INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 20, Issue: 5, Month: October, Year: 2025
Article Number: 6869, https://doi.org/10.15837/ijccc.2025.5.6869

CCC Publications
AGORA UNIVERSITY PRESS

Enhanced YOLOv8-based Lightweight Small Personnel Detection Algorithm for UAV Flood Emergency Rescue

Yunfan Bu

Yunfan Bu*
1.Department of Electronic Information Engineering
Hebei University of Technology, China
5340 Xiping Road, Beichen District, Tianjin, China
2.Innovation and Research Institute
Hebei University of Technology, China
9 Jingwu Road, Shijiazhuang, Hebei, China
*Corresponding author:byf13937162630@163.com

Abstract

This study proposes an enhanced lightweight small-target detection algorithm tailored for UAV-based flood emergency rescues, building upon YOLOv8. By introducing a Linear Deformable Convolution kernel and a redesigned bottleneck structure with partial convolution, the algorithm not only captures personnel target features of different scales and shapes more efficiently and achieves higher detection accuracy, but also reduces the number of model parameters. In addition, by improving the structure of the detection head and adding the ResNeXt-SENet fusion layer, the algorithm is able to suppress the interference of the complex background in emergency rescue scenarios and focus more on detecting small-targeted people, while improving the global information integration capability of the model, so that the algorithm is better applicable to different small-targeted detection datasets. Evaluation on custom flood-rescue datasets and VisDrone2019 demonstrates a significant increase in detection accuracy for small targets and reduction in the number of model parameters. The detection accuracy and model size also compare favorably with other state-of-the-art target detection algorithm models under the same experimental conditions, highlighting the suitability of the model for resource-constrained real-time UAV applications in challenging environments.

Keywords: UAV, Flood Emergency rescue, Small-target personnel detection, YOLOv8, Lightweight.

1 Introduction

The intensity and frequency of natural disasters suffered globally have been increasing in recent years due to geographic and climatic conditions. Table1 shows the frequency of global disasters during the four-year period 2020-2023, with disaster data from EM-DAT, the global disaster database of the University of Leuven, Belgium.

In these severe disasters such as storms (typhoons, hurricanes), forest fires, floods, earthquakes, etc., traditional terrestrial communication systems and basic services are destroyed, people facing

Table 1: Global frequency of natural disasters, 2020-2023

Timespan	Frequency	Number of people(million)	Wildfire	Flood	Earthquake	Storm	Volcano	Drought	Others
2020.1.1-2020.12.31	313	9896.67	6	193	14	69	3	7	21
2021.1.1-2021.12.31	367	10416.76	19	206	25	82	8	13	14
2022.1.1-2022.12.31	321	18595.51	15	163	30	66	4	20	23
2023.1.1-2023.12.31	326	9305.24	16	152	27	88	4	9	30

problems such as information barriers, shortage of supplies, and insufficient rescue forces. In such cases, UAVs are proving to be a better solution by virtue of being fast, economical, and easy to deploy [1]. When a flood disaster occurs, the UAV can use the camera to conduct a wide range of aerial reconnaissance of the disaster area, and use deep learning-based small target personnel detection algorithms to quickly process the collected data, which can detect the location of the affected people in a timely manner, thus guiding the rescue forces on the ground and protecting the lives of the affected people. In daily life, the use of UAVs to detect the distribution of people in rivers and seashores in a timely manner, so that emergency departments and rescuers can take targeted protective measures, which has an important application value for improving the efficiency and quality of rescue work.

Personnel detection in UAV flood emergency rescue essentially belongs to the category of small target detection, which is a technology that utilizes images or videos taken by UAVs for personnel identification and localization. Current target detection methods are mainly divided into two categories: feature-based methods and deep learning-based methods. The feature-based method first selects the candidate region in the image, and after obtaining the candidate frame, the candidate region is selected using Scale-Invariant Feature Transform (SIFT) [2], Histogram of Oriented Gradient (HOG) [3], and Integral Channel Feature (ICF) [4] to extract features from the target, and the extracted feature information is used to train a classifier using Support Vector Machine (SVM) to determine whether the window contains the target object of interest or not. Finally, the Non-Maximum Suppression (NMS) [5] algorithm is used to eliminate redundant candidate frames to achieve the detection of the target. However, these methods require manual feature extraction, which is a cumbersome and computationally redundant process that hinders the efficiency and accuracy of feature extraction and classification. Since 2012, convolutional neural networks have gained the favor of researchers by virtue of their powerful feature extraction capability, strong robustness and good adaptability to different datasets, and target detection techniques have begun to gradually shift from feature-based methods to deep learning. Deep learning-based target detection algorithms are categorized into single-stage and two-stage. The two-stage target detection algorithm was first proposed in 2014 by Ross Girshick [6] et al. in the R-CNN target detection algorithm, but the algorithm needs to extract features through CNN for the generation of each candidate region, which generates a large number of repetitive calculations, resulting in slower detection speed. To address the shortcomings of the former, researchers have successively proposed Fast R-CNN [7], Faster R-CNN [8] and other series of algorithms. The two-stage algorithm has higher detection accuracy, but the algorithm is more complex and the model is not lightweight enough to meet the resource-constrained real-time UAV applications. In 2015, in order to achieve the purpose of real-time detection, the typical single-stage target detection method YOLO (You Only Look Once) algorithm [9] came into being. YOLO transforms the target detection problem into a regression problem and uses a separate neural network to predict the class and location of the target with the advantages of speed and real-time performance. After continuous improvement and evolution by researchers, as of 2024, YOLO has evolved to v8, which is able to better balance the algorithm's detection speed and detection accuracy.

Although target detection based on deep learning already has good results, there are still some limitations for flood small target personnel detection. On the one hand, UAV aerial images are not only characterized by large scenes, multiple scales, variable environments, complex backgrounds and mutual occlusion, but also fewer appearance and geometric cues of people on the flood, and the lack of a large-scale dataset of small targets make it difficult to accurately identify targets. On the other hand, the design of lightweight networks sacrifices a certain level of accuracy to reduce the space occupied by the network parameters, and complex models are difficult to deploy in resource-constrained UAV platforms.To address the problem of illumination and mutual occlusion, Wu [10]et al. used the Low Level Feature Attention (LFA) module to learn to focus on the regional feature information of objects in low illumination environments to improve the performance of end-to-end detection algorithms in low illumination images. Huang [11] efficiently detected moving objects in a low-luminance scene by designing a Deep Adaptive Network (DSA-Net). Gilroy [12]et al. provided an overview of occlusion reasoning in computer vision and summarized occlusion handling strategies for object detection applications. He [13]et al. proposed a novel Distribution-based Mutually Supervised Feature Learning Network (DMSFLN) and a two-branch network architecture trained in a mutually supervised manner, which achieved excellent performance on four challenging pedestrian datasets Caltech, CityPersons, CrowdHuman, and CUHK occlusion, especially on the heavily occluded subsets. Aiming at the problems of low accuracy and lightweight algorithms for detecting small targets in UAVs, Li [14] et al. proposed a two-channel feature fusion YOLOv8 improvement network, which introduces the idea of Bi-PAN-FPN and uses the GhostblockV2 structure to replace part of the C2f module to improve the model's ability to detect small targets. Hu [15] et al. proposed the PC-YOLOv8-n network, which performs convolutional computation on some of the feature layers of Bottleneck, and introduces a two-channel feature extraction network that incorporates a lightweight attention mechanism to increase the feature fusion capability of the network.

In order to solve the lack of large-scale datasets for detecting small target personnel in UAV flood emergency rescue, while facing technical challenges such as low detection accuracy and lightweight algorithms, this paper proposes a lightweight small target detection algorithm with improved YOLOv8.

2 YOLOv8 Algorithm Process

YOLO is a popular real-time target detection algorithm, which divides the image into grid cells and predicts both the bounding box and the category of the target within each cell to achieve fast and accurate target detection. YOLO has faster speed and better accuracy than traditional target detection methods and has evolved to YOLOv8 [16].The overall structure of the YOLOv8 algorithm consists of three parts: the Backbone network (Backbone), the Neck network (Neck) and the Detection Head (Head). This algorithm can divide the model into N/S/M/L/X different sizes based on the scaling factor according to the detection needs . As the latest target detection algorithm, it is very suitable for target detection of UAV images, and the network structure is shown in Figure1, the specific improvements are as follows.

