Article | Received 27 March 2026; Revised 22 April 2026; Accepted 23 April 2026; Published 11 June 2026
https://doi.org/10.55092/aic20260007

PAS-YOLO: an enhanced algorithm for high-precision non-cooperative UAV recognition

Kai Zhou and Yanbin Zhang*

School of Electrical and Information Engineering, Zhengzhou University, Zhengzhou 450001, China

* Correspondence author; E-mail: ieybzhang@zzu.edu.cn.

Highlights:

- Investigates precise detection and identification of UAV remote control signals to enhance low-altitude security against unauthorized flights.
- Proposes PAS-YOLO, an improved YOLOv12 based algorithm integrating a parallelized patch-aware attention module, ASF-YOLO neck structure, and switchable atrous convolution. These features enhance multi-scale feature fusion and receptive field.
- Simulation experiments show that the PAS-YOLO model strikes an effective balance between precision and inference speed, achieving positive detection accuracy and robustness.

Abstract: In light of the recent proliferation of unmanned aerial vehicles (UAVs) and the challenges posed by unauthorized flights interfering with air traffic, the precise detection and identification of UAV have become critical for ensuring security at low altitudes. This study introduces PAS-YOLO, an enhanced algorithm for detecting and recognizing UAV remote control signals, built upon the You Only Look Once version 12 (YOLOv12) framework, with the objective of improving UAV target identification capabilities. To augment the detection of small target signals and reduce the risk of remote control signal loss, a parallelized patch-aware attention (PPA) module is integrated into the backbone network. Addressing the limited feature representation capacity of YOLOv12, particularly the difficulty in distinguishing similar remote control signals through fine-grained features, the neck network is redesigned based on the Attentional Scale Sequence Fusion YOLO (ASF-YOLO) architecture. Furthermore, to broaden the receptive field and enhance the contextual extraction capability for signals with diverse time-frequency characteristics, the original area attention with C2f (A2C2f) module is refined by incorporating a switchable atrous convolution (SAConv) module. Experimental evaluations are performed using the publicly available Radio Frequency (RF) signal dataset DroneRFa, wherein the Short-Time Fourier Transform (STFT) is employed to generate a UAV time-frequency spectrum dataset. The results indicate that the proposed PAS-YOLO algorithm attains an average detection accuracy of 99.36% for mAP@50 and 75.38% for mAP@50:95 across 22 UAV remote control signal models. Compared to the baseline YOLOv12 model, these metrics represent improvements of 0.23% and 3.45%, respectively.

Keywords: UAV; UAV classification; deep learning; YOLOv12; radio frequency identification

1. Introduction

With the explosive growth of unmanned aerial vehicles (UAVs), they have been widely applied in military, civil, and other fields [1,2]. However, unauthorized UAV operations have also posed serious safety hazards, with frequent incidents resulting from illegal and reckless UAV flights. Therefore, developing efficient and accurate UAV detection algorithms to counter unauthorized operations has become an urgent priority [3,4]. Frequency-hopping communication technology, widely adopted for UAV remote control signals, significantly enhances transmission security through its low interception probability, strong anti-jamming capabilities, and high confidentiality. Consequently, research on remote control signal monitoring and identification methods holds both significant theoretical importance and practical value.

Currently, UAV detection methods include: acoustic detection [5]; optical detection [6]; radar detection [7]; radio frequency (RF) detection [8]. RF detection is a passive detection method, meaning it does not transmit signals but only receives, extracts, analyzes, and determines the characteristics and direction of UAV targets. RF detection is more advantageous for analyzing and studying UAVs.The remote control signals for UAVs employ frequency-hopping communication. Different UAV models exhibit significant variations in their carrier frequency points and frequency interval parameters, and the modulation generation mechanism for their frequency-hopping signals is fixed at the factory. Therefore, the extraction and identification of frequency-hopping signal characteristic parameters can serve as a key technological approach for UAV model classification.

Methods for detecting and identifying UAV remote control signals can be classified into four main categories: time domain, frequency domain, time-frequency domain, and hybrid domain.

Time domain based UAV signal detection and recognition methods: Reference [9] achieves high-precision recognition by extracting time-domain RF fingerprint features combined with dimensionality reduction and classifiers, but is only applicable to single-target scenarios. Reference [10] employs a one-dimensional convolutional neural network (1D-CNN) approach, achieving 99.8% detection accuracy but only 88.4% aircraft model recognition rate. Reference [11] proposed the lightweight RF-NeuralNet framework, which performed well in resource-constrained scenarios but achieved recognition accuracies below 90% across all metrics. The RF-UAVNet architecture introduced in Reference [12] achieved high accuracy in detection (99.85%), aircraft model recognition (98.53%), and flight mode recognition (95.33%). References [10–12] all rely on the same dataset containing only three aircraft models, limiting their scale and generalization capabilities.

Frequency domain based UAV signal detection and recognition method: Reference [13] combines fixed-boundary empirical wavelet transform (FBREWT) with lightweight CNN, achieving 97.25% classification accuracy for 15 UAV types. However, it exhibits high misclassification rates between certain models (e.g., 15.9% misclassification of Matrice 600 as Phantom 4 Pro). Reference [14] extracts features like power spectral density, Mel-frequency cepstral coefficients (MFCC), and linear frequency cepstral coefficients (LFCC) using support vector machine (SVM) classification, offering high computational efficiency but unclear physical significance of features. Reference [15] enhances features using minimum

variance distortionless response (MVDR) spectrum and overlapping sliding window segmentation (OSWS) techniques, classified by lightweight networks, yet requires fine-tuning window parameters and suffers from false detections. Reference [16] demodulates bit streams via compressed sensing and orthogonal matching pursuit (OMP) without prior information but demands extremely high synchronization accuracy. Reference [17] proposed a GA-Bagging-KNN approach achieving 98% accuracy in binary classification but dropping to 79% in four-class tasks, with limited capability to distinguish similar models.

Time frequency domain based UAV signal detection and recognition method: Reference [18] preprocesses signals using complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) and wavelet envelope threshold denoising, then combined with Short-Time Fourier Transform (STFT) to extract multi-channel time-frequency features. These features are fed into an improved EfficientNet-UAV model, achieving over 95% accuracy in multiple scenarios but with high computational complexity. Reference [19] employs an enhanced YOLOv5-7.0 model to detect frequency-hopping feature images generated by STFT and wavelet transforms, achieving an average accuracy of 96.7%. However, its multi-UAV detection capability is limited. Reference [20] utilizes Mel-spectrograms to characterize time-frequency features and applies YAMNet transfer learning for classification. Yet, it remains sensitive to noise and has limited validated aircraft models. Reference [21] distinguishes remote-control and Bluetooth signals by combining cyclostationarity with STFT, but cannot identify specific UAV models. Reference [22] utilizes full-duplex radio reception signals, extracts features via STFT, and employs CNN classification, maintaining good recognition capability even in noisy environments. Reference [23] compared multiple CNNs' classification performance on STFT spectrograms under low signal-to-noise ratios (SNR), achieving an average accuracy exceeding 80%, but incurred high computational overhead. Reference [24] employed a lightweight CNN to process power spectrograms, achieving 99.17% accuracy at low SNR (-15 dB), yet struggled with multi-UAV classification. Reference [25] employs a Deep Recurrent Neural Network (DRNN) for device feature extraction from STFT spectrograms, enabling simultaneous classification of seven UAV. However, experiments rely on signal superposition simulations without accounting for real-world frequency band overlap. Reference [26] proposes an RF fingerprinting method based on STFT and PCA dimension reduction, achieving over 98% accuracy for 1–3 UAV. Yet its detection range is limited to a maximum of 7 meters, restricting practical application.

Hybrid domain based UAV signal detection and recognition method: Reference [27] proposed a time-frequency multiscale convolutional neural network (TFMS-CNN), which enhances robustness by fusing transient and steady-state features through dual-branch convolutional integration in both time and frequency domains. Reference [28] employs compressed sensing (CS) and multi-stage deep learning, achieving over 99% accuracy across three-stage tasks but is limited to three UAV models, resulting in insufficient generalization capability; Reference [29] combined multi-level RF fingerprinting with machine learning, achieving 99.8% detection accuracy and 98.13% classification accuracy for 15 UAV types at high SNR. However, performance degraded at low SNR with misclassifications occurring; Reference [30] employs stacked denoising autoencoder-local outlier factor (SDAE-LOF) with hierarchical classification to distinguish UAVs from interference signals. However, the method is hindered by the high computational cost of time-frequency analysis, inadequate real-time response, and a flight-controller

classification accuracy that reaches just 73.19%.

To address the common shortcomings of existing time-frequency domain based UAV detection methods, such as high computational complexity leading to poor real-time performance, sensitivity to noise, limited model validation, and degraded recognition performance under frequency overlap interference in real world scenarios. This paper proposes PAS-YOLO, an improved UAV remote control signal detection and recognition algorithm based on YOLOv12. The main contributions of this work are summarized as follows:

- A UAV time-frequency spectrum dataset covering 22 distinct UAV models was constructed from open source RF signal datasets by applying STFT. This provides a valuable data foundation for deep learning based UAV RF identification research.
- In order to deal with the issue of extracting small, multi-scale features in UAV time-frequency spectrum maps, the PAS-YOLO model was proposed. The model integrates a parallelised PPA module into the backbone network to strengthen the perception of small objects; adopts an ASF-YOLO neck structure to enable fusion of features across multiple scales; and incorporates SAConv within the A2C2f module to enlarge the receptive field, thus enhancing the model's ability to represent features effectively.
- Experiments demonstrate that PAS-YOLO achieves improved detection accuracy on our self built dataset (mAP@50: 99.36%, mAP@50:95: 75.38%) compared to You Only Look Once (YOLO) series models, while maintaining high inference efficiency (224 FPS). This balances accuracy and speed under controlled model complexity. Further noise resistance experiments validate the model's robustness in low SNR environments, demonstrating its potential for practical application in complex electromagnetic scenarios.

The structure of this paper is as follows: Section 2 analyzes UAV signals and the limitations of YOLOv12. Section 3 details the improved modules and overall model architecture. Section 4 describes the dataset construction and experimental results. Section 5 summarizes the entire paper.

2. Signal model and problem description

This section first explores methods for time-frequency characterization of UAV frequency-hopping signals, demonstrating the feasibility of converting them into time-frequency spectrograms and applying target detection models. Subsequently, it systematically introduces the fundamental architecture of the YOLOv12 algorithm and analyzes its limitations, providing a theoretical basis for the algorithmic improvements presented in Section 3.

2.1. UAV remote control signal

The remote control signals for UAVs employ a frequency-hopping mechanism, rapidly switching carrier frequencies to enhance anti-interference capabilities. Different UAV models may exhibit similar frequency-hopping patterns, necessitating the identification of subtle signal characteristics for differentiation. STFT converts the signal into a time-frequency spectrum, from which unique details such as modulation characteristics and frequency deviation can be extracted for different models. Taking the DJI Phantom 4 Pro as an example, the RF signal and its STFT-derived time-frequency spectrum are shown in

Figure 1. This paper transforms the UAV identification problem into a fine-grained object detection and classification task on the time-frequency spectrum. By employing the YOLOv12 framework and optimizing the model's sensitivity to detailed features, high-precision identification of various UAV models is achieved.

