ISSN 2616-5775 Vol. 7, Issue 1: 9-17, DOI: 10.25236/AJCIS.2024.070102

A Method of Lightweight Pedestrian Detection in Rainy and Snowy Weather Based on Improved YOLOv5

Liuyang Gao[1,a,*], Yun Yang[1,b]

[1]*School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi'an, China*
[a]*supreme1060399828@163.com,* [b]*yangyun11@163.com*
[*]*Corresponding author*

Abstract: *With the rapid development of artificial intelligence technology, pedestrian detection tasks have been widely used in intelligent security and other fields. In rainy and snowy weather, pedestrian detection models may be severely affected, resulting in missed detections and false alarms. This article proposes a lightweight pedestrian detection method based on improved YOLOv5 for rainy and snowy weather conditions. Firstly, to address the issue of insufficient datasets, artificial rain and snow noise is added to the collected pedestrian data. This increases the diversity of the data and improves the robustness of the model in handling pedestrian detection tasks in rainy and snowy weather. Next, a lightweight pedestrian detection model called MAC_YOLOv5s is designed based on the YOLOv5 network model. The MAC_YOLOv5s model replaces the backbone network of the original network with the Mobilenetv3 network to achieve light weight. In addition, the CBAM attention mechanism is integrated to enhance the model's anti-interference ability. The progressive feature pyramid network AFPN is introduced to fuse feature information between adjacent layers, extracting more important features. Experimental results show that the improved MAC_YOLOv5s lightweight pedestrian detection network model reduces the model size to one-third of the original YOLOv5s model, with a file size of only 5.35MB. The mAP value reaches 0.683. These results indicate that the improved MAC_YOLOv5s model has fewer parameters and a smaller model size compared to the original model, making it suitable for pedestrian detection tasks in rainy and snowy weather conditions.*

Keywords: *Pedestrian Detection, YOLOv5s network, Mobilenetv3 Network, Attention Mechanism*

1. Introduction

In recent years, with the rapid development of computer technology, artificial intelligence has gradually begun to replace manual completion of various tedious tasks due to its excellent work efficiency and high-quality work results. In many work scenarios, reliable machine vision technology has become a necessary condition for the successful completion of tasks by artificial intelligence. Therefore, the target detection problem has become one of the research directions attracting attention in the field of machine vision. Compared with general object detection tasks, the appearance and posture of pedestrians are more uncertain, which poses more difficulties and challenges for pedestrian detection technology. Pedestrian detection technology is mainly divided into pedestrian detection methods based on traditional handcrafted features and pedestrian detection methods based on deep learning[1]. Pedestrian detection methods based on traditional handcrafted features use artificially designed feature descriptors for feature extraction, and then implement classification, while pedestrian detection methods based on deep learning obtain feature extraction methods that are most conducive to classification through learning, and then perform classification on this basis[2]. Currently, with the continuous development of computer hardware devices and convolutional neural networks, pedestrian detection methods based on deep learning have been more widely used due to their advantages in detection speed and accuracy. These applications mainly include the fields of security monitoring technology and intelligent driving. Usually, in rainy or snowy weather, pedestrian detection models can be severely affected and may have problems such as missed detection and false detection. Although the performance of computer hardware has made significant breakthroughs with the continuous development of technology, the computing power of most embedded devices still has certain limitations. Many application scenarios, such as security monitoring systems and intelligent driving systems, require the use of embedded devices for control. Therefore, lightweight pedestrian detection models have become particularly important.

In China, lightweight pedestrian detection algorithms have been widely applied. For example, a research team from Kunming University of Science and Technology proposed a lightweight pedestrian detection algorithm based on YOLOv5s, which can accurately and quickly identify pedestrians while reducing model parameters[3]. In addition, a research team from Nanjing University of Information Science and Technology proposed a lightweight pedestrian detection algorithm based on EfficientNet[4], which can improve the detection speed of the network while maintaining the same accuracy, meeting the real-time requirements of embedded devices.

