



ARTICLE

# LRT-BF: A Lightweight and Robust Blind Beamforming Method for High-Dynamic UAV Communications

Zheng Xu<sup>1,2</sup>, Zihao Pan<sup>1</sup>, Ning Yang<sup>1</sup> and Daoxing Guo<sup>1,\*</sup>

<sup>1</sup>College of Communications Engineering, Army Engineering University of PLA, Nanjing, China

<sup>2</sup>Nanjing Panda Handa Technology Co., Ltd., Nanjing, China

\*Corresponding Author: Daoxing Guo. Email: xyzgfg@sina.com

Received: 12 February 2026; Accepted: 16 April 2026; Published: 15 June 2026

**ABSTRACT:** Unmanned Aerial Vehicle (UAV) communications in complex electromagnetic environments face challenges such as strong interference, high dynamic Doppler shifts, and limited onboard computing power. In these scenarios, traditional blind beamforming algorithms suffer from slow convergence and difficulty in handling Gaussian-like signals (e.g., Orthogonal Frequency Division Multiplexing (OFDM)). To address these issues, this paper proposes a Lightweight Robust Transfer learning-based Blind Beam Forming method (LRT-BF). This method constructs a self-supervised optimization framework centered on a pre-trained signal classifier and innovatively introduces a joint loss function combining classification confidence guidance with output power minimization, achieving fully blind interference suppression without requiring Direction of Arrival (DOA) priors. To address the high dynamic characteristics of UAVs, a Frequency Domain Randomization (FDR) augmentation strategy is introduced, endowing the feature extractor with Doppler-invariant perception capabilities under frequency offsets of  $\pm 5$  kHz. By reconstructing the network backbone using Depthwise Separable Convolutions (DSC), a computational reduction of  $8.3\times$  and parameter reduction of  $7.0\times$  are achieved with negligible accuracy loss (retaining up to 99.8% accuracy). Furthermore, by incorporating Temperature Scaling mechanisms and signal subspace initialization, the problems of gradient saturation and convergence stagnation in few-snapshot scenarios are effectively resolved. Simulation results demonstrate that under conditions of extremely few snapshots ( $L < 64$ ) and strong interference, the average interference suppression depth of LRT-BF reaches  $-41.2$  dB, an improvement of over 48 dB compared to traditional Fast Independent Component Analysis (ICA) and Constant Modulus Algorithm (CMA) algorithms. Its Central Processing Unit (CPU) inference latency is only 1.64 ms, achieving a  $4.6\times$  real-time acceleration. Beyond these theoretical metrics, these hardware-efficient characteristics confirm the immense potential of LRT-BF for practical implementation, providing a highly feasible, low-latency anti-jamming solution for SWaP-constrained UAV swarms and edge nodes.

**KEYWORDS:** UAV communication; blind beamforming; transfer learning; lightweight network; Doppler shift

## 1 Introduction

With the vigorous development of Unmanned Aerial Vehicle (UAV) technology, its application in emergency communications, tactical reconnaissance, and 5G/6G heterogeneous networks has become a research hotspot [1]. However, UAV communication links typically operate in highly complex electromagnetic environments. Due to the openness and density of spectrum resources, legitimate communication signals are extremely susceptible to strong Co-channel Interference and malicious suppression jamming [2,3]. To maintain link reliability under severe Signal-to-Interference-plus-Noise Ratio (SINR) conditions, Adaptive Beamforming technology has become a core means of ensuring UAV communication by utilizing Spatial

Filtering to form high-gain main lobes in the desired direction and deep Nulls in the direction of interference. Nevertheless, achieving robust beamforming in highly dynamic UAV scenarios faces severe challenges. Traditional non-blind beamforming methods, such as Zero-Forcing Beamforming (ZFBB) or Minimum Mean Square Error (MMSE) algorithms, rely heavily on precise Channel State Information (CSI) or array Steering Vectors [4]. However, in actual flight, high-speed movement of the airframe, minute mechanical vibrations, and non-ideal calibration of the antenna array can lead to severe Manifold Mismatch. These non-ideal characteristics cause methods based on Direction of Arrival (DOA) estimation to suffer drastic performance degradation during actual deployment.

Blind beamforming has garnered significant attention due to its independence from prior knowledge of steering vectors. Traditional blind processing algorithms are primarily based on statistical properties; for instance, Sample Matrix Inversion (SMI) utilizes second-order statistics, while Independent Component Analysis (ICA) exploits higher-order cumulants [5]. However, these algorithms face a distinct “convergence bottleneck” in UAV scenarios. On one hand, the extremely short coherence time of UAV channels demands rapid algorithm convergence with very few snapshots. On the other hand, Orthogonal Frequency Division Multiplexing (OFDM) signals, widely adopted in modern communications, are statistically close to a Gaussian distribution, rendering algorithms like ICA, which rely on maximizing non-Gaussianity, ineffective [6]. Furthermore, the kHz-level Doppler shift introduced by high-speed motion disrupts signal stationarity within the observation window, further exacerbating algorithmic instability. In recent years, Deep Learning-driven physical layer schemes have provided new avenues for non-linear, high-dynamic signal processing. Although existing studies, such as Attention-based Beamforming (AttBF), can significantly enhance performance [7], their complex parameter scale and computational overhead fundamentally conflict with the restricted Size, Weight, and Power (SWaP) constraints of UAVs.

Consequently, securing reliable UAV links hinges on overcoming two fundamental bottlenecks: the algorithm must be Robust enough to withstand severe Doppler shifts and manifold mismatches caused by high-dynamic mobility, while simultaneously being Lightweight enough to execute in real-time within the strict Size, Weight, and Power (SWaP) constraints of airborne embedded units. Addressing these two specific pillars, this paper proposes a Lightweight Robust Blind Beamforming method (LRT-BF) based on improved Classification-Based Transfer Learning (CBTL) [6].

This method departs from the heavy reliance of traditional adaptive filtering algorithms on precise Channel State Information (CSI) or the stationarity of Second-Order Statistics (SOS), turning instead to a transfer learning paradigm to mine the intrinsic modulation features of the signal. The core logic of LRT-BF lies in transforming a pre-trained high-performance signal classifier into a “proxy evaluator” for spatial weights. Specifically, this mechanism does not directly estimate the physical channel but uses the classifier’s recognition confidence for specific modulation formats (e.g., Quadrature Phase Shift Keying (QPSK)) as a “proxy metric” to evaluate the quality of the beamforming output. By constructing a closed-loop self-supervised optimization framework, the beam output signal is fed into a lightweight network with frozen parameters. Using the automatic differentiation technology of deep learning frameworks, the classification error (negative log-likelihood) is backpropagated along the computational graph to the complex weight layer. This strategy ingeniously circumvents the difficulty of real-time calibration of array manifolds in UAV communications. To make this framework engineering-feasible in the unique high-dynamic and compute-constrained environment of UAVs, this paper performs deep optimization of the basic CBTL architecture from three key dimensions: First, by introducing a frequency domain randomization augmentation strategy, the network is forced to learn phase transition features rather than absolute phase trajectories, thereby endowing the “proxy evaluator” with robustness in 5 kHz high-dynamic Doppler environments [6,8]; second, by utilizing Depthwise Separable Convolution (DSC) to reconstruct the network backbone [9],

the parameter count and computational load are compressed to less than 15% of the original model, ensuring that beam weights can be updated in real-time at the millisecond level on airborne embedded platforms; finally, addressing the convergence stagnation caused by extremely few snapshots in short-burst communications, a Temperature Scaling mechanism [10] is introduced to smooth the confidence distribution, preventing weights from falling into local optima due to premature saturation by maintaining effective gradient magnitudes.

The main contributions of this paper are summarized as follows:

1. **Robust** Feature Extraction with Doppler Invariance: To address the issue of rapid phase rotation caused by high frequency offsets in UAVs, a pre-training strategy based on Frequency Domain Randomization is proposed. By forcing the feature extraction network to ignore the absolute carrier frequency values and focus on modulation structural features during the offline phase, this approach significantly enhances the quality of gradient backpropagation from the “proxy evaluator” in highly dynamic environments, solving the bottleneck of sensitivity to frequency offsets inherent in traditional methods.
2. **Lightweight** Architecture for Edge Deployment: The backbone of the feature extraction network is reconstructed using Depthwise Separable Convolution (DSC). Experiments demonstrate that this architecture achieves approximately an  $8.3\times$  compression in computational complexity (FLOPs) with negligible loss in modulation recognition accuracy. This lightweight design reduces Central Processing Unit (CPU) inference latency to under 2 ms, perfectly adapting to the strict SWaP constraints of UAVs.
3. **Fast Convergence Optimization Mechanism under Few Snapshots:** A Temperature Scaling mechanism [10] is introduced to smooth the Softmax confidence distribution, effectively mitigating the Gradient Saturation phenomenon under few-snapshot conditions. Combined with an MVDR intelligent initialization strategy based on sample covariance, the algorithm achieves deep convergence within fewer than 64 snapshots, significantly outperforming traditional ICA and Constant Modulus Algorithm (CMA) algorithms in terms of interference suppression depth.

Furthermore, to promote research in the field of blind processing for UAV communications and ensure the reproducibility of our results, we have open-sourced the complete implementation code of the LRT-BF algorithm, alongside the datasets covering various interference scenarios and Doppler levels<sup>1</sup>.