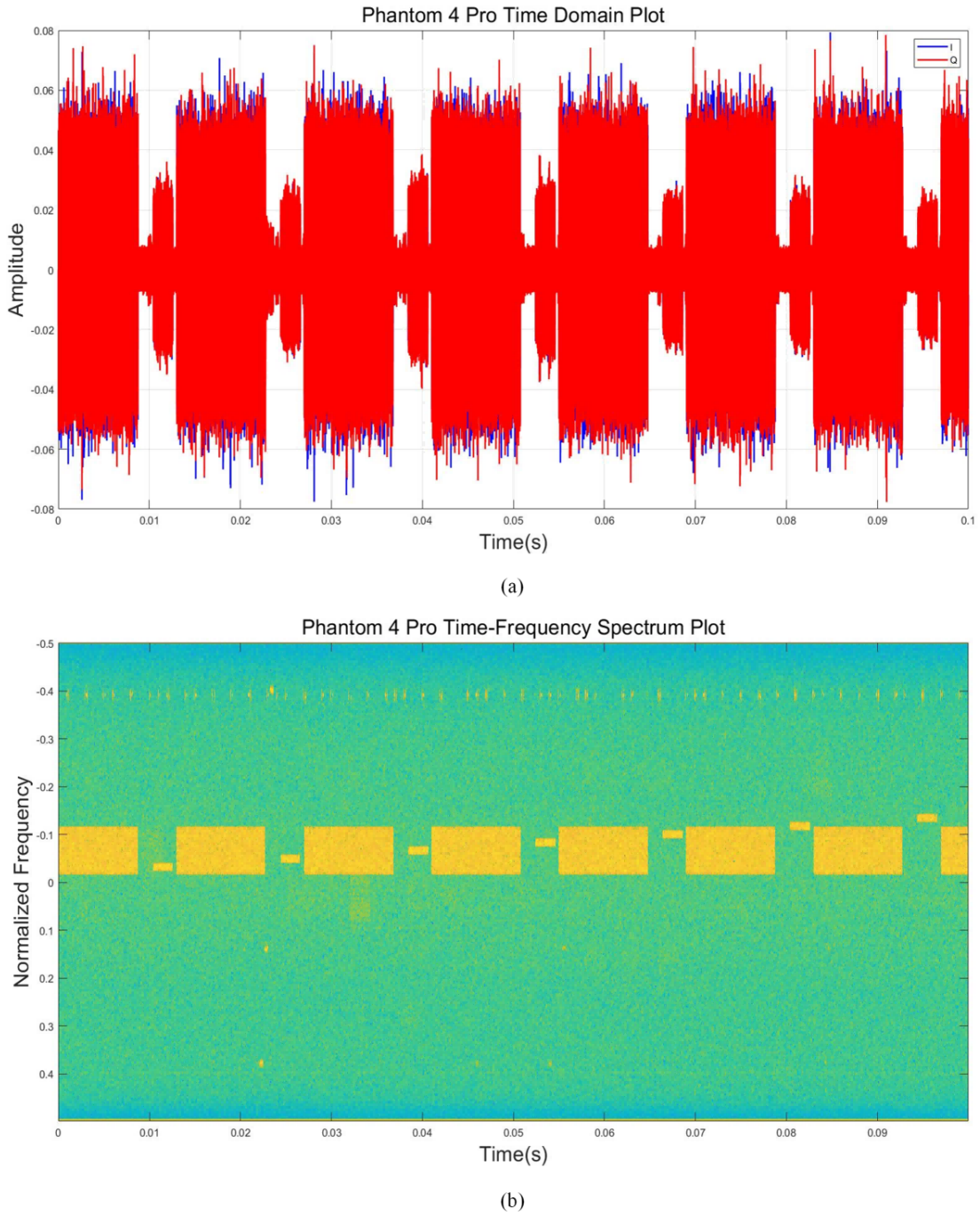

Figure 1. Time domain and time-frequency spectrum plot of the Phantom 4 Pro. **(a)** Time Domain Plot; **(b)** Time-Frequency Spectrum Plot.

2.2. YOLOv12

YOLO is a machine-learning algorithm that detects objects throughout the entire process. The present paper proposes an enhancement to the network architecture, based on the YOLOv12n model. The

fundamental network architecture of YOLOv12 is illustrated in Figure 2. YOLOv12 has been shown to strike a good balance between detection accuracy and real-time performance. However, the basic YOLOv12 model still has the following shortcomings that require improvement:

- Limited field of vision: the model largely depends on standard convolutional layers for learning features, thus constraining its ability to capture long-range context.
- Limited feature characterization capabilities. A relatively fixed parameter budget restricts the model from capturing sufficiently rich feature information when dealing with complex image patterns, especially during pre-training on large-scale datasets, thereby limiting its expressive power.
- Limited small object detection capability. The Complete Intersection over Union (CIoU) loss employed by the model places excessive weight on geometric factors, such as the bounding box's width-to-height ratio and the distance between its center points. This results in excessive penalties for low-quality samples with large deviations in predicted box positions, failing to effectively balance the training weights of samples of varying difficulty and curtailing the model's capacity for performance gains in small-object detection.

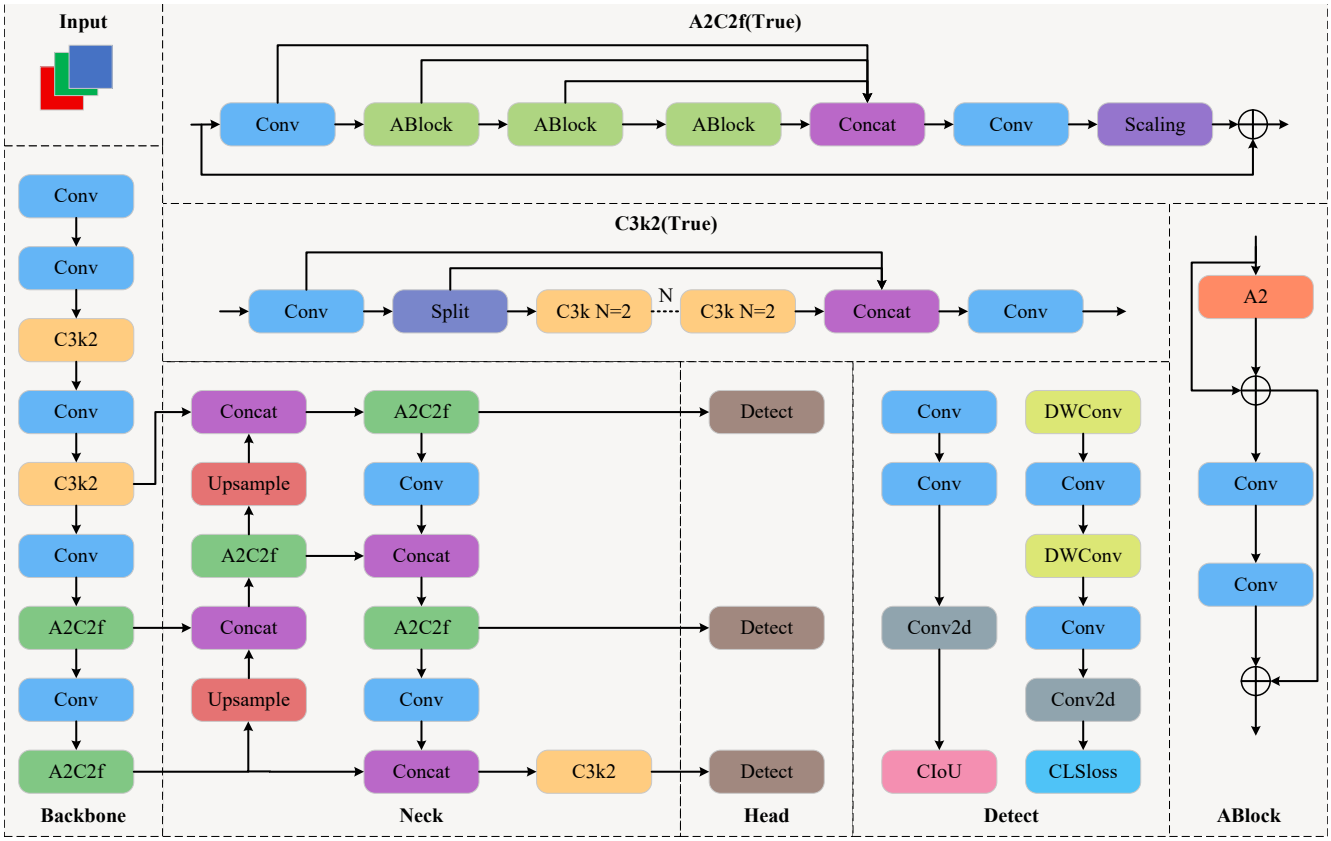

Figure 2. YOLOv12 model architecture diagram.

3. PAS-YOLO model

To overcome the limitations of the YOLOv12 model in handling UAV spectrograms—including restricted receptive fields, weak feature representation capabilities, and small object detection failures. This section introduces the PAS-YOLO enhanced model through innovations across three layers: the backbone network, neck network, and A2C2f module.

3.1. Parallelized patch-aware attention module

UAV remote control signals typically occupy very few pixels in the spatial domain. The original YOLOv12 backbone network relies primarily on standard convolutional downsampling operations, which are inherently insensitive to small objects. Consequently, during multiple downsampling stages, key signal features are easily overwhelmed by background noise, resulting in the failure to detect weak frequency-hopping signals. To address this specific limitation, this paper introduces the Parallelized PPA module [31] into the backbone network. Originally, this module was developed for the Hierarchical Context Fusion Network (HCF-Net), which targets infrared small object detection—a task that likewise encounters the difficulty of preserving small objects during hierarchical downsampling. The PPA module utilises a multi-branch feature extraction strategy combined with an attention mechanism to preserve and strengthen the representations of small objects. This strategy guarantees the preservation of critical information, even in the face of multiple rounds of spatial compression.

The PPA module's main advantage stems from the strategy it uses to extract branch features. Figure 3 illustrates that the PPA module employs a parallel multi-branch design, with each branch dedicated to extracting features at different scales and levels. Such a multi-branch design has proven effective in extracting multi-scale object characteristics, thereby boosting small-object detection precision. The discussed strategy consists of three parallel pathways: one for local information, one for global context, and one based on serial convolution. Given an input feature tensor $\mathrm{F} \in \mathbb{R}^{H' \times W' \times C}$, a point-wise convolution is first applied to produce $\mathrm{F}' \in \mathbb{R}^{H' \times W' \times C'}$. From this, the three branches independently generate their respective outputs $\mathrm{F}_{\mathrm{local}}$, $\mathrm{F}_{\mathrm{global}}$, and $\mathrm{F}_{\mathrm{conv}}$, all of dimension $\mathbb{R}^{H' \times W' \times C'}$. These branch outputs are then summed element-wise to form $\tilde{\mathrm{F}} \in \mathbb{R}^{H' \times W' \times C'}$. To process local information, F' is first partitioned into a grid of contiguous spatial blocks of size $(p \times p, \frac{H'}{p}, \frac{W'}{p}, C')$. Channel-wise averaging is subsequently performed on each block, reducing the feature to $(p \times p, \frac{H'}{p}, \frac{W'}{p})$, after which a Feed-Forward Network (FFN) carries out linear transformations. The outcomes of the linear calculations are then employed, using the activation function to generate the spatial probability distribution of feature responses, with the weights being recalibrated accordingly. In the weighted results, the PPA module uses feature selection to select task-related features from channels and labels, generating two features, $\mathrm{F}_{\mathrm{local}} \in \mathbb{R}^{H' \times W' \times C'}$ and $\mathrm{F}_{\mathrm{global}} \in \mathbb{R}^{H' \times W' \times C'}$. At the same time, a sequence of convolutions consisting of three 3×3 convolutional layers replaces the traditional three convolutional layers of 7×7, 5×5, and 3×3, resulting in three different outputs: $\mathrm{F}_{\mathrm{conv1}} \in \mathbb{R}^{H' \times W' \times C'}$, $\mathrm{F}_{\mathrm{conv2}} \in \mathbb{R}^{H' \times W' \times C'}$, and $\mathrm{F}_{\mathrm{conv3}} \in \mathbb{R}^{H' \times W' \times C'}$, add the three output results to obtain the sequence convolution output $\mathrm{F}_{\mathrm{conv}} \in \mathbb{R}^{H' \times W' \times C'}$.