In foreign countries, lightweight pedestrian detection algorithms have also been widely used. For example, Google team proposed MobileNetv1, which divides traditional convolution into two steps to reduce computational complexity and model parameters, considering the limited computational resources of mobile devices[5]. Following that, the Google team introduced MobileNetv2 with depth-wise convolutions, using adaptive activation and introduced the Squeeze-and-Excitation module in MobileNetv3. Since then, the MobileNet series models have been widely used in embedded devices. Joseph et al. proposed the YOLOv3-tiny algorithm, which uses a similar structure to YOLOv3 but with reduced network depth and feature layers to meet the requirements of embedded devices[6].

In conclusion, deep learning-based pedestrian detection technology has been widely applied both domestically and internationally, and continues to be researched and innovated. As time goes on, new technologies will continue to emerge, digital management will become more prevalent, and pedestrian detection technology in the field of embedded systems will see broader application and development.

In the current mainstream pedestrian detection algorithms, the single-stage object detection algorithm YOLOv5 is widely used. YOLOv5 has four versions: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5X[7]. These versions of the model have basically the same structure, mainly differentiated by the two parameters: depth_multiple and width_multiple. Among them, YOLOv5s, with its fewer weight parameters, smaller model size, and relatively higher accuracy, is widely used in object detection tasks.

Given the widespread application of the MobileNetv3 network model on embedded devices and the excellent performance of YOLOv5s, this paper aims to improve the YOLOv5s model for pedestrian detection in lightweight rainy and snowy weather conditions. By replacing the original backbone network of YOLOv5s with the MobileNetv3 network, the model parameters of the original network are reduced. The AFPN is used to replace the original FPN structure to fuse more feature information, and the CBAM attention mechanism is introduced. The MAC_YOLOv5s model is proposed.

2. Related work

2.1. Summary of YOLOv5

YOLOv5 is similar to previous generations of YOLO algorithms in that it adopts the concept of grids, dividing the image into multiple grids. Each grid is responsible for predicting one or more objects, in simple terms, each grid can generate prediction boxes. The reason why grids can generate prediction boxes is simple. Each grid has several (usually three) template prediction boxes, each with pre-defined width, height, coordinates, and confidence[8]. Confidence represents the probability of the presence of an object within the grid.

In YOLOv5s, the network structure is divided into three parts: backbone, neck, and head. The main function of the backbone is to extract features and continuously reduce the size of the feature map. The backbone consists of Conv modules, C3 modules, and SPPF modules. The Conv module extracts and organizes the feature map, and performs operations such as downsampling, dimensionality reduction, and normalization. The C3 module consists of three Conv modules and one Bottleneck module. In the backbone, the C3 module is the most important for feature extraction. SPP stands for Spatial Pyramid Pooling, and YOLOv5 improves it to SPPF, which achieves higher running speed while maintaining the same output as SPP. The main function of SPPF in YOLOv5 is to fuse multi-scale features by concatenating the features at different scales of the same feature map. The neck of YOLOv5 obtains relatively shallow features from the backbone and concatenates them with deep semantic features. The purpose of the neck layer is to combine shallow graphical features with deep semantic features to obtain more complete features. The head layer is the Detect module, which has a simple network structure consisting of three 1×1 convolutions corresponding to three detection feature layers. Figure 1 shows the model structure of YOLOv5.