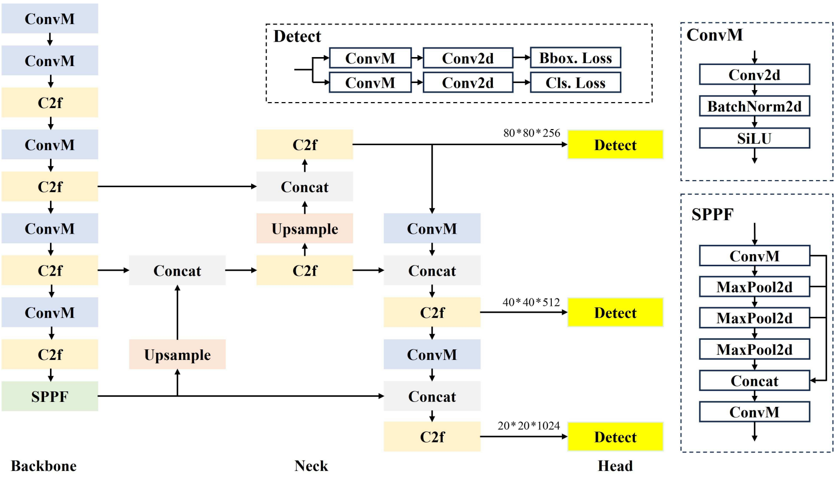


Figure 1: YOLOv8 network architecture

–The Backbone part extracts image features by convolution and pooling, and the structure used is Darknet53, which refers to the design of ELAN in YOLOv7 [18], and replaces the C3 structure of YOLOv5 with the C2f structure, which is richer in gradient streams, to achieve further lightweighting. At the same time, YOLOv8 uses the spatial pyramid pool module SPPF integrated in YOLOv5 and other architectures.

–The Neck part uses the idea of PAN to realize the feature fusion of multiple feature maps of different sizes and uses the C2f module as the main module for feature extraction, compared to YOLOv5, YOLOv8 removes the convolutional structure in the upsampling stage of PAN-FPN.

–Head is replaced by the current mainstream Decoupled-Head structure, and the two tasks of classification and detection no longer share parameters, while the Anchor-Free target detection idea with better results is adopted in the detection process.

–The loss function calculation adopts the TaskAlignedAssigner positive sample allocation strategy, introduces the distribution focal loss, takes VFL loss as the classification loss, and uses DFL loss + CIOU loss as the regression loss.

3 Improvement of YOLOv8 Algorithm

In this paper, YOLOv8s is chosen as the base model, which has a moderate number of parameters, faster detection speed and higher detection accuracy, which is suitable for the research scenario of this paper. We have accomplished the improvement of four aspects of the YOLOv8s network, and the improved network structure is shown in Figure2. By designing a kind of Linear Deformable Convolution kernel (LDConv) of size 3 in the downsampling part of the neck network, the dynamic size and shape changes of small target personnel are captured more effectively. The Bottleneck structure of the CSPLayer is reconstructed to ensure that the network effectively extracts features for better application to resource-constrained UAV platforms. Optimize the existing detection head of YOLOv8, which makes the model more focused on small targets, and the detection performance of small targets is improved. A ResNeXt-SENet fusion layer is added to the output part of the detection layer to make the algorithm better applicable to different small target detection datasets.

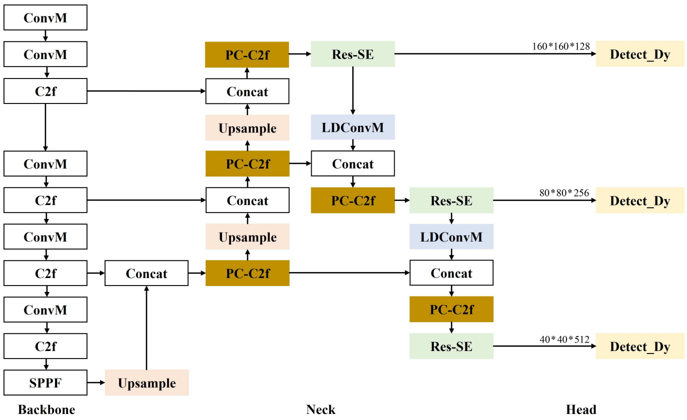


Figure 2: Overall structure of the improved YOLOv8

3.1 Linear Deformable Convolution Kernel

The standard convolutional kernel size is a fixed square shape k*k, and the number of parameters is proportional to the size. Therefore, it has good performance in dealing with some static or more regular low-complexity targets. However, small target personnel detection for flood emergency rescue is often a dynamic detection process with different shapes and sizes of targets. The height of the UAV flight, the distance between the personnel and the UAV, the lens zoom magnification, the various postures of the personnel, and whether there is occlusion, etc., all affect their size and shape in the image, resulting in the square standard convolutional kernel with a fixed sample shape not being able to adapt well to these size and shape changes. Deformable convolutional network [19, 20] by introducing offsets to the standard convolution for sample shape adjustment, so that the convolution kernel is no longer confined to the regular square shape, which can be adapted to a certain extent to the different personnel targets, enhancing the performance of the network, as shown in Figure3.

However, Deformable convolution kernel is only deformed on the basis of the standard convolutional kernel, which is not flexible enough to choose the size of the convolutional kernel, and its initial

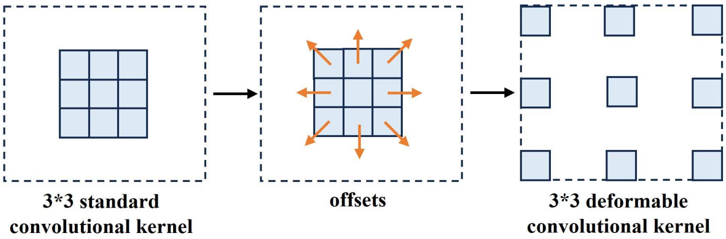

3*3 standard convolutional kernel
offsets
3*3 deformable convolutional kernel

Figure 3: Deformable Convolution process

sampling shape is still limited to a square shape k*k. As the size of the convolution kernel increases, k*k will show a square scale growth, it is not a friendly way of growth for UAV platforms with limited computational resources. Linear Deformable Convolution kernel (LDConv) adopts a convolution generation algorithm that can generate the initial positions of convolution kernels with any number of arbitrary initial sampling shapes, e.g., the number of convolution kernels can be 1,2,3,4,5,6,7..., which cannot be realized by standard convolution and deformable convolution. After generating the initial sampled shape of the convolutional kernel, offsets are still used for shape adjustment to adapt to changes in the dynamic size and shape of the person to better capture the target features, providing a new way of thinking about the trade-off between performance and network overhead.

In the process of generating the initial sampling locations of the convolution kernel, the standard convolution operation uses a regular sampling grid to localize the features at the corresponding locations, where the regular sampling grid R for a convolution kernel size of 3*3 can be used in the following equation.

$$R = \{(-1, -1), (-1, 0), ..., (0, 1), (1, 1)\}$$

The sampling grid of a standard convolutional kernel is centered at the point (0,0), however, the Linear Deformable Convolution kernel targets irregularly shaped convolutional kernels, resulting in a sampling grid without a center in many cases. Therefore, Linear Deformable Convolution first generates a regular sampling grid P_0, followed by a convolution generation algorithm that generates the initial sampling coordinates P_n of any number of convolution kernels and uses the upper left corner as the sampling origin (0,0). After defining the initial sampling coordinates P_n of the irregular convolutional kernels, P_0 and P_n are stitched together to form an overall sampling grid. Because the irregular sampling coordinates cannot be matched to the convolution operation of the corresponding size, e.g., convolution of size 5, 7, and 13, Linear Deformable Convolution achieves this by stitching into an overall sampling grid. The convolution operation corresponding to the position of P_0 can be expressed by the following equation, where w is the convolution parameter.

$$Conv(P_0) = \sum w \times (P_0 + P_n)$$

The Linear Deformable Convolution process is similar to Deformable convolutional network. First, the input feature map is passed through the standard convolution operation to obtain a tensor offset of the corresponding convolution kernel of dimension (B, 2N, H, W), where N is the size of the convolution kernel, and N = 3 in Figure4. Then the obtained tensor offsets are added to the overall sampling grid (P_0 + P_n) after stitching the initial sampling coordinates of the generated arbitrary convolutional kernel sizes to obtain the corrected specific offset coordinates. These tensor offsets are mainly applied to the stitched overall sampling grid by adjusting the shape of the sampling grid, allowing the convolution kernel to be dynamically adjusted to better fit the features of the target in the image. Next, the adjusted sampling positions are bilinearly interpolated and resampled to obtain the feature values at the corresponding positions in the original input feature map. Finally, the feature values at the corresponding locations are reshaped into shapes suitable for subsequent convolution operations and the features are extracted using standard convolution operations, as shown in Figure4. In summary, the core of the Linear Deformable Convolution operation is to utilize the initial sampling coordinates P_n, the regular sampling grid P_0, and the learned offsets to compute the final sampling point locations and obtain the corresponding feature values in the input feature map.