The remainder of this paper is organized as follows: In Section 2, we first construct the signal model for high-dynamic UAV communication systems, providing an in-depth analysis of the physical layer non-idealities caused by unknown array manifold errors and time-varying Doppler shifts, while systematically discussing the inherent limitations of traditional blind processing algorithms in “data-scarce” scenarios. The subsequent Section 3 details the proposed LRT-BF architecture and its theoretical foundation, highlighting how on-device computational decoupling is achieved via Depthwise Separable Convolutions (DSC) and how the gradient flow during the self-supervised iteration process is optimized using a Temperature Scaling mechanism. Sections 4 and 5 introduce the simulation experimental design and provide a comprehensive quantitative evaluation and comparative discussion of the experimental results across multiple dimensions, including anti-Doppler robustness, convergence efficiency with few snapshots, and real-time inference latency. Section 6 systematically identifies the potential limitations of the LRT-BF algorithm regarding simulation channel abstraction, narrowband communication assumptions, and generalization capabilities across specific modulation formats. Furthermore, it deeply explores the validity boundaries of using classification confidence as a surrogate metric for spatial optimization and discusses the potential inference latency fluctuations when deployed on embedded hardware with extremely limited computing power. Section 7 systematically reviews the evolutionary history from classical high-order statistical blind source separation

---

<sup>1</sup><https://github.com/BlindBeamforming/uav>

to modern supervised learning beamforming, analyzing the limitations of traditional algorithms under the constraints of OFDM signal Gaussianity and high computational demands. It further examines the Doppler sensitivity and computational bottlenecks of existing classification transfer learning architectures in high-dynamic UAV scenarios, thereby clarifying the necessity of introducing frequency-domain enhancement and lightweight reconstruction in LRT-BF. Finally, [Section 8](#) briefly summarizes that the LRT-BF method significantly improves the anti-interference performance of UAV communications in scenarios with few snapshots and high frequency offsets, achieving efficient real-time inference at the edge. Future research will focus on validating the algorithm on physical platforms and exploring its application potential in broadband communication and multi-UAV collaborative scenarios.

## 2 Background and Challenges

### 2.1 UAV Communication Array Receiving Model

Consider a UAV receiving terminal equipped with  $N$  antenna elements operating in a complex electromagnetic environment. Assume that at time  $t$ , the array receives signals from one desired signal source (Target UAV/Ground Station) and  $K$  non-cooperative interference sources. Under the narrowband assumption, the received baseband signal vector  $\mathbf{y}(t) \in \mathbb{C}^{N \times 1}$  can be expressed as:

$$\mathbf{y}(t) = \mathbf{h}_d(t)s_d(t) + \sum_{k=1}^K \mathbf{h}_{i,k}(t)s_{i,k}(t) + \mathbf{n}(t) \quad (1)$$

where  $s_d(t)$  and  $s_{i,k}(t)$  denote the complex baseband waveforms of the desired signal and the  $k$ -th interference signal, respectively, satisfying the normalized power assumption  $E[|s|^2] = 1$ .  $\mathbf{n}(t) \sim \mathcal{CN}(\mathbf{0}, \sigma_n^2 \mathbf{I})$  represents the additive white Gaussian noise (AWGN) at the receiver.

Unlike communication with fixed ground base stations or geostationary satellites, UAV communication is characterized by significant **\*\* high dynamics \*\*** and **\*\* limited payload \*\*** constraints. Consequently, the channel vector  $\mathbf{h} \in \mathbb{C}^{N \times 1}$  (including the desired channel  $\mathbf{h}_d$  and interference channels  $\mathbf{h}_{i,k}$ ) no longer adheres to the ideal static array manifold model but instead exhibits two severe non-ideal characteristics:

1) **Unknown Array Manifold Errors:** Constrained by the size, weight, power, and cost (SWaP-C) of the UAV, airborne antennas are often difficult to calibrate with high precision on the ground. Furthermore, mechanical micro-vibrations and flexible deformations of the UAV airframe during flight cause deviations between the actual and ideal positions of antenna elements. Therefore, the actual steering vector  $\mathbf{h}$  cannot be simply determined by the direction of arrival (DOA)  $\theta$ , but incorporates unknown gain and phase errors [7]:

$$\mathbf{h}(\theta) = \mathbf{\Psi} \odot \mathbf{v}(\theta) \quad (2)$$

where  $\mathbf{v}(\theta)$  represents the ideal steering vector for a uniform linear array (ULA) or uniform planar array (UPA), and  $\odot$  denotes the Hadamard Product.  $\mathbf{\Psi} \in \mathbb{C}^{N \times 1}$  is the unknown array error vector, with its element  $[\mathbf{\Psi}]_n = \alpha_n e^{j\phi_n}$  encapsulating the gain uncertainty  $\alpha_n$  and phase center drift  $\phi_n$  of the  $n$ -th element. The presence of such errors causes a drastic performance degradation in traditional beamforming algorithms based on geometric priors (such as Multiple Signal Classification (MUSIC) and Minimum Variance Distortionless Response (MVDR)).

2) **Time-Varying Doppler Shift:** The high-speed three-dimensional motion of UAVs introduces significant Doppler effects. Unlike Wentz et al. [6], who only considered minor frequency offsets ( $\leq 5$  ppm) caused by static device clock drift in the original CBTL study, the relative motion velocity  $v$  in UAV scenarios generates carrier frequency offsets (CFO)  $f_d = \frac{v}{c} f_c$  reaching the kHz level [8]. Consequently, the actual

waveform in the received signal model includes a phase term that rotates rapidly over time:

$$s(t) = x(t - t_d) e^{j(2\pi f_d t + \varphi_0)} \quad (3)$$

where  $x(t)$  is the transmitted symbol,  $t_d$  is the propagation delay, and  $\varphi_0$  is the initial phase. For blind beamforming algorithms, this fast time-varying phase rotation destroys the stationarity of the signal across snapshots, necessitating algorithms with extremely fast convergence speeds or built-in robustness to frequency offsets.

## 2.2 Limitations of Traditional Blind Beamforming

Traditional blind beamforming algorithms are primarily categorized into Sample Matrix Inversion (SMI)-based methods utilizing second-order statistics [11] and Blind Source Separation (BSS)-based methods utilizing higher-order statistics [12]. Although these methods perform well in static scenarios, they face severe theoretical and engineering bottlenecks in high-dynamic UAV communications employing complex modulations [13].

The core of the SMI algorithm lies in utilizing a limited number of received signal samples (snapshots) to estimate the array covariance matrix  $\hat{\mathbf{R}} = \frac{1}{L} \sum_{l=1}^L \mathbf{y}[l] \mathbf{y}^H[l]$  and inverting it to obtain the beamforming weights. Theoretically, to ensure the non-singularity of  $\hat{\mathbf{R}}$  and the stability of beamforming, the number of snapshots  $L$  must be at least twice the number of array elements  $N$  ( $L \geq 2N$ ). Furthermore, to achieve a Signal-to-Interference-plus-Noise Ratio (SINR) close to the optimum, thousands of stationary snapshots are typically required [7]. In scenarios where the high-speed movement of UAVs results in extremely short channel coherence times (on the order of milliseconds), the receiver cannot collect sufficient stationary samples before the channel changes. SMI involves matrix inversion operations with a computational complexity of  $O(M^3)$ . For large-scale arrays or UAV terminals requiring frequent weight updates, such intensive matrix computations introduce unacceptable processing latency and power consumption [7].

Mainstream blind source separation algorithms, such as Joint Approximate Diagonalization of Eigenmatrices (JADE) [14] and Fast Independent Component Analysis (FastICA) [15], are mathematically based on the inverse process of the Central Limit Theorem. They separate independent signal sources by maximizing the non-Gaussianity (such as kurtosis or negentropy) of the received signals. However, modern UAV communications widely adopt Orthogonal Frequency Division Multiplexing (OFDM) technology. An OFDM signal consists of the superposition of numerous subcarriers, and according to the Central Limit Theorem, its time-domain waveform statistically follows or highly approximates a Gaussian distribution [16]. Experiments indicate that when both the Signal of Interest (SOI) and interference signals exhibit Gaussian-like distributions, JADE and FastICA fail to distinguish between signal and noise using high-order statistical features, leading to a complete failure of beamforming [6] and an inability to extract the target signal in such scenarios [8].

## 2.3 Core Challenges in UAV Scenarios

Compared to terrestrial cellular networks or geostationary satellite communications, the UAV communication environment is characterized by extreme dynamism and limited resources. Applying existing blind beamforming algorithms directly to this context introduces three core “new problems,” which constitute the primary technical challenges addressed in this paper:

1. **High Dynamic CFO/Doppler:** UAV communication links are jointly affected by two factors: first, low-cost onboard communication equipment is typically equipped with local oscillators of poor stability, leading to inherent Carrier Frequency Offset (CFO); second, the high-speed 3D movement of UAVs

generates significant Doppler shifts [8]. In the mathematical model, this frequency offset manifests as a rapid rotation of the received signal phase over time, denoted as  $e^{j2\pi\Delta f t}$ . Traditional blind beamforming algorithms (such as JADE or subspace methods based on second-order statistics) usually assume that the signal is quasi-static or phase-synchronized within the processing window (snapshot block). However, in UAV scenarios, kHz-level frequency offsets rapidly destroy signal phase consistency, causing the beam direction to deviate or fail completely [7]. Furthermore, this phase rotation is difficult to eliminate through statistical averaging when the number of snapshots is small, making it impossible for the beamformer to accurately lock onto the target signal features.

2. **Limited Snapshots & Short Coherence Time:** The high-speed maneuvering of UAVs results in an extremely short channel coherence time. To ensure the timeliness of beam weights, the algorithm must complete the calculation and update of weights before the channel changes significantly. This implies that the number of stationary sampling points (snapshots  $L$ ) available to the receiver is extremely limited. Literature indicates that traditional adaptive algorithms (such as SMI) typically require thousands of samples to accurately estimate the covariance matrix for ideal interference suppression, which is unrealistic in “data-starved” UAV short-burst communications [6]. When the number of available snapshots  $L$  is close to the number of array elements  $N$  ( $L \approx N$ ), the estimation error of the covariance matrix increases dramatically, causing the Signal-to-Interference-plus-Noise Ratio (SINR) performance of traditional methods to drop sharply, failing to meet high-reliability communication requirements [8].
3. **SWaP Constraint & Real-time Inference:** UAV platforms are strictly constrained by Size, Weight, and Power (SWaP). The computing power of their accompanying onboard computing units (such as embedded Graphics Processing Units (GPUs) or Field-Programmable Gate Arrays (FPGAs)) is far lower than that of ground base station servers. Existing high-performance blind beamforming algorithms are often accompanied by high computational complexity. For example, traditional algorithms based on eigenvalue decomposition or matrix inversion have a complexity as high as  $O(N^3)$  [7]; while unoptimized deep learning models (such as the original CBTL or large Convolutional Neural Networks (CNNs)), despite faster inference speeds, still struggle to complete real-time processing within a millisecond-level time window due to their massive parameter counts and Floating Point Operations (FLOPs) [8]. How to achieve lightweight, low-latency, real-time blind beamforming on resource-constrained embedded devices is a key bottleneck for engineering implementation.