ISSN 2616-5775 Vol. 7, Issue 1: 9-17, DOI: 10.25236/AJCIS.2024.070102

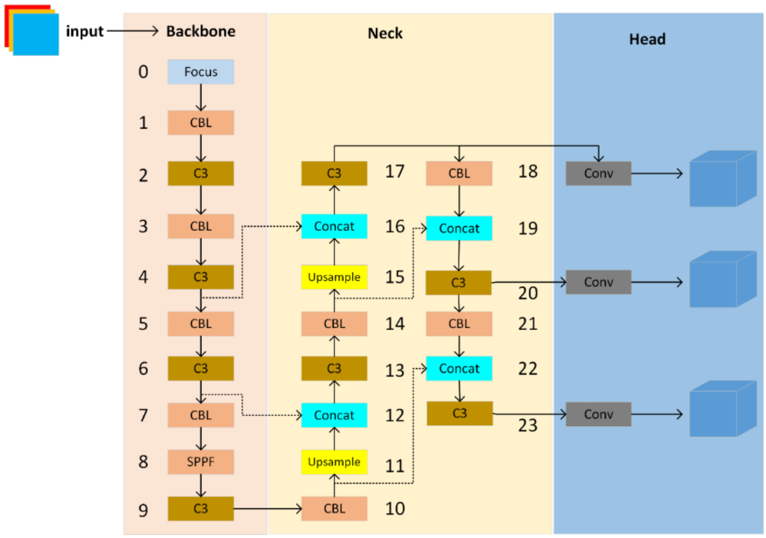

Figure 1: Model architecture of YOLOv5.

2.2. Summary of MonileNetV3

MobileNetV3 is a lightweight network model proposed by the Google team in 2019. Traditional convolutional neural networks have a large number of parameters and computations, making it difficult to run efficiently on mobile and embedded devices. To solve this problem, the MobileNet network model was introduced. MobileNetV3's overall architecture is based on the design of MobileNetV2, using lightweight depthwise separable convolutions and residual blocks, and is still composed of multiple modules. However, each module has been optimized and upgraded, including bottleneck structure, SE module, and NL module. MobileNetV3 achieved a 3.2% increase in accuracy and a 20% decrease in computational latency in the ImageNet classification task. MobileNetV3 has two versions, MobileNetV3-Small and MobileNetV3-Large, which correspond to low and high computational and storage requirements respectively[9]. The network structure of MobileNetv3-Small is shown in Table 1.

Table 1: Network architecture of MobileNetv3-Small.

Input	Operator	exp size	Out	SE	NL	S
224*224*3	Conv2d,3*3	-	16	-	HS	2
112*112*16	bneck,3*3	16	16	√	RE	2
56*56*16	bneck,3*3	72	24	-	RE	2
28*28*24	bneck,3*3	88	24	-	RE	1
28*28*24	bneck,5*5	96	40	√	HS	2
14*14*40	bneck,5*5	240	40	√	HS	1
14*14*40	bneck,5*5	240	40	√	HS	1
14*14*40	bneck,5*5	120	48	√	HS	1
14*14*48	bneck,5*5	144	48	√	HS	1
14*14*48	bneck,5*5	288	96	√	HS	2
7*7*96	bneck,5*5	576	96	√	HS	1
7*7*96	bneck,5*5	576	96	√	HS	1
7*7*96	Conv2d,1*1	-	576	√	HS	1
7*7*576	Pool,7*7	-	-	-	-	1
1*1*576	Conv2d,1*1,NBN	-	1280	-	HS	1
1*1*1280	Conv2d,1*1,NBN	-	k	-	-	1

The research objective of this paper is to develop a lightweight YOLOv5 model framework in the field of pedestrian detection. Based on the specific parameters in Table 1, it can be concluded that the MobileNetv3-Small model has fewer parameters and lower computational complexity, making it suitable for faster inference speed and lower resource consumption. Therefore, the MobileNetv3-Small backbone network is more suitable for the design objective of this paper. In conclusion, this paper replaces the backbone network of the YOLOv5s model with the MobileNetv3-Small backbone network to achieve the goal of a lightweight network model.