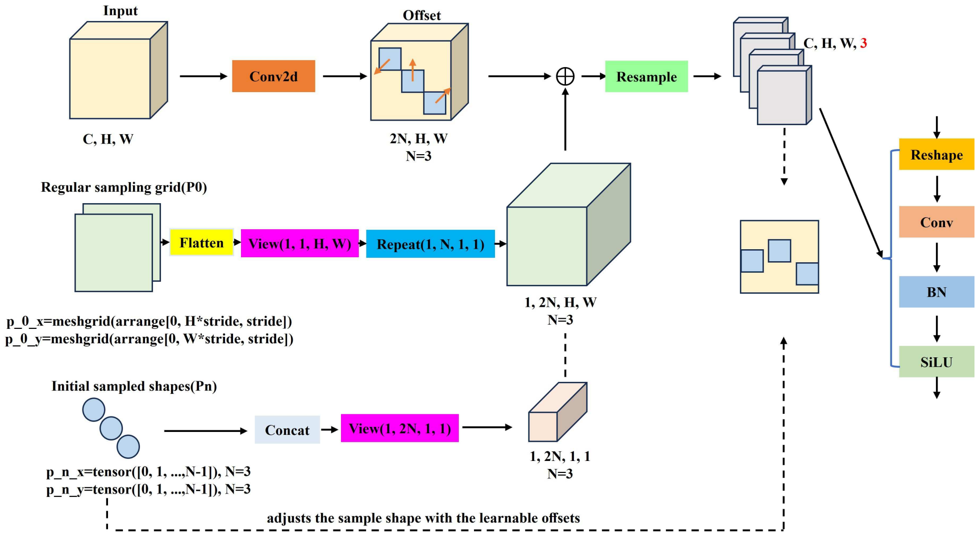

Input
C, H, W
Conv2d
Offset
2N, H, W
N=3
Resample
C, H, W, 3
Reshape
Conv
BN
SiLU
Regular sampling grid(P0)
Flatten
View(1, 1, H, W)
Repeat(1, N, 1, 1)
1, 2N, H, W
N=3
p_0_x=meshgrid(arrange[0, H*stride, stride])
p_0_y=meshgrid(arrange[0, W*stride, stride])
Initial sampled shapes(Pn)
Concat
View(1, 2N, 1, 1)
1, 2N, 1, 1
N=3
p_n_x=tensor([0, 1, ...,N-1]), N=3
p_n_y=tensor([0, 1, ...,N-1]), N=3
adjusts the sample shape with the learnable offsets

Figure 4: Linear Deformable Convolution process

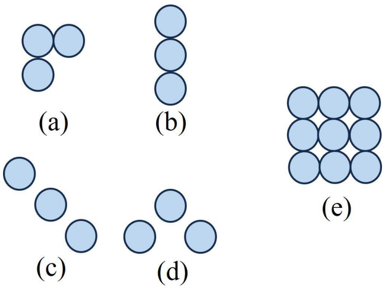


Figure 5: Initial sampling shape of Linear Deformable Convolution kernel of size 3 and 3*3 standard convolution kernel

Combined with practical application to resource-constrained UAV platforms, the algorithm needs to be lightweight. In this paper, a Linear Deformable Convolution kernel of size 3 is chosen to design four different initial sampling shapes, as shown in Figure5. The standard convolutional module is replaced with the designed Linear Deformable Convolution module in the last two downsampling stages of the YOLOv8 neck network, and the effect of LDConv with different initial sampling shapes on the network is explored through experiments. From the data in Table2, it can be seen that the initial sampling shape of the Linear Deformable Convolution kernel is very important in the customized flood rescue dataset person detection task. Designing the corresponding convolutional kernel according to the shape of the dataset person can enable the network to more fully learn the characteristics of the target person and improve the recognition accuracy. Moreover, the number of convolutional kernels can be linearly adjusted according to the actual application needs, compared to the traditional convolutional network in which the number grows with the convolutional kernel size of the square level, reducing unnecessary parameters and alleviating the complexity and computational cost of the model while maintaining the detection accuracy. In Figure5, this paper only shows some examples of size 3. However, the size of the Linear Deformable Convolution kernel is arbitrary, and the initial sampling shapes will be more diverse as the size increases.

Table 2: LDConv with different initial sampling shapes obtain YOLOv8s performance

Model	Number	Shape	P	R	mAP50	mAP50:95	Params
YOLOv8s(MyDataset)	–	–	0.728	0.638	0.693	0.296	11135987
YOLOv8s(MyDataset)	3	a	0.737	0.645	0.702	0.299	10665215
YOLOv8s(MyDataset)	3	b	0.740	0.659	0.709	0.303	10665215
YOLOv8s(MyDataset)	3	c	0.738	0.640	0.703	0.297	10665215
YOLOv8s(MyDataset)	3	d	0.746	0.644	0.708	0.301	10665215

3.2 Improvement of the C2f module

In YOLOv8 the C2f can be categorized into two structures based on the presence or absence of residual structures in the Bottleneck component, as shown in Figure6.

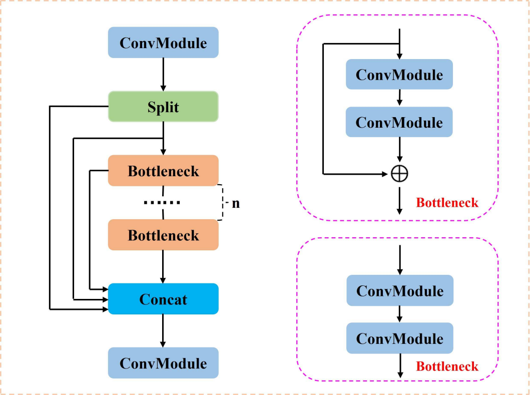


Figure 6: The two C2f structures

The C2f structure branches the series-connected Bottleneck components for cross-layer linking, adds more layer-hopping connections and Split operations, and removes the convolution operations from the branches. However, a large amount of feature map redundancy will occur in this process, and when performing network forward propagation, very similar or large amounts of duplicated information between different channels of the feature map will be processed multiple times, resulting in a waste of resources. In fact, each additional Bottleneck in the C2f structure will generate more feature mapping redundancy because the neural network performs convolution operations on all channels, including those that do not have a significant impact on the performance of the network, and this redundancy may increase computational and memory access overhead. So this paper adopts a partial convolution idea, as shown in Figure7, to utilize the redundancy in feature mapping to systematically apply regular convolution to some input channels while keeping other channels unchanged. This design advantage not only can effectively extract spatial features, but also can reduce the feature map redundancy and memory access, improve the operational efficiency and reduce the scale of parameter computation.

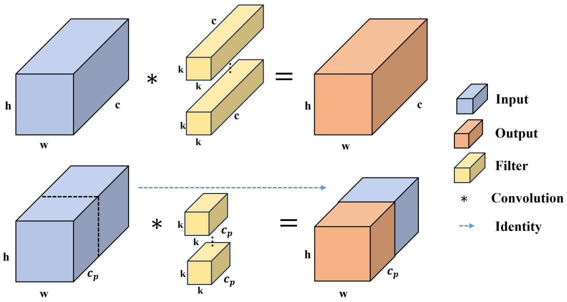


Figure 7: Comparison of partial convolution and standard convolution operation

According to the above advantages of partial convolution, this paper redesigns the lightweight C2f structure in Neck network and retains the extended feature extraction function of C2f. The improved C2f structure is shown in Figure8, which is denoted as PC-C2f. Bottleneck consists of two PConvModel modules containing partial convolutional layers, applying regular convolution to only some of the input channels for spatial feature extraction, leaving the remaining channels unchanged, and finally connect the two channels.