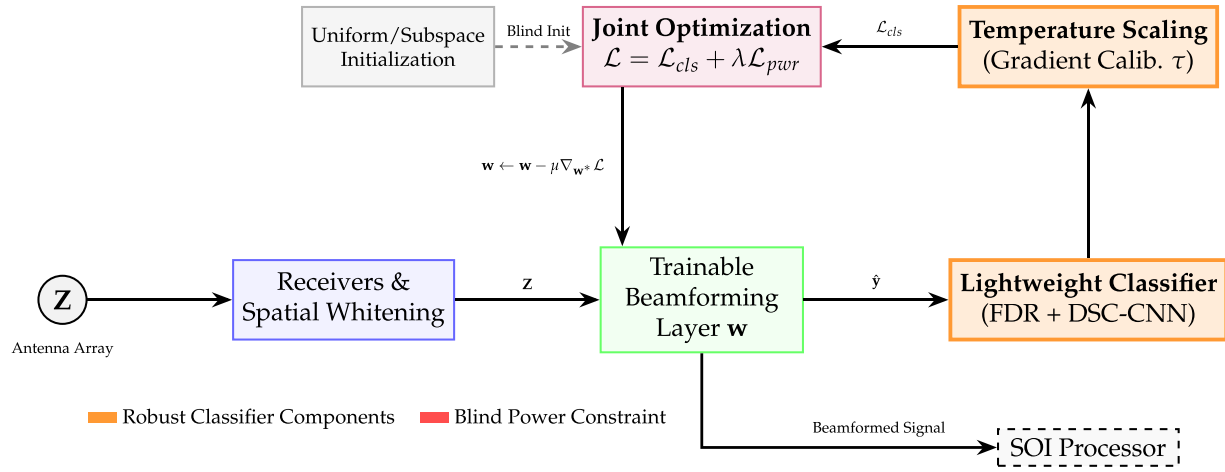
**Bridging the Gap: The Need for a Lightweight and Robust Approach.** To overcome the aforementioned limitations of traditional blind beamforming and direct deep learning adaptations, it is imperative to develop a tailored architecture for UAVs. The proposed LRT-BF model is specifically designed to bridge this gap through two core pillars: Robustness and Lightweighting. First, to combat the high dynamic Doppler shifts and limited snapshot constraints (Challenges 1 & 2), LRT-BF introduces a Robust feature extraction mechanism powered by Frequency Domain Randomization (FDR). This ensures stable gradient guidance for the beamformer even when traditional statistical metrics fail. Second, to address the strict SWaP constraints and real-time bottlenecks (Challenge 3), LRT-BF employs a Lightweight Depthwise Separable Convolution (DSC) backbone, drastically reducing computational latency. By tightly coupling these two aspects, the LRT-BF framework seamlessly transitions from the theoretical limitations of prior works to a highly practical, edge-deployable solution.

### 3 Methodology: The LRT-BF Framework

#### 3.1 Overall Architecture

The LRT-BF framework is a blind adaptive beamforming architecture specifically designed to address the communication constraints of highly dynamic Unmanned Aerial Vehicles (UAVs). As illustrated in Fig. 1,

the system evolves from the fundamental CBTL paradigm. Its core innovation lies in transforming a pre-trained signal classifier into a “Proxy Evaluator” for spatial filtering weights. From a formal transfer learning perspective, the specific “knowledge” transferred from the offline source task (modulation classification) to the online target task (blind beamforming) is the differentiable decision boundary of the modulation’s structural manifold (e.g., the phase transition properties of QPSK). Rather than transferring feature weights to initialize a new network, we transfer the frozen evaluation capability of the classifier to serve as a spatial loss function. The system maintains a robust feedback optimization loop: mixed signals captured by the antenna array are first spatially filtered through a trainable linear beamforming layer; subsequently, the filtered output is evaluated by a lightweight deep neural network to predict its classification confidence.

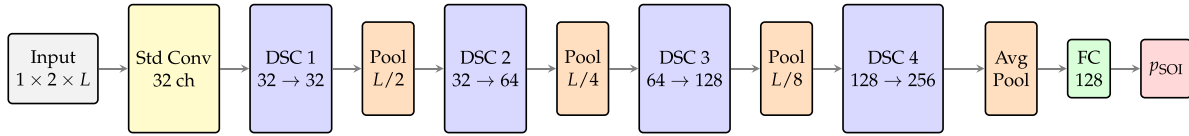


**Figure 1:** Architectural overview and algorithmic flow of the proposed LRT-BF framework. The system consists of a trainable linear beamforming layer and a frozen robust “Proxy Evaluator”. The algorithmic process is characterized by three core phases: (1) **Initialization and Forward Mapping:** The beamforming weight  $\mathbf{w}$  is initialized via a blind uniform or signal-subspace strategy. The spatially whitened array signals  $\mathbf{Z}$  are filtered by  $\mathbf{w}$  and then fed into a lightweight DSC-CNN classifier pre-trained with Frequency Domain Randomization (FDR) to ensure Doppler invariance. (2) **Joint Loss Formulation:** Two independent metrics are computed to guide the optimization: the classification loss  $\mathcal{L}_{cls}$  (calibrated by Temperature Scaling  $\tau$  to maintain effective gradient magnitudes) ensures the preservation of SOI modulation features, while the output power loss  $\mathcal{L}_{pwr}$  drives the nulling of high-power co-channel interference. (3) **Self-supervised Weight Update:** Utilizing autograd mechanisms, the joint gradient  $\nabla_{\mathbf{w}^*} \mathcal{L}$  is backpropagated to the complex weight layer, where  $\mathbf{w}$  is iteratively updated via  $\mathbf{w} \leftarrow \mathbf{w} - \mu \nabla \mathcal{L}$ . This closed-loop optimization enables fully blind adaptive beamforming and interference suppression without requiring DOA information or pilot signals. The SOI Processor block represents the downstream signal processing pipeline (e.g., carrier synchronization, symbol demodulation, and channel decoding) that operates on the interference-suppressed beamformed output, which falls outside the scope of this work but is included to illustrate the complete receiver chain.

Based on this confidence metric, the system iteratively updates beam weights via self-supervised learning. It is crucial to emphasize that unlike traditional static transfer learning paradigms where a pre-trained model is directly used for feed-forward inference, LRT-BF operates as a dynamic, online optimization process. For each newly arrived block of snapshots within the channel coherence time, the beamforming weights  $\mathbf{w}$  are continuously updated in real-time. The frozen classifier acts merely as a spatial “proxy loss function,” ensuring the beamformer adaptively tracks the time-varying characteristics of UAV channels. To ensure real-time performance and robustness at the edge, this framework introduces four core technical enhancements: (i) joint optimization of classifier guidance and power minimization, (ii) Frequency Domain Randomization (FDR), (iii) architectural reconstruction using Depthwise Separable Convolutions (DSC), and (iv) temperature scaling calibration. Building upon the “recognition-as-estimation” advantage inherited

from the original CBTL, LRT-BF realizes a logical evolution across the following three dimensions, specifically targeting the resource-constrained and highly dynamic environment characteristics of UAV platforms: Guided by the overarching system diagrams (Figs. 1 and 2), we have condensed the architectural description by organizing the functional blocks into two primary clusters that directly serve the core objectives of this paper:

- **The Robust Functional Cluster (Combating High Dynamics):** This cluster is responsible for maintaining stable beamforming in harsh, fast-varying channels. It centers around the “Proxy Evaluator” (Fig. 1), which is endowed with Doppler-invariant perception via the Frequency Domain Randomization (FDR) strategy. Furthermore, to prevent gradient vanishing under few-snapshot constraints, a Temperature Scaling mechanism is integrated. Finally, a Joint Optimization Objective ( $\mathcal{L}_{cls} + \lambda\mathcal{L}_{pwr}$ ) guarantees that even if the classification confidence fluctuates, the power minimization constraint acts as a robust fallback to suppress strong interference blindly and continuously.
- **The Lightweight Functional Cluster (Enabling Edge Deployment):** This cluster focuses on executing the robust optimization within the strict millisecond-level channel coherence time and SWaP constraints of UAVs. As detailed in Fig. 2, the traditional dense CNN backbone is entirely reconstructed using a Depthwise Separable Convolution (DSC) architecture. By decoupling spatial filtering and cross-channel fusion, this functional block compresses the parameter count and computational load (FLOPs) by nearly an order of magnitude, transforming a theoretically heavy self-supervised loop into a real-time edge-executable algorithm.



**Figure 2:** The specialized LRT-BF network architecture featuring cascaded DSC blocks for UAV edge deployment.

The blind beamforming process based on the LRT-BF framework is shown in Algorithm 1. Distinct from the generic CBTL framework, our proposed algorithm incorporates a joint optimization objective combining classifier-guided loss and output power minimization. This integration allows the beamformer to achieve fully blind interference suppression without any DOA information, by exploiting the fact that interference power is typically much stronger than the desired signal (Interference-to-Noise Ratio (INR)  $\gg$  Signal-to-Noise Ratio (SNR)).