After multi-branch feature extraction, an attention mechanism is used for adaptive feature enhancement. An efficient channel attention is cascaded with a spatial attention component to form the attention mechanism module. The feature tensor $\tilde{\mathrm{F}} \in \mathbb{R}^{H' \times W' \times C'}$ is obtained by sequentially processing the one-dimensional channel attention map followed by the two-dimensional spatial attention map, yielding the final output feature representation $\mathrm{F}'' \in \mathbb{R}^{H' \times W' \times C'}$ of the PPA module.

The PPA module's parallel local and global branches enable it to simultaneously extract fine-grained local textures along with global sequence features. This multi-scale perception mechanism mitigates the risk of YOLOv12 losing weak drone remote control signals during downsampling.

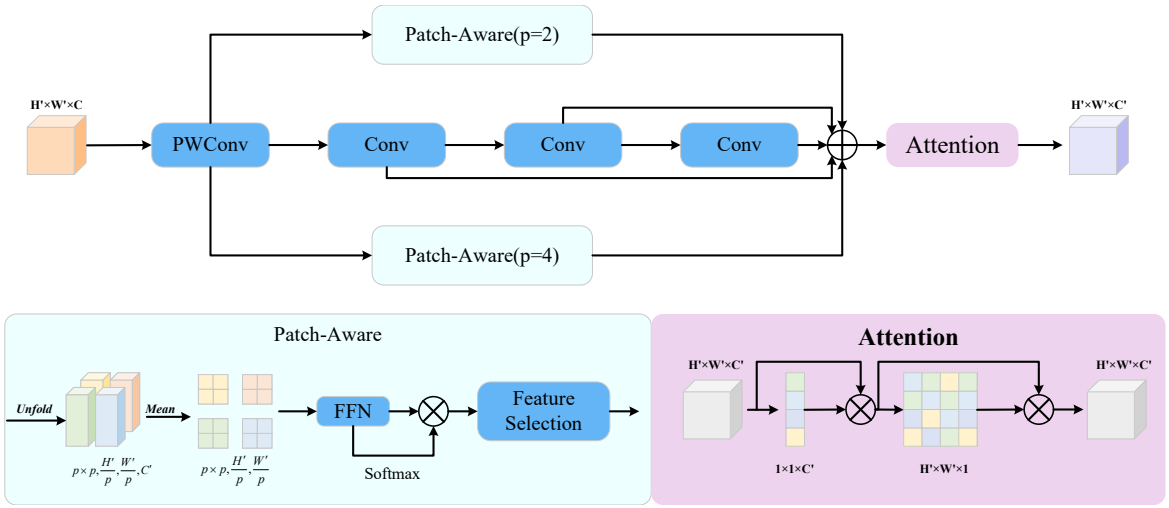

Figure 3. Parallelized patch-aware attention module structure diagram. In patch-aware, p = 2 and p = 4 represent local branches and global branches, respectively.

3.2. ASF-YOLO neck network structure

The frequency hopping patterns of different UAVs exhibit subtle differences, and the bandwidth and dwell time of individual remote control signals are highly similar. Coupled with the lack of color information in single-channel observation, distinguishing between them is difficult, thus requiring strong feature extraction capabilities. Consequently, to address the insufficient feature representation capabilities of YOLOv12, this paper introduces the neck network structure of ASF-YOLO, effectively fusing multi-scale features. The ASF-YOLO [32] framework is shown in Figure 4. ASF-YOLO has developed a novel feature fusion network architecture that combines spatial and multi-scale features to capture detailed information about small objects. ASF-YOLO is an improvement over YOLOv5, with its backbone network remaining unchanged. The neck network consists of three main modules: (1) Scale Sequence Feature Fusion (SSFF) module, (2) Triple Feature Encoder (TFE) module, and (3) Channel and Position Attention Mechanism (CPAM) module.

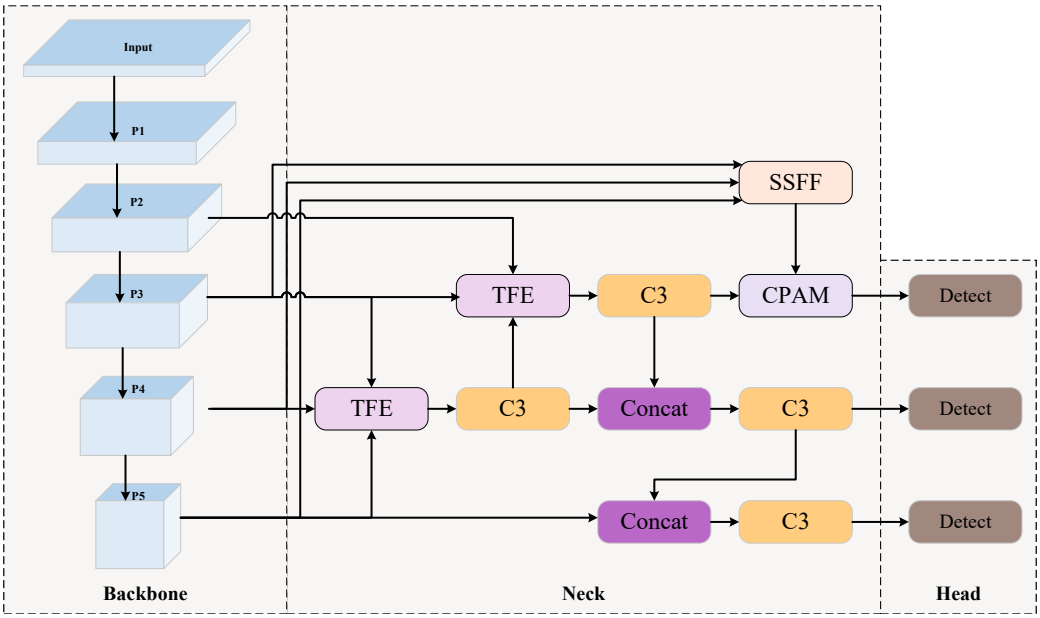

Figure 4. ASF-YOLO model architecture diagram.

3.2.1. Scale sequence feature fusion module

Existing literature employs feature pyramids to fuse features, typically utilising summation or cascading methods to fuse pyramid features. Nevertheless, conventional feature pyramid networks often struggle to exploit inter-layer dependencies among pyramid feature maps. By contrast, the SSFF module has proven effective at merging multi-scale features—specifically, by integrating high-level semantics from deep layers with fine-grained spatial details from shallow layers—all while keeping feature map aspect ratios consistent.

The SSFF module structure diagram can be observed in Figure 5, where the P3, P4, and P5 feature maps undergo convolution with a set of Gaussian kernels whose standard deviations increase progressively, as formalized in the equation:

$$F_\sigma(h,w) = f(h,w) \times G_\sigma(h,w) \tag{1}$$

$$G_\sigma(h,w) = \frac{1}{2\pi\sigma^2} e^{-(h^2+w^2)/2\sigma^2} \tag{2}$$

where $f(h,w)$ represents a two-dimensional input image of size $h \times w$. The filtered version $F_\sigma(h,w)$ is obtained by convolving f with a 2D Gaussian kernel $G_\sigma(h,w)$, in which σ determines the kernel's standard deviation during the convolution process.

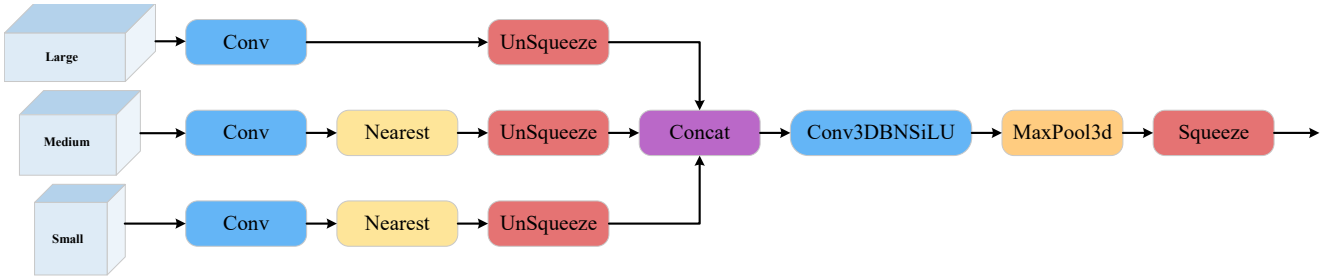

Figure 5. Scale sequence feature fusion module structure diagram.

The feature maps derived from the model are generated across disparate levels, possess differing resolutions, and align with varying receptive field scales. Fusing multi-scale information requires the feature maps at various resolutions to be first upsampled to a common resolution. Motivated by 2D/3D convolution approaches for video sequences (continuous frames), we stack the aligned multi-scale feature maps along a new dimension to construct a 3D volume. Subsequently, 3D convolution is applied to capture cross-scale dependencies among the feature maps. Among them, high-resolution shallow features (such as P3) typically retain rich spatial detail information, making them especially important for small-object detection and segmentation. Therefore, the design concept of the SSFF module builds upon the P3 layer, which enhances the feature representation of P3 by aggregating contextual information from other scales through the above-mentioned 3D convolution operation. The module mainly consists of the following parts:

- The number of channels in the P4 and P5 feature layers is to be adjusted by means of a 1×1 convolution.
- Apply nearest neighbor interpolation to resize its spatial dimensions to the size of the P3 layer.
- Unsqueeze is applied to each feature layer, converting the 3D tensor (height, width, channel) into a 4D tensor with an extra leading dimension (depth, height, width, channel).

- The 4D feature maps are concatenated along the depth dimension, forming 3D feature maps that are subsequently used in the following convolution.
- Through 3D convolution (merging depth information), 3D batch normalization, Sigmoid Linear Unit (SiLU) activation function, and 3D max pooling (compressing depth dimensions).
- The depth dimension is removed via squeeze, thereby performing the extraction of scale-sequential features.

3.2.2. Triple feature encoder module

To improve the detection of densely overlapping small objects, one must exploit multi-scale features. The backbone network generates feature maps at different resolutions across its layers. However, conventional feature pyramid networks merely upsample small-scale maps and add them to the preceding layer's features, thus ignoring the potential of deep, large-scale feature maps, which are replete with intricate detail information. The triple-scale feature enhancement (TFE) module enhances the detail representation capabilities of deep-layer features by synergistically processing high, medium, and low resolution features. As shown in Figure 6, the TFE module processing flow is as follows:

- Channel alignment: Perform 1×1 convolution on the input features to unify the number of channels.
- Large feature processing: Use max pooling together with average pooling to downsample and reduce the size of feature maps.
- Small feature processing: Use nearest neighbor upscaling to increase resolution and enlarge feature maps.
- Feature fusion: Concatenate the processed high, medium, low resolution feature maps along the channel dimension, and finally use convolution to achieve cross-scale feature interaction.