In the process of performing convolutional computation, the calculation of the ordinary convolution and partial convolution is as follows.

$$FLOPs = h * w * k^2 * c^2$$

$$FLOPs = h * w * k^2 * c_p^2$$

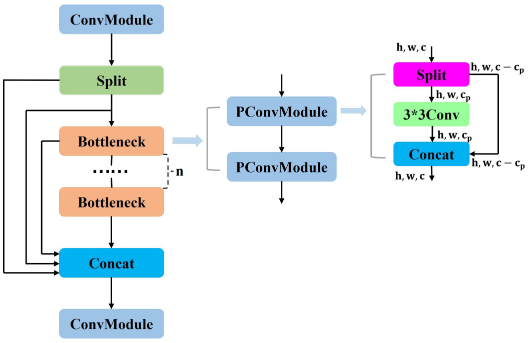


Figure 8: Improved structure of C2f

Table 3: PC-C2f with different partial convolution rates obtain YOLOv8s performance

Model	Partial convolution rate	P	R	mAP50	mAP50:95	Params
YOLOv8s(MyDataset)	1	0.728	0.638	0.693	0.296	11135987
YOLOv8s(MyDataset)	1/2	0.724	0.643	0.694	0.296	9751283
YOLOv8s(MyDataset)	1/3	0.730	0.652	0.699	0.297	9491975
YOLOv8s(MyDataset)	1/4	0.741	0.645	0.698	0.295	9405683
YOLOv8s(MyDataset)	1/5	0.731	0.642	0.699	0.298	9362393

where h , w , c are the height, width, and number of channels, k is the convolution kernel size, and c_p is the number of channels for partial convolution operation. The calculation required for partial convolution is proportional to c_p. If the partial convolution rate is set to 1/4 , the computation is only 1/16 of the regular convolution, so the problem of network model computation can be solved by controlling this variable. At the same time, reducing the convolution operation will reduce the memory access, which can effectively alleviate the memory pressure in the context of edge computing.

In this paper, for the practical application to resource-constrained UAV platforms, the partial convolution rate is set to 1/2, 1/3, 1/4, and 1/5, respectively, to experimentally explore the impact of using PC-C2f with different partial convolution rates on the performance in Neck networks. As can be seen from the data in Table3, the setting of the PC-C2f partial convolution rate is very important in the task of detecting people in the customized flood rescue dataset. Comprehensively comparing all the indicators, when the partial convolution rate is set to 1/4, the algorithm does not cause feature loss during operation, reduces unnecessary parameters while maintaining the performance, reduces the complexity and computational cost of the model, and realizes lightweighting.

3.3 Improved detection head

In the YOLOv8 detection header section, the sizes of the final detection three feature maps are 80*80*256 , 40*40*512 and 20*20*1024, as shown in Figure9(a), corresponding to the small target, medium target and large target detection headers. When the dataset application scenarios are generally small targets, the feature map size is minimally compressed to 20*20*1024. Excessive deep convolution will cause the feature information of small targets to be ignored, thus affecting the detection accuracy. Directly adding the tiny target detection head on top of the small target detection head will lead to a sharp rise in the number of parameters of the network, which affects the model detection rate and does not meet the lightweight design requirements. Therefore, this paper removes the large target detection head, and then increases the tiny target detection head, so that the sizes of the three feature maps to be detected are 160*160*128, 80*80*256, 40*40*512, corresponding to the tiny, small, and medium target detection heads, and the improved detection head is shown in Figure9(b).By improving the detection head structure, the model can better capture the features of small targets, reduce the missed detection due to too small target personnel, reducing the number of network parameters and complex algorithmic overhead.

For the person detection scenario in the water, the images presented by the UAV aerial photography usually have complex backgrounds, including dense small target data, easy to be occluded and unclear detail information, resulting in increased detection difficulty. The YOLOv8 detection head

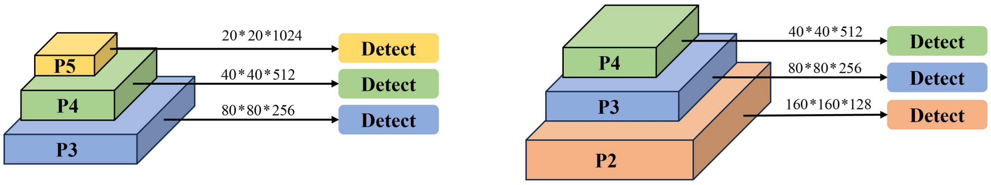


Figure 9: Detection head structure.(a)Standard detection head;(b)Improved triple detection head

adopts a single-scale prediction structure, which ignores the contribution of its scale features to the detection, and each prediction position is performed independently without considering the contextual information, which lacks the global field of view. In addition, due to the number of parameter, it is difficult for the detection head to deeply excavate spatial structural information in the features, and its expressive ability is limited.

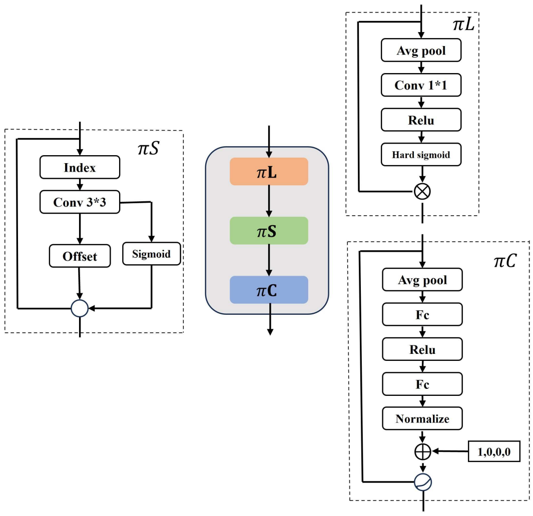


Figure 10: Framework structure of DynamicHead

The dynamic head framework uses a dynamic routing mechanism to unify scale-aware, spatial-aware and task-aware attention, and treats the input to the target detection head as a tensor with three dimensions $F \in R^{L*S*C}$ of Level(L)-Space(S)-Channel(C), as shown in Figure10. The scale-aware attention module (πL) focuses on the feature hierarchy, enabling the detection head to deal with multiple targets of different scales coexisting in the image, using average pooling, 1*1 convolution, ReLU activation function, and hard Sigmoid function; The spatial-aware attention module (πS) focuses on spatial location, applying attention to each spatial location, adaptively aggregating multiple feature levels to learn more discriminative representations, including offset learning and 3*3 convolution; The task-aware attention module (πC) focuses on the channel and adaptively supports a variety of tasks, which are processed through fully connected layers, ReLU activation functions, and normalization operations, the calculation of the attention is as follows.

$$W(F) = \pi C(\pi S(\pi L(F) * F) * F) * F$$

In this paper, the dynamic head framework is applied in the improved three-detection head of YOLOv8 to make the feature map clearer and more focused. It is able to dynamically adjust the input channel weights according to the characteristics of the target and the changes of the environment, focusing on the region where the personnel target is located and suppressing the background interference, thus adapting to the task of target detection in water rescue scenarios, and further improving the performance of the model in recognizing the personnel of different scales, complex backgrounds, and different features of small targets. As can be seen from Table4, in the customized flood rescue dataset personnel detection task, both adjusting the small target detection head and applying the dynamic head framework enhance the network performance to different degrees, improving the small

Table 4: Small target detection heads and dynamic head framework obtain YOLOv8s performance

Model	Detection head	P	R	mAP50	mAP50:95	Params
YOLOv8s(MyDataset)	Small-Medium-Large	0.728	0.638	0.693	0.296	11135987
YOLOv8s(MyDataset)	Tiny-Small-Medium	0.755	0.644	0.724	0.318	7409459
YOLOv8s(MyDataset)	T-S-M (DynamicHead)	0.758	0.685	0.745	0.326	9155495

target personnel recognition problem faced in UAV flood rescue with complex backgrounds and mutual occlusion, and also reduce the number of model parameters to a certain extent.

3.4 Introduction of the ResNeXt-SENet fusion structure

Squeeze-Excitation Networks consist of a superposition of Squeeze-Excitation (SE) blocks, the core idea is to adaptively recalibrate the feature response of each channel to enhance the network's ability to evaluate the importance of features. The adoption of the SENet architecture can effectively improve the generalization of the model over different datasets, which is helpful for the overall performance improvement of the existing model with a minimal increase in computational cost, as shown in Figure11(a). After the standard convolution operation, the features are first squeezed using global average pooling, and then the channel weights are obtained by two fully connected layer of size 1*1 (the first one is downscaled using the ReLU activation function, and the second one is upscaled using the Sigmoid activation function). Finally, the convolved features are scaled. This unique structural design helps the network to learn the dependencies between channels and adaptively strengthen or suppress certain feature channels. The ResNeXt module is characterized by a multi-branch CNN structure, and the feature maps of different branches perform convolution-merge-convolution operation, which can improve the generalization ability of the model, as shown in Figure11(b).