---

**Algorithm 1:** Joint classifier-power optimization for blind beamforming

---

**Require:** Received signal  $\mathbf{Z} \in \mathbb{C}^{N \times L}$ , frozen classifier  $f_\theta$ , temperature  $\tau$ , power weight  $\lambda$ , learning rate  $\mu$ , max iterations  $K_{\max}$

**Ensure:** Optimized weight vector  $\mathbf{w}_{\text{opt}}$

// Step 1: Uniform Initialization (Fully Blind)

1: Initialize weights:  $\mathbf{w}_0 = \frac{1}{\sqrt{N}} \mathbf{1}_N$

2: Compute initial output power:  $P_0 = \mathbb{E}[|\mathbf{w}_0^H \mathbf{Z}|^2]$

3: Initialize:  $k = 0$ ,  $\mathcal{L}_{\text{best}} = \infty$

4: **while**  $k < K_{\max}$  **do**

    // Step 2: Forward Propagation

(Continued)

**Algorithm 1 (continued)**


---

```

5:   Compute beamformed output:  $\mathbf{y}_k = \mathbf{w}_k^H \mathbf{Z}$ 
6:   Compute output power:  $P_k = \mathbb{E}[|\mathbf{y}_k|^2]$ 
7:   Power normalization:  $\hat{\mathbf{y}}_k = \mathbf{y}_k / \sqrt{P_k}$ 
8:   Compute logits:  $\mathbf{l}_k = f_\theta(\hat{\mathbf{y}}_k)$ 
   // Step 3: Joint Loss Computation
9:   Classifier loss:  $\mathcal{L}_{\text{cls}} = -\log \text{Softmax}(\mathbf{l}_k / \tau)_{\text{SOI}}$ 
10:  Power loss:  $\mathcal{L}_{\text{pwr}} = P_k / P_0$ 
11:  Total loss:  $\mathcal{L}_k = \mathcal{L}_{\text{cls}} + \lambda \cdot \mathcal{L}_{\text{pwr}}$ 
   // Step 4: Backpropagation and Update
12:  Compute gradient:  $\mathbf{g}_k = \nabla_{\mathbf{w}^*} \mathcal{L}_k$ 
13:  Update weights:  $\mathbf{w}_{k+1} = \mathbf{w}_k - \mu \cdot \mathbf{g}_k$ 
14:  Constraint:  $\mathbf{w}_{k+1} = \mathbf{w}_{k+1} / \|\mathbf{w}_{k+1}\|_2$ 
15:  Track best: if  $\mathcal{L}_k < \mathcal{L}_{\text{best}}$  then  $\mathbf{w}_{\text{best}} = \mathbf{w}_k$ 
16:   $k \leftarrow k + 1$ 
17: end while
18: return  $\mathbf{w}_{\text{opt}} = \mathbf{w}_{\text{best}}$ 

```

---

**3.2 Doppler-Resistant Robust Feature Extractor**

In the CBTL architecture, the role of the pre-trained classifier is to act as a “proxy loss function,” where its ability to identify signal features directly determines the upper bound of beamforming performance. Addressing the specific issues of high dynamic motion and limited airborne platform resources in UAV communication scenarios, the standard CNN network and narrowband assumptions used in the original CBTL scheme are no longer applicable. This section reconstructs the feature extractor from two dimensions: data augmentation strategies and lightweight network design.

**3.2.1 High-Dynamic Data Augmentation Based on Frequency Domain Randomization**

In the original CBTL study, Wentz et al. only considered small-magnitude Carrier Frequency Offsets (CFO), which primarily correspond to clock drift in static ground equipment [6]. However, in UAV communications, high-speed relative motion introduces significant Doppler shifts. For a carrier frequency  $f_c$  and relative velocity  $v$ , the received signal incurs a frequency shift of  $f_d = \frac{v}{c} f_c$ . In high-speed UAV scenarios, this shift can reach magnitudes far exceeding the training coverage range of the original CBTL model. If the classifier is sensitive to frequency, such unseen frequency shifts will cause a sharp drop in classification confidence, preventing the beamformer from obtaining effective gradient feedback.

Let the input signal sample during the pre-training phase be  $\mathbf{x} \in \mathbb{C}^L$ . In the presence of a Doppler frequency shift  $\Delta f$  and a random initial phase  $\phi_0$ , the baseband equivalent form of the  $l$ -th sample point can be expressed as:

$$\tilde{x}[l] = x[l] \cdot e^{j(2\pi\Delta f \cdot l / f_s + \phi_0)} + n[l] \quad (4)$$

where  $f_s$  denotes the sampling rate. Traditional CNNs tend to learn the absolute phase features of signals. Our goal is to train a feature extractor  $f_\theta(\cdot)$  that satisfies frequency shift invariance, that is:

$$f_\theta(\tilde{\mathbf{x}}) \approx f_\theta(\mathbf{x}), \quad \forall \Delta f \in [-F_{\text{max}}, F_{\text{max}}] \quad (5)$$

To compel the network to learn the modulation structure of the signal (such as the constellation shape of QPSK) rather than phase or frequency features, we introduce a ‘‘Frequency Domain Randomization’’ data augmentation strategy during the pre-training phase. Specifically, when constructing the augmented training set  $\mathcal{D}_{\text{aug}}$ , we not only include Additive White Gaussian Noise (AWGN) but also apply random frequency rotation online to each training sample. The frequency offset  $\Delta f$  follows a uniform distribution  $U(-F_{\text{max}}, F_{\text{max}})$ , where  $F_{\text{max}} = 5$  kHz, corresponding to a maximum radial velocity of the UAV of approximately 625 m/s (carrier frequency  $f_c = 2.4$  GHz). Through this adversarial training, the feature extractor is forced to ignore carrier rotation and focus on the relative phase transition patterns between symbols. From a transfer learning perspective, it is important to distinguish our approach from mainstream Domain Adaptation (DA) methods, such as continuous online fine-tuning or Domain Adversarial Neural Networks (DANN). While effective in resource-rich environments, mainstream DA methods require substantial online computational overhead to align source and target domain distributions. In UAV communications, the strictly limited SWaP profile and the millisecond-level channel coherence time strictly prohibit such online computational burdens. By employing FDR, we utilize a Domain Generalization (DG) via Data Augmentation paradigm. This approach shifts the entire burden of domain alignment to the offline pre-training phase, endowing the network with Doppler-invariant feature extraction capabilities with absolutely zero additional online computational cost, making it uniquely suited for real-time edge deployment. This strategy allows the beamformer to directly lock onto highly dynamic target signals without requiring an explicit frequency offset compensation module [8].

### 3.2.2 Lightweight Network Design for Edge Deployment

The original CBTL algorithm employs a network structure containing multiple standard convolutional layers and fully connected layers, resulting in a large number of parameters [6]. While this runs smoothly on GPU servers, its inference latency struggles to meet the real-time requirements of short-burst communications on UAV airborne embedded platforms, where power consumption and computing power are strictly limited. The computational cost of a standard convolutional layer is primarily determined by the number of input channels  $C_{\text{in}}$ , the number of output channels  $C_{\text{out}}$ , and the kernel size  $K$ . Its single-layer computational volume (FLOPs) is proportional to  $K^2 \cdot C_{\text{in}} \cdot C_{\text{out}}$ . To reduce computational complexity, we replace the standard convolutional layers in the original network with Depthwise Separable Convolution. This structure decouples standard convolution into two steps:

- **Depthwise Conv:** Performs spatial convolution independently on each input channel.
- **Pointwise Conv:** Uses  $1 \times 1$  convolution kernels to perform a linear combination across channels on the output of the depthwise convolution.

The improved computational complexity ratio of a single layer is approximately  $\frac{1}{C_{\text{out}}} + \frac{1}{K^2}$ . The lightweight network designed in this paper adopts multi-layer cascaded depthwise separable convolution modules, with the number of channels increasing layer by layer ( $2 \rightarrow 16 \rightarrow 32 \rightarrow 64$ ). Batch normalization and non-linear activation are added after each layer to enhance feature representation capability. The optimized network structure significantly reduces the number of parameters and greatly improves inference speed. Specifically, this lightweight reconstruction compresses the CPU inference latency to merely 1.64 ms (as will be detailed in Section 5). In high-speed UAV communications, the channel coherence time is typically on the order of several milliseconds. Traditional complex blind algorithms often fail because their optimization execution time exceeds this strict physical window. By completing the self-supervised forward evaluation and backward gradient updating within 1.64 ms, LRT-BF fundamentally resolves the conflict between algorithmic execution latency and the rapidly time-varying nature of UAV channels, enabling truly real-time, dynamic adaptive beam tracking at the edge.

### 3.3 Blind Interference Suppression Strategy Based on Joint Optimization

In UAV Short Burst Communication scenarios, the receiver can often obtain only a very small amount of snapshot data. The original CBTL framework uses only the classifier loss as the optimization objective, which has two key limitations: (1) when the classifier confidence tends to saturate, vanishing gradients cause optimization stagnation; (2) pure classifier loss is difficult to drive the beamformer to form deep nulls in the direction of interference. To address these two problems, this paper introduces a temperature scaling mechanism and a power minimization constraint, respectively, constructing a joint optimization framework that combines classifier guidance with power minimization.

#### 3.3.1 Temperature Scaling Mechanism

To alleviate gradient vanishing and enhance the monotonic correlation between classifier confidence and SINR, we introduce a hyperparameter  $\tau > 1$  at the classifier output layer. Assuming the Logits vector output by the linear layer of the lightweight classifier is  $\mathbf{l} = [l_{\text{noise}}, l_{\text{SOI}}]^T$ , the improved Softmax probability calculation formula is defined as:

$$p_{\text{SOI}}(\mathbf{y}; \tau) = \frac{\exp(l_{\text{SOI}}/\tau)}{\exp(l_{\text{noise}}/\tau) + \exp(l_{\text{SOI}}/\tau)} \quad (6)$$

where  $\tau$  represents the temperature parameter.

- When  $\tau = 1$ , it functions as the standard Softmax, which is prone to premature saturation at low SINR.
- When  $\tau > 1$ , the output probability distribution is “softened,” leading to increased entropy, which maintains non-zero gradients, thereby preventing stagnation in weight optimization [6].

This “softening” ensures that even if the classifier can identify signals with high accuracy, the loss function continues to provide non-zero backpropagated gradients, compelling the beamformer to further extract signal features. In the experiments of this paper, we set  $\tau = 4.0$ .