ISSN 2616-5775 Vol. 7, Issue 1: 9-17, DOI: 10.25236/AJCIS.2024.070102

2.3. Data pre-processing

This article aims to propose a lightweight pedestrian detection model suitable for rainy and snowy weather conditions. However, existing pedestrian datasets mainly consist of pedestrian images taken under good lighting conditions. Therefore, in the data preprocessing stage, we plan to simulate the pedestrian detection task in rainy and snowy weather environments by adding rain and snow noise. Firstly, to simulate different amounts of rain and snow, we use the get_noise function in OpenCV to generate random noise with different densities. This function mainly uses uniform random numbers and thresholds to control the noise level. Then, we stretch and rotate the noise to simulate different sizes and directions of rain and snow effects. Finally, we overlay the generated raindrops and snowflake noise on the original image to obtain simulated rainy and snowy weather scene images. Figure 2 shows the comparison of images after adding noise.

Figure 2: Comparison of images with added noise for data analysis.

2.4. Image dataset of pedestrians in rainy and snowy weather

In this article, approximately 5000 pedestrian images were selected from the COCO dataset, Pascal VOC2007 dataset[10], and Pascal VOC2012 dataset. Then, the aforementioned data preprocessing method was applied to add noise to these pedestrian images. Finally, 4000 images with rain and snow noise were selected as the dataset for this experiment. These images were divided into 2800 images for the training set, 800 images for the test set, and 400 images for the validation set.

3. Design of the MAC_YOLOv5 model

3.1. Introducing the CBAM module

In neural networks, attention mechanism is a resource allocation scheme. Typically, attention mechanism can be divided into two categories: channel attention mechanism and spatial attention mechanism. Channel attention mechanism focuses on the weights of each pixel, while spatial attention mechanism focuses on the weights of each channel. The combined attention mechanism, CBAM (Convolutional Block Attention Module), emerges by integrating these two mechanisms[11]. CBAM attention mechanism achieves better results compared to SENet (Squeeze-and-Excitation Networks) which only focuses on channel attention[12]. The structure of CBAM attention mechanism is shown in Figure 3. The specific process is as follows: given the input feature map F, the channel attention vector $M_1(F)$ is computed based on the original feature map. The feature map F is multiplied by the channel attention vector $M_1(F)$ to obtain the feature map $M_1(F)$. The spatial attention vector $M_2(F_1)$ is then calculated based on the feature map F_1. Finally, the feature map F_1 is multiplied by the spatial attention vector $M_2(F_1)$ to obtain the CBAM output result F_2 as shown in equations (1) and (2).

ISSN 2616-5775 Vol. 7, Issue 1: 9-17, DOI: 10.25236/AJCIS.2024.070102

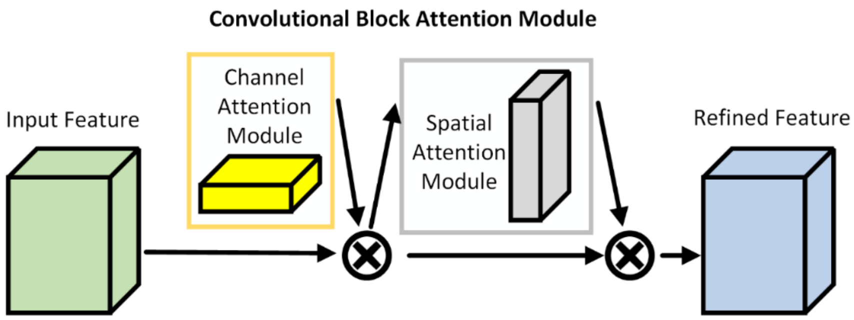

Figure 3: Structure of the CBMA attention mechanism

$$F_1 = M_1(F) \times F \tag{1}$$

$$F_2 = M_2(F_1) \times F_1 \tag{2}$$

The channel attention module architecture, as shown in Figure 4, performs global average pooling (AvgPool) and global max pooling (MaxPool) on the input feature map F. Both poolingTranslation: "The channel attention module structure is shown in Figure 4. It applies global average pooling (AvgPool) and global max pooling (MaxPool) operations to the input feature map F. Both pooling operations are performed on the height and width dimensions of the input feature map. The results of average pooling and max pooling are then processed by a shared multi-layer perceptron (Shared MLP), and the results are added together. The sigmoid activation function is applied to obtain the channel attention map, which represents the weights assigned to each channel in the input feature map. These weights range from 0 to 1. The calculation formula is shown in equation (3).