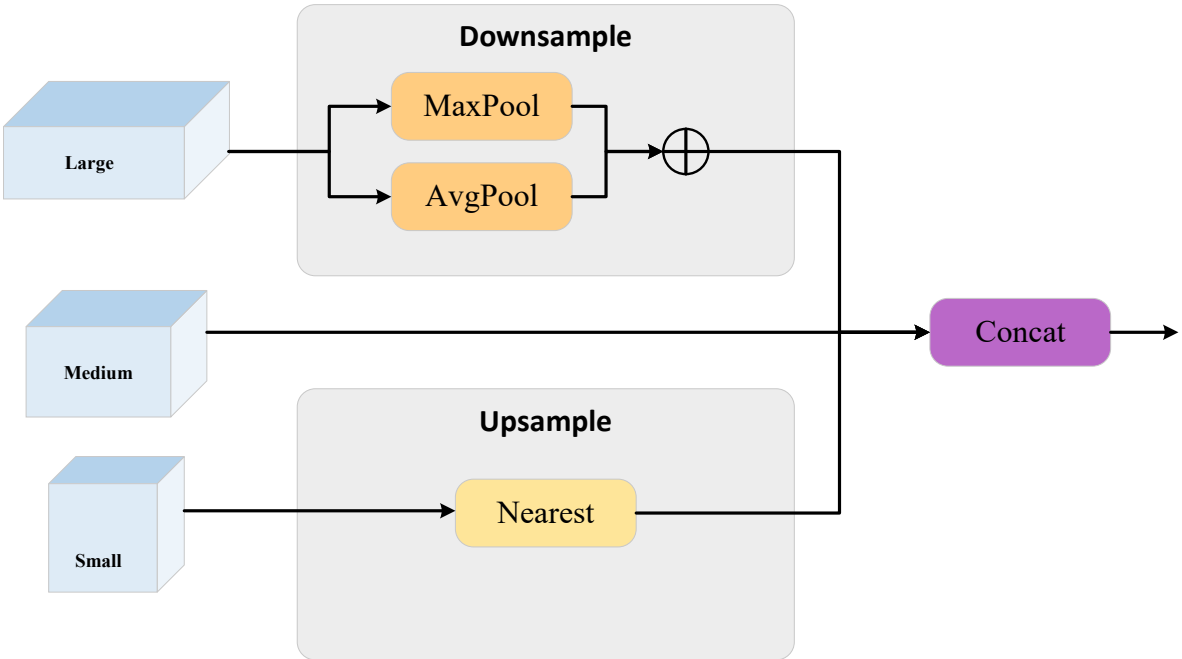

Figure 6. Triple feature encoder module structure diagram.

3.2.3. Channel and position attention mechanism module

In order to effectively extract discriminative features in the channel dimension, a channel-position attention module (CPAM) was designed. This module fuses the TFE module's detailed features with the multi-scale contextual information from the SSFF module. The CPAM module, depicted in Figure 7, comprises two cascaded branches: (1) Channel Attention Branch: Receives the detailed features output from the TFE module and learns feature weights in the channel dimension; (2) Position Attention Branch: Receives the output from the Channel Attention Branch, as well as the fusion result of multi-scale features from the SSFF module.

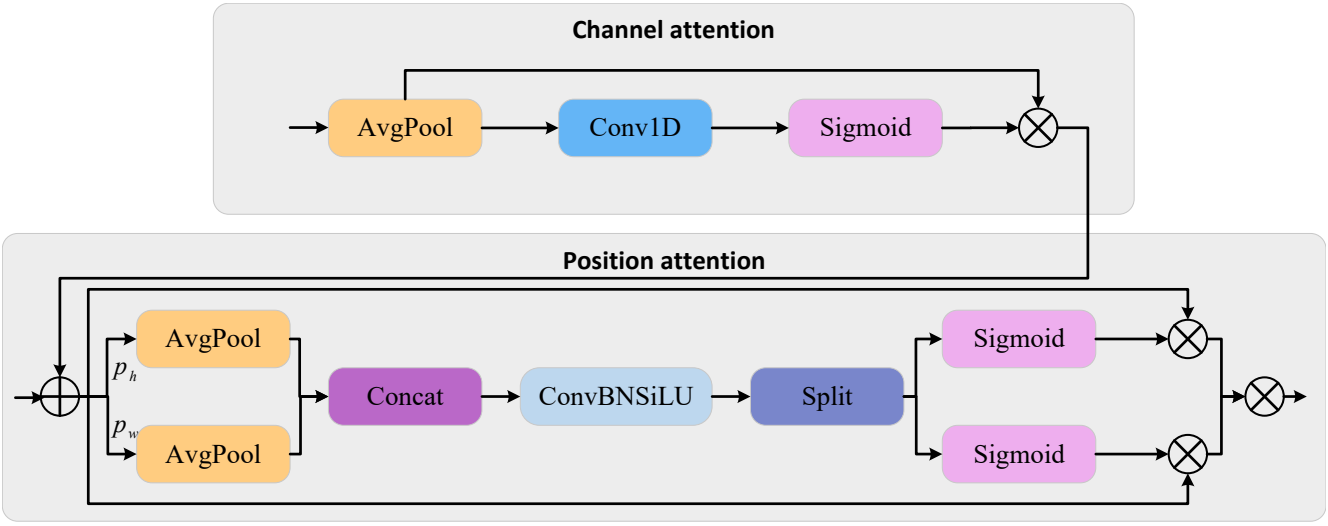

Figure 7. Channel and position attention mechanism module structure diagram.

The enhanced feature map constitutes the input source for the channel attention branch, and this incorporates the more detailed features extracted by the TFE module. The CPAM module introduces a dimension-preserving attention mechanism to efficiently capture cross-channel interaction information: after global average pooling is performed while preserving the dimension, each channel is enhanced by aggregating information from its k nearest neighboring channels, thereby capturing local cross-channel interactions. This process is implemented via a one-dimensional convolution whose kernel size k governs the range of such interactions—that is, the number of adjacent channels that participate in computing the attention weight for a single channel. Determining the best k demands manual adjustment across different network architectures and varying numbers of convolutional modules, which is quite laborious. However, since k is proportional to the channel dimension C, and C is typically an integer power of two, we can determine k as:

$$C = \psi(k) = 2^{(\gamma \times k - b)} \tag{3}$$

$$k = \Psi(C) = \left| \frac{\log_2(C) + b}{\gamma} \right|_{odd} \tag{4}$$

where $|\cdot|_{\text{odd}}$ denotes the nearest odd integer; γ and b are fixed to 2 and 1, respectively. This nonlinear mapping ensures that channel groups with a larger number of channels enjoy a wider cross-channel interaction scope, while those with fewer channels operate over a narrower range. As a result, the channel attention mechanism enables deeper extraction of features across multiple channels.

The outputs of the SSFF module and the channel attention module are combined, and the fused feature is then passed to the positional attention network. This integration supplies supplementary information that enriches the key positional cues computed for each spatial cell. The positional attention mechanism operates differently from its channel-oriented counterpart: it begins by splitting the input feature map into two separate pathways along the height and width dimensions, respectively. It then performs average pooling along the vertical axis (p_h) and horizontal axis (p_w) to preserve the spatial structural information of the feature map. It then applies concatenation and convolution operations, followed by dividing the attention feature map and using the Sigmoid activation function to obtain the final output result F_{CPAM}. The calculation process is as follows:

$$p_h(i) = \frac{1}{W}\sum_{0\le j\le W} E(i,j) \tag{5}$$

$$p_w(i) = \frac{1}{H}\sum_{0\le j\le H} E(i,j) \tag{6}$$

$$p(a_h,a_w) = Conv[Concat(p_h,p_w)] \tag{7}$$

$$s_h = Split(a_h) \tag{8}$$

$$s_w = Split(a_w) \tag{9}$$

$$F_{CPAM} = E \times s_h \times s_w \tag{10}$$

where H and W represent the height and width of the input feature map, respectively. $E(i,j)$ denotes the value of the input feature map at position (i,j). $p(a_h,a_w)$ represents the output of the positional attention coordinates. *Conv* denotes a 1×1 convolution kernel. s_h and s_w denote the height and width of the output after segmentation. E denotes the weight matrix for channel attention and positional attention.

Within the ASF-YOLO neck structure, the SSFF module constructs a scale-space representation by applying 3D convolutions over normalized multi-scale feature maps, thereby effectively capturing the cross-scale temporal coherence of frequency-hopping sequences. Concurrently, the TFE module enhances high-resolution spatial details through triple-stream encoding, preserving the minute discriminative features required to differentiate visually similar remote control signals. Subsequently, the CPAM module selectively integrates these complementary cues, consisting of channel-wise discriminative features from the TFE module and spatial contextual information from the SSFF module, yielding a fused representation that is both scale-aware and detail-sensitive.

3.3. A2C2f-SAConv module

Standard convolution has a fixed receptive field size, making it difficult to adaptively adjust based on the bandwidth and dwell time of remote control signals, resulting in poor adaptability to signals with varying time-frequency characteristics. In addressing the constrained receptive field of YOLOv12, this paper proposes the integration of the SAConv module to enhance the A2C2f module. SAConv dilated convolution [33] is a technique that aims to expand the filter field of view by inserting dilated kernels between convolution kernels without increasing the number of parameters or computational complexity. As shown in Figure 8, the structure of the A2C2f-SAConv module is obtained by incorporating the

SAConv module into the innovative A2C2f module from YOLOv12. Compared to the A2C2f module, replacing the ABlock module with the SAConv module enables better capture of multi-scale information about objects by combining the results of convolutions with different dilation rates, thus enhancing object detection performance.

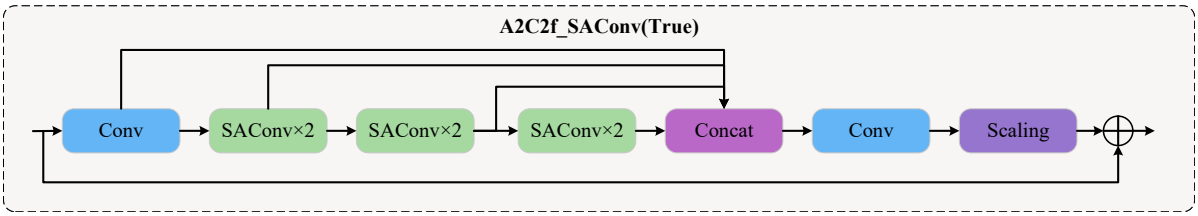

Figure 8. A2C2f-SAConv module structure diagram.

SAConv, based on dilated convolution, effectively enlarges the receptive field of convolutional filters. When the dilation rate is set to r, the filter is expanded by inserting $r-1$ zeros between its adjacent elements. This expands the effective kernel size from $k \times k$ to $k_e \times k_e$, where $k_e = k + (k-1)(r-1)$. The number of parameters and the computational cost remain unchanged. The SAConv module is capable of performing convolution on the same set of input features, utilising divergent dilation rates. Moreover, the employment of switch functions enables the aggregation of results. Switch functions are known to be spatially dependent, each position within the feature map may contain distinct switches which govern the output of SAConv. Consequently, the model is able to adaptively select the appropriate dilated convolution results based on the target scale at varying positions within the image. As demonstrated in Figure 9, the SAConv module is comprised of two constituent elements: the intermediate SAConv component and the global context components that precede and succeed it.

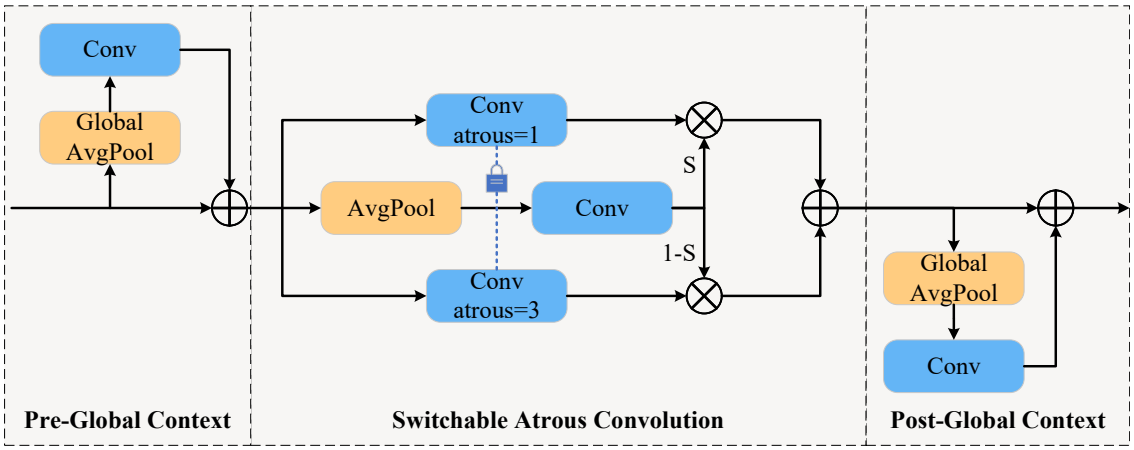

Figure 9. SAConv module structure diagram.

