scientific reports

Check for updates

OPEN

YOLOv5_mamba: unmanned aerial vehicle object detection based on bidirectional dense feedback network and adaptive gate feature fusion

Shixiao Wu[1,5], Xingyuan Lu[2,5]✉ & Chengcheng Guo[3,4,5]✉

Addressing the problem that the object size in Unmanned Aerial Vehicles (UAVs) aerial images is too small and contains limited feature information, leading to existing detection algorithms having less than ideal performance in small object detection, we propose a UAV aerial object detection system named YOLv5_mamba based on bidirectional dense feedback network and adaptive gate feature fusion. This paper improves the You Only Look Once Version 5 (YOLOv5) algorithm by firstly introducing the Faster Implementation of CSP Bottleneck with 2 convolutions (C2f) module from YOLOv8 into the backbone network to enhance the feature extraction capability of the backbone network. Furthermore, the mamba module and C2f module are introduced to construct a bidirectional dense feedback network to enhance the transfer of contextual information in the neck part. Thirdly, an adaptive gate feature fusion network is proposed to improve the head part of YOLOv5 and enhance its final detection capability. Experimental results on the public UAV aerial dataset VisDrone2019 demonstrate that the proposed algorithm improves the detection accuracy by 9.3% compared to the original YOLOv5 baseline network, showing better detection performance for small objects. For the UCAS_AOD dataset, the proposed algorithm outperforms YOLOv5-s by 9%. In the case of the DIOR dataset, the proposed algorithm exceeds YOLOv5-s by 12%.

Keywords UAV, Object detection, Mamba, YOLOv5, Adaptive gate feature fusion

UAVs and remote sensing technologies have been continuously evolving. By harnessing their combined visual perception capabilities, they can perform a wider range of tasks. Through intelligent analysis and processing of aerial images, object features can be quickly and effectively captured, thereby enhancing the UAVs' understanding of the scene. Object detection technology can automatically identify and locate objects in images, improving the UAVs' perceptual capabilities under conditions of limited human interaction, and providing essential technical support for autonomous detection and flight. Object detection in aerial images has garnered widespread research attention in civilian and military fields, such as UAV reconnaissance, traffic monitoring, precision agriculture, wildlife tracking, and personnel rescue[1–4].

Currently, object detection is primarily focused on natural scene images and is relatively mature in applications like face recognition and pedestrian detection. However, aerial images differ from natural scene images in that the objects are typically small, densely distributed, exhibit large scale variations, may be occluded, and can be oriented in any direction. When existing algorithms are directly applied to aerial remote sensing images, the model performance tends to be poor.

Traditional object detection algorithms typically involve several key steps: image preprocessing, candidate region selection, feature extraction, dimensionality reduction, and classification. The goal of feature extraction is to leverage the expressive and deformation-resistant properties of features, with the accuracy of feature classification positively linked to the algorithm's quality. Key feature extraction methods include the Scale-

[1]School of Information Engineering, Wuhan Business University, Wuhan 430056, China. [2]School of Computer Science and Engineering, Tianjin University of Technology, Tianjin 300384, China. [3]School of Electronic Information, Wuhan University, Wuhan 430072, China. [4]School of Information Engineering, Wuhan College, Wuhan 430212, China. [5]These authors contributed equally: Shixiao Wu, Xingyuan Lu and Chengcheng Guo. ✉email: lxy1466792674@stud.tjut.edu.cn; netccg@whu.edu.cn

Invariant Feature Transform (SIFT), the Viola-Jones (VJ) algorithm, the Histogram of Oriented Gradients (HOG), and Deformable Part Models (DPM). Common classifiers in this context are support vector machines, AdaBoost, and others[6–10]. However, the SIFT algorithm struggles to extract edge-smooth object features and exhibits high computational complexity. While the VJ algorithm is straightforward to implement, it still has significant computational demands. The HOG algorithm does not effectively handle occlusion, and the DPM algorithm necessitates manual feature design, resulting in a heavy workload, lack of rotational invariance, and poor stability.

Many researchers are leveraging deep learning to address detection challenges in UAV aerial images. Inderpreet Singh et al. enhanced the accuracy of small object detection in UAV aerial images by introducing a new feature fusion layer in the feature pyramid section of YOLOv5 and employing composite scaling to increase the input size[11]. Hang Zhang et al. proposed an improved triple loss function to effectively assess the similarity among multiple objects detected by UAVs, which helped reduce false positives and negatives[12]. They also presented a consistent recognition algorithm for objects in multi-angle scenes using distributed computing and established a multi-UAV multi-object detection database to mitigate training and validation issues in complex scenarios. Minglei Du et al. developed an error model for the positioning algorithm, provided a method for calculating the shared Position Dilution of Precision (PDOP), and validated the accuracy of the error model through simulations using Monte Carlo statistical methods[13–15]. Jinguang Chen et al. designed a module that integrates oscillation transformation and convolution to better capture the global contextual information of small objects in images[16]. Tang, Shiyi et al. proposed the HIC-YOLOv5 model, which was validated using only one public dataset, Visdrone[17]. In contrast, Benjumea, Aduen et al. introduced the YOLO-Z model, which utilizes the autonomous vehicles dataset, differing from our approach[18]. Murat Bakirci integrates YOLOv8 with intelligent transportation systems, noting that YOLOv8x outperforms YOLOv8n in terms of F1 score and mean Average Precision (mAP)[19,20].

This article focuses on enhancing the YOLOv5 algorithm. First, the C2f module from YOLOv8 is integrated into the backbone network to improve feature extraction capabilities. Additionally, the mamba module is introduced alongside the C2f module to create a bidirectional dense feedback network, which enhances the transmission of contextual information in the neck section. Third, an adaptive gate feature fusion network is proposed to refine the head section of YOLOv5, thereby boosting its final detection capabilities. Experimental results on the VisDrone2019 public UAV aerial dataset demonstrate that the proposed algorithm achieves a 9.3% improvement in detection accuracy compared to the original YOLOv5 baseline network, particularly showing enhanced performance in detecting small targets.

Methods

The YOLOv5 algorithm is structured into four main components: the input section, backbone network, neck network, and detection head. The backbone consists of a CSP Darknet53, which is built on the C3 module, integrating the CSP architecture with the Darknet53 network structure. The CSP architecture splits the base layer features into a gradient flow main branch and a sub-branch, merging them within a cross-stage hierarchical framework. This design achieves richer gradient combinations while minimizing computational load. Darknet53 is a 53-layer convolutional neural network utilized for feature extraction from input images, capturing edges, textures, and other low-level attributes.

In YOLOv5, the SPPF module is employed in the primary feature extraction network, performing max-pooling with varying kernel sizes to expand the network's receptive field, thus enhancing its feature extraction capabilities. During feature extraction, YOLOv5 extracts three feature layers for object detection, positioned at different levels within the CSP-Darknet backbone: middle, mid-bottom, and bottom layers. When the input size is (640, 640, 3), the shapes of these three feature layers are as follows: feat1 = (80, 80, 256), feat2 = (40, 40, 512), and feat3 = (20, 20, 1024). Both YOLOv5 and YOLOv4 employ an FPN + PAN structure in the neck network. In contrast to the regular convolution operations used in the neck structure of YOLOv4, YOLOv5's neck introduces the CSP2 structure based on CSPNet design to enhance the network's feature fusion capability.

This paper introduces the YOLOv5_mamba algorithm, which enhances the backbone and neck components of YOLOv5 through the implementation of bidirectional dense feedback networks and gated feature fusion techniques.

Backbone network

YOLOv5_mamba incorporates the C2f module from YOLOv8, substituting the C3 module in YOLOv5, as illustrated in Fig. 1. By integrating the C2f module into the backbone, YOLOv5_mamba increases the number of skip connections and adds extra Split operations, thereby enhancing the network's feature extraction capabilities. The C2f module features more branches and connections, which promote better information propagation and facilitate effective feature fusion. This modification aims to improve the overall performance of the network in object detection tasks.

YOLOv5_mamba's neck network

The neck of YOLOv5_mamba features 10 concatenation (Concat) operations, in contrast to YOLOv5s' 4 Concat operations. This enhancement results in 2.5 times greater information fusion capability and improved information transmission. Additionally, the neck incorporates the mamba module, which integrates selective state space models (SSMs) into a streamlined end-to-end neural network architecture. This module provides rapid inference capabilities, achieving throughput that is 5 times higher than that of Transformers, while also offering linear scalability.