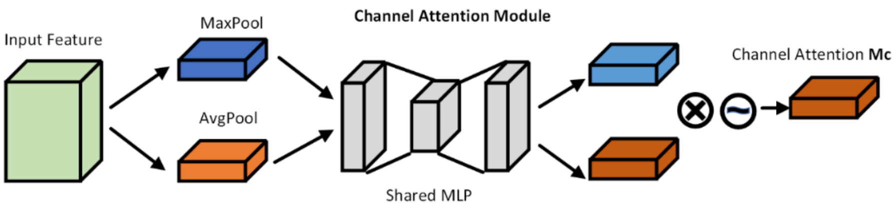

Figure 4: Channel Attention Module

$$M_C(F) = \sigma(MLP(AvgPool(F))) + MLP(MaxPool(F))) = \sigma\left(W_1(W_0(F_{avg}^c)) + W_1\left(W_0(F_{max}^c)\right)\right) \tag{3}$$

Unlike the Channel Attention Module, the focus of the Spatial Attention Module is to determine important regions of the image. The structure of the Spatial Attention Module is shown in Figure 5. First, the input feature map is subjected to max pooling and average pooling, and then stacked to generate an effective feature descriptor. Afterwards, a convolution with channel number 1 is used for concatenation and convolution to adjust the channel number. Then, a Sigmoid activation function is applied to obtain the spatial attention vector. The calculation formula is shown in equation (4).

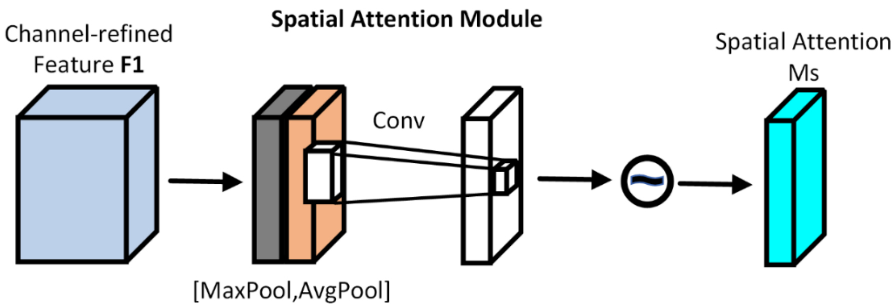

Figure 5: Module for spatial attention network

$$M_S(F') = \delta\left(f^{7\times7}([AvgPool(F'); MaxPool(F')])\right) = \delta\left(f^{7\times7}\left([F'_{avg;}F'_{max;}]\right)\right) \tag{4}$$

The Head part of YOLOv5s corresponds to three different scales of feature maps. The feature map of

ISSN 2616-5775 Vol. 7, Issue 1: 9-17, DOI: 10.25236/AJCIS.2024.070102

size 80×80 mainly detects small objects, the feature map of size 40×40 mainly detects medium objects, and the feature map of size 20×20 mainly detects large objects. Through multiple experiments, it has been found that adding the CBAM module before the prediction output layer for medium objects can enhance the extraction of key features of pedestrians during the network training process and suppress unimportant feature information, thereby improving pedestrian detection tasks. The position of incorporating CBAM into the YOLOv5 network is shown in Figure 6.