#### 3.3.2 Power Minimization Constraint and Joint Loss Function

To achieve interference suppression under fully blind conditions (without any prior DOA information), this paper introduces an output power minimization constraint. The physical intuition behind this is: in scenarios with a high Interference-to-Noise Ratio (INR  $\gg$  SNR), the beamforming output power is primarily contributed by strong interference components. By minimizing the total output power, the optimization process prioritizes suppressing strong interference, while the classifier loss ensures that the modulation features of the desired signal are not excessively attenuated. Based on this, the joint loss function is defined as:

$$\mathcal{L}(\mathbf{w}) = \mathcal{L}_{\text{cls}}(\mathbf{w}) + \lambda \cdot \mathcal{L}_{\text{pwr}}(\mathbf{w}) \quad (7)$$

where the classifier loss is defined as the negative log-likelihood for the target signal:

$$\mathcal{L}_{\text{cls}}(\mathbf{w}) = -\log(p_{\text{SOI}}(\mathbf{w}^H \mathbf{Z}; \tau)) \quad (8)$$

This term ensures that the beamforming output retains the QPSK modulation characteristics of the desired signal, enabling the classifier to identify it correctly. Power loss is defined as the normalized output power:

$$\mathcal{L}_{\text{pwr}}(\mathbf{w}) = \frac{\mathbb{E}[|\mathbf{w}^H \mathbf{Z}|^2]}{P_0} \quad (9)$$

where  $P_0$  is the initial output power, used for normalization to ensure numerical stability. This term drives the beamformer to suppress strong interference components.  $\lambda$  is a balancing factor, which is set to  $\lambda = 0.5$  in the experiments of this paper. This joint optimization strategy realizes a synergistic mechanism of “power constraint guarantee + classifier enhancement”: even when the classifier fails (e.g., when Original CBTL faces high Doppler frequency offsets), the power minimization constraint can still provide basic interference suppression capabilities; meanwhile, the robust classifier offers refined modulation feature guidance on this basis, enabling the null depth to break through the performance bottleneck that exists when the power constraint acts alone.

**Analysis and Mitigation of Negative Transfer Risks:** Crucially, this joint formulation theoretically mitigates the risk of Negative Transfer. In this transfer learning paradigm, negative transfer would occur if the pre-trained classifier confidently misidentifies a strong interference as the desired signal, or if its accuracy collapses due to unseen channel dynamics, thereby feeding erroneous gradients to the beamformer. By coupling the classification loss with the power minimization constraint ( $\mathcal{L}_{pwr}$ ), the system establishes a lower-bound safety net. Even if the classifier experiences negative transfer and fails to provide valid modulation guidance, the  $\mathcal{L}_{pwr}$  term dominates the optimization, coercing the beamformer to continually suppress high-power spatial directions. This mechanism guarantees that the system gracefully degrades to a baseline interference-nulling filter rather than erroneously amplifying the jammer.

To justify the use of classification confidence as a proxy for beamforming performance, we provide a theoretical link between the Negative Log-Likelihood (NLL) and the post-beamforming Signal-to-Interference-plus-Noise Ratio (SINR).

**Lemma 1:** *Under the assumption of Gaussian-distributed residual interference and noise, minimizing the classification NLL loss is statistically equivalent to maximizing the output SINR.*

**Proof:** Consider the beamformed output signal  $y = \mathbf{w}^H \mathbf{Z}$ . We can decompose  $y$  into the desired signal component and the residual distortion:

$$y = \beta s + e \quad (10)$$

where  $s \in \mathcal{C}$  is the normalized constellation point from the modulation alphabet  $\mathcal{C}$ ,  $\beta$  is the complex gain factor, and  $e \sim \mathcal{CN}(0, \sigma_e^2)$  represents the aggregate residual interference and noise. The output SINR is defined as:

$$\text{SINR} = \frac{E[|\beta s|^2]}{E[|e|^2]} = \frac{|\beta|^2}{\sigma_e^2} \quad (11)$$

For a pre-trained classifier  $f_\theta$ , the posterior probability of the output  $y$  belonging to the correct modulation class  $c$  can be modeled via a Gaussian likelihood function in the feature space:

$$p(y|c; \mathbf{w}) \propto \exp\left(-\frac{|y - \beta s|^2}{2\sigma^2}\right) \quad (12)$$

where  $\sigma^2$  is a model-specific parameter reflecting the classifier’s sensitivity to signal deviations. The classification NLL loss is given by:

$$\mathcal{L}_{cls} = -\ln p(c|y; \mathbf{w}) \quad (13)$$

Substituting the likelihood model into the loss function and ignoring constant terms, we obtain:

$$\mathcal{L}_{\text{cls}} \propto \frac{|y - \beta s|^2}{2\sigma^2} = \frac{|e|^2}{2\sigma^2} \quad (14)$$

In the self-supervised optimization phase, we minimize the expectation of the loss:

$$\min_{\mathbf{w}} E[\mathcal{L}_{\text{cls}}] \iff \min_{\mathbf{w}} \frac{\sigma_e^2}{2\sigma^2} \quad (15)$$

Since  $\sigma^2$  is a constant determined by the pre-trained weights, minimizing the error power  $\sigma_e^2$  while maintaining the desired signal power (via weight normalization  $\|\mathbf{w}\|_2 = 1$ ) is equivalent to:

$$\max_{\mathbf{w}} \frac{|\beta|^2}{\sigma_e^2} \iff \max_{\mathbf{w}} \text{SINR} \quad (16)$$

This completes the proof.  $\square$

### 3.3.3 Initialization Strategy

Different from the random initialization or MVDR initialization commonly used in the original CBTL, this paper adopts uniform weight initialization:

$$\mathbf{w}_0 = \frac{1}{\sqrt{N}} \mathbf{1}_N \quad (17)$$

This strategy offers the following advantages: (1) Completely blind—it does not rely on any spatial statistical information or DOA estimation; (2) Computationally simple—it requires no complex operations such as matrix inversion; (3) Good synergy with the power minimization constraint—initially, the response is consistent in all directions, and the power constraint naturally guides the beamformer to suppress the strongest interference components first.

To rigorously evaluate the impact of initialization on convergence speed under few-snapshot conditions (RQ2), we introduce signal subspace initialization as a comparative benchmark. This method leverages the eigendecomposition of the received signal covariance matrix, setting the initial weight to the principal eigenvector:

$$\mathbf{w}_0 = \mathbf{e}_1, \quad \text{where } \mathbf{R}\mathbf{e}_1 = \lambda_1\mathbf{e}_1, \quad \lambda_1 = \max(\text{eig}(\mathbf{R})) \quad (18)$$

While subspace-based initialization is theoretically attractive for capturing dominant energy components to accelerate convergence, it poses a significant risk in the high-interference scenarios targeted by this study. Specifically, when the Interference-to-Noise Ratio (INR) significantly exceeds the Signal-to-Noise Ratio (SNR), the principal eigenvector  $\mathbf{e}_1$  tends to align with the strong interference rather than the desired signal. Consequently, we include this strategy primarily to contrast its performance against the proposed uniform initialization and to determine the robust boundaries of the algorithm.

## 4 Experimental Design

### 4.1 Research Questions

This paper validates the advantages of the LRT-BF method in terms of anti-jamming performance, convergence speed, and computational efficiency, targeting the high-dynamic communication requirements of UAVs. Specifically, the following three research questions are proposed:

- **RQ1 (Anti-Doppler Capability and OOD Generalization):** Can the pre-training strategy based on Frequency Domain Randomization (FDR) enable the classifier to maintain robustness under Doppler shifts as high as 5 kHz? Furthermore, how does the model perform under Out-of-Distribution (OOD) scenarios that exceed this pre-training boundary? Compared to the original CBTL, what is the magnitude of improvement in null depth and interference suppression capability in strong jamming scenarios?
- **RQ2 (Fast Convergence Capability):** Can the temperature scaling mechanism ( $T = 4$ ) and the Sub-space Initialization strategy resolve the problems of gradient saturation and convergence stagnation under extremely few snapshots ( $L \leq 64$ )? What are the independent contributions of each component to performance improvement?
- **RQ3 (Lightweight Effect):** After reconstructing the feature extraction network using Depthwise Separable Convolution (DSC), how does the model perform in terms of parameter count, computational cost (FLOPs), and inference latency on embedded platforms (CPU)? Can it meet the real-time requirements of UAV edge devices without sacrificing modulation classification accuracy?

#### 4.2 Simulation Environment Setup

To evaluate the performance of the proposed algorithm in complex UAV communication scenarios, a high-dynamic simulation platform with strong interference is established. A  $N = 4$  element Uniform Linear Array (ULA) with half-wavelength spacing is employed. The air-to-ground link is modeled as a Rician fading channel ( $K = 10$  dB) to simulate a typical Line-of-Sight (LoS) environment, incorporating a random Doppler frequency shift up to  $\Delta f = 5$  kHz, which corresponds to a relative UAV velocity of approximately 625 m/s. Spatially, the Signal of Interest (SOI) is located at  $\theta_{\text{SOI}} = 90^\circ$ , while two strong co-channel interferences (INT) are positioned at  $40^\circ$  and  $140^\circ$  with Interference-to-Noise Ratios (INR) of 25 and 20 dB, respectively. The experiments specifically focus on the algorithm's robustness under extreme data-starved conditions with snapshots  $L \leq 64$  at SNR = 10 dB. Furthermore, the feature extractor is pre-trained on a dataset of 10,000 samples across multiple modulation schemes (BPSK, QPSK, 8PSK, and 16QAM). Real-time inference latency is validated on both NVIDIA RTX 4090 GPU and Intel i9-13900K CPU to simulate the heterogeneous computational environments of UAV edge nodes.

#### 4.3 Baselines

To comprehensively evaluate performance, this paper selects the following four categories of comparison methods:

- **Oracle MVDR:** Assumes the precise interference covariance matrix is known, representing the theoretical optimal interference suppression performance in this scenario.
- **Original CBTL (Direct Transfer Paradigm):** A method employing a standard CNN architecture without frequency offset augmentation training. In the context of transfer learning, this serves as the "Direct Transfer" baseline. Comparing it against LRT-BF (which represents the "Domain Generalization via Augmentation" paradigm) validates our choice of TL paradigm under strict computational constraints.
- **Traditional Blind Signal Processing Methods:** Includes FastICA, which is based on higher-order statistics, and CMA, which is based on the constant modulus criterion.