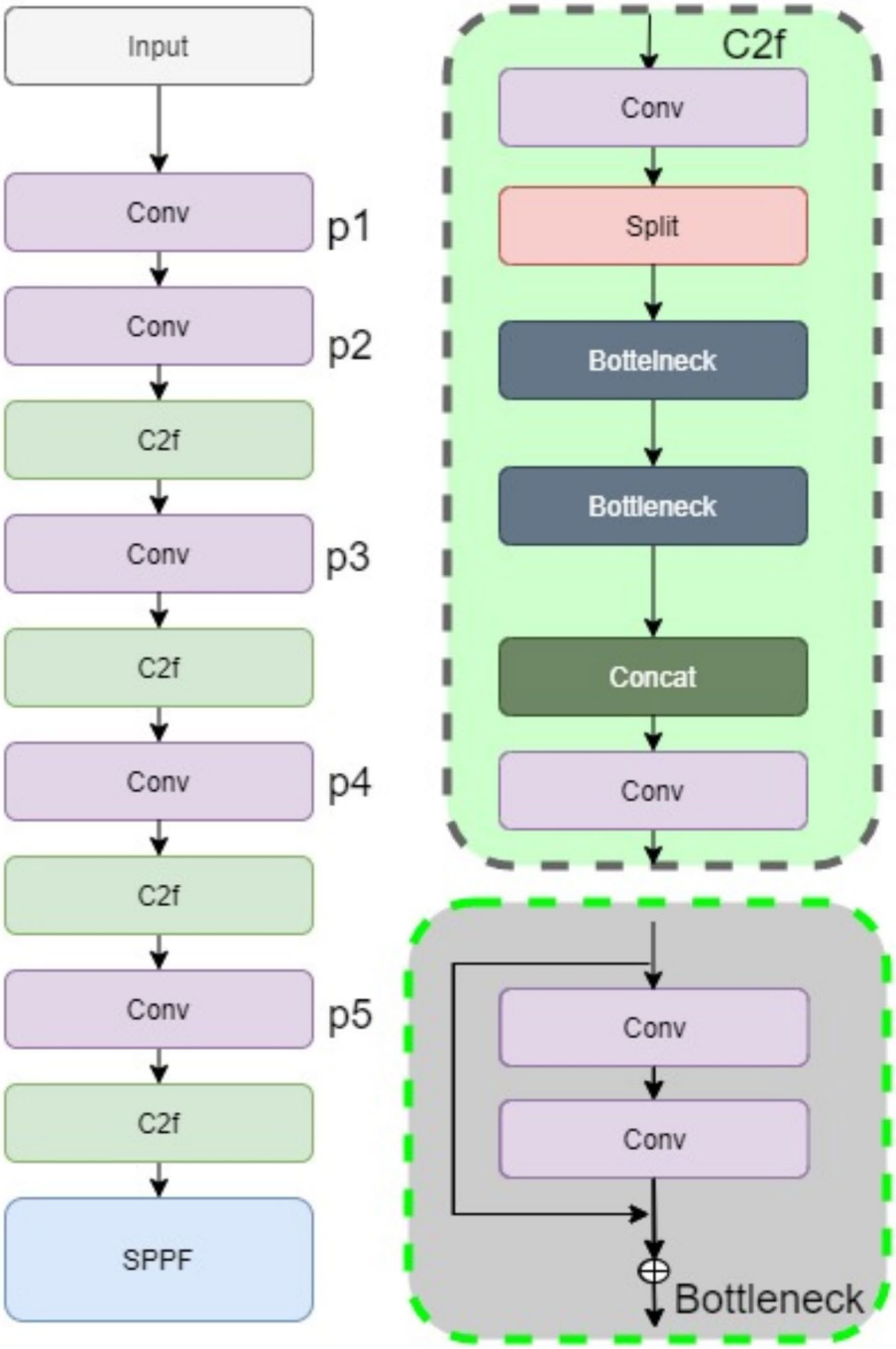

Fig. 1. YOLOv5_mamba's backbone network.

Mamba module

In recent years, Structured State Space Sequence Models (SSMs) have emerged as a promising class of sequence modeling architectures[21,22]. These models can be interpreted as a fusion of Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), drawing inspiration from classical state space models. Albert Gu et al. integrated prior SSM architectures with the MLP blocks from transformers to develop a new architectural block. This innovation simplifies the earlier deep sequence model structures, leading to the creation of a streamlined and homogeneous architecture design known as mamba, which incorporates selective state space

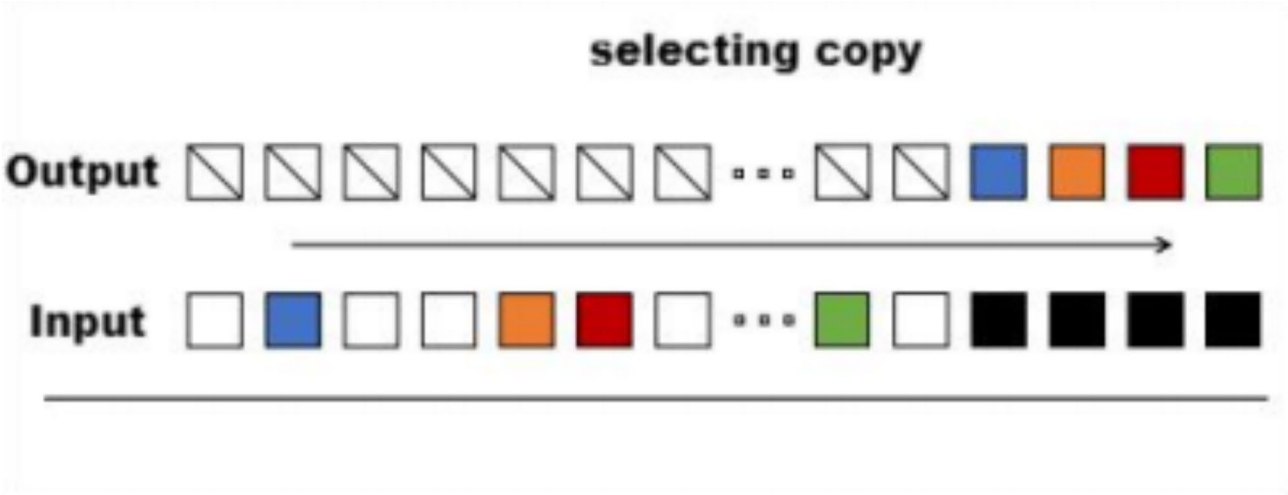

Fig. 2. Selecting copy (SSM selectively chooses useful information, such as the colored areas, while filtering out irrelevant information).

Algorithm 1 SSM + Selection (S6)
Input: *x: (B, L, D)*
Output: *y: (B, L, D)*
1: *A: (D, N)* ← *Parameter*
2: *B: (B, L, N)* ← *SB(x)*
3: *C: (B, L, N)* ← *SC(x)*
4: *Δ: (B, L, D)* ← τ_{Δ}*(Parameter* + S_{Δ}*(x))*
5: *A, B: (B, L, D, N)* ← *discretize(Δ, A, B)*
6: *y* ← *S SM (* $\bar{A}$, $\bar{B}$, *C)(X)*
7: return *y*

Table 1. The selective state space model construction algorithm of mamba.

mechanisms. This design aims to enhance the efficiency and effectiveness of sequence modeling in various applications.

From a convolutional perspective, global convolutions are capable of managing ordinary replication tasks since they only necessitate time awareness. However, they struggle with selective replication tasks due to a lack of content awareness. In selective replication, the distances between inputs and outputs can vary, making it challenging to model these relationships with static convolutional kernels, as illustrated in Fig. 2. This variability in distances requires a more dynamic approach to effectively capture the nuances of selective replication.

In contrast to SSMs, which compress all historical information (while Transformers do not perform compression), the mamba architecture employs a simplified selective mechanism. This mechanism enables the model to focus on or exclude specific inputs by "parameterizing the inputs of the SSM". By doing so, mamba can effectively filter out irrelevant information while preserving long-term memory of relevant data. The mamba algorithm is built upon the principles of S6 to construct selective state space models, as illustrated in Table 1. This approach enhances the model's ability to concentrate on significant information, improving task performance and efficiency.