Figure 6: Network architecture diagram integrating CBAM module

3.2. AFPN replaces FPN structure

AFPN (Asymptotic Feature Pyramid Network)[13] is a progressive feature pyramid network published at CVPR in 2023. Its structure is initiated by fusing two adjacent LOW-Level features and progressively integrating High-Level features into the fusion process. This approach avoids significant semantic gaps between non-adjacent levels and prevents loss or degradation of feature information during transmission and interaction. In the FPN structure pyramid of the YOLOv5 model, the High-Level features at the top need to propagate through multiple intermediate scales and interact with these scales before being fused with the Low-Level features at the bottom. Considering that the semantic information from High-Level features may be lost or degraded during this propagation and interaction process, this paper proposes to replace the FPN structure with the APFN structure to extract more feature information. The structure of AFPN is shown in Figure 7.

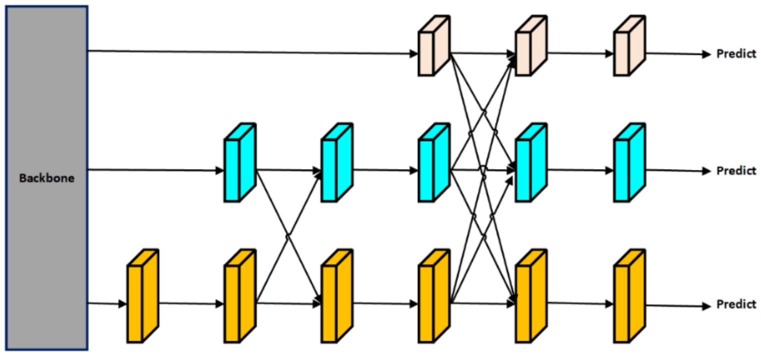

Figure 7: AFPN structure

3.3. Loss function

Alpha-IOU is a loss function based on the IOU Loss that is generalized to a new Power IOU series[14], with a single Power parameter α. Multiple experimental results have shown that the Alpha-IOU loss has many characteristics in object detection models. Firstly, it outperforms all existing IOU-based loss functions significantly. Secondly, by adjusting the parameter α, we can provide greater flexibility to the detector, resulting in better performance at different levels of bounding box regression accuracy. Additionally, the Alpha-IOU loss exhibits stronger robustness and adaptability to different datasets and noise. In the experiments, it was found that setting the parameter α to 3 yields good performance. Leveraging the advantages of Alpha-IOU mentioned above, this paper replaces the original GIOU function in YOLOv5s with Alpha-IOU for experimental analysis. The original formula for GIOU is shown in equation (5), while the replaced Alpha-IOU is shown in equation (6).

$$\mathcal{L}_{GIou} = 1 - IoU + \frac{|C\backslash(B \cup B^{gt})|}{|C|} \tag{5}$$

$$\mathcal{L}_{\alpha-GIou} = 1 - IoU^{\alpha} + \left(\frac{|C\backslash(B \cup B^{gt})|}{|C|}\right)^{\alpha} \tag{6}$$

4. Experimental Results and Analysis

4.1. Evaluation criteria

The evaluation metrics used for pedestrian detection in this experiment include detection accuracy (the proportion of correctly predicted positive samples among predicted positive samples, Precision, P), mean average precision (mAP), and recall (the proportion of correctly predicted true positive samples among original samples, recall, R).

Precision is the accuracy (also known as the precision rate), which refers to the proportion of correctly identified positive samples among all samples identified as positive. It is calculated using formula (7):

$$P = \frac{TP}{TP+FP} \tag{7}$$

Recall is the recall rate (also known as sensitivity), which refers to the proportion of positive samples that are correctly identified as positive among all actual positive samples. The formula is given by equation (8):

$$R = \frac{TP}{TP+FN} \tag{8}$$

mAP stands for Mean Average Precision, which is a metric used to measure the accuracy of object detection. It is calculated by formula (9).

$$mAP = \frac{1}{C}\sum_{i=1}^{C} AP_I \tag{9}$$

TP (True Positive) refers to cases that are predicted as positive samples but are actually negative samples; FP (False Positive) refers to cases that are predicted as positive samples but are actually negative samples; FN (False Negative) refers to cases that are predicted as negative samples but are actually positive samples; AP represents Average Precision, and C represents the total number of categories. mAP is obtained by summing the AP of each individual category and then taking the mean.