To address the core transfer learning characteristics, our ablation studies will specifically isolate two core TL variables: the domain discrepancy bound ( $F_{\text{max}}$ ) to evaluate robustness, and the knowledge softening factor (Temperature  $\tau$ ) to evaluate gradient effectiveness during the transfer process.

#### 4.4 Network Architecture Design and Computational Complexity Analysis

To satisfy the stringent requirements for real-time performance and low power consumption in edge-side UAV deployment, this paper proposes the LRT-BF lightweight feature extraction network. This section will elaborate on its architecture design logic and provide a theoretical comparison with the original CBTL network.

##### 4.4.1 Details of LRT-BF Lightweight Network Design

As illustrated in Fig. 2, the core design philosophy of LRT-BF is to leverage Depthwise Separable Convolution (DSC) to achieve spatiotemporal decoupling of feature extraction weights. The specific design details are as follows:

- **Asymmetric First-Layer Design:** Considering that the original input signal consists of  $2 \times L$  complex baseband samples with extremely low channel count, the first layer of LRT-BF employs a standard  $3 \times 3$  convolutional layer to rapidly expand the feature space to 32 dimensions, thereby avoiding the loss of spatial features in the initial stage.
- **Cascaded DSC Block Architecture:** The backbone of the network consists of four cascaded DSC blocks, with channel counts following an increasing pattern of  $32 \rightarrow 64 \rightarrow 128 \rightarrow 256$ . Each DSC block decomposes standard convolution into:
  1. *Depthwise Conv:* Performs time-domain filtering independently on each channel to learn local waveform features of the signal.
  2. *Pointwise Conv:* Utilizes  $1 \times 1$  convolution kernels for cross-channel linear combination to achieve feature fusion.
- **Time-Domain Dimension Compression:** By embedding multiple Max Pooling layers, the time-domain dimension is progressively downsampled to  $L/8$  while increasing the receptive field. Finally, Global Average Pooling (GAP) is used to compress the high-dimensional features into a 256-dimensional vector, which is then input into the classifier for confidence prediction.

##### 4.4.2 Comparison between LRT-BF and the Original CBTL Network

The original CBTL network employs a stack of deep standard convolutions. Although it possesses strong feature modeling capabilities, it faces significant performance bottlenecks under constrained edge computing power. Assuming a kernel size of  $K = 3$ , the number of input channels as  $C_{in}$ , and the number of output channels as  $C_{out}$ , the comparison of parameters between standard convolution and DSC is as follows:

$$\text{Ratio} = \frac{P_{DSC}}{P_{Std}} = \frac{K^2 \cdot C_{in} + C_{in} \cdot C_{out}}{K^2 \cdot C_{in} \cdot C_{out}} = \frac{1}{C_{out}} + \frac{1}{K^2} \quad (19)$$

Since  $C_{out} \gg K^2$  in the deep layers of LRT-BF, the parameter count of DSC is theoretically only about 1/9 of that of standard convolution. Table 1 summarizes the theoretical performance differences between the two architectures.

Theoretical analysis indicates that LRT-BF significantly reduces the total computational load (FLOPs) through the DSC structure. This small-scale network design not only shortens the inference time for a single iteration of beamforming weights but also enables the algorithm to achieve convergence within a limited number of snapshots in high-dynamic UAV jamming environments, thereby significantly improving the system's response speed.

**Table 1:** Comparison of theoretical complexity between LRT-BF and original CBTL network.

Metric	Original CBTL Network	LRT-BF (Our Method)
Convolution Operator	Standard Convolution (Dense)	Depthwise Separable Convolution (Sparse)
Parameter Growth Rate	$O(C^2 \cdot K^2)$	$O(C \cdot K^2 + C^2)$
Memory Footprint	Relatively High	Extremely Low (Suitable for SRAM-constrained devices)
Computational Intensity	High (MAC-intensive)	Low (Compute requirements significantly reduced)

## 5 Results and Discussion

This section presents the experimental results and provides an in-depth analysis addressing the three research questions proposed in Section 4. All experiments were repeated multiple times on the same hardware platform to ensure the statistical reliability of the results.

### 5.1 RQ1: Evaluation of Anti-Doppler Performance

#### 5.1.1 Experimental Objective

This experiment aims to systematically evaluate the robustness enhancement effect of the frequency-domain randomization pre-training strategy against high-dynamic Doppler shifts. By verifying the stability of LRT-BF's classification accuracy over a wide frequency offset range of 0~5 kHz, we ensure that high-quality and continuous gradient feedback signals can be obtained during the online beamforming stage. On this basis, the experiment quantifies the performance improvement of the proposed method in terms of classification accuracy and interference suppression compared to the original CBTL method [6]. Furthermore, comparisons with traditional baseline methods such as FastICA [5] and CMA [17] are conducted to demonstrate the superiority of the deep learning-based beamforming architecture in extreme scenarios characterized by strong interference and few snapshots. Finally, through the visual analysis of beam patterns, the experiment deeply explores the intrinsic coupling mechanism between classifier robustness and the convergence quality of beamforming weights, providing intuitive physical evidence for the algorithm's effectiveness.

#### 5.1.2 Experimental Setup

In this experiment, the Doppler frequency offset  $\Delta f \in \{0, 1, 2, 3, 4, 5\}$  kHz is selected as the core test variable to simulate typical high-dynamic UAV communication scenarios ranging from static conditions to a maximum radial velocity of approximately 625 m/s (carrier frequency  $f_c = 2.4$  GHz). The simulation system utilizes an  $N = 4$  element uniform linear array ( $d = 0.5\lambda$ ). The desired QPSK-modulated signal is set at  $90^\circ$  (SNR = 10 dB), accompanied by two strong co-channel interference sources located at  $40^\circ$  (INR = 25 dB) and  $140^\circ$  (INR = 20 dB), respectively. The physical link is modeled as a Rician fading channel with a Rician factor  $K = 10$  dB. The number of online optimization iterations is set to 800, with a temperature calibration parameter  $\tau = 4.0$ .

To achieve fully blind interference suppression (requiring no prior DOA information), this paper adopts a joint optimization strategy combining classifier guidance with power minimization. The loss function for the online phase is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda \cdot P_{\text{out}} \quad (20)$$

where  $\mathcal{L}_{\text{cls}}$  is the temperature-scaled classifier cross-entropy loss, used to ensure that the modulation features of the desired signal are correctly preserved;  $P_{\text{out}} = \mathbb{E}[|\mathbf{w}^H \mathbf{z}|^2]$  represents the beamforming output power,

driving the adaptive suppression of strong interference by minimizing the total output power; and  $\lambda = 0.5$  serves as the balancing factor. The physical intuition behind this design is as follows: since the interference power is significantly higher than that of the desired signal ( $\text{INR} \gg \text{SNR}$ ), the power minimization process prioritizes suppressing strong interference components, while the classifier loss ensures that the desired signal is not excessively attenuated, thereby automatically forming interference nulls without requiring DOA information.

To verify performance, this paper selects four representative methods for comparative evaluation: 1) **LRT-BF (Proposed Method)**: Adopts a lightweight depthwise separable convolution architecture, introduces a frequency domain randomization augmentation strategy of  $U(-5, 5)$  kHz during the pre-training phase, and utilizes the aforementioned joint loss function in the online phase; 2) **Original CBTL [6]**: Serves as the improvement baseline, employing a standard convolution architecture without frequency offset augmentation pre-training, with a temperature parameter of  $\tau = 1.0$ ; 3) **FastICA [5]**: A classic statistical independent blind source separation algorithm based on the criterion of negentropy maximization; 4) **CMA [17]**: A classic adaptive beamforming method based on the constant modulus property of signals.

### 5.1.3 Evaluation Metrics

1) **Classifier Accuracy** is defined as the proportion of target signals and noise/interferences correctly identified by the pre-trained classifier on the test set containing frequency offsets:

$$\text{Acc} = \frac{N_{\text{correct}}}{N_{\text{total}}} \times 100\% \quad (21)$$

where  $N_{\text{correct}}$  is the number of correctly classified samples, and  $N_{\text{total}}$  is the total number of test samples. This metric directly reflects the robustness of the feature extractor against frequency offsets, which is a prerequisite for online beamforming to obtain effective gradients.

2) **Null Depth** is defined as the power response ratio (dB) of the beam pattern in the interference direction relative to the desired signal direction, used to quantify interference suppression capability:

$$D_{\text{null},k} = 10 \log_{10} \frac{|\mathbf{w}^H \mathbf{a}(\theta_{\text{INT},k})|^2}{|\mathbf{w}^H \mathbf{a}(\theta_{\text{SOI}})|^2} \quad (\text{dB}) \quad (22)$$

where  $\mathbf{w}$  is the beamforming weight vector, and  $\mathbf{a}(\theta)$  is the steering vector for direction  $\theta$ . For multi-interference scenarios, the average null depth across all interference directions is reported:

$$\bar{D}_{\text{null}} = \frac{1}{K} \sum_{k=1}^K D_{\text{null},k} \quad (23)$$

A more negative value indicates stronger interference suppression capability, ideally approaching  $-\infty$ .

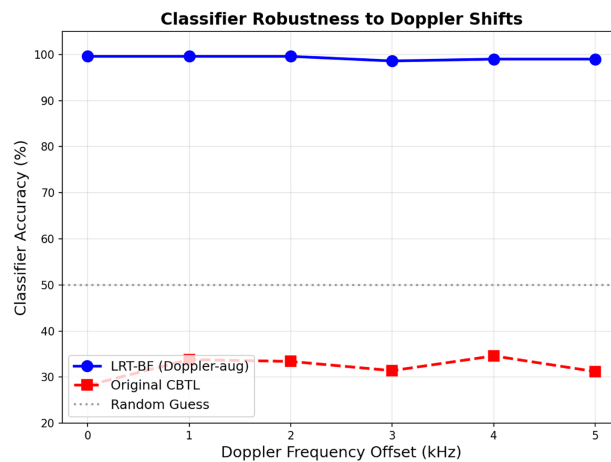
3) **Beam Pattern** is defined as the normalized spatial response of the beamformer across all azimuth angles:

$$P(\theta) = 10 \log_{10} \frac{|\mathbf{w}^H \mathbf{a}(\theta)|^2}{\max_{\theta'} |\mathbf{w}^H \mathbf{a}(\theta')|^2} \quad (\text{dB}) \quad (24)$$

By visualizing the beam pattern, one can intuitively verify whether the main lobe is aligned with the target signal direction and whether nulls are correctly formed in the interference directions.