In the mamba architecture, the *B* matrix and *C* matrix at each position are distinct, meaning that each input token has its corresponding *B* and *C* matrices. This design addresses the issue of content perception effectively. In Table 1, *x: (B, L,D)* represents operations on input sequences x with a batch size *B*, length *L*, and *D* channels. The architecture allows developers to define *B* matrices, *C* matrices, and *Δ* parameters as functions of the input data, enabling these matrices to be adjusted through learning. Here, *N* denotes the dimensionality of the state space. This adaptive modification allows the model to tailor its responses based on the actual input information, enhancing its ability to process and understand varying contexts effectively.

Dual-dense feedback network (DDFN)

The dual-dense feedback network consists of modules such as Concat, Upsample and C2f, as shown in Fig. 3.

YOLOv5_mamba's neck section features 10 Concat operations, providing 2.5 times the information fusion capability compared to YOLOv5s'4 Concat operations, which results in stronger information transmission.

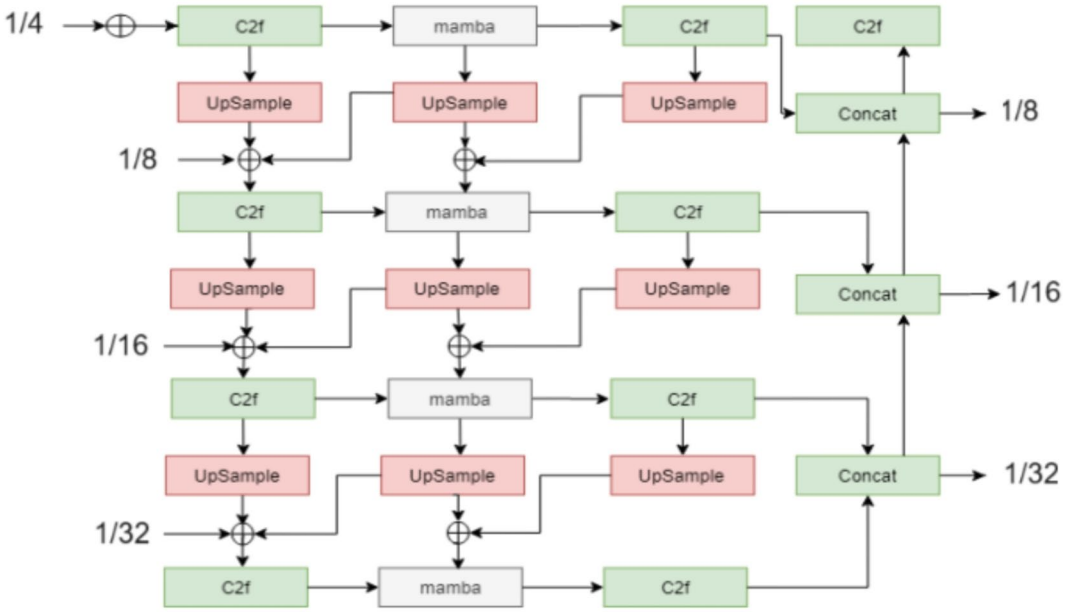

Fig. 3. The Dual-Dense Feedback Network. (In the backbone, there are several convolution operations, each reducing the output feature map size to 1/4, 1/8, 1/16, and 1/32 of the input feature map size. Before detection, the feature maps undergo a concatenation operation. After each concatenation, the feature maps are sent to the detection head, resulting in a total of three outputs, with sizes corresponding to 1/8, 1/16, and 1/32 of the input image size.)

This section incorporates the mamba module, integrating SSMs into a simplified end-to-end neural network architecture, enabling rapid inference capabilities.

In addition to the Mamba module, YOLOv5_mamba extensively utilizes the C2f structure. This structure not only lightens the model but also significantly enhances performance by improving gradient flow and adjusting the number of channels based on the model's scale. The C2f module introduces more inter-layer connections compared to the C3 module, along with additional splitting operations, while eliminating convolution steps within the branches. This design enhances gradient propagation through auxiliary paths and improves feature representation while reducing computational complexity.

Detection head

In deep learning, "gating" is a commonly used technique, particularly in recurrent neural networks (RNNs) and their variants such as long short-term memory networks (LSTMs) and gated recurrent units (GRUs). Gating mechanisms enhance network performance by controlling the flow of information, allowing the network to learn when to update or forget information, thereby addressing the issue of long-term dependencies in sequential data. Gating mechanisms also improve model efficiency and performance by reducing unnecessary computations and reinforcing the transmission of crucial information.

Gating mechanisms typically involve one or more gating units, which dynamically adjust the flow of information based on input data and the current state of the network. Each gating unit outputs a value between 0 and 1, which can be seen as a switch or regulator controlling the flow of information. Following the bidirectional dense feedback network and preceding the detection heads, we design a gate-controlled feature fusion module to significantly enhance the performance and efficiency of neural network models in object detection tasks by finely controlling the flow of information. This module is illustrated in Fig. 4. YOLOv5_mamba includes three gate-controlled feature fusion modules.

In Fig. 4, each input undergoes a linear transformation (Linear) for gate-controlled information processing. The primary scale input (input1) also utilizes a depthwise separable convolution (dwconv), which helps in learning richer information. Additional inputs (input 2 and input 3) are multiplied with the primary input, and the results are then summed together. This aggregated information undergoes another linear transformation.

Following this, the output of the addition and linear transformation is residual connected to the initial primary input. Finally, a further linear transformation is applied to obtain the fused information. This process facilitates effective information integration while maintaining the essential features of the primary input.

YOLOv5_mamba

Combining the improved backbone network with two enhancements in the neck section—the dual-dense feedback network and the gate-controlled feature fusion module—we propose the YOLOv5_mamba algorithm. The overall structure of this algorithm is illustrated in Fig. 5.

In the YOLOv5_mamba algorithm, multiple basic block units are employed, including the c2f module derived from YOLOv8. The improvements made to the backbone network of YOLOv5_mamba are primarily

