets that are correctly identified as non-pedestrian by the model. Specifically, as shown in Table II. Such categorization helps us to comprehensively evaluate the performance of the model on the pedestrian classification problem.

Table II
The four classifications of labels

	Positive	Negative
Positive	TP	FN
Negative	FP	TN

This chapter quantitatively evaluates the effectiveness of the pedestrian detection model based on the following evaluation criteria: true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN).

Recall: indicates the proportion of correctly predicted as pedestrian targets to actual pedestrian targets.

$$Recall = \frac{TP}{TP + FN} \#(1)$$

Precision: Indicates the percentage of correctly predicted pedestrian targets out of all detected pedestrian targets.

$$Precision = \frac{TP}{TP + FP} \#(2)$$

By setting thresholds for different confidence scores, different precision and recall rates can be generated, and the precision recall curve (P-R curve) can be obtained by taking the recall rate as the horizontal coordinate and the precision rate as in the vertical coordinate.

Average Precision (AP): the area under the PR curve is calculated to measure the detection precision of the model.

$$AP = \int_0^1 p(r)dr \#(3)$$

Since the dataset used in this paper merges the labels uniformly into human, the AP value in this paper is the mAP value. In addition, the pedestrian detection speed is an important indicator when measuring the real-time performance of the model, in this paper, we choose FPS (frames per second) as the detection speed measure, which stands for the number of images that can be detected per second, and the final inference detection is executed on the embedded device Jetson Xavier NX.

C. Experimental Results

Based on the improvement work mentioned above and configuring the related hardware and software environment as well as the platform, it comes to the final part of the experiment. In this chapter, the performance of the optimized pedestrian detection network model is verified through both ablation experiments based on each improvement point and side-by-side comparison tests of the improved and optimized network model with similar state-of-the-art network models.

From the results of the ablation experiments in TableIII, it can be seen that the original YOLOv7 model with 36.9M in the number of parameters has a mAP performance of 68.0% on the CityPersons dataset with a speedup of 25.1 FPS.The first improvement in this paper based on YOLOv7 is to replace the ELAN module in the original model with the RepGhost module. The data proves that after replacing the RepGhost module, the model has a 1.4% improvement in mAP with a 6.5M drop in the number of parameters. Secondly, the second improvement point is in the neck network part of the original model, based on the RepPAN neck structure, the neck network of YOLOv7 is redesigned, i.e., the RepFPN is proposed. This improvement point reaps a slightly different effect from the RepGhost module, although the decrease in the number of parameters is not very large, the mAP is sufficiently increased by 2.1%. Finally, the performance of the new YOLOv7_RepFPN_RepGhost network model was tested again after integrating the two improvements. The measured experimental results maintained more than 70% in the mAP metric, reaching 70.4%. In addition, while the accuracy is improved, the number of parameters is reduced by 12.5% compared with the original model, and the speed is also improved to a certain extent compared with the original network model, in fact, this model can satisfy a fairly wide range of application scenarios.

Table III
Ablation experiments on the test set of CityPersons

Method	mAP	Para	FPS
YOLOv7	68.0	36.9M	25.1
YOLOV7_RepGhost	69.4	30.4M	25.8
YOLOv7_RepFPN	70.1	36.6M	18.5
YOLOv7_RepFPN_RepGhost	70.4	32.3M	29.1
YOLOv7_RepGFPN_RepGhost	68.6	20.5M	32.9

However, this paper further integrates the RepFPN and RepGhost modules by replacing the original Repblock module in RepFPN with RepGhost module to form another innovative RepGFPN module. Through experiments, it can be seen that the accuracy of YOLOv7_RepGFPN_RepGhost is slightly lower than that of YOLOv7_RepFPN_RepGhost, but it is still 0.6% higher than the original YOLOv7 model, and it is worth mentioning that the number of parameters is directly reduced to 55% of the original model, and the corresponding speed reaches 29.1FPS. In summary, the improvements made in this paper in this part are very effective, and if the frame rate requirement is high in practical use, then the YOLOv7_RepFPN_RepGhost model can be used, and if the resources at the edge are very limited and there are difficulties in the deployment of the model, then the YOLOv7_RepGFPN_RepGhost model proposed in this chapter can be considered.

Table IV
Accuracy comparison with advanced detectors

method	mAP	Para	FPS
Faster-R-CNN	66.5	-	15
SSD	64.5	-	27
YOLOv4	54.9	64.0M	26.6
YOLOv5-L	59.6	46.5M	37
CornerNet	54.6	-	11
ALFNet	54.4	-	11
YOLOv7_RepGFPN_RepGhost	68.6	20.5M	32.9

After the ablation experiments were completed, in order to further verify whether the performance of the improved network model is excellent or not, this paper also made an all-round comparison between the improved network model and similar state-of-the-art models, see Table IV.

Through the final measured experimental data, it can be seen that both the YOLOv7_RepGFPN_RepGhost network model and the YOLOv7_RepFPN_RepGhost network model have a greater advantage in the two indicators of accuracy and speed when comparing with the models such as the Faster-R-CNN, SSD, CornerNet and ALFNet [21]. In addition, comparing with YOLOv5-L and YOLOv4 network models of the same series, the modified model has made a big breakthrough in the number of model parameters and mAP. It can be said that the improved network model in this paper is the most comprehensive network model with the best balance of various indexes and the most comprehensive performance. In other words, the two models proposed in this paper are the most lightweight models in terms of model size among the models with the same accuracy, and due to the lightweight design, the detection speed of the models can also reach the level of nearly 35FPS.