The SAConv component is the core part of the SAConv module, used to convert the convolution layer to SAConv. The conversion formula is:

$$Conv(x,w,1) \rightarrow S(x) \cdot Conv(x,w,1) + (1-S(x)) \cdot Conv(x,w+\Delta w,r) \tag{11}$$

where r denotes the dilation rate of the SAConv module, and Δw stands for the learnable weight. The switch function $S(\cdot)$ outputs either 0 or 1, and is computed by first applying an average pooling layer with a 5×5 kernel, followed by a 1×1 convolution.

Two global context modules are placed by the global context component: one before and one after the SAConv component. The two modules under discussion first compress the input features through a global average pooling layer, then pass through a 1×1 convolution layer, and finally add the output directly to the mainstream.

The SAConv module endows the network with the capability to dynamically switch between different atrous rates, enabling the receptive field to adaptively adjust according to local signal characteristics. In the context of single-channel time-frequency spectrograms, UAV remote control signals exhibit varying scale characteristics. SAConv addresses this by automatically favoring larger atrous rates when processing wideband signal components, thereby capturing global spectral envelopes, while defaulting to smaller atrous rates for narrowband elements to extract precise frequency edges. This content-aware adaptive mechanism mitigates the limitation of fixed receptive fields in the original YOLOv12 architecture, which may struggle to maintain consistent detection performance when handling signals of differing bandwidths.

3.4. PAS-YOLO model

The present paper puts forward a model that has been enhanced in comparison to the original YOLOv12 model. The structural configuration of the enhanced model is illustrated in Figure 10. The following improvements have been made to the model:

- The PPA module is incorporated into the YOLOv12 backbone as a novel enhancement. Its multi-branch design extracts object features at multiple scales, thereby boosting the detection accuracy for small targets.
- The ASF-YOLO neck network structure is introduced to innovatively improve the YOLOv12 neck network. The SSFF module effectively fuses multi-scale features, the TFE module captures fine-grained information of small targets, and the CPAM module's attention mechanism focuses on features related to small targets, thereby improving target detection accuracy.
- The SAConv module is introduced into the A2C2f module to innovatively improve the attention mechanism module A2C2f of YOLOv12. The A2C2f-SAConv module combines the results of convolutions with different hole rates to better capture the multi-scale information of objects, thereby improving object detection performance. Due to the spatial correlation of the switch function, the convolution operation is adaptively modulated by the model according to the target's location and size within the image, aiding in the detection of targets of varying sizes.

The performance gains achieved by PAS-YOLO stem not from the isolated operation of individual modules, but from their coordinated collaboration across different stages of the network. At the backbone level, the PPA module, through its parallel branch design, reduces the risk of weak signal information loss during downsampling, thereby providing a robust feature foundation for subsequent processing. These features are then passed to the neck network, where the ASF-YOLO neck structure performs sequence-aware multi-scale feature fusion. At the deeper stages of feature refinement, the SAConv module embedded within A2C2f is responsible for adaptively adjusting the receptive field to accommodate scale variations among signals of differing bandwidths. In summary, with the PPA module responsible for robust small-object feature extraction, the ASF-YOLO neck structure for multi-scale feature fusion, and SAConv for adaptive receptive field refinement, this three-tiered synergistic mechanism enables PAS-YOLO to achieve higher detection accuracy and classification robustness across diverse UAV model conditions.

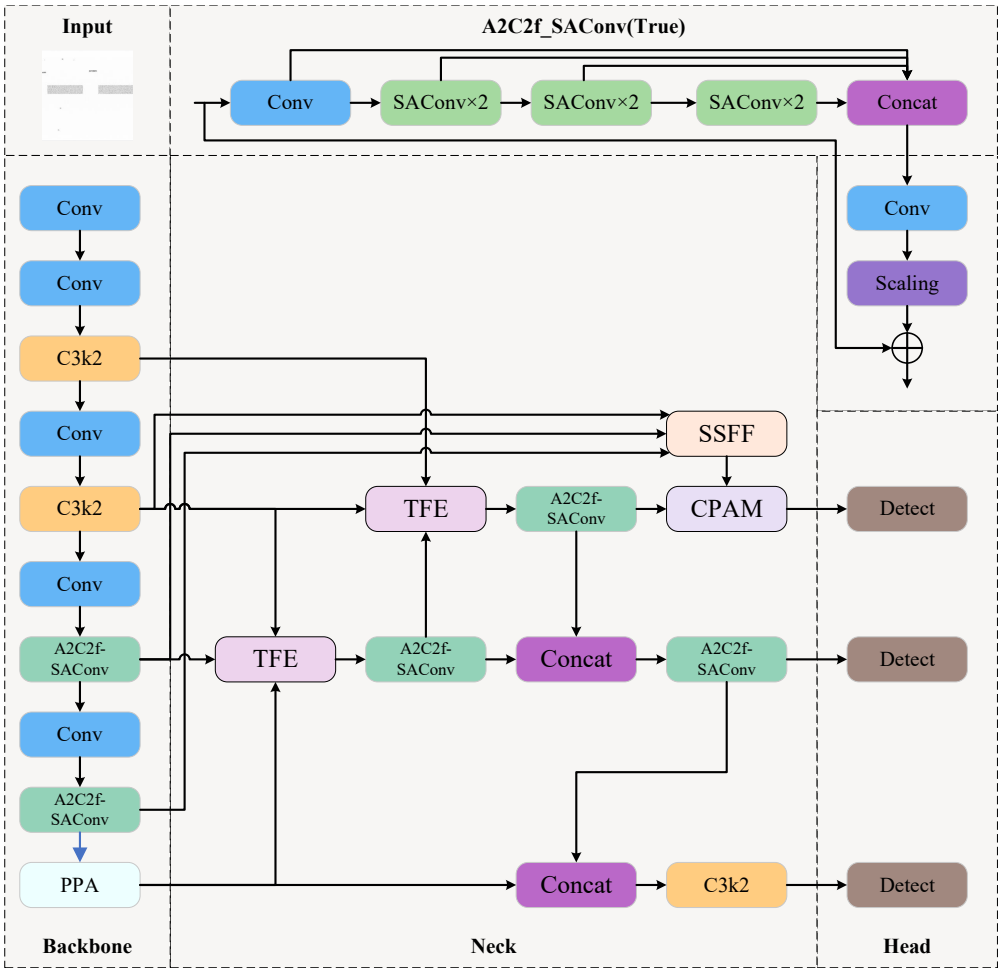

Figure 10. PAS-YOLO module structure diagram.

4. Experiment results

4.1. Dataset and experimental environment

To address the lack of standard datasets for UAV detection tasks, this paper constructs a custom time-frequency spectrum dataset based on the open-source DroneRFa dataset [34]. The DroneRFa dataset was collected in both outdoor and indoor environments, aiming to capture real drone communication signals. Outdoor collection was conducted in the suburbs of Hangzhou, China, with distances between the drone and receiver ranging from 20 to 150 meters. Indoor data acquisition was conducted in a controlled laboratory environment, with the drone remote controller placed approximately 2 meters away from the receiving antenna. The RF signals were recorded by a National Instruments USRP-2955 software-defined radio, which offers a frequency span of 10 MHz to 6 GHz and a maximum I/Q sampling rate of 100 MS/s. The receiver employed a VERT2450 omnidirectional vertical antenna, providing 3 dBi gain and operating across both the 2.4 GHz and 5.8 GHz bands. The captured baseband signals were saved as I/Q samples, and the analog-to-digital converter (ADC) had a 14-bit resolution.

This dataset utilizes the RF signals from 22 UAVs of different brands and models collected in the DroneRFa dataset. Each 0.025s signal segment underwent STFT processing to generate 640×640 grayscale spectrograms. The remote control signals within these images were annotated by model using labelImg, producing corresponding label files. The final dataset comprises 8694 images, with each

model having at least 350 samples. The data were split into training, validation, and test sets at an 8:1:1 ratio. To accommodate grayscale input, the model input channels were adjusted from Red Green Blue (RGB) to a single channel. Selected spectrograms are shown in Figure 11. While the current dataset provides a foundational basis for UAV RF signal recognition, it primarily consists of single-source RF signals without incorporating common real-world electromagnetic interference (e.g., Wi-Fi or Bluetooth signals) or cross-regional signal variations. In future work, we plan to expand the dataset by introducing multi-interference scenarios and cross-domain validation to further assess the model's generalization capability in complex electromagnetic environments.

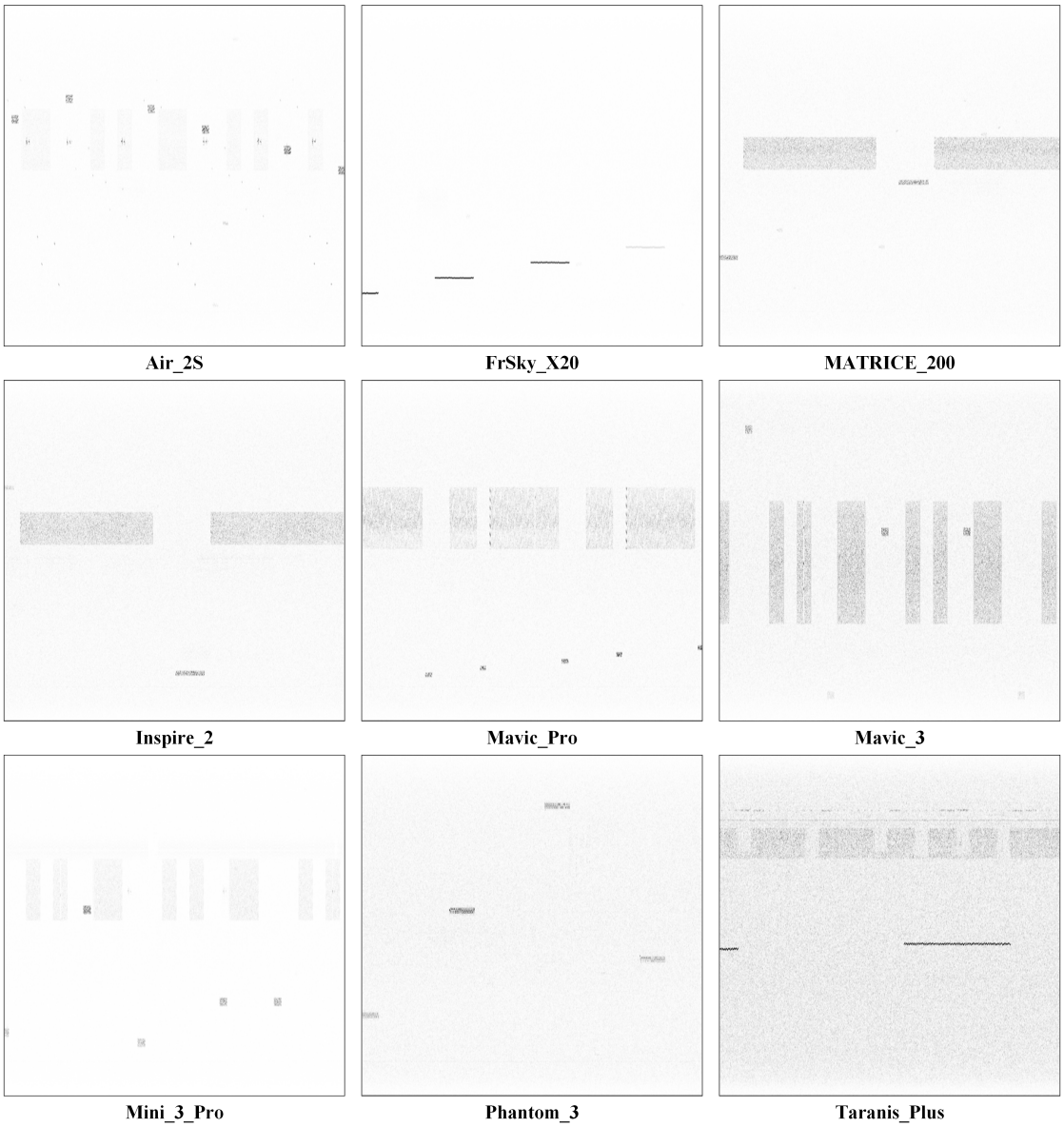

Figure 11. Partial UAV time-frequency spectrum diagram, with rectangular regions in different frequency bands representing the UAV remote control signals.