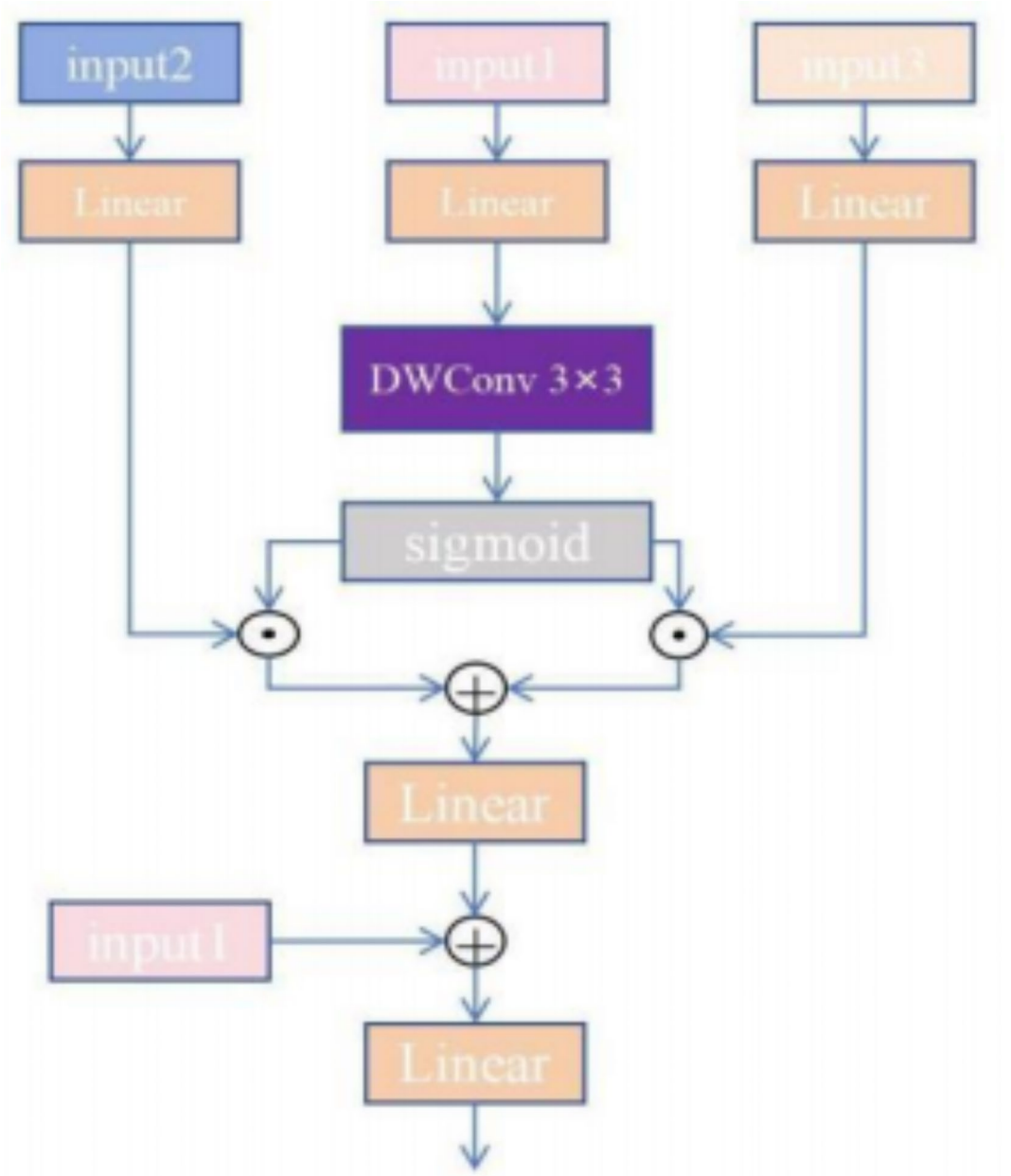

Fig. 4. YOLOv5_mamba's gate-controlled feature fusion module.

built upon the c2f module, which serves as a key component due to its enhanced information processing capabilities. Enhancements to the neck section feature a dual-dense feedback network alongside three gate-controlled feature fusion modules.

In the following sections, we will present results from ablation experiments to justify the selection of the c2f module as the foundational block. These experiments will demonstrate that the network exhibits superior performance when utilizing the c2f module compared to other blocks, such as c3, bottleneck, and single conv3×3 layers. Within the dual-dense feedback network, feature maps from three different scales are directed towards the detection heads. Prior to entering the detection heads, we incorporate gate-controlled feature fusion modules to finely regulate the flow of information, ensuring that the most relevant features are emphasized for improved detection accuracy.

Results

The VisDrone2019 dataset

The VisDrone2019 dataset presents significant challenges due to scale variations, occlusions and class imbalances. This benchmark dataset consists of 400 video segments, totaling 265,228 frames, alongside 10,209 static images captured by cameras mounted on various drones. It encompasses a broad range of factors, including diverse locations across 14 different cities in China, covering thousands of kilometers.

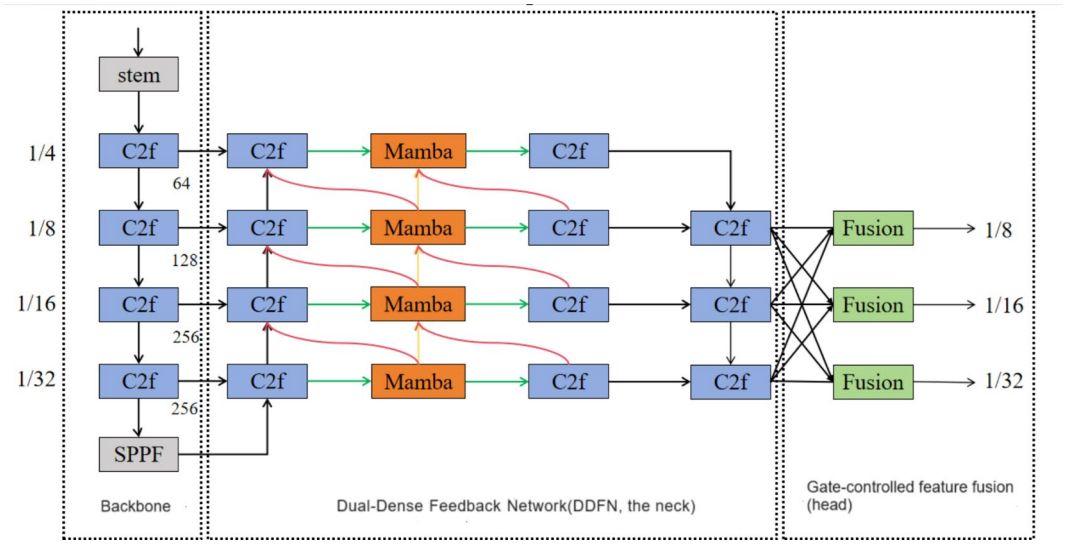

Fig. 5. YOLOv5_mamba (the proposed network).

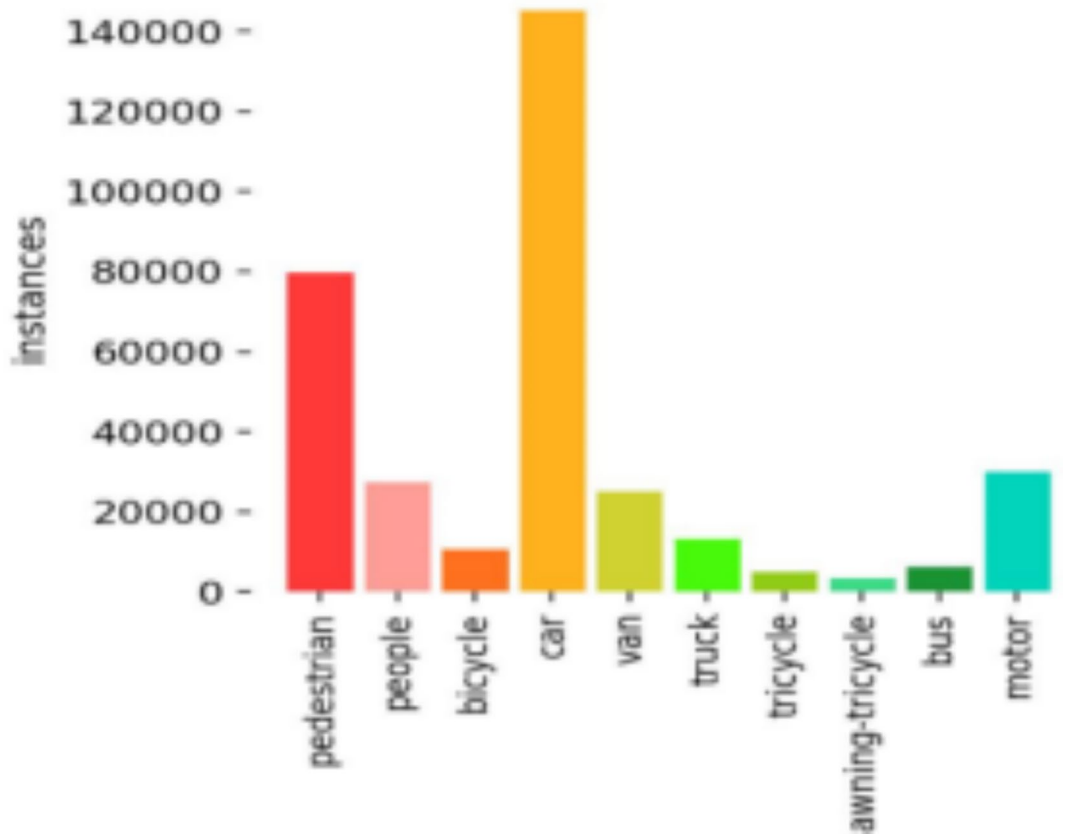

Fig. 6. The VisDrone dataset.

The dataset features a variety of environments, such as urban and rural areas, and includes multiple object types, including pedestrians, vehicles, bicycles, and more. It also captures scenes with varying densities, from sparse to crowded. Additionally, the dataset has been using various drone platforms (i.e., different drone models) under diverse scene, weather, and lighting conditions. The distribution of each class is illustrated in Fig. 6, highlighting the complexity and richness of the dataset for training and evaluating object detection models.

Evaluation metrics

This algorithm primarily utilizes several evaluation metrics to assess model performance, including model parameter size, computational complexity, F1 score, and mean Average Precision (mAP). These metrics provide a comprehensive understanding of the model's efficiency and effectiveness in object detection tasks. Model parameter size and computational complexity help evaluate the resource requirements, while the F1 score and mAP measure the accuracy and robustness of the model's predictions.