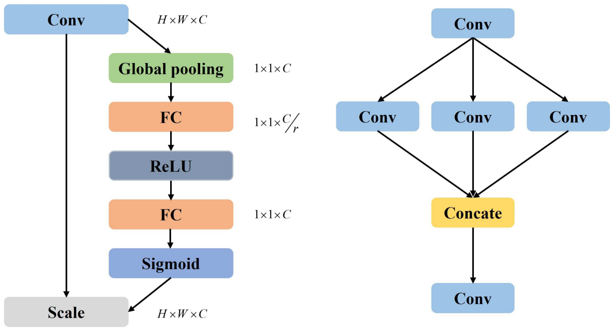


Figure 11: SENet and ResNeXt structure.(a)SENet structure;(b)ResNeXt structure

In this paper, we combine the features of ResNeXt and SENet, using a multi-branch fully connected layer to perform squeezing, excitation, and feature scaling operations, and finally the segmented feature maps in restoring their original shapes. This design allows the network to learn the different features of the input data more efficiently, and consider the dependency between different channels when performing feature transformation, as shown in Figure12. Placing this module in the output part of the detection layer further improves the fineness of the model's feature representation and the integration of global information through the multi-branch structure, making the algorithm better applicable to different small target detection datasets.

4 Experimental results and analysis

4.1 Experimental environment and parameter configuration

All experimental environments in this paper were conducted under Ubuntu 20.04 based system. Hardware configuration for GPU: NVIDIA RTX 2080Ti 11GB; CPU: 12 vCPU Intel(R) Xeon(R) Platinum 8255C CPU @ 2.50GHz. Development environment is Python 3.8; CUDA11.8; PyTorch2.0.0. The experimental parameters are set as shown in Table5.

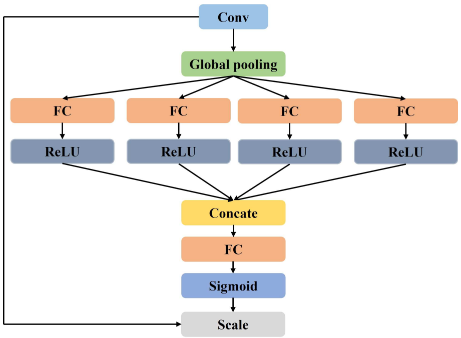


Figure 12: ResNeXt-SENet fusion structure

Table 5: Experimental model parameters

Training parameters	Values
Image Size	640*640
Momentum	0.937
Weight Decay	0.0005
Batch Size	8
Epochs	200
Learning Rate	0.01
Optimizer	SGD

4.2 Experimental datasets and evaluation indicators

The source of the water emergency rescue scenario personnel detection dataset in this paper consists of a portion of the open-source TinyPerson dataset[26], aerial photographs related to flood relief on the Internet, aerial photographs of swimmers, and actual data collected by UAVs.

The Life Search and Rescue dataset TinyPerson focuses on people near the seaside, and the size of the target figures is small. Due to the diversity of activities, the postures of the target figures show a variety of characteristics. TinyPerson is mainly applied to life rescue at sea and maritime defense, it is related to the application scenarios in this paper. Therefore, the TinyPerson dataset can effectively supplement the dataset in this paper in terms of posture and view diversity, as shown in Figure13.

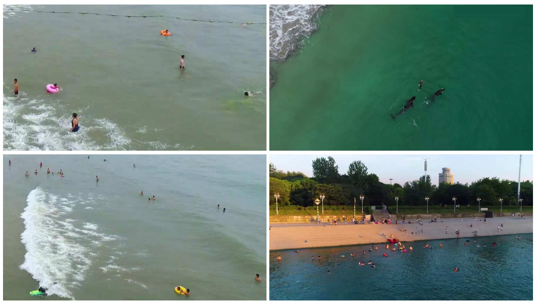


Figure 13: Example of a partial image from the TinyPerson dataset

When floods occur around the world, many rescuers and refugees are recorded by UAVs. In daily life, there are also many swimmers using UAVs to record their swimming routine. In this paper, these video or image data that have been open-sourced on the Internet are collected and organized to form a dataset, as shown in Figure14.

The fourth part of the data was obtained from the actual collection by a UAV, the model is DJI MINI3, as shown in Figure15(a). The acquisition location is along the Hai River in Tianjin, the flight altitude is 20-25m, the camera pitch angle is 45 degrees, the flight control interface is shown in Figure15(b), and some examples of the acquired images are shown in Figure16. The effectiveness of the algorithm improvement in this paper is verified by the data obtained from actual collection.

To produce labeled data for the dataset, this paper uses the annotation tool Labelimg to label personnel targets in the image. Labelimg is a powerful and easy to use open source image annotation

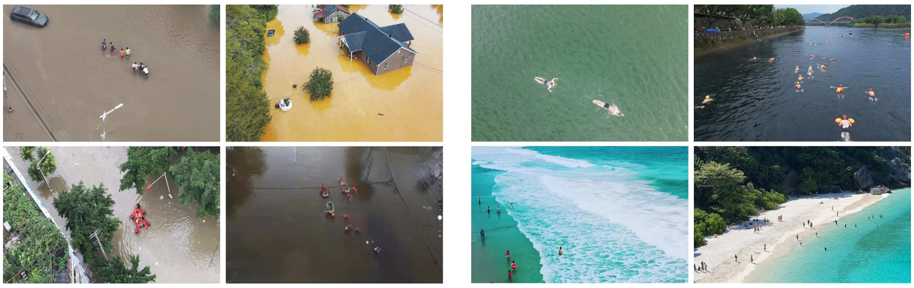


Figure 14: Subtargets for flood emergency rescue and swimmers.(a)Subtargets for flood emergency rescue;(b)Subtargets for swimmers

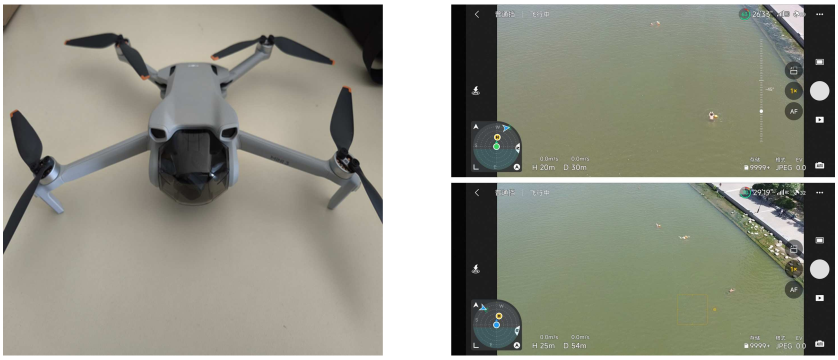


Figure 15: DJI MINI3 and flight control interface.(a) DJI MINI3;(b) Flight control interface

tool that can generate Pascal VOC (XML format) and YOLO (TXT format) and other formats of the labeled file, providing users with a simple and intuitive graphical user interface (GUI), making the labeling process more efficient and convenient. The specific annotation process is as follows: first, import the dataset folder into the tool Labelimg, and add the categories to be labeled in the label list. Since this paper is a small target personnel detection for flood emergency rescue, it only needs to focus on the person category. Figure 17 shows the working interface of labeling dataset, and Figure18 shows the labeling process and result interface.

In Figure18, the target character is selected using the labeling box, and the highlighted area is the target character area. When the labeling of all targets in the image is completed, Labeling will generate a .txt file for each labeled image, which contains the image label category and label pixel location information, as shown in Figure19, at this time the obtained label file can be used directly for model training and validation.

The dataset in this paper is characterized by very small and dense labeling of people swimming in the water and on the shore, which exists less appearance and geometric cues, and has the characteristics of large scene, multi-scale, small targets, complex background and mutual occlusion, which makes it difficult to accurately identify the targets. In this paper, we additionally add some low altitude shots of people targets in the water in the training set, especially medium-sized targets, which enables the

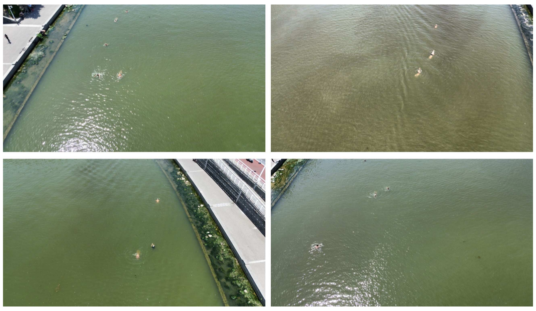


Figure 16: Pictures of actual sampling along the Hai River in Tianjin

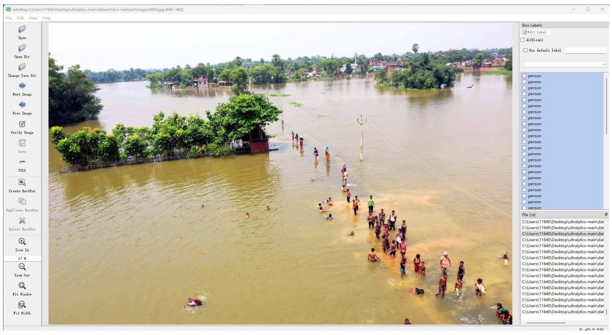


Figure 17: Labeling tool user interface

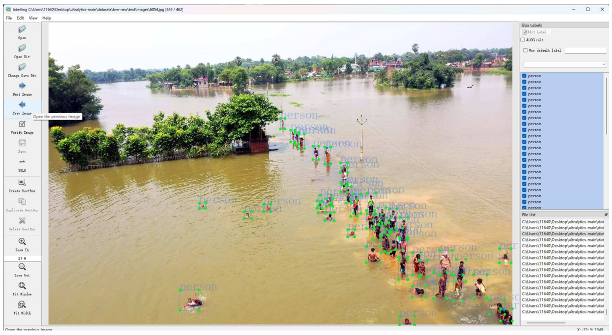


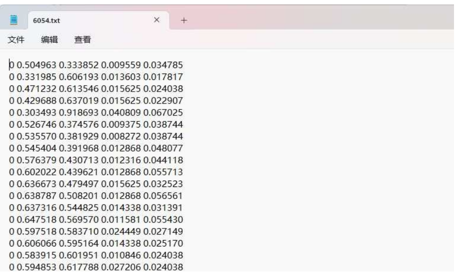
Figure 18: Labeling process and results interface