4.2. Experimental environment

The MAC_YOLOv5s algorithm proposed in this paper is implemented in the deep learning framework Pytorch. The hardware used includes a single NVIDIA GeForce RTX 3060 GPU and 16GB of memory, running on the Windows 11 operating system. The algorithm development is based on the Python language. In the experiments conducted in this paper, the network model parameters are set as follows: Batch_size is set to 16, weight decay factor is 0.0001, the training iteration is set to 300, and the initial learning rate is 0.001.

4.3. Comparative analysis of experimental results

This article compares the proposed MAC_YOLOv5s model with other commonly used lightweight object detection methods, including yolov3-tiny, yolov4tiny, and EfficientDet. Table 2 shows the performance comparison of these models on a pedestrian dataset with rain and snow noise introduced. From Table 2, it can be seen that the proposed MAC_YOLOv5s model outperforms the other three lightweight models in terms of model size, with a size of only 5.35MB. The mean average precision (mAP) is also higher than the other three lightweight models, reaching 0.686. Although the mAP of the MAC_YOLOv5s model has slightly decreased compared to the original YOLOv5s model, it still meets the requirements of practical applications.

Table 2: Experimental Data for Performance Comparison

Model	mAP	Model size
YOLOv3-tiny	0.391	33.19MB
YOLOv4-tiny	0.583	22.45MB
EfficientDet	0.627	17.1MB
MAC_YOLOv5s	**0.686**	**5.35MB**

4.4. Melting experiment

To verify the effectiveness of the MobileNetv3 backbone model, CBAM module, and AFPN module,

ISSN 2616-5775 Vol. 7, Issue 1: 9-17, DOI: 10.25236/AJCIS.2024.070102

ablation experiments were conducted on a dataset with added noise. The experimental results are shown in Table 3. By replacing the backbone network of YOLOv5s with MobileNetv3 for lightweight processing, introducing the CBAM module to the original YOLOv5s, and introducing the AFPN module to the original YOLOv5s, improvements were observed. As shown in Table 3, replacing the backbone network with MobileNetv3 reduced the overall network model size by 11.28MB. Introducing the CBAM module to the original model resulted in improvements of 0.053, 0.004, and 0.011 in terms of precision (P), recall (R), and mean average precision (mAP), respectively. Introducing the AFPN structure resulted in improvements of 0.008, 0.028, and 0.009 in P, R, and mAP, respectively. By incorporating the CBAM module, AFPN structure, and MobileNetv3 backbone model into YOLOv5s, the model size decreased by 9.05MB, with a mAP value of 0.686, meeting the requirements for pedestrian detection tasks.

Table 3: MAC-YOLOv5s ablation experiment results

Mobilenetv3	CBAM	AFPN	Precision	Recall	mAP	Model size
-	-	-	0.806	0.717	0.809	14.4MB
✓	-	-	0.721	0.585	0.628	**3.12MB**
-	✓	-	**0.859**	0.721	**0.822**	14.91MB
-	-	✓	0.814	**0.745**	0.818	13.18MB
✓	✓	✓	**0.734**	**0.602**	**0.686**	**5.35MB**

5. Conclusion

This article proposes a lightweight pedestrian detection algorithm model based on improved YOLOv5, which can detect pedestrian images in rainy and snowy weather. The improvements to the YOLOv5 network model are as follows: first, replace the backbone network of YOLOv5s model with Mobilenetv3; then add CBAM attention mechanism to the neck of the YOLOv5 network to improve the model's anti-interference ability and key feature extraction ability; next, replace the PAN structure in YOLOv5 with AFPN to enhance the fusion of features between different layers and improve the effectiveness of object detection; finally, replace the GIOU loss function in the original network with alphaIOU loss function to improve the accuracy of pedestrian object detection.