F1 score

The F1 score is the harmonic mean of precision and recall, serving as a metric that comprehensively evaluates both aspects of model performance. By balancing precision and recall, the F1 score is particularly suitable for situations involving imbalanced classes, where one class may significantly outweigh others. This metric helps ensure that both false positives and false negatives are accounted for, providing a more nuanced understanding of a model's effectiveness in detecting objects across various categories.

$$\mathrm{Pr}\,ecison = \frac{TP}{TP + FP} \tag{1}$$

$$\mathrm{Re}\,call = \frac{TP}{TP + FP} \tag{2}$$

$$\mathrm{F1} = 2 \cdot \frac{\mathrm{Pr}\,ecision \cdot \mathrm{Re}\,call}{\mathrm{Pr}\,ecision + \mathrm{Re}\,call} \tag{3}$$

Precision refers to the proportion of correctly detected positive samples (true positives, TP) among all samples identified as positive, which includes true positives (TP) and false positives (FP). This metric indicates how many of the detected positive samples are actually correct, thus providing insight into the accuracy of the positive detections made by the model.

Recall, on the other hand, indicates the proportion of correctly detected positive samples (TP) among all actual positive samples, which includes TP and false negatives (FN). Recall measures the model's ability to identify all relevant positive cases, highlighting its effectiveness in capturing true instances of the target class.

mAP

Average Precision (AP) is a key performance metric in evaluating the quality of object detection models. It represents the area under the Precision-Recall (P-R) curve, providing a comprehensive measure of a model's precision and recall across various thresholds.

One of the most commonly used methods for estimating this area is the 11-point interpolated average precision method. In this approach, precision values are averaged at 11 specific points of recall: 0, 0.1, 0.2, …, up to 1.0. For each of these recall levels, the highest precision observed is taken, ensuring that the precision values are non-decreasing as recall increases. This interpolation helps smooth out the curve and provides a more robust estimate of the model's performance.

Once the AP is calculated for each class, the mAP can be derived by averaging the AP values across all classes. This metric offers a single score that reflects the overall performance of the model across multiple categories, making it particularly useful for comparing different models or algorithms in tasks such as object detection.

$$\mathrm{mAP} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{AP_i} \tag{4}$$

Ablation experiment

Input resolution selection (input)

This paper conducted ablation experiments on the resolution of input images, and the results are shown in Table 2.

According to Table 2, it can be observed that the model performs better when the resolution of the input images is 1,280* 1,280. Therefore, the algorithm selects the input image resolution to be 1,280* 1,280.

Basic model selection (neck)

The algorithm utilizes the C2f module as the basic building block. The performance of the model was compared when using other block modules such as c3, bottleneck, and single conv3 × 3, as shown in Table 3.

Resolution selection	map0.5	map0.5:0.95	F1
640*640	0.312	0.170	0.370
960*960	0.315	0.171	0.373
1,280*1,280	0.482	0.291	0.521

Table 2. Input resolution selection.

Basic module	map0.5	map0.5:0.95	F1	Parameter	FPS
C2f	0.549	0.343	0.582	9.4	53
C3	0.537	0.331	0.554	9	50
Bottleneck	0.518	0.318	0.535	9.5	62
Single conv3 × 3	0.505	0.305	0.519	16.1	74

Table 3. Basic module selection (neck).

FPN	map0.5	map0.5:0.95	F1	Parameter	FPS
PANet	0.519	0.323	0.556	7.5	115
FPN	0.444	0.241	0.502	6.4	99
BiFPN	0.486	0.298	0.515	7.5	82
DDFN	0.549	0.343	0.582	9.4	53

Table 4. Different feature pyramid network (neck).

Method	map0.5	map0.5:0.95	F1	Parameter	FPS
Bottleneck	0.542	0.337	0.560	7.1	56
Mamba	0.549	0.343	0.582	9.4	53
Transformer	0.531	0.323	0.557	7.8	42

Table 5. The neck network feature propagation methods comparison.

From Table 3, it can be seen that in YOLOv5 mamba, selecting C2f as the basic module for both backbone and neck construction yields the best results. Although there is some decrease in speed, it still achieves real-time performance.

Different feature pyramid network (neck)

We compared the impact of various feature pyramid networks (FPNs) such as Path Aggregation Network (PANet), FPN, Bidirectional Feature Pyramid Network (BiFPN), and the proposed dual-dense feedback network on network performance, as shown in Table 4.

From the results presented in Table 4, it is evident that the proposed dual-dense feedback network achieves the highest mAP and F1 scores compared to the other networks. The networks referenced in Tables 3 and 4 highlight improvements made to the backbone and neck sections of YOLOv5, utilizing C2f and mamba enhancements, respectively. These modifications contribute to the overall performance gains observed in the proposed architecture.

Feature propagation method

We compared the information propagation capabilities of bottleneck, mamba, and transformer modules in the.

proposed network, further explaining why the mamba module was chosen for improving the neck network (Table 5).

From Table 5, it is clear that the feature information propagation method based on mamba achieves greater efficiency in information transfer, leading to improved the mAP and F1 scores. The subpar performance of the transformer model may be attributed to its stringent data requirements; specifically, the dataset used in this study may not be sufficiently large, causing the transformer to suffer from underfitting.

While mamba does incur higher computational costs compared to transformers, it compensates for this with faster processing speeds, making it a more effective choice in scenarios where rapid inference is critical. This balance between efficiency and performance highlights the advantages of the mamba approach in feature propagation tasks.

Gate-controlled feature fusion method (GFFM)

Following the dual-dense feedback network, we introduced three sets of gate-controlled feature fusion modules to finely control the flow of information. Comparative experiments based on gate-controlled feature fusion are shown in Table 6.

Module stacking ablation experiment

The proposed algorithm integrates the C2f module, enhances the neck module, and adopts the detection head from YOLOX. Additionally, gate-controlled feature fusion is implemented prior to the detection head to optimize feature integration.

GFFM	map0.5	map0.5:0.95	F1	Parameter	FPS
w/o	0.561	0.355	0.591	12.2	47
w/	0.575	0.364	0.601	12.9	45

Table 6. Gate-controlled feature fusion method.

Method	map0.5	map0.5:0.95	F1	Parameter	FPS
Baseline	0.482	0.291	0.521	7.0	93
c2f	0.489	0.303	0.536	7.5	115
DDFN	0.549	0.343	0.582	9.4	53
DDFN + anchor free	0.561	0.355	0.591	12.2	47
DDFN + anchor free + Gated Fusion	0.575	0.364	0.601	12.9	45

Table 7. Comparative analysis of ablation experiments for each module.

Methods	mAP50	mAP50:95	Params	GFLOPs	FPS
Basic (only black connections)	52.2	32.6	7.3	30.8	92.6
Basic + green(mamba block and c2f)	57.2	36.2	11.5	44.9	49.3
Basic + green + yellow	57.2	36.3	11.7	46.6	47.4
Basic + green + red	57.3	36.7	11.8	47.3	46.5
ours	57.5	36.9	12.5	50.5	45.0

Table 8. The concat effects.

The results of the ablation experiments, which evaluate the contribution of each module's stacking, are presented in Table 7. These results demonstrate the effectiveness of the individual components and their combined impact on overall model performance, highlighting the importance of each enhancement in the proposed architecture.

It's worth mentioning that while the addition of the C2f module in YOLOv5 led to only a slight improvement in accuracy, it resulted in a noticeable increase in processing speed. However, by replacing C2f with the dual-dense feedback network in the neck module, adding gate-controlled feature fusion modules, and incorporating the anchor-free head from YOLOX, the model's overall performance experienced a significant enhancement. These modifications not only improved accuracy but also maintained efficient processing, showcasing the effectiveness of the proposed architecture.

Concat effects

We examine whether the increased number of concatenation operations in your YOLOv5_mamba model impacts detection speed. Given that UAV-based applications frequently demand real-time performance, this is a crucial aspect to consider.

In Fig. 5, different colors represent different connections. After all the connections are added, the model's speed decreases, but the mAP improves (Table 8).

Comparison with SOTA methods

This paper conducts a comparative analysis with several other methods, including YOLOv5s, YOLOv5m, YOLOv6s, YOLOv6m, YOLOv7tiny, YOLOv7, YOLOv8s, YOLOXs and YOLOXm. The results of this comparison are presented in Table 9.

Feature visualization

The paper visually compares different outputs of the mamba module, the neck part of YOLOv5, and the neck part of YOLOv5_mamba, as shown in Figs. 7 and 8.

From Figs. 7 and 8, it is evident that the mamba module significantly enhances the model's ability to focus on or exclude specific inputs, thereby allowing for the extraction of more detailed information in the regions of interest.