network to learn more people features, so the dataset is very suitable for the research of small target detection algorithms. The dataset has a total of 4602 images, of which 3424 are used for training, 716 are used for validation, and 462 are used for testing, including person a predefined category.

To further evaluate the effectiveness of the improved algorithm proposed in this paper, experimental validation is performed on the VisDrone2019-DET[21] public dataset collected and produced by the AISKYEYE team at Tianjin University. The dataset was captured by camera-equipped UAVs at different locations and altitudes, with a total of 10,209 images, including 6,471 training set, 548 validation set, and 3,190 test set. With ten categories of pedestrian, people, car, van, bus, truck, motor, bicycle, awning-tricycle, and tricycle. Pedestrians and distant objects are characterized by clustering of small targets in large numbers. The dataset image and labeled image are shown in Figure20. By analyzing the dataset, it can be observed that it mainly consists of small targets, which is very suitable for this paper to conduct the comparison validation experiments.

In order to demonstrate the improvement effect of YOLOv8 and analyze the various performances of the network, this paper selects the precision (P), recall (R), mean average precision (mAP), number of model parameters (Params), and the size of the weight file (Size) as the model evaluation indexes.

Precision is the rate of correct predictions out of all results predicted as positive samples, which is used to assess the accuracy of the model's predictions, the calculation of the precision is as follows. Recall is the proportion of all actual positive examples that the model correctly identifies as positive,



```
0 0.504963 0.333852 0.009559 0.034785
0 0.331985 0.606193 0.013603 0.017817
0 0.471232 0.613546 0.015625 0.024038
0 0.429688 0.637019 0.015625 0.022907
0 0.303493 0.918693 0.040809 0.067025
0 0.526746 0.374576 0.009375 0.038744
0 0.535570 0.381929 0.008272 0.038744
0 0.545404 0.391968 0.012868 0.048077
0 0.576379 0.430713 0.012316 0.044118
0 0.602022 0.439621 0.012868 0.055713
0 0.636673 0.479497 0.015625 0.032523
0 0.638787 0.508201 0.012868 0.056561
0 0.637316 0.544825 0.014338 0.031391
0 0.647518 0.569570 0.011581 0.055430
0 0.597518 0.583710 0.024449 0.027149
0 0.606066 0.595164 0.014338 0.025170
0 0.583915 0.601951 0.010846 0.024038
0 0.594853 0.617788 0.027206 0.024038
```

Figure 19: Image Labeling Information

Figure 20: VisDrone2019-DET public dataset raw and labeled images

Table 6: Ablation experiment

LDConv	PC-C2f	Res-SE	DyHead	P	R	mAP50	mAP50:95	Params	Size
–	–	–	–	0.728	0.638	0.693	0.296	11135987	21.4
✓	–	–	–	0.737	0.645	0.702	0.299	10665215	20.5
–	✓	–	–	0.741	0.645	0.698	0.295	9405683	18.1
–	–	✓	–	0.740	0.635	0.694	0.301	11308019	21.7
–	–	–	✓	0.758	0.685	0.745	0.326	9155495	17.7
✓	✓	–	–	0.734	0.654	0.706	0.301	8934911	17.2
✓	✓	✓	–	0.755	0.646	0.709	0.303	9106943	17.5
✓	✓	✓	✓	0.764	0.683	0.749	0.334	8491955	16.5

which is used to assess the model's ability to find the correct sample, the calculation of the precision and recall is as follows.

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

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

TP is the number of correctly detected samples in the test results, FP is the number of incorrectly detected samples in the test results, FN is the number of undetected samples in all correct targets.

Mean average precision (mAP) is a combination of precision (P) and recall (R) ,which used to measure the performance of the model on all categories, the calculation of the mean average precision is as follows.

$$mAP = \frac{1}{m}\sum_{i=1}^{m}\int_0^1 P(R)dR$$

Where mAP50 is the average detection accuracy of m categories when the IOU threshold is equal to 0.5, and mAP50:95 is the average value of detection accuracy at different IOU thresholds ranging from 0.5 to 0.95 in steps of 0.05.

The number of model parameters (Params) and the size of the weight file (Size) are measures of model complexity, with smaller sizes indicating that the model requires less computational power.

4.3 Analysis of ablation experiment results

Using YOLOv8s as the base algorithm, ablation experiments are conducted by adding LDConv, PC-C2f, Res-SE, and DyHead module after adjusting the detection head to the YOLOv8s network, to evaluate the performance of each improvement in the detection of small-targeted people in the customized UAV flooding emergency rescue dataset, as shown in Table6.

The analysis of the experimental results in the table shows that the detection performance of the network is improved to different degrees with the addition of each improvement module:

–Lightweight design by replacing the standard convolutional module with a Linear Deformable Convolution kernel (LDConv) module of sample shape (a) in the last two downsamplings of the Neck network reduces the number of parameters in the network by 470772 and the weight file by 0.9MB, while the mAP50 value and the mAP50:95 value are improved by 0.9% and 0.3%.

–The lightweight C2f structure PC-C2f with partial convolution rate of 1/4 is used in Neck network, comparing with the base algorithm, except for the mAP50:95 which is decreased by 0.1%, the rest of

Table 7: Comparative experiments with different datasets

Method	Params	P	R	mAP50	mAP50:95	Size
MyDataset(YOLOv8s)	11135987	0.728	0.638	0.693	0.296	21.4
MyDataset(Ours)	8491955	0.764	0.683	0.749	0.334	16.5
VisDrone2019(YOLOv8s)	11139470	0.504	0.375	0.388	0.231	21.4
VisDrone2019(Ours)	8495438	0.583	0.473	0.497	0.306	16.5

the evaluation indexes have been improved, which ensures the accuracy and realizes the lightweight of the model.

–The addition of a multi-branch fully-connected layer that combines the features of ResNeXt and SENet in the output portion of the small, medium, and large detection layers, with a small increase in the number of network parameters, and an improvement of 0.1% and 0.5% in mAP50 and mAP50:95.

–Applying DynamicHead in the improved triple-detection head of YOLOv8 to unify scale-awareness, spatial-awareness and task-awareness attention not only reduces the number of parameters by 1980492, but also improves the accuracy, recall, and mean average precision, with mAP50 and mAP50:95 increasing by 5.2% and 3.0%, further improve the performance of the model in recognizing people with different scales, complex backgrounds and small targets.

–Using a combination of this paper's improvement measures (LDConv, PC-C2f, Res-SE, DyHead) in YOLOv8s, compared to the base algorithm, the network improves the mAP50 and mAP50:95 values by 5.6% and 3.8%, the number of network parameters is reduced by 23.8%, the model size is reduced by 4.9MB, the accuracy and recall are improved by 3.6% and 4.5%.

Experimental results show that the improved algorithm of this paper is better than YOLOv8s in all indicators, proving that the design scheme of this paper has a more significant effect on the detection and identification of people, which not only improves the detection accuracy, but also reduces the number of parameters, realizes the lightweight of the model.

4.4 Comparison experiment

In order to be able to better show the effectiveness of the improved algorithm in this paper, two groups of comparison experiments are conducted in this paper. In the first group of comparison experiments, YOLOv8s was used as the base network, and the experimental dataset was replaced with the VisDrone2019-DET public dataset from the AISKYEYE team of Tianjin University, to compare the changes in precision, recall, mean average precision and other indexes, as shown in Table7. Meanwhile, the proportion of correct detection for each category is listed through the confusion matrix, as shown in Figure21, to further evaluate the effectiveness and generalization of the improved algorithmic model of this paper in the field of small target detection dataset.