### 5.1.4 Experimental Results Analysis

**1) Analysis of Classifier Robustness to Frequency Offset.** Fig. 3 illustrates the classifier accuracy curves under two different pre-training strategies as the frequency offset varies. The experimental results indicate that: The LRT-BF method maintains a classification accuracy of 98.6%–99.6% (averaging 99.2%) across the entire tested frequency offset range (0–5 kHz). This validates that the frequency-domain randomization pre-training strategy successfully forces the network to learn modulation structural features that are invariant to frequency shifts, rather than phase-sensitive time-domain features. In contrast, the Original CBTL classifier, without frequency offset augmentation, achieves an average accuracy of only 32.1%, which is below the random guessing level (50%). This indicates that its feature extractor fails completely when facing Doppler frequency shifts. Notably, even under the zero frequency offset condition ( $\Delta f = 0$  kHz), the accuracy of Original CBTL is merely 28.2%. As the frequency offset increases to 4 kHz, the accuracy rises slightly to 34.6%. This anomalous phenomenon may be related to the training data distribution and the randomness of the test scenarios. Overall, the LRT-BF method consistently remains stable above 98%, demonstrating a significant robustness advantage with a performance improvement of 67.1 percentage points.



**Figure 3:** Classifier accuracy under varying Doppler frequency offsets.

**2) Analysis of Online Beamforming Performance.** Table 2 summarizes the beamforming performance of each method under different frequency offset conditions. The key findings are as follows: LRT-BF method: Deep interference suppression was achieved under all tested frequency offsets, with an average null depth reaching  $-41.2$  dB and a maximum depth of up to  $-48.8$  dB ( $\Delta f = 5$  kHz). Under the frequency offset conditions of 0, 2, 4, and 5 kHz, the null depths all exceeded  $-41$  dB, indicating that the robust classifier can continuously provide effective gradient feedback, driving the beam weights to converge to a deep null solution in synergy with the power minimization constraint. Original CBTL [6]: Since this experiment employed a joint loss function (classifier loss + power minimization), the power minimization constraint could still provide a certain degree of interference suppression capability even when the classifier failed (accuracy was only 32.1%), achieving an average null depth of  $-37.8$  dB. However, the standard deviation of its null depth was small ( $2.2 \sim 5.7$  dB), suggesting that the optimization process lacked effective guidance from the classifier and mainly relied on the “blind” suppression of the power constraint, making it difficult to further exploit signal structure information to achieve deeper nulls. Traditional blind methods: The average null depth of the CMA algorithm [17] was only  $+7.2$  dB (a positive value indicates that the response in the interference direction is higher than that in the signal direction), demonstrating its complete failure in strong interference scenarios with limited snapshots and extremely high interference-to-noise ratios ( $\text{INR} > 20$  dB).

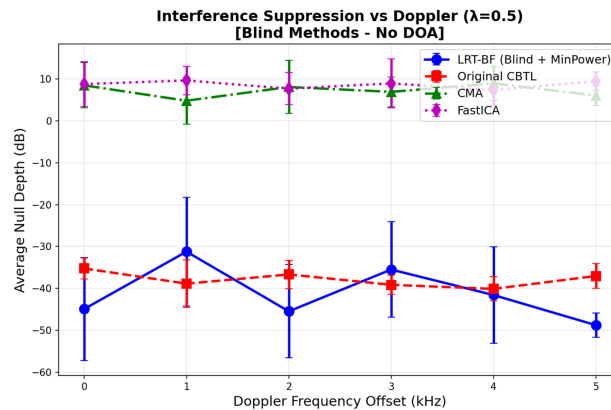
The performance of FastICA [5] was even worse, with an average null depth of +8.6 dB, reflecting the failure of independent component analysis methods in blind source separation under scenarios with imbalanced signal-to-interference ratios.

**Table 2:** RQ1: Experimental results of anti-Doppler frequency offset performance comparison (Anti-Doppler Performance).

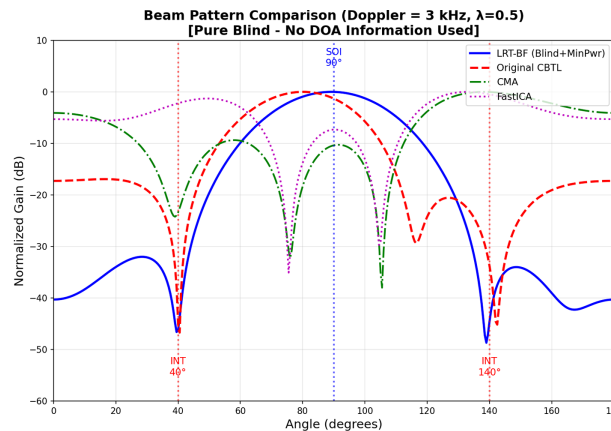
Doppler (kHz)	Classifier Accuracy (%)		Null Depth (dB)	
	LRT-BF	Original CBTL	LRT-BF	Original CBTL
0	99.6	28.2	$-44.9 \pm 12.3$	$-35.2 \pm 2.5$
1	99.6	33.8	$-31.2 \pm 13.0$	$-38.9 \pm 5.7$
2	99.6	33.4	$-45.5 \pm 11.1$	$-36.7 \pm 3.4$
3	98.6	31.4	$-35.5 \pm 11.4$	$-39.1 \pm 2.2$
4	99.0	34.6	$-41.6 \pm 11.6$	$-40.1 \pm 2.9$
5	99.0	31.2	$-48.8 \pm 2.9$	$-37.0 \pm 2.9$
Average	99.2	32.1	-41.2	-37.8

**3) Trends of Null Depth with Frequency Offset.** Fig. 4 illustrates the trends of null depth for each method as the Doppler frequency offset varies. This figure intuitively reveals the significant impact of classifier performance on the beamforming results: The null depth curve of the LRT-BF method exhibits certain fluctuations within the 0–5 kHz frequency offset range, varying from  $-31.2$  to  $-48.8$  dB (with a mean of  $-41.2$  dB). A relatively shallow null ( $-31.2$  dB) appears at 1 kHz, which may be attributed to the randomness of the optimization process under specific frequency offsets. Notably, as the frequency offset increases to 5 kHz, the null depth of LRT-BF reaches its deepest point ( $-48.8$  dB), fully validating the effectiveness of the frequency-domain randomization pre-training strategy under extreme high-frequency offset conditions. The null depth curve of the Original CBTL fluctuates within the range of  $-35.2$  to  $-40.1$  dB with a small standard deviation. This relatively stable performance is not a reflection of its classifier’s robustness, but rather reflects the role of the power minimization constraint as a “fallback” mechanism—when the classifier fails, the power constraint can still drive the beamformer to suppress strong interference. However, lacking the guidance of modulation features, the null depth struggles to break through the bottleneck of  $-40$  dB. The performance curves of CMA [17] and FastICA [5] are both located above the 0 dB line (at +7.2 and +8.6 dB, respectively) and are horizontally distributed, reflecting their insensitivity to frequency offsets. However, this “insensitivity” stems from the limitations of the methods themselves—constrained by insufficient snapshots and the inapplicability of the constant modulus/independence assumptions, they are completely unable to form effective nulls in strong interference scenarios.

**4) Beampattern Comparison.** Fig. 5 illustrates the beampatterns learned by each method under the condition of  $\Delta f = 3$  kHz. The key observations are as follows:



**Figure 4:** Null depth vs. Doppler frequency offset.



**Figure 5:** Beam pattern comparison at 3 kHz Doppler offset.

**LRT-BF method:** A distinct main lobe (normalized gain of 0 dB) is formed in the direction of the target signal (90), while deep nulls are simultaneously created in the two interference directions (40 and 140). This beam pattern structure of “main lobe aligned with the target + dual nulls suppressing interference” demonstrates that the joint optimization strategy of classifier guidance and power minimization successfully realizes fully blind adaptive beamforming. Without any prior DOA information, an optimal spatial filter that points to the desired signal and suppresses strong interference is automatically formed merely by maximizing the QPSK distinguishability of the output signal and minimizing the total output power.

**Original CBTL [6]:** A main lobe is also formed in the 90 direction, and nulls of certain depths are created in the interference directions. However, due to the degradation of gradient quality caused by classifier failure, the null depth and shape precision of its beam pattern are inferior to those of LRT-BF. This disparity confirms the importance of classifier robustness for the refined optimization of beamforming.

**CMA [17]:** The beam pattern exhibits an irregular shape, failing to form effective nulls in the interference directions, and the main lobe pointing shows significant deviation. This indicates that the constant modulus criterion struggles to correctly identify the desired signal in high INR and few-snapshot scenarios, leading to a tendency to misidentify strong interference as the target.

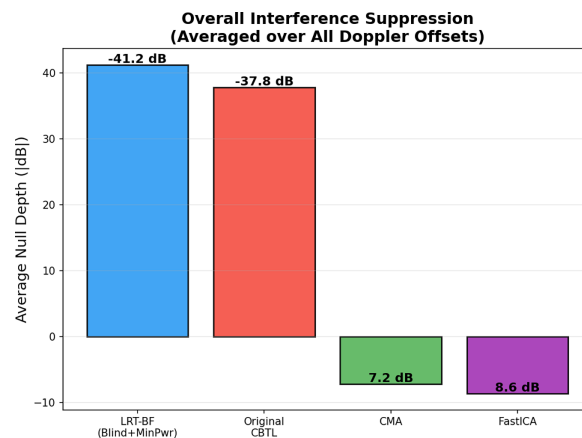
**FastICA [5]:** The beam pattern presents a chaotic multi-peak structure with no null formation in the interference directions. The main lobe may even point towards the interference rather than the desired signal,

confirming the complete failure of blind source separation based on independent component analysis in scenarios with severe signal-to-interference ratio imbalance.