Detection results comparison

The comparison of detection results for YOLOv7, YOLOv8-m and YOLOv5_mamba is illustrated in Fig. 9.

From Fig. 9, it can be observed that the proposed algorithm, YOLOv5_mamba, exhibits stronger capabilities in detecting small objects.

Model	mAP (%)	Pedestrian	People	Bicycle	Car	Van	Truck	Tricycle	Awning-tricycle	Bus	Motor
YOLOv5s[11]	30.7	29.8	20.3	13.0	63.5	39.3	28.8	21.3	14.0	47.0	29.6
YOLOv6s[23]	30.3	27.5	19.4	12.4	62.4	38.1	30.6	22.0	14.2	47.8	28.8
YOLOv7-tiny[24]	29.9	28.2	19.3	12.1	62.8	38.0	30.0	21.1	13.5	45.9	28.1
YOLOv8s[20]	31.0	30.0	20.4	13.8	63.6	38.7	29.5	23.2	13.8	47.3	29.8
YOLOXs[25]	31.3	29.9	20.5	13.6	63.9	40.3	30.3	22.9	15.0	46.7	30.0
YOLOv5m[11]	34.4	32.6	22.8	16.5	66.0	42.5	35.1	26.2	15.6	53.8	32.6
YOLOv6m[23]	33.7	32.8	23.1	17.0	64.0	41.4	35.5	26.1	17.0	49.8	30.6
YOLOv7[24]	36.3	36.9	25.4	19.7	66.8	44.4	37.6	26.2	16.7	55.5	33.7
YOLOv8m[20]	34.5	34.6	23.9	17.8	65.1	42.0	36.5	25.4	16.1	51.4	32.2
YOLOXm[25]	34.2	34.7	24.3	17.9	65.5	42.3	33.5	25.0	17.1	49.3	32.0
Ours	36.9	37.0	25.1	19.4	68.2	44.8	37.0	28.0	16.9	56.9	35.7

Table 9. Comparison of YOLOv5_mamba with other algorithms in the YOLO series (Visdrone 2019).

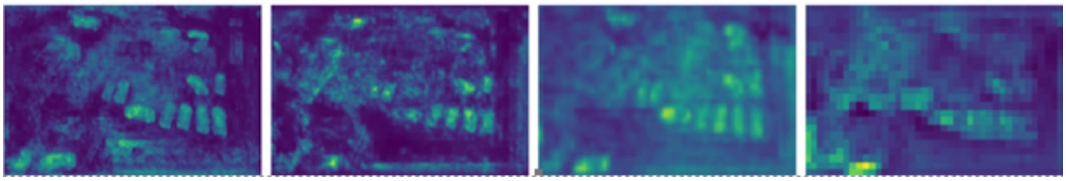

Fig. 7. Feature visualization in DDFN (1/4,1/8,1/16,1/32).

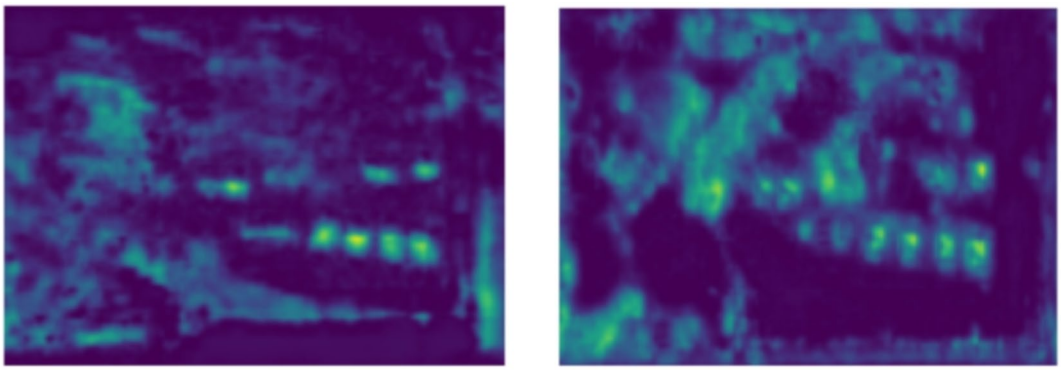

Fig. 8. Visualization of the neck part in YOLOv5 and YOLOv5_mamba (1/8).

Expansion experiment

UCAS_AOD Dataset is a well-known dataset in the field of aerial object detection, specifically designed for the detection of objects in aerial images. Developed by the University of Chinese Academy of Sciences (UCAS), this dataset is often utilized for training and evaluating object detection algorithms, particularly in applications related to remote sensing and UAV imagery. The UCAS-AOD dataset is a remote sensing image dataset that includes two target classes: cars and airplanes, as well as background negative samples. This dataset contains 1,800 images, with 1,200 images in the training set and 600 images in the testing set.

In addition to the VisDrone2019 dataset, we also conducted tests on the UCAS-AOD dataset. The comparative testing results are shown in Table 10; Fig. 10:

The research team, led by Han Junwei at Xi'an University of Technology, introduced the large-scale benchmark dataset known as "DIOR" for object detection in optical remote sensing images. This dataset includes a total of 23,463 images and contains 190,288 object instances. The mAP comparisons for different classes within the DIOR dataset are detailed in Tables 11, 12 and 13.

Figure 11 gives the comparative analysis of detection results among YOLOv6-m, YOLOv8-m and YOLOv5_mamba (DIOR).

Discussion

The performance of YOLOv5_mamba was collectively enhanced through several key components: the backbone network based on C2f, the bidirectional dense feedback network based on mamba, and the head utilizing

Fig. 9. Comparative analysis of detection results among YOLOv7, YOLOv8-m, and YOLOv5_mamba.

Method	mAP (%)	Plane	Car
YOLOv5s[11]	61.1	73.4	48.7
YOLOv6s[23]	63.8	72.3	55.2
YOLOv7-tiny[24]	61.3	69.4	53.2
YOLOv8s[20]	61.8	70.6	53.0
YOLOXs[25]	62.5	70.1	54.8
YOLOv5m[11]	64.5	72.7	56.4
YOLOv6m[23]	64.7	72.4	57.1
YOLOv7[24]	64.6	70.9	58.3
YOLOv8m[20]	65.2	71.9	58.5
YOLOXm[25]	63.9	72.2	55.5
Ours	70.1	74.9	65.4

Table 10. The UCAS_AOD dataset's sota experiments.

anchor-free and gate-controlled feature fusion. The algorithm evaluated the effectiveness of various modules, such as c3, bottleneck, and single conv3 × 3, as shown in Table 3. From the results, it is clear that using C2f as the foundational module for both the backbone and neck networks in YOLOv5_mamba yields the best performance outcomes.

Although there is a slight decrease in speed, the model still achieves real-time performance. In the neck component, we compared different feature pyramid networks, including PANet, FPN and BiFPN, alongside the proposed bidirectional dense feedback network, ultimately selecting the latter for its superior performance.

We also investigated the capabilities of the bottleneck, mamba, and transformer modules in information propagation within the network, providing justification for the choice of the mamba module to enhance the neck network. Furthermore, three sets of gate-controlled feature fusion modules were incorporated to finely regulate the flow of information, optimizing the overall architecture for improved detection capabilities.

The backbone network of YOLOv5_mamba continues to utilize the YOLOv5 backbone; however, there is potential for exploring more advanced state-of-the-art (SOTA) networks in future iterations. While comparisons were made among various feature pyramid networks, including PANet, FPN, and BiFPN, the study did not extend to additional feature pyramid networks or investigate the integration of attention mechanisms such as SENet, simAM, CBAM, or Coord Attention.

In terms of enhancements to the head, the model incorporates both anchor-free detection and gate-controlled feature fusion, which contribute to improved performance. Future research could benefit from a broader exploration of these advanced architectures and attention mechanisms to further refine the model's capabilities.

Since our improvements are based on YOLOv5, we only compared YOLOv8-m and YOLOv8-s, without evaluating YOLOv8n, YOLOv8l, and YOLOv8x. Future work could focus on enhancing the other YOLOv8 models. Additionally, the backbone of our model is quite simple, and exploring more complex architectures such as dual-branch models could be beneficial. The feature flow in the neck section could also be expanded to consider additional directions. This experiment was limited to three datasets; hence, incorporating more remote sensing datasets could provide further insights.