For VisDrone2019-DET public dataset, all the evaluation metrics of the improved algorithm in this paper compared to the base algorithm YOLOv8s the number of parameters is reduced by 23.7%, the model size is reduced by 4.9MB, the mAP50 and the mAP50:95 are increased by 10.9% and 7.5%. At the same time, the proportion of small target pedestrian and people detecting correctly increases by 0.17 and 0.19, which has the largest increase compared with other categories, and has a more excellent small target detection performance. Therefore, the improved algorithm in this paper has good small target detection ability for UAV aerial image dataset, which fully verifies the superiority and applicability of the algorithm in this paper.

In the second set of comparison experiments, considering the characteristics of UAV target detection, the model is compared with the current mainstream classical target detection algorithms such as Faster-RCNN[8], YOLOv5s[22], YOLOv6s[23], SSD[24], and YOLOv7[18]under the same hardware and software conditions, as shown in Table8.

YOLOv5 uses the CSPDarknet53 for extracting the features of the input image, and its structure mainly consists of Conv Module, C3 Module, and SPPF Module, and adopts CIOU_Loss as the loss function of the bounding box. The algorithm utilizes the Focus mechanism to compress and combine the information in the input feature map to extract a higher level of feature representation and increase the sensory field of the network. YOLOv6 has redesigned both Backbone and Neck with RepVGG style structure, which is more friendly to hardware computational power, compilation optimization features,

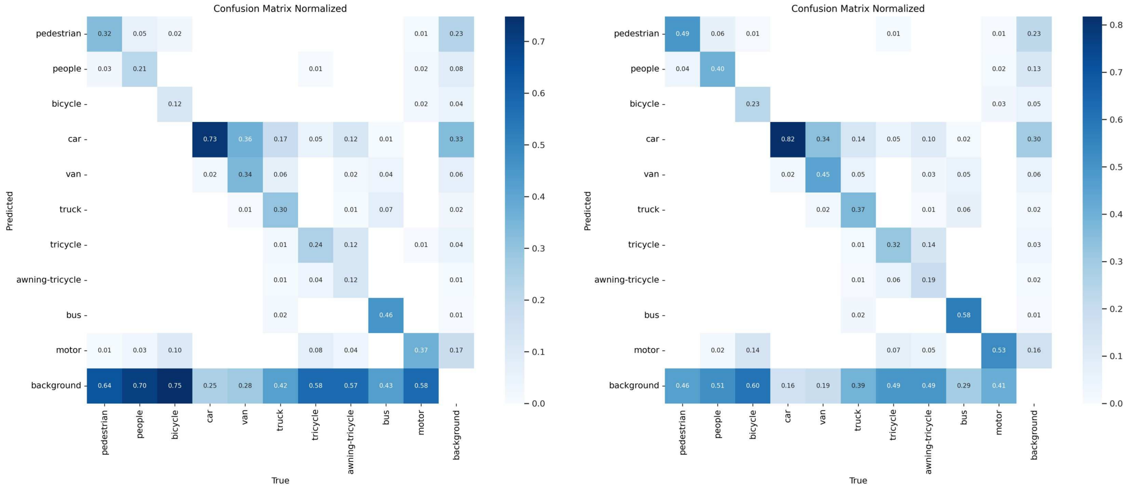


Figure 21: Normalized confusion matrix.(a)YOLOv8s;(b)ours

Table 8: Network model comparison experiment

Method	Params	P	R	mAP50	mAP50:95	Size
YOLOv5s	9122579	0.749	0.634	0.695	0.294	17.6
SSD	–	–	–	0.405	0.148	100
YOLOv6s	16306035	0.745	0.626	0.682	0.286	31.2
Faster-RCNN-FPN	–	–	–	0.711	0.309	315
Faster-RCNN	–	–	–	0.325	0.111	521
YOLOv8n	3011043	0.708	0.594	0.651	0.275	5.94
YOLOv7	9320380	0.731	0.726	0.747	0.302	18.0
YOLOv8m	25856899	0.772	0.661	0.730	0.315	49.5
Ours	8491955	0.764	0.683	0.749	0.334	16.5

network characterization ability, etc. Meanwhile, YOLOv6 also improves the training strategy by applying Anchor-free anchorless paradigm, SimOTA label assignment strategy, and SIOU bounding box regression loss. YOLOv7 employs an extended efficient layer aggregation network (E-ELAN), which enhances the information interaction and fusion capabilities between feature maps of different scales through a clever design, improving the model's detection performance on targets of various sizes. The typical two-stage detection model Faster-RCNN integrates candidate region generation, feature extraction, target classification and target frame regression in a single network, which can be viewed as a combination of Region Proposal Network (RPN) and Fast RCNN. Where Region Proposal Network (RPN) replaces selective search to generate candidate regions, Fast RCNN is used for target detection, resulting in a large improvement in comprehensive performance, especially in detection speed. Meanwhile, Faster-RCNN constructs a pyramidal feature map by combining with FPN to extract target features at different scales, which improves the detection accuracy. SSD directly predicts at multiple scales of the image without candidate frame generation and screening. The core idea is to set multiple feature maps for predicting targets on different layers of the convolutional neural network, which have different scales in space and can detect targets of different sizes. From the comparative experimental results in Table8, the improved algorithm proposed in this paper has the best comprehensive evaluation indexes compared to other excellent models, achieves higher detection accuracy with a low model size.

Summarizing the two sets of comparison experiments, the improved algorithm in this paper shows greater advantages over the base algorithm YOLOv8s in all evaluation indexes on different datasets, and has better detection performance compared with other excellent models under the same conditions. The designed PC-C2f partial convolution module reduces computational complexity and memory access while ensuring effective spatial feature extraction, making the algorithm better suited for resource-constrained real-time UAV platform applications. Introducing a Linear Deformable Convolution kernel (LDConv) to replace the original standard convolution kernel reduces model parameters and computational overhead, enhancing flexibility and feature extraction accuracy when dealing with personnel targets of various sizes in flood rescue. Improving the detection head module and adding the Dyhead module reduces the number of parameters while enabling the detection accuracy to be improved, which further enhances the performance of the model in recognizing small target personnel at different scales, complex backgrounds, and different features. The multi-branch Fully Connected Layer (FC) that incorporates the features of ResNeXt and SENet for squeezing excitation and feature scaling can improve the fineness of the model's feature expression and the integration of global information, which is more applicable to different types of small target detection datasets.

4.5 Experiment results visualization experiment

In order to show the detection effect of the improved algorithm of this paper more intuitively, the actual data of the UAV is detected with the YOLOv8s base algorithm and the improved algorithm of this paper, and the experimental results are shown in Figure22. Where the left side is the original image, the center is detected by YOLOv8s basic algorithm, and the right side is detected by the improved algorithm of this paper.

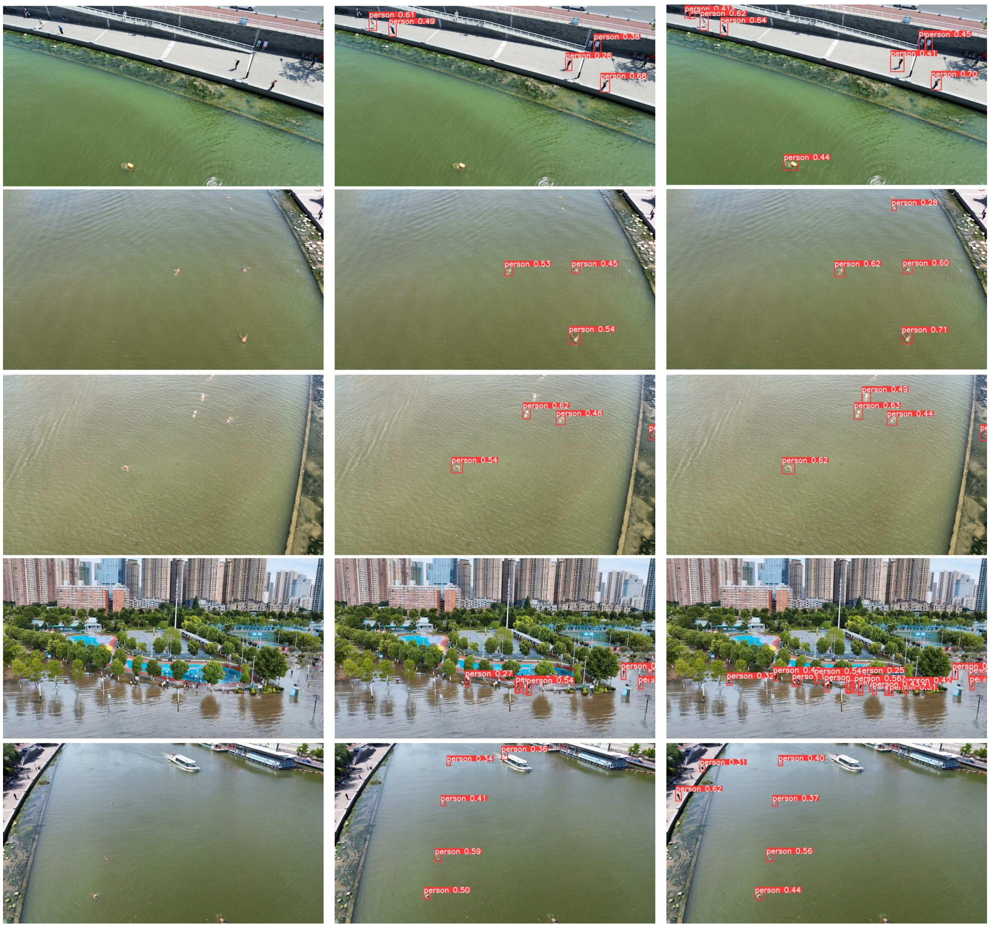


Figure 22: Visualization of the experiment, with the original graph on the left, the YOLOv8s benchmark algorithm in the middle, and the improved algorithm in this paper on the right side