All experiments were carried out on a workstation running Ubuntu 22.04, equipped with an Intel(R) Xeon(R) Platinum 8474C CPU and an NVIDIA GeForce RTX 4090D GPU. The software environment comprised Python 3.12, PyTorch 2.5.1, and CUDA 12.4. Training was performed for 500 epochs with a batch size of 32.

4.2. Performance metrics

We assess model performance using three standard metrics: precision, recall, and mean average precision (mAP). Precision is the proportion of correctly predicted positive instances out of all positive predictions; recall is the proportion of true objects that are correctly detected; and mAP is the mean of the per-class average precision values. The notation mAP@50 refers to the mAP computed at an IoU threshold of 0.5, whereas mAP@50:95 denotes the average mAP across IoU thresholds from 0.5 to 0.95 in steps of 0.05. The IoU metric measures the intersection-over-union between the predicted and ground-truth bounding boxes. The relevant formulas are provided below:

$$\mathrm{P} = \frac{TP}{TP+FP} \tag{12}$$

$$R = \frac{TP}{TP+FN} \tag{13}$$

$$mAP@50 = \frac{1}{N}\sum_{i=1}^{N} AP_i^{IoU=0.5} \tag{14}$$

$$mAP@50:95 = \frac{1}{10}\sum_{t=5}^{9.5} mAP^{IoU=t/10} \tag{15}$$

$$IoU = \frac{A \bigcap B}{A \bigcup B} \tag{16}$$

where P denotes precision, R denotes recall, and TP, FP, FN stand for True Positive, False Positive, and False Negative, respectively. Specifically, a TP occurs when both the ground truth and the prediction are positive; an FP occurs when the ground truth is negative but the prediction is positive; an FN occurs when the ground truth is negative and the prediction is also negative. The total number of UAV classes is represented by N, while AP gives the average precision for a single class. The term $A \cap B$ is the intersection area between the predicted and ground-truth bounding boxes, and $A \cup B$ is their union area.

4.3. Ablation experiment

This paper presents ablation studies designed to assess the contribution of each improved module to UAV remote control signal detection. The experimental setup was carefully controlled to ensure consistent model parameters across all runs. The corresponding results are reported in Table 1.

The ablation results presented in Table 1 indicate that each of the proposed modules plays an effective role in improving model performance. Among the individual modules, the PPA module shows the most notable improvement. In Experiment 2, it achieves a mAP@50:95 of 74.61%, significantly increasing recall albeit with a slight decrease in precision. The ASF-YOLO neck structure, adopted in Experiment 3, improves precision, recall, and mAP@50:95 comprehensively while only moderately increasing model complexity. Experiment 4 incorporates the A2C2f-SAConv module, which reduces computational cost by 20.6% while still improving accuracy. For two module combinations, the integration of PPA and ASF in Experiment 5 yields the best mAP@50:95 and precision. Meanwhile, the combination of PPA

and A2C2f-SAConv in Experiment 6 stands out in computational efficiency, increasing computational load by only 4.8%. Experiment 7, which combines ASF and A2C2f-SAConv, achieves the highest recall rate of 99.33% with relatively low computational overhead, making it a suitable option for lightweight models. The full three module integration in Experiment 8 leads to the best overall performance, reaching a mAP@50:95 of 75.38%, which is a 3.45% improvement over the baseline. This outcome serves to substantiate the efficacy of the overall architecture.

Table 1. Ablation experiment results of the model.

Experiment	PPA	ASF	A2C2f-SAConv	Par/M	GFLOPs	P/%	R/%	mAP@50/%	mAP@50:95/%
1	×	×	×	2.56	6.3	98.15	98.07	99.13	71.93
2	✓	×	×	4.73	7.9	97.14	99.10	99.24	74.61
3	×	✓	×	2.89	7.7	97.41	98.47	99.25	74.44
4	×	×	✓	3.47	5.0	97.97	98.52	99.34	73.78
5	✓	✓	×	5.06	9.2	98.27	98.42	99.41	74.82
6	✓	×	✓	5.64	6.6	98.35	98.48	99.35	74.88
7	×	✓	✓	3.50	5.7	97.49	99.33	99.32	74.89
8	✓	✓	✓	5.67	7.3	98.38	98.70	99.36	75.38

The ✓ symbol in the table indicates that the module has been added.

To further evaluate the trade-off between model complexity and detection accuracy, we conducted a joint analysis of mAP@50:95 and GFLOPs using the results from Table 1. While the fully integrated model in Experiment 8 achieved the optimal detection accuracy of 75.38%, it incurred a 15.9% increase in computational cost relative to the baseline. Notably, Experiment 7, which combines only the ASF and A2C2f-SAConv modules, demonstrates an optimal accuracy-efficiency balance. This configuration improves mAP@50:95 to 74.89% while maintaining a relatively low computational budget of 5.7 GFLOPs, yielding a superior efficiency ratio compared to other configurations. Furthermore, Experiment 6 achieves a 2.95% gain in mAP@50:95 with a negligible 4.8% increase in GFLOPs, underscoring the critical role of the A2C2f-SAConv module in mitigating computational redundancy and compressing model overhead.

In summary, the full PAS-YOLO architecture is well-suited for ground-station analysis scenarios demanding maximum precision. Conversely, for resource-constrained onboard embedded deployment, the modular combinations represented by Experiments 6 and 7 offer compelling lightweight alternatives. This analysis confirms that the modular design of PAS-YOLO provides flexible scalability in navigating the complexity-performance trade off.

4.4. Comparative experiment

To verify the superiority of our proposed algorithm, we performed a comparative evaluation against several YOLO-based object detectors and RT-DETR models. All model parameters were strictly controlled to ensure a fair comparison. The results are summarized in Table 2.

As shown in the table, the PAS-YOLO improved model proposed in this paper outperforms eight classic YOLO series models as well as two RT-DETR variants in multiple key metrics, particularly achieving breakthrough performance in detection accuracy and overall performance. The mAP@50:95 score is higher than all comparison models, with a relative improvement of 3% compared to YOLOv10n, and the model maintains a lower parameter count and computational cost than the evaluated RT-DETR

models. The algorithm's mAP@50, precision, and recall rates all reach the highest levels, validating its effectiveness. The algorithm has been demonstrated to achieve a detection speed of 224 FPS, thereby maintaining a high level of both speed and accuracy in its operations. Experimental results confirm that the PAS-YOLO algorithm delivers rapid inference while preserving high accuracy, thus achieving an effective speed–accuracy trade-off and attesting to its overall effectiveness.

Table 2. Comparison of experimental results for different models.

Model	Par/M	GFLOPs	P/%	R/%	mAP@50/%	mAP@50:95/%
YOLOv3t	12.1	18.7	90.34	85.12	93.30	64.58
YOLOv5n	2.51	6.9	96.32	97.89	98.79	69.84
YOLOv6n	4.24	11.7	97.35	97.95	98.93	71.64
YOLOv8n	3.01	8.0	97.42	97.93	98.83	71.49
YOLOv9t	1.97	7.6	97.58	98.14	99.07	72.23
YOLOv10n	2.70	8.2	98.01	97.32	98.95	72.39
YOLOv11n	2.59	6.3	98.00	98.05	99.06	71.24
YOLOv12n	2.56	6.3	98.15	98.07	99.13	71.93
RT-DETR-R18	20.0	60.0	76.93	85.23	87.59	56.74
RT-DETR-L	32.0	104.0	87.99	90.56	94.88	61.50
PAS-YOLO	5.67	7.3	98.38	98.70	99.36	75.38

To further evaluate the model's ability to distinguish between similar UAV models, this study presents its confusion matrix on the test set. Figure 12 and Figure 13 show that the PAS-YOLO model achieves high classification accuracy on most UAVs. Although recognition confidence is relatively low for the FrSky X20 and V Bar categories, the model exhibits no cross-category misclassifications, indicating stable category discrimination capabilities. Additionally, some background elements were classified as remote control signals. This primarily stems from incomplete remote control signals in the dataset that were not annotated, yet the model effectively detected such features. This phenomenon further validates the model's robustness in processing diverse remote control signals.

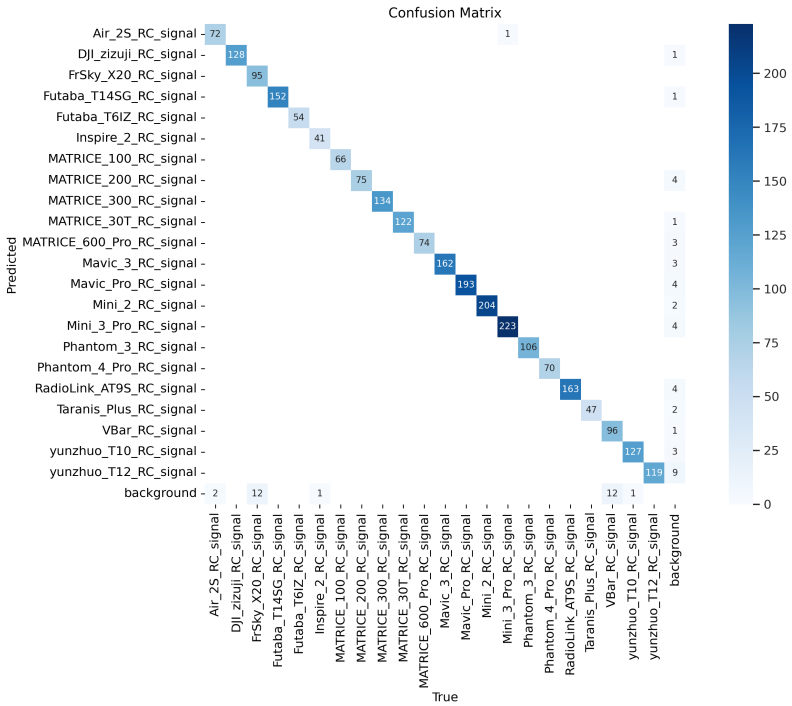

Figure 12. Confusion matrix of PAS-YOLO model.