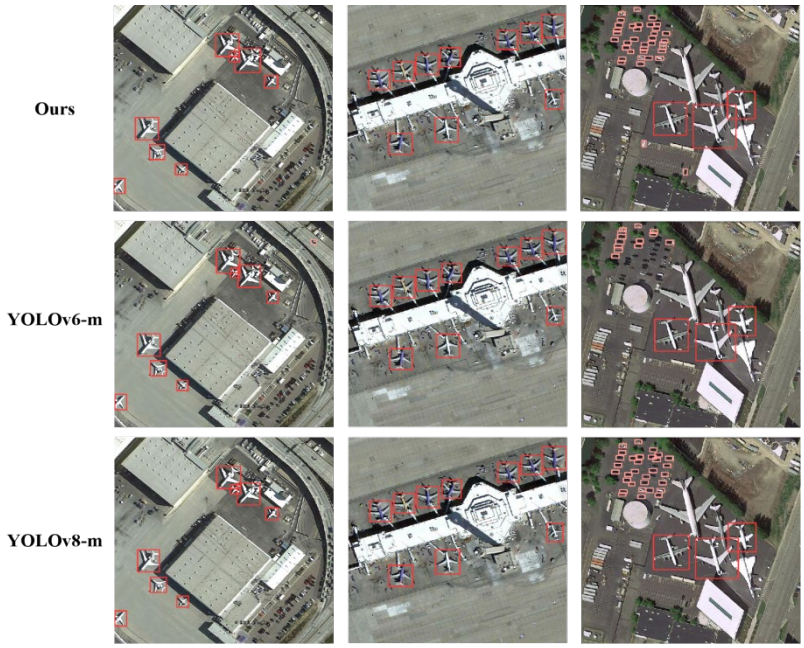

Fig. 10. Comparative analysis of detection results among YOLOv6-m, YOLOv8-m and YOLOv5_mamba (UCAS_AOD. From the comparison in the last column of the images, it can be observed that the proposed algorithm demonstrates better detection capabilities for small objects.

Method	mAP (%)	Airplane	Airport	Baseball field	Basketball court	Bridge	Chimney
YOLOv5s[11]	54.0	70.7	33.3	75.2	67.7	25.8	67.4
YOLOv6s[23]	53.1	72.5	31.2	76.0	69.2	26.3	66.5
YOLOv7-tiny[24]	56.6	71.8	38.1	75.0	68.5	30.0	69.9
YOLOv8s[20]	57.0	72.7	36.8	76.8	70.9	29.3	62.9
YOLOXs[25]	53.8	71.1	34.5	74.1	68.2	27.8	56.3
YOLOv5m[11]	56.3	72.1	50.3	78.4	77.2	27.5	72.6
YOLOv6m[23]	55.0	71.9	47.7	77.5	76.1	25.4	73.6
YOLOv7[24]	58.8	73.9	60.0	79.8	78.9	29.2	75.5
YOLOv8m[20]	65.8	78.2	70.7	82.4	83.3	37.4	81.7
YOLOXm[25]	64.6	77.4	66.4	82.1	82.1	38.3	80.0
Ours	66.0	78.9	71.2	83.7	83.2	38.7	82.4

Table 11. The DIOR dataset's sota experiments (part 1).

Method	mAP (%)	Dam	Expressway-service-area	Expressway-toll-station	Golf field	Groundbreaking	Harbor	Overpass	Ship
YOLOv5s[11]	54.0	36.4	60.9	44.7	59.9	73.1	45.5	42.2	58.4
YOLOv6s[23]	53.1	34.4	55.6	52.9	57.5	69.1	42.4	41.4	58.2
YOLOv7-tiny[24]	56.6	41.6	65.8	46.2	66.9	71.8	48.2	46.0	58.7
YOLOv8s[20]	57.0	42.1	66.4	49.4	66.4	71.3	50.2	47.5	59.2
YOLOXs[25]	53.8	35.2	61.5	45.9	60.2	72.5	45.7	42.3	58.6
YOLOv5m[11]	56.3	36.0	62.9	51.2	59.7	70.1	45.8	42.2	57.0
YOLOv6m[23]	55.0	34.2	60.4	52.3	53.3	69.3	42.5	40.7	55.7
YOLOv7[24]	58.8	37.4	65.1	55.7	64.9	72.5	46.5	42.6	57.6
YOLOv8m[20]	65.8	53.1	75.1	65.8	75.2	78.3	55.3	50.5	61.5
YOLOXm[25]	64.6	50.1	74.6	64.4	71.1	76.9	55.4	48.9	61.0
Ours	66.0	53.1	73.2	67.2	72.9	72.1	54.5	49.2	61.6

Table 12. The DIOR dataset's sota experiments (part 2).

Method	mAP (%)	Stadium	Storage-tank	Tennis-court	Train station	Vehicle	Windmill
YOLOv5s[11]	54.0	70.4	59.5	81.7	27.6	36.7	42.0
YOLOv6s[23]	53.1	68.1	59.0	73.4	27.5	36.4	44.1
YOLOv7-tiny[24]	56.6	74.7	58.8	82.0	32.5	37.4	47.2
YOLOv8s[20]	57.0	73.9	59.7	83.0	40.3	36.9	44.3
YOLOXs[25]	53.8	71.9	59.1	81.7	31.8	36.2	41.9
YOLOv5m[11]	56.3	76.7	56.7	83.7	25.4	34.1	45.8
YOLOv6m[23]	55.0	74.7	53.6	82.8	27.5	33.3	46.6
YOLOv7[24]	58.8	77.8	57.7	84.6	33.2	36.0	47.7
YOLOv8m[20]	65.8	84.6	60.9	87.0	41.0	40.5	54.3
YOLOXm[25]	64.6	83.1	60.5	86.4	41.2	40.1	52.9
Ours	66.0	84.7	61.9	87.5	45.3	42.8	55.6

Table 13. The DIOR dataset's sota experiments (part 3).