It can be seen that the improved algorithm in this paper is able to detect more small targets at a longer distance compared to the basic algorithm detection, and the confidence scores of each labeled box are improved, which can be more finely adapted to the detection task in this paper, and the effectiveness of the improved algorithm is verified.

5 Conclusion

Improved algorithms based on YOLOv8 enhance the ability of UAVs to detect small-targeted people in flood rescue environments by improving detection accuracy and reducing the number of network parameters. These innovations include the use of a Linear Deformable Convolution kernel of size 3, the design of a partial convolutional bottleneck structure, detection head optimization, and the addition of a ResNeXt-SENet fusion layer, which can reliably identify small targets of people in flood emergency rescue environments. Experimental results on a customized flood rescue dataset and the VisDrone2019 small target dataset validate the algorithm's potential for real-world UAV rescue missions, striking a balance between model accuracy and resource efficiency.

However, there are still some limitations in our study which need to be further explored and addressed in future work. Firstly, in terms of dealing with occlusions, our model may face challenges when confronted with complex terrain and obstacles. For example, during flooding, trees, buildings, and other obstacles may block the UAV's line of sight, thus affecting its detection accuracy. To overcome this limitation, future research could explore combining multiple sensors to enhance the UAV's environmental sensing capabilities. Secondly, UAVs may also face other challenging environmental factors in flood emergency response, such as strong winds, nighttime environments, fog, etc., which may adversely affect the accuracy of UAV target detection. Therefore, some additional image enhancement networks can be added in future experiments, such as the low-light enhancement network Retinexformer[27] to discuss in detail the detection of UAV emergency rescue personnel targets in the dark night. Finally, for the problem of complex background and mutual occlusion of personnel targets, this paper only discusses the improvement of the overall detection accuracy, and lacks a separate discussion of the improvement effect on the personnel targets when they appear to be occluded. In the future, the SEAM attention mechanism and exclusion loss function proposed by YOLO-Face[28] can be added to optimize the target detection during emergency rescue when people are occluded from each other by compensating for the loss of response in the occluded part, enhancing the response in the unoccluded part.

In summary, although our research has achieved some preliminary results in UAV flood emergency rescue scenarios, there are still many limitations and challenges that need to be addressed. Future research should focus on overcoming these limitations and exploring more innovative technologies and methods to promote the application and development of UAVs in flood rescue. Through continuous research and improvement, we are expected to provide more accurate, efficient and reliable UAV detection and rescue services in disaster-stricken areas.

Funding

This work was supported by Science and Technology Cooperation Special Project of Shijiazhuang (SJZZXA23005).

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Khan A.; Gupta S.; Gupta S K. (2022). Emerging UAV technology for disaster detection, mitigation, response, and preparedness *Journal of Field Robotics*, 39(6), 905-955, 2022

[2] Lowe D G. (2004). Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*, 60, 91-110, 2004.

[3] Dalal N.;Triggs B. (2005). Histograms of oriented gradients for human detection, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 886-893, 2005.

[4] Kieritz H.;Becker S,; Hübner W.(2016). Online multi-person tracking using integral channel features, *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance*, 122-130, 2016.

[5] Felzenszwalb P F.; Girshick R B.; McAllester D.(2009). Object detection with discriminatively trained part-based models, *IEEE transactions on pattern analysis and machine intelligence*, 32(9), 1627-1645, 2009.

[6] Girshick R.; Donahue J.; Darrell T.(2014). Rich feature hierarchies for accurate object detection and semantic segmentation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 580-587, 2014.

[7] Girshick R.(2015). Fast R-CNN, *Computer Science*, arXiv:1504.08083, 2015.

[8] Ren S.; He K.; Girshick R.(2016). Faster R-CNN: Towards real-time object detection with region proposal networks, *IEEE transactions on pattern analysis and machine intelligence*, 39(6), 1137-1149, 2016.

[9] Redmon J.(2016). You only look once: Unified, real-time object detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[10] Wu S.; Liu Z.; Lu H.(2023). Shadow Hunter: Low-Illumination Object-Detection Algorithm, *Applied Sciences*, 13(16), 9261, 2023.

[11] Huang X.(2023). Moving object detection in low-luminance images, *The Visual Computer*, 39(1), 183-195, 2023.

[12] Gilroy S.; Jones E.; Glavin M.(2019). Overcoming occlusion in the automotive environment—A review, *IEEE Transactions on Intelligent Transportation Systems*, 22(1), 23-35, 2019.

[13] He Y.; Zhu C.; Yin X C.(2021). Occluded pedestrian detection via distribution-based mutual-supervised feature learning, *IEEE Transactions on Intelligent Transportation Systems*, 23(8), 10514-10529, 2021.

[14] Li Y.; Fan Q.; Huang H.(2023). A modified YOLOv8 detection network for UAV aerial image recognition, *Drones*, 7(5), 304, 2023.

[15] Hu J.; Li B.; Zhu H. (2024). Improved lightweight UAV target detection algorithm for YOLOv8, *Computer Engineering and Applications*, 60(08), 182-191, 2024.

[16] Liu W.; Liu D.; Wang L. (2023). A review of research on deformable convolutional networks, *Computer Science and Exploration*, 17(7), 1549-1564, 2023.

[17] [Online]. Available: www.github.com/ultralytics/, Accesed on 10 January 2023.

[18] Wang C Y.; Bochkovskiy A.; Liao H Y M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7464-7475, 2023.

[19] Dai J.; Qi H.; Xiong Y. (2017). Deformable convolutional networks, *Proceedings of the IEEE international conference on computer vision*, 764–773, 2017.

[20] Zhu X.; Hu H.; Lin S. (2019). Deformable convnets v2: More deformable, better results, *Proceedings of the IEEE/CVF conference on computer vision and patternrecognition*, 9308–9316, 2019.

[21] Du D W.; Zhu P F.; Wen L Y. (2019). VisDrone-DET2019: The Vision Meets Drone Object Detection in Image Challenge Results, *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop*, 213-226, 2019.

[22] [Online]. Available: www.github.com/ultralytics/yolov5/, Accesed on 5 September 2022.

[23] Li C.; Li L.; Jiang H. (2022). YOLOv6: A single-stage object detection framework for industrial applications, *arxiv preprint*, arxiv:2209.02976, 2022.

[24] Liu W.; Anguelov D.; Erhan D. (2016). SSD: Single shot multibox detector, *Proceedings of the European Conference on Computer Vision*, 21-37, 2016.

[25] Wang C Y.; Bochkovskiy A.; Liao H-Y M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7464-7475, 2023.

[26] Yu X.; Gong Y.; Jiang N. (2020). Scale match for tiny person detection, *Proceedings of The IEEE/CVF Winter Conference on Applications of Computer Vision*, 1257-1265, 2020.

[27] Cai Y.; Bian H.; Lin J. (2023). Retinexformer: One-stage retinex-based transformer for low-light image enhancement, *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12504-12513, 2023.

[28] Chen W.; Huang H.; Peng S. (2023). YOLO-face: a real-time face detector, *The Visual Computer*, 37, 805-813, 2023.

Cite this paper as:

Yunfan, Bu. (2025). Enhanced YOLOv8-based Lightweight Small Personnel Detection Algorithm for UAV Flood Emergency Rescue, *International Journal of Computers Communications & Control*, 20(5), 6869, 2025.
https://doi.org/10.15837/ijccc.2025.5.6869