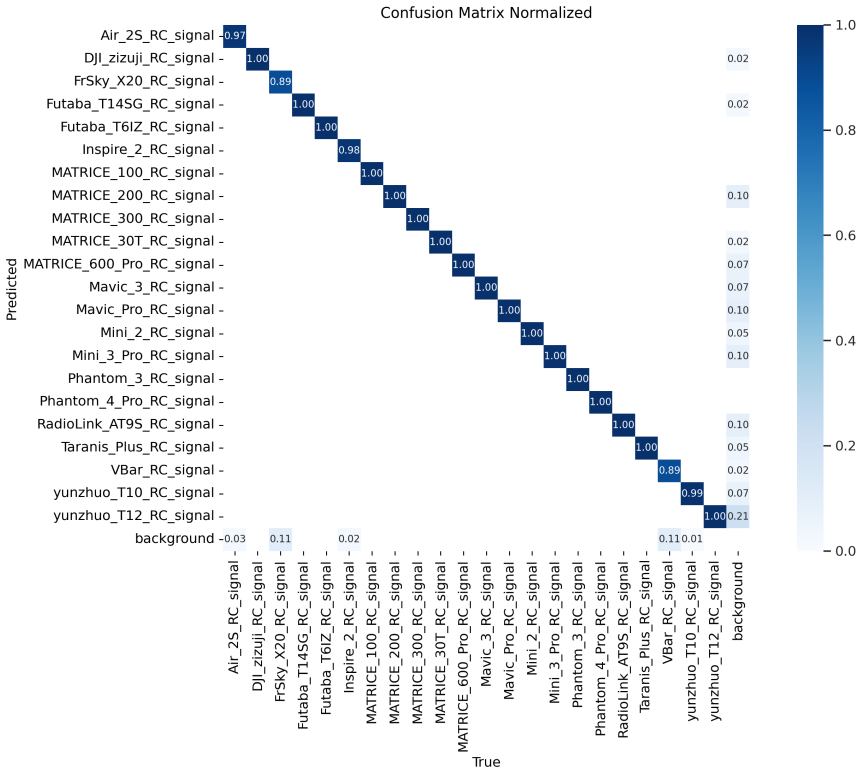

Figure 13. Confusion matrix normalized of PAS-YOLO model.

4.5. Noise resistance evaluation experiment

This section examines the robustness of the UAV time-frequency spectrum detection model based on YOLOv12 in low-SNR regimes, and assesses its applicability to real-world complex electromagnetic interference. A low-SNR test set was built by adding Gaussian white noise to the original dataset, so that the SNR was degraded by 5 dB, 10 dB, 15 dB, and 20 dB. The model's performance under these degraded conditions was then compared to that of the original SNR setting. The time-frequency spectrograms of UAV signals at different SNRs are illustrated in Figure 14.

The performance comparison of the algorithm under different SNR is shown in Table 3.

Experiments show that the algorithm demonstrates high robustness in low SNR UAV time-frequency spectrum detection. When the SNR decreases by 10 dB relative to the initial remote control signal SNR at the receiving end, the mAP@50 decreases by 0.55%, with an accuracy rate of 96.81% and a recall rate of 97.13%. When the SNR decreases by 15 dB, the algorithm still maintains good detection capability, with an accuracy rate of 96.13%, a recall rate of 93.64%, and an mAP@50 of 96.53%, but the mAP@50:95 has dropped to 66.29%. However, when the SNR further decreases by 20 dB, the detection performance shows a significant decline, with accuracy dropping to 90.51%, recall rate decreasing to 83.82%, mAP@50 falling to 86.63%, and mAP@50:95 dropping to 56.39%. These results indicate that the method is suitable for real-time detection scenarios with strong interference, providing a feasible approach for passive detection of UAVs in complex electromagnetic environments. However, detection performance is limited under low SNR conditions.

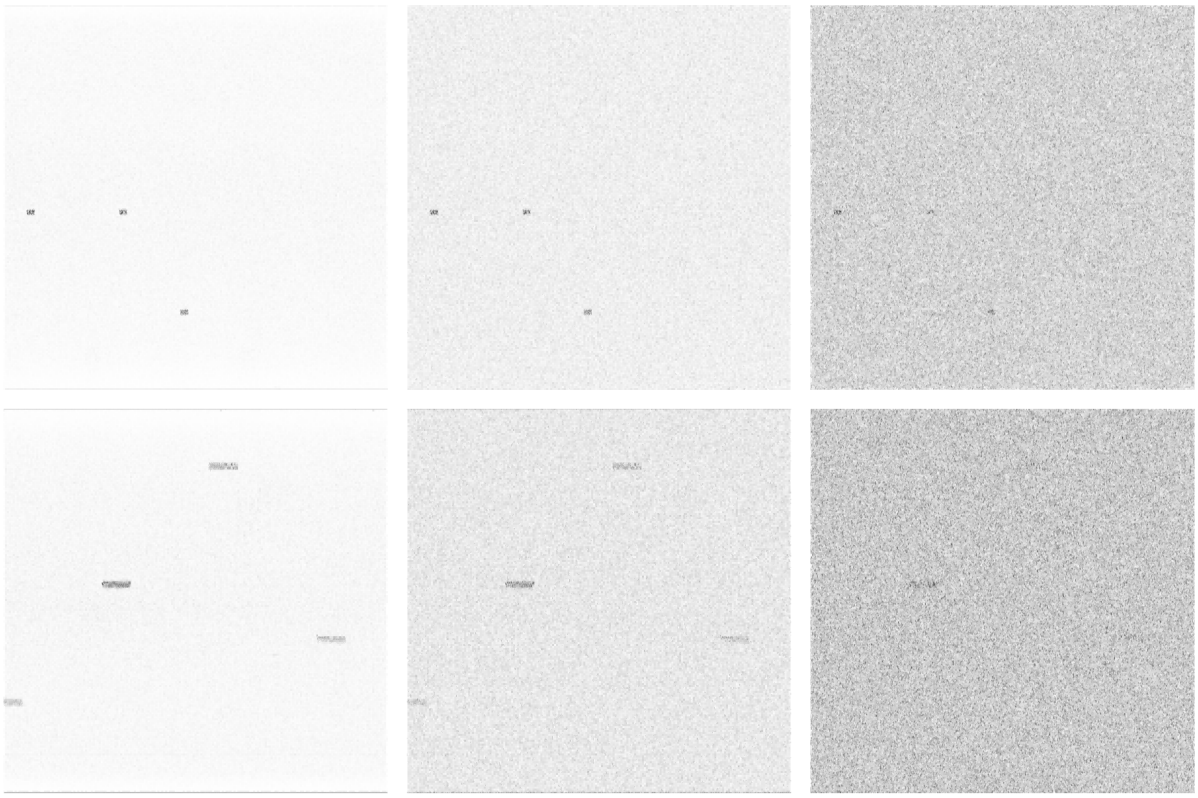

Figure 14. Time-Frequency spectrum diagrams of UAV signals at different SNR. **(a)** Time-frequency spectrum of the UAV signal without added noise; **(b)** Time-frequency spectrum of the UAV signal with a SNR reduced by 10 dB; **(c)** Time-frequency spectrum of the UAV signal with a SNR reduced by 20 dB.

Table 3. Comparison of experimental results under different SNR.

SNR Reduction (dB)	P/%	R/%	mAP@50/%	mAP@50:95/%
0	98.38	98.70	99.36	75.38
5	98.10	98.26	99.32	73.47
10	96.81	97.13	98.81	70.13
15	96.13	93.64	96.53	66.29
20	90.51	83.82	86.63	56.39

4.6. Comparison of PAS-YOLO with baseline model

This study aimed to provide a visual comparison of detection performance between the enhanced PAS-YOLO model and the baseline. Moreover, it was necessary to confirm that PAS-YOLO maintains robust performance in multi-UAV environments. To this end, test images were sampled from the evaluation set to perform comparative experiments, as illustrated in Figure 15. To simulate concurrent signals, composite spectrograms were generated by overlaying the time-frequency spectrograms of two UAVs for comparative testing, as shown in Figure 16. Utilizing detection performance visualization methods, the superiority of the PAS-YOLO model was demonstrated.

In comparative tests between YOLOv12n and PAS-YOLO, PAS-YOLO demonstrated comprehensively enhanced detection performance. Figure 15a demonstrates higher sensitivity for small object detection, particularly in complex scenarios involving multiple signals, such as Figure 15b,d, Figure 16e,h,

PAS-YOLO effectively identifies targets missed by YOLOv12n and improves recognition accuracy for different remote control signal models. When confronted with the complex scenario depicted in Figure 15c, PAS-YOLO successfully identifies targets missed by YOLOv12n and enhances recognition accuracy for various remote control signal models. In Figure 16g,h, PAS-YOLO effectively identifies targets missed by YOLOv12n and improves recognition accuracy for different remote control signal models. When confronted with incomplete signals in Figure 15c, PAS-YOLO maintains stable recognition. In Figure 16f, this model avoids the misclassifications observed in YOLOv12n and successfully detects missed targets. Experiments demonstrate that PAS-YOLO outperforms baseline models in false negative recovery, false positive control, and complex signal adaptation. Notably, it maintains reliable recognition capabilities even in composite spectra with overlapping signals from multiple UAVs, further proving its strong detection adaptability and robustness in multi-UAV scenarios.

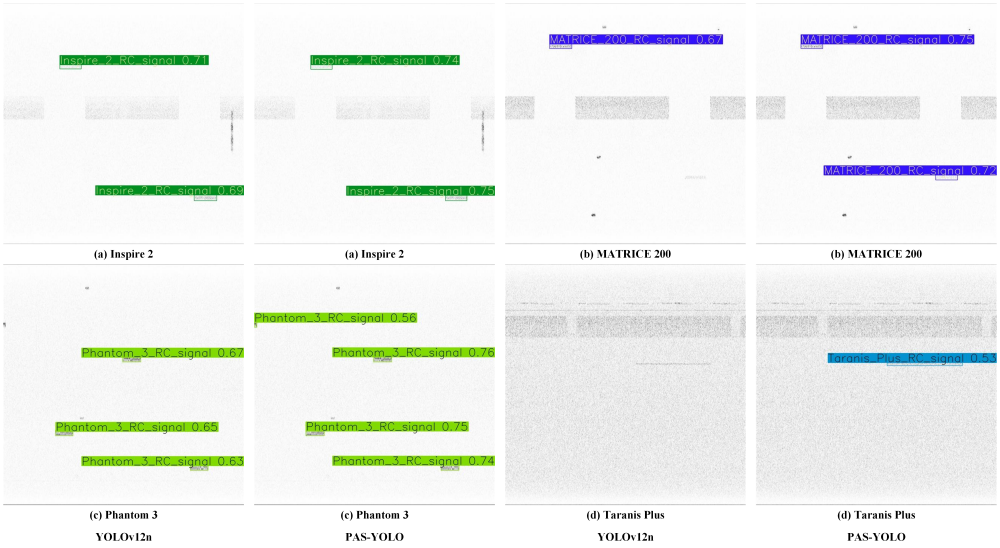

Figure 15. The left and right sides show the detection results for the same single-UAV signal image using YOLOv12n and PAS-YOLO respectively.