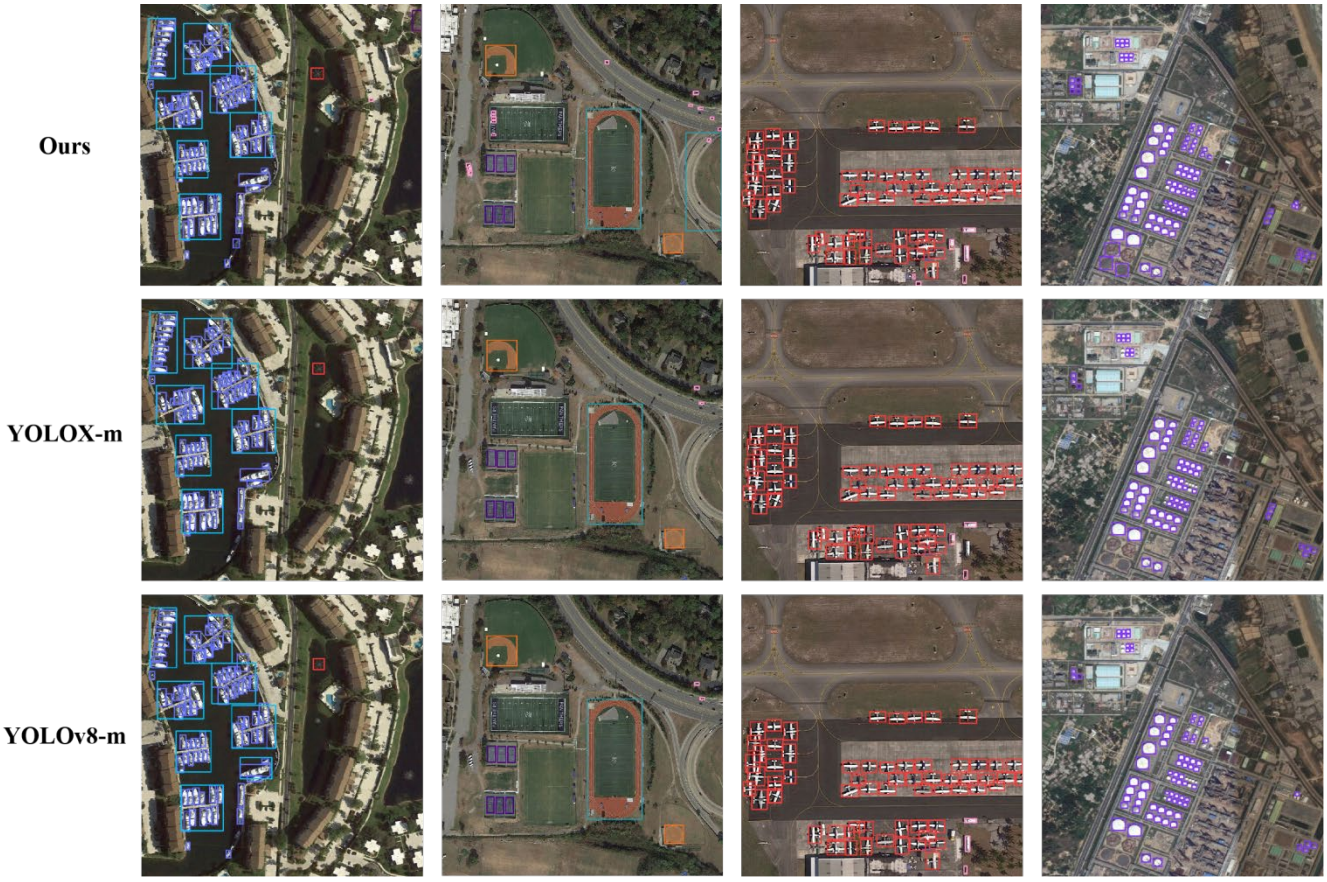

Fig. 11. Comparative analysis of detection results among YOLOX-m, YOLOv8-m and YOLOv5_mamba (DIOR). From the comparison in the last column of the images, it can be observed that the proposed algorithm demonstrates better detection capabilities for small objects.

Conclusions

The paper presents significant improvements to the YOLOv5 algorithm through several innovative modifications. The introduction of the C2f module from YOLOv8 into the backbone network aims to enhance feature extraction capabilities, allowing for better representation of input data. By combining the mamba module with the C2f module, a bidirectional dense feedback network is constructed, which enhances the transmission of contextual information and improves focus capabilities within the neck part of the architecture. An adaptive gate-controlled feature fusion network is employed in the head part of YOLOv5. This mechanism finely controls the flow of information, leading to improved detection performance.

The experimental results indicate that on the publicly available VisDrone2019 dataset, the proposed algorithm achieves a 9.3% improvement in detection accuracy compared to the original YOLOv5 baseline network,

culminating in a detection accuracy of 57.5%. Notably, the model performs exceptionally well in detecting small objects. For the UCAS_AOD dataset, the proposed algorithm outperforms YOLOv5-s by 9%. In the case of the DIOR dataset, the proposed algorithm exceeds YOLOv5-s by 12%.

Looking ahead, future work will focus on replacing basic modules and downsampling modules in YOLOv5_mamba and enhancing functionalities to further boost detection performance. This exploration aims to leverage advanced architectures and techniques to continue improving the effectiveness of the model in diverse detection scenarios.

Data availability

These data were available at: https://github.com/VisDrone, https://hyper.ai/datasets/5419, https://pan.baidu.com/s/1w8iq2WvgXORb3ZEGtmRGOw. The code is available at: https://github.com/xgyutu/yolo_mamba.

Received: 29 May 2024; Accepted: 16 September 2024
Published online: 27 September 2024

References

1. Zhou, H., Ma, A., Niu, Y. & Ma, Z. Small-object detection for uav-based images using a distance metric method. *Drones*. **6**(10), 308 (2022).
2. Khosravi, M. R., Rezaee, K., Moghimi, M. K., Wan, S. & Menon, V. G. Crowd emotion prediction for human-vehicle interaction through modified transfer learning and fuzzy logic ranking. *IEEE Trans. Intell. Transp. Syst.* (2023).
3. Roy, A. M. & Bhaduri, J. Real-time growth stage detection model for high degree of occultation using densenet-fused yolov4. *Comput. Electron. Agric.***193**, 106694 (2022).
4. Feng, J. & Xiao, X. Multiobject tracking of wildlife in videos using few-shot learning. *Animals*. **12**(9), 1223 (2022).
5. Ren, X. et al. An improved mask rcnn algorithm for uav tir video stream target detection. *Int. J. Appl. Earth Obs. Geoinf.***106**, 102660 (2022).
6. Smitha, J. C. & Babu, S. S. Mri brain image classification using haar wavelet and artificial neural network. *Artif. Intell. Evol. Algorithms Eng. Syst.***325**, 253–261 (2015).
7. Violap, A. & Jones, M. J. Rapid object detection using a boosted cascade of simplefeatures. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1–9. (IEEE, 2001).
8. Dalal, N. & Triggs, B. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. 886–893. (IEEE, 2005).
9. Felzenszwalb, P., Mcallester, D. & Ramanan, D. A discriminatively trained, multiscale, deformable part model. In *2008 IEEE Conference on Computer Vision and Pattern Recognition: volume 8*. 1–8. (IEEE, 2008).
10. Felzenszwalb, P. F., Girshick, R. B. & Mcallester, D.. Cascade object detection with deformable part models. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2241–2248. (IEEE, 2010).
11. Singh, I. & Munjal, G. Modified YOLOv5 for small target detection in aerial images. *Multimed. Tools Appl.***2023**, 1–22. (2023).
12. Zhang, H., Zheng, J. & Song, C. Multiple-target matching Algorithm for SAR and visible light Image data captured by multiple unmanned aerial vehicles. *Drones*. **8**(83), 1–17 (2024).
13. Du, M., Zou, H., Wang, T. & Zhu, K. A cooperative target localization method based on UAV aerial images. *Aerospace*. **10**(943), 1–24. (2023).
14. Yue, J., Wang, H., Zhu, D. & Aleksandr, C. UAV formation cooperative navigation algorithm based on improved particle filtering. *Chin. J. f Aeronaut.***44**, 251–262 (2023).
15. Liu, Y. et al. Autonomous navigation and localization in IMU/UWB group domain based on particle filtering. *Transducer Microsyst. Technol.***41**, 47–50 (2022).
16. Chen, J., Wen, R. & Ma Lili. Small object detection model for UAV aerial image based on YOLOv7. *Signal Image Video Process.* 1–13 (2023).
17. Tang, S. et al. HIC-YOLOv5: Improved YOLOv5 for small object detection. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 6614–6619. (2023).
18. Benjumea, A. et al. YOLO-Z: improving small object detection in YOLOv5 for autonomous vehicles. ArXiv abs/2112.11798 (2021).
19. Bakirci, M. Enhancing vehicle detection in intelligent transportation systems via autonomous UAV platform and YOLOv8 integration. *Appl. Soft Comput.***164**, 112015 (2024).
20. Bakirci, M. U. YOLOv8 for enhanced traffic monitoring in intelligent transportation systems (ITS) applications. *Digit. Signal Process.* 152. (2024).
21. Gupta, A., Gu, A. & Berant, J. Diagonal state spaces areas Eectiveas Structured State spaces. *Adv. Neural. Inf. Process. Syst.***35**, 22982–22994 (2022).
22. Ugidos, M., Nuño-Cabanes, C., Tarazona, S., Ferrer, A. & Nielsen, L. K. MAMBA: a model-driven, constraint-based multiomic integration method. (2022).
23. Li, C. et al. YOLOv6: a single-stage object detection framework for industrial applications. arXiv preprint arXiv:2209.02976 (2022).
24. Wang, C. Y., Bochkovskiy, A. & Liao, H. Y. M. YOLOv7: Trainable NMS, Model Scaling, and Inference Optimization, arXiv preprint arXiv:2207.00778 (2022).
25. Ge, Z., Liu, S., Wang, F., Li, Z. & Sun, J. YOLOX: Exceeding YOLO Series in 2021, rXiv e-prints. https://doi.org/10.48550/arXiv.2107.08430 (2021).

Author contributions

Conceptualization, Sxw. and Xyl.; methodology, Xyl.; validation, Sxw and Xyl.; formal analysis, Sxw and Ccg.; investigation, Sxw and Xyl.; data curation, Sxw and Xyl.; writing—original draft preparation, Sxw.; writing—review and editing, Sxw and Xyl.; visualization, Xyl.; supervision, Ccg.; project administration, Ccg.; funding acquisition, Ccg. All authors have read and agreed to the published version of the manuscript.

Funding

This paper is funded by the creation of "5G + artificial intelligence" remote treatment and diagnosis platform for major aortic diseases (grant number: 2022BCA035).

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to X.L. or C.G.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.