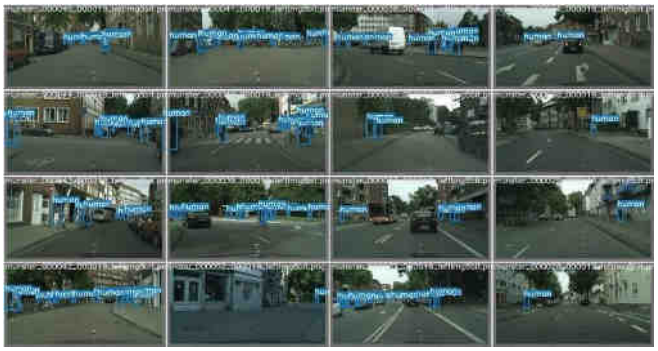

Fig. 7. Examples of labels

As shown in Fig.7, it is the case of valid targets in the original image of the CityPersons dataset, and as shown in Fig.8, it is the detection sample after using the YOLOv7_RepGFPN_RepGhost model. As can be seen from the sample, the pedestrian targets in the picture are basically recognized perfectly, which once again proves that the improvement in this paper is effective and Successful realization of model lightweight and accuracy at the same time.

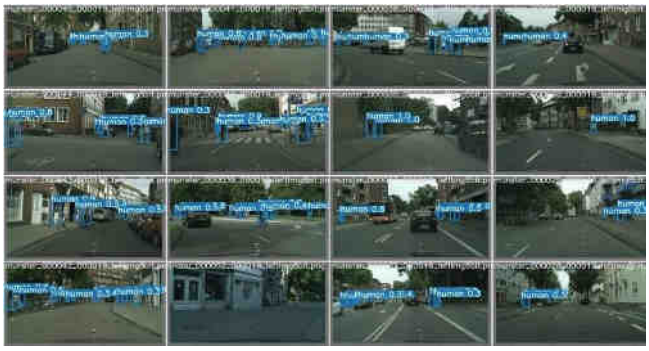

Fig. 8. Test Sample

V. CONCLUSION

In this paper, we design a solution for neural network acceleration algorithms in low-power application scenarios, which solves the problem of the contradiction between high computational demand and low-power demand that has been difficult to overcome for a long time by the RCNN series of methods and SSD algorithms in the field of embedded applications. Using the reparameterization technique, a lightweight neural network is re-designed, which can still achieve high detection accuracy while ensuring a small number of model parameters, in addition to a considerable degree of improvement in detection speed. The pedestrian detection algorithm studied in this paper meets the expectations in terms of performance, power consumption and operational efficiency, and can effectively recognize pedestrian targets with occlusion in low-power scenarios, realizing both accuracy and speed, and pushing the field of pedestrian detection to a new stage.

REFERENCES

[1] Voulodimos A, Doulamis N, Doulamis A, et al, Deep learning for computer vision: A brief review, 2018.

[2] Girshick R, Donahue J, Darrell T, et al, Rich feature hierarchies for accurate object detection and semantic segmentation, Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, 580-587.

[3] Girshick R Fast r-cnn, Proceedings of the IEEE international conference on computer vision, 2015, 1440-1448.

[4] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems, 2015, 28.

[5] Redmon J, Divvala S, Girshick R, et al, You only look once: Unified, real-time object detection, Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, 779-788.

[6] Liu W, Anguelov D, Erhan D, et al, Ssd: Single shot multibox detector, Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing, 2016: 21-37.

[7] Zhang S, Benenson R, Schiele B, Citypersons: A diverse dataset for pedestrian detection, Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, 3213-3221.

[8] Tian Y, Luo P, et al, Deep learning strong parts for pedestrian detection, IEEE Internation Conference on Commputer Vision, 2017, 3213-3221.

[9] C. Zhou and J. Yuan, Multi-label learning of part detectors for heavily occluded pedestrian detection, Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3486-3495.

[10] Zhang S, Yang J, Schiele B, Occluded pedestrian detection through guided attention in cnns, Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2018, 708–717.

[11] Pang Y, Cao J, Wang J, et al, JCS-Net: Joint Classification and Super-Resolution Network for Small-Scale Pedestrian Detection in Surveillance Images, IEEE Transactions on Information Forensics and Security,2019, 14(12): 3322-3331.

[12] Law H, Deng J, Cornernet: Detecting objects as paired keypoints, Proceedings of the European conference on computer vision (ECCV), 2018, 734-750.

[13] Chen C, Guo Z, Zeng H, et al, RepGhost: A Hardware-Efficient Ghost Module via Re-Parameterization. arXiv 2022, 2211.06088.

[14] Han K, Wang Y, Tian Q, et al, Ghostnet: More features from cheap operations, Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, 1580-1589.

[15] Howard A, Sandler M, Chu G, et al, Searching for mobilenetv3, Proceedings of the IEEE/CVF international conference on computer vision, 2019, 1314-1324.

[16] Huang G, Liu Z, Van Der Maaten L, et al, Densely connected convolutional networks, Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, 4700-4708.

[17] Ding X, Zhang X, Ma N, et al, Repvgg: Making vgg-style convnets great again, Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, 13733-13742.

[18] Szegedy C, Ioffe S, Vanhoucke V, et al, nception-v4, inception-resnet and the impact of residual connections on learning, In Proceedings of the 56th Annual Meeting of the As sociation for Computational Linguistics (Volume 1: Long Papers), 1980–1989.

[19] He K, Zhang X, Ren S, et al, Deep residual learning for image recognition, Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, 770-778.

[20] Cordts M, Omran M, Ramos S, et al, The cityscapes dataset for semantic urban scene understanding. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, 3213-3223.

[21] Liu W, Liao S, Hu W, et al, Learning efficient single-stage pedestrian detectors by asymptotic localization fitting, Proceedings of the European Conference on Computer Vision (ECCV),2018, 618-634.