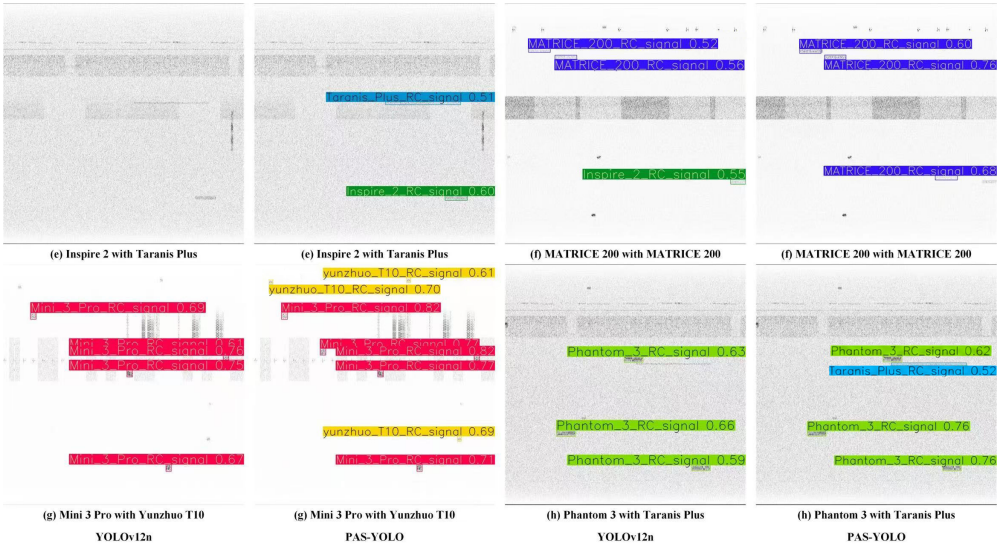

Figure 16. The left and right sides show the detection results for the same composite spectrogram image with overlapping signals from two UAVs using YOLOv12n and PAS-YOLO respectively.

5. Conclusion

This paper addresses the demand for precise detection of UAV remote control signals by proposing PAS-YOLO, a detection algorithm based on an improved YOLOv12. The algorithm incorporates a PPA module into the backbone network to enhance small object detection capabilities, adopts the ASF-YOLO neck structure to fuse multi-scale features, and embeds a SAConv module to expand the receptive field and improve feature extraction. Experiments show that PAS-YOLO attains 99.36% mAP@50 and 75.38% mAP@50:95 across 22 UAV models, outperforming the original model. The model maintains high accuracy even with a 10dB reduction in SNR and achieves an inference speed of 224 FPS. Despite increased model complexity, it achieves a good trade-off between accuracy and speed. The current work employs CIoU for its balanced performance and compatibility with the YOLOv12 framework, future enhancements may incorporate advanced loss functions to further refine localization accuracy in challenging detection environments. On the other hand, while this work focuses on high precision recognition within the RF modality, future systems achieving comprehensive airspace security may require multimodal fusion. The proposed PAS-YOLO model can serve as a key component within a hierarchical fusion framework. Subsequent research will explore integrating this RF classification with complementary modalities such as acoustic and optical sensing. This will enable operation across diverse scenarios, ranging from clear line-of-sight conditions to cluttered urban environments. Future work will focus on incorporating temporal features, expanding the dataset, advancing model lightweighting, and validating practical system deployment. Furthermore, future investigations will incorporate more extensive statistical validation, including repeated trials and confidence interval analysis, to further strengthen the reliability evaluation and methodological soundness of our approach.

Data availability statement

The data or datasets that support the findings of this study are available from the corresponding author upon reasonable request.

Declaration of generative AI and AI-assisted technologies

During the preparation of this manuscript, the authors used generative AI tools only to improve language and readability. Specifically, the authors used DeepSeek and DeepL for language polishing and readability enhancement in limited sections of the manuscript. The authors take full responsibility for the content of the manuscript.

Acknowledgments

This work was supported in part by the Science and Technology Research Project of Henan Province, China, under Grant 242102210210.

Authors' contribution

Kai Zhou: conceptualization, methodology, investigation, software, validation, writing—original and draft preparation; Yanbin Zhang: methodology, writing—reviewing and editing, supervision, funding acquisition. All authors have read and agreed to the published version of the manuscript.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Tauseef M, Reddy TSS, Sravani K. A comprehensive survey on unmanned aerial vehicles (UAVs): types, structural components, communication systems, and operating platforms. In *Proceedings of 2025 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*, Bangalore, India, January 16–17, 2025, pp. 1–6.

[2] Ahmad BI, Rogers C, Harman S, Dale H, Jahangir M, *et al.* A review of automatic classification of drones using radar: key considerations, performance evaluation, and prospects. *IEEE Aerosp. Electron. Syst. Mag.* 2023, 39(2):18–33.

[3] Khan MA, Menouar H, Eldeeb A, Abu-Dayya A, Salim FD. On the detection of unauthorized drones—techniques and future perspectives: a review. *IEEE Sens. J.* 2022, 22(12):11439–11455.

[4] Shi X, Yang C, Xie W, Liang C, Shi Z, *et al.* Anti-drone system with multiple surveillance technologies: architecture, implementation, and challenges. *IEEE Commun. Mag.* 2018, 56(4):68–74.

[5] Chevtchenko SF, Rodriguez BJ, Do Vale RF, Soti A, Bethi Y, *et al.* Drone-based sound source localisation: a systematic literature review. *IEEE Access* 2025, 13:94256–94274.

[6] Coluccia A, Fascista A, Sommer L, Schumann A, Dimou A, *et al.* The drone-vs-bird detection grand challenge at icassp 2023: a review of methods and results. *IEEE Open J. Signal Process.* 2024, 5:766–779.

[7] Ahmad BI, Rogers C, Harman S, Dale H, Jahangir M, *et al.* A review of automatic classification of drones using radar: key considerations, performance evaluation, and prospects. *IEEE Aerosp. Electron. Syst. Mag.* 2023, 39(2):18–33.

[8] Jurn YN, Ibraheem ZT, Zaki ND. A review of RF based drone detection, direction of arrival and identification techniques. In *Proceedings of 2024 IEEE 14th International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang, Malaysia, August 23–24, 2024, pp. 162–167.

[9] Li Q, Wang F, Zhang Y, Li R, Shi S, *et al.* Novel micro-UAV identification approach assisted by combined RF fingerprint. *IEEE Sens. J.* 2024, 24(16):26802–26813.

[10] Al-Emadi S, Al-Senaid F. Drone detection approach based on radio-frequency using convolutional neural network. In *Proceedings of 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, Doha, Qatar, February 2–5, 2020, pp. 29–34.

[11] Misbah M, Dil M, Khalid W, Kaleem Z. RF-NeuralNet: lightweight deep learning framework for

detecting rogue drones from radio frequency signatures. In *Proceedings of 2023 7th International Conference on Automation, Control and Robots (ICACR)*, Shanghai, China, August 4–6, 2023, pp. 163–167.

[12] Huynh-The T, Pham QV, Nguyen TV, Da Costa DB, Kim DS. RF-UAVNet: high-performance convolutional network for RF-based drone surveillance systems. *IEEE Access* 2022, 10:49696–49707.

[13] Bremnes K, Moen R, Yeduri SR, Yakkati RR, Cenkeramaddi LR. Classification of UAVs utilizing fixed boundary empirical wavelet sub-bands of RF fingerprints and deep convolutional neural network. *IEEE Sens. J.* 2022, 22(21):21248–21256.

[14] Kılıç R, Kumbasar N, Oral EA, Ozbek IY. Drone classification using RF signal based spectral features. *Eng. Sci. Technol. Int. J.* 2022, 28:101028.

[15] Wang Q, Yang P, Yan X, Wu HC, He L. Radio frequency-based UAV sensing using novel hybrid lightweight learning network. *IEEE Sens. J.* 2024, 24(4):4841–4850.

[16] Mototolea D, Youssef R, Radoi E, Nicolaescu I. Non-cooperative low-complexity detection approach for FHSS-GFSK drone control signals. *IEEE Open J. Commun. Soc.* 2020, 1:401–412.

[17] Xue Y, Chang Y, Zhang Y, Sun J, Ji Z, *et al.* UAV signal recognition of heterogeneous integrated KNN based on genetic algorithm. *Telecommun. Syst.* 2024, 85(4):591–599.

[18] Wang S, Luo Y, Zheng Y, Sun Z, Zheng Y, *et al.* Detection and recognition of UAV radio frequency signals based on time–frequency processing and transfer learning with multi-channel input. *Signal Image Video Process.* 2025, 19(10):868.

[19] Li M, Hao D, Wang J, Wang S, Zhong Z, *et al.* Intelligent identification and classification of small UAV remote control signals based on improved Yolov5-7.0. *IEEE Access* 2024, 12:41688–41703.

[20] Mohammed KK, Abd El-Latif EI, El-Sayad NE, Darwish A, Hassanien AE. Radio frequency fingerprint-based drone identification and classification using Mel spectrograms and pre-trained YAMNet neural. *Internet Things* 2023, 23:100879.

[21] Kaplan B, Kahraman I, Ekti A, Yarkan S, Gorcin A, *et al.* Detection, identification, and direction of arrival estimation of drone FHSS signals with uniform linear antenna array. *IEEE Access* 2021, 9:152057–152069.

[22] Pärlin K, Riihonen T, Karm G, Turunen M. Jamming and classification of drones using full-duplex radios and deep learning. In *Proceedings of 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, London, UK, August 31–September 3, 2020, pp. 1–5.

[23] Glüge S, Nyfeler M, Aghaebrahimian A, Ramagnano N, Schüpbach C. Robust low-cost drone detection and classification using convolutional neural networks in low SNR environments. *IEEE J. Radio Freq. Identif.* 2024, 8:821–830.

[24] Noh DI, Jeong SG, Hoang HT, Pham QV, Huynh-The T, *et al.* Signal preprocessing technique with noise-tolerant for RF-based UAV signal classification. *IEEE Access* 2022, 10:134785–134798.

[25] Basak S, Rajendran S, Pollin S, Scheers B. Drone classification from RF fingerprints using deep residual nets. In *Proceedings of 2021 International Conference on COMmunication Systems & NETworkS (COMSNETS)*, Bangalore, India, January 5–9, 2021, pp. 548–555.

[26] Xu C, Chen B, Liu Y, He F, Song H. RF fingerprint measurement for detecting multiple amateur

drones based on STFT and feature reduction. In *Proceedings of 2020 Integrated Communications Navigation and Surveillance Conference (ICNS)*, Herndon, USA, September 8–10, 2020.

[27] Mandal S, Satija U. Time–frequency multiscale convolutional neural network for RF-based drone detection and identification. *IEEE Sens. Lett.* 2023, 7(7):1–4.

[28] Mo Y, Huang J, Qian G. Deep learning approach to UAV detection and classification by using compressively sensed RF signal. *Sensors* 2022, 22(8):3072.

[29] Ezuma M, Erden F, Anjinappa CK, Ozdemir O, Guvenc I. Detection and classification of UAVs using RF fingerprints in the presence of Wi-Fi and Bluetooth interference. *IEEE Open J. Commun. Soc.* 2019, 1:60–76.

[30] Medaiyese OO, Ezuma M, Lauf AP, Adeniran AA. Hierarchical learning framework for UAV detection and identification. *IEEE J. Radio Freq. Identif.* 2022, 6:176–188.

[31] Xu S, Zheng S, Xu W, Xu R, Wang C, *et al.* Hcf-net: hierarchical context fusion network for infrared small object detection. In *Proceedings of 2024 IEEE International Conference on Multimedia and Expo (ICME), Niagara Falls*, Niagara Falls, Canada, July 15–19, 2024, pp. 1–6.

[32] Kang M, Ting CM, Ting FF, Phan RCW. ASF-YOLO: a novel YOLO model with attentional scale sequence fusion for cell instance segmentation. *Image Vision Comput.* 2024, 147:105057.

[33] Qiao S, Chen LC, Yuille A. DetectoRS: detecting objects with recursive feature pyramid and switchable atrous convolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Nashville, USA, June 20–25, 2021, pp. 10213–10224.

[34] Yu N, Mao S, Zhou C, Sun G, Shi Z, *et al.* DroneRFa: a large-scale dataset of drone radio frequency signals for detecting low-altitude drones. *J. Electron. Inf. Technol.* 2023, 45(11):1–9.