The experimental results show that the size of the MAC_YOLOv5s model is reduced to 5.35MB compared to the original YOLOv5s model, and the mAP can still reach 0.686. Compared to the YOLOv3-tiny model, the size is reduced by 27.84MB, and the mAP is improved by 0.295. Compared to the YOLOv4-tiny model, the size is reduced by 17.1MB, and the mAP is improved by 0.103. Compared to the EfficientDet model, the size is reduced by 11.75MB, and the mAP is improved by 0.059. Due to the limited dataset collected in this study, the learning of pedestrian images in different scenarios and environments is also limited. The next step is to collect more pedestrian datasets to expand the dataset used in this study and strive to further improve the accuracy of pedestrian detection.

References

[1] Ragesh N K, Rajesh R. Pedestrian Detection in Automotive Safety: Understanding State-of-the-Art [J]. IEEE Access, 2019, 7:47864-47890.

[2] Bootorabi F, Haapasalo J, Smith E, et al. Carbonic anhydrase VII–a potential prognostic marker in gliomas [J]. Health, 2011, 3(01): 6.

[3] Gao Y, WU Y H. YOLOv5s-based Lightweight Pedestrian Detection Algorithm [J]. Modern Electronics Technology, 2023, 46(22): 151-158.

[4] Chen Juan, Li Yan, Kan Xi, et al. Based on EfficientNet, a lightweight pedestrian detection algorithm [J]. Foreign Electronic Measurement Technology, 2023, 42(06): 1-9.

[5] Yi Z, Yu S Z, Xiong W L, et al. In-Field Tobacco Leaf Maturity Detection with an Enhanced MobileNetV1: Incorporating a Feature Pyramid Network and Attention Mechanism [J]. Sensors (Basel, Switzerland), 2023, 23(13).

[6] Liu Y X, Gao Chen, Wu Zhen, et al. Improved YOLOv3 Embedded Real-time Vehicle Detection Algorithm [J]. Single-Chip Microcomputer and Embedded System Application, 2023, 23(02): 33-37.

[7] Su T, Wang Y, Deng Q Y, et al. Improved foggy weather pedestrian and vehicle detection algorithm based on YOLOv5 [J]. Journal of System Simulation, 1-11 [2023-12-14].

[8] Yang H, Wang J. Efficient Detection of Forest Fire Smoke in UAV Aerial Imagery Based on an Improved Yolov5 Model and Transfer Learning[J]. Remote Sensing,2023,15(23).

[9] Wang M, Mei Q S, Song X Y, et al. A Machine Anomalous Sound Detection Method Using the lMS

ISSN 2616-5775 Vol. 7, Issue 1: 9-17, DOI: 10.25236/AJCIS.2024.070102

Spectrogram and ES-MobileNetV3 Network [J]. Applied Sciences, 2023, 13(23).
[10] Gui X Q, Liu S Q, Li L, et al. Based on the improved YOLOv8 algorithm, a pedestrian detection algorithm for scenic areas [J]. Computer Engineering: 1-11 [2023-12-14].
[11] Niu S Q, Ma R, Xun X L, et al. Research on corn seed variety recognition based on improved CBAM attention mechanism in MobileNetV2 [J]. Journal of Chinese Cereals and Oils: 1-12 [2023-12-14].
[12] Jia L, Wang Y, Zang Y, Li Q, Leng H, Xiao Z & Jiang L. (2022). MobileNetV3 with CBAM for bamboo stick counting. IEEE Access, 10, 53963-53971.
[13] Yang G, Lei J, Zhu Z, Cheng S, Feng Z, & Liang R. AFPN: Asymptotic Feature Pyramid Network for Object Detection. arXiv 2023. arXiv preprint arXiv:2306.15988.
[14] He J, Erfani S, Ma X, Bailey J, Chi Y, & Hua X S. (2021). α-IoU: A family of power intersection over union losses for bounding box regression. Advances in Neural Information Processing Systems, 34, 20230-20242.