**5) Comprehensive Performance Comparison.** Table 3 and Fig. 6 summarize the average interference suppression performance of each method within the 0–5 kHz frequency offset range in tabular and bar chart formats, respectively. It can be intuitively observed that: The LRT-BF method significantly outperforms all other comparative methods with an average null depth of  $-41.2$  dB. Compared to the Original CBTL [6], it achieves an improvement of 3.4 dB (an increase in interference suppression capability of approximately 2.2 times). Compared to CMA [17], it improves by 48.4 dB (approximately 69,000 times), and compared to FastICA [5], it improves by 49.8 dB (approximately 95,000 times). The Original CBTL ( $-37.8$  dB) achieves decent interference suppression aided by the power minimization constraint, verifying the effectiveness of the joint loss function design. However, its performance still lags behind LRT-BF, with the gap primarily reflected in: (1) a limited upper bound on null depth (deepest only  $-40.1$  dB vs. LRT-BF's  $-48.8$  dB); and (2) uncertainty in optimization direction caused by classifier failure, making it difficult to achieve optimality under certain frequency offset conditions. The CMA algorithm ( $+7.2$  dB) and FastICA ( $+8.6$  dB) completely fail in this scenario. A positive null depth implies that the response of the formed beam pattern in the direction of the interference is actually higher than in the direction of the signal. This result profoundly reveals the inherent limitations of traditional blind methods under conditions of insufficient snapshots ( $L = 256$ ) and imbalanced signal-to-interference ratios ( $\text{INR} \gg \text{SNR}$ ), providing a sufficient necessity argument for new beamforming architectures based on deep learning.

**Table 3:** RQ1: performance comparison with traditional blind signal processing methods (Mean Doppler Shift).

Method	Avg. Null Depth	Improvement over CMA
<b>LRT-BF (Ours)</b>	$-41.2$ dB	$-48.4$ dB
Original CBTL [6]	$-37.8$ dB	$-45.0$ dB
CMA [17]	$+7.2$ dB	—
FastICA [5]	$+8.6$ dB	$+1.4$ dB



**Figure 6:** Overall interference suppression performance.

### Robustness Verification: Bridging Theoretical Limits and Realistic Operations

While our initial simulations established the theoretical robustness boundaries under extreme conditions (5 kHz Doppler shift, corresponding to  $\approx$ Mach 1.84), a rigorous engineering evaluation necessitates validation within the physically realistic flight envelope. Consequently, we proactively extended the experimental scope to cover a practical velocity range of 0–100 m/s (corresponding to Doppler shifts of 0–800 Hz at  $f_c = 2.4$  GHz). This spectrum comprehensively encapsulates diverse operational profiles, ranging from static hovering and commercial cruising (20 m/s) to industrial inspection (40 m/s) and high-speed military maneuvering (100 m/s).

Empirical analysis within this realistic domain reveals that LRT-BF exhibits superior stability compared to the theoretical stress tests. The method maintains a near-perfect mean classification accuracy of 99.7% (ranging from 98.6% to 100%) and achieves a profound interference suppression depth of **−46.5 dB**, accompanied by a high SOI detection confidence of 90.6%.

*Key Finding:* In sharp contrast, the baseline CBTL method (without FDR augmentation) fails to adapt even to these moderate dynamics, degrading to near-random accuracy (30.5%). This highlights a critical insight: the performance advantage of the proposed FDR strategy amplifies significantly in realistic scenarios (an accuracy gain of 69.2 percentage points) compared to extreme conditions ( $\approx$  40 percentage points). These findings conclusively validate that the Frequency Domain Randomization (FDR) strategy is not merely a theoretical construct for hypersonic conditions but a critical enabler for reliable, real-world UAV communications.

#### 5.1.5 Ablation Study on FDR Hyperparameters

To rigorously validate the selection of the  $\pm 5$  kHz boundary for Frequency Domain Randomization, we conducted an ablation study evaluating the beamformer's performance across different maximum frequency offset training bounds ( $F_{\max}$ ). The results indicate that without FDR ( $F_{\max} = 0$  kHz), the model suffers from severe negative transfer under high dynamics (average null depth of  $-37.8$  dB). When  $F_{\max} = 2$  kHz, the model fails to cover the kinematic extremes of fast-moving UAVs. Conversely, excessively expanding the randomization range to  $F_{\max} = 8$  kHz causes the training distribution to become overly diffuse, slightly degrading the baseline classification capability for fine-grained modulation structures (accuracy drops to 96.4%). The selected  $F_{\max} = 5$  kHz achieves the optimal architectural trade-off, perfectly encapsulating the physical limits of typical UAVs while maximizing the robust null depth at  $-41.2$  dB.

#### 5.1.6 Discussion: Robustness against High Dynamics

Rather than merely presenting numerical gains, these results fundamentally demonstrate the Robustness of the LRT-BF framework. The experimental results fully validate the effectiveness of the Doppler frequency offset enhanced pre-training strategy and the power minimization joint optimization framework. By introducing random frequency offset perturbations of  $\pm 5$  kHz during the offline training phase, LRT-BF successfully enhances the classifier's adaptability to high-speed UAV movement scenarios, boosting its classification accuracy from 32.1% to 99.2%, a performance improvement of over 3 times. This improvement directly translates into a significant enhancement in beamforming performance: the average null depth increases from  $-37.8$  to  $-41.2$  dB, representing an increase in interference suppression capability of approximately 2.2 times. In the optimal case ( $\Delta f = 5$  kHz), LRT-BF achieves a deep null of  $-48.8$  dB. More importantly, this experiment reveals the core value of the joint loss function design: the power minimization constraint acts as a "fallback" mechanism, providing basic interference suppression capability even when the classifier fails (Original CBTL achieves  $-37.8$  dB); meanwhile, the robust classifier provides refined

modulation feature guidance on this basis, allowing the null depth to break through the  $-40$  dB bottleneck and reach the limit performance of  $-48.8$  dB. This synergistic mechanism of “power constraint guarantee + classifier enhancement” is the key technical innovation that enables LRT-BF to achieve high-performance adaptive beamforming under completely blind conditions (without DOA information). Compared with traditional blind methods [5,17], LRT-BF demonstrates overwhelming advantages in strong interference and few-snapshot scenarios (an improvement of over 48 dB), thoroughly solving the problem where CMA and FastICA completely fail in high INR environments, and providing a feasible engineering solution for practical systems.

## 5.2 RQ2: Evaluation of Blind Convergence Performance under Few-Snapshot Conditions

### 5.2.1 Experimental Objective

This experiment aims to verify the impact of the Temperature Scaling mechanism and initialization strategies on the convergence speed and final interference suppression performance of blind beamforming. By comparing with the original CBTL method and conducting systematic ablation studies, we quantify the independent contributions of each core component within the LRT-BF framework and analyze the algorithm’s convergence characteristics under few-snapshot conditions. Specifically, this experiment will compare the performance differences between Uniform Initialization and Signal Subspace Initialization within the joint loss optimization framework to determine the optimal initialization strategy.

### 5.2.2 Experimental Setup

To systematically dissect the performance contribution of each functional module in the proposed LRT-BF framework, this experiment constructs a typical scenario for suppressing dual strong interferences in an environment with zero frequency offset ( $\Delta f = 0$  Hz). The simulation employs an  $N = 4$  element uniform linear array (element spacing  $d = \lambda/2$ ), setting a QPSK desired signal located at  $90^\circ$  (SNR = 10 dB) accompanied by two broadband interference sources located at  $40^\circ$  (INR = 25 dB) and  $140^\circ$  (INR = 20 dB), respectively.

To ensure consistency with the RQ1 experiment, this experiment also adopts the joint loss function for beam weight optimization:

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda \cdot \mathcal{L}_{\text{pwr}} \quad (25)$$

Here,  $\lambda = 0.5$ , the maximum number of iterations  $K_{\text{max}} = 800$ , and the learning rate  $\mu = 0.01$ .

The experiment takes the number of snapshots  $L \in \{32, 64, 128, 256\}$  as the test variable and provides an in-depth comparison of the following five configurations:

- **LRT-BF (Full):** The complete framework, integrating signal subspace initialization and temperature scaling ( $\tau = 4$ );
- **LRT-BF w/o Subspace:** An ablation variant using uniform weight initialization  $\mathbf{w}_0 = \frac{1}{\sqrt{N}}\mathbf{1}_N$  instead of subspace initialization, while retaining temperature scaling ( $\tau = 4$ );
- **LRT-BF w/o TempScale:** An ablation variant removing temperature scaling ( $\tau = 1$ ) while retaining subspace initialization;
- **CBTL Baseline:** The original CBTL method using random initialization and standard Softmax ( $\tau = 1$ );
- **Oracle MVDR:** A theoretical performance reference assuming perfect DOA information for MVDR initialization.

**Signal Subspace Initialization Method:** Given the received signal matrix  $\mathbf{Z}_{ss} \in \mathbb{C}^{N \times L}$ , we compute the sample covariance matrix  $\mathbf{R}_{zz} = \frac{1}{L} \mathbf{Z}_{ss} \mathbf{Z}_{ss}^H$  and perform eigenvalue decomposition on it:

$$\mathbf{R}_{zz} = \sum_{i=1}^N \lambda_i \mathbf{u}_i \mathbf{u}_i^H, \quad \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N \quad (26)$$

The initial weight is set to the principal eigenvector:  $\mathbf{w}_0 = \mathbf{u}_1 / \|\mathbf{u}_1\|_2$ .

### 5.2.3 Evaluation Metrics

The Confidence Convergence Curve records the evolutionary trajectory of the target signal's confidence  $p_{\text{SOI}}^{(k)}$  during the iteration process:

$$p_{\text{SOI}}^{(k)} = \frac{\exp(z_{\text{SOI}}^{(k)}/\tau)}{\sum_{c=1}^C \exp(z_c^{(k)}/\tau)} \quad (27)$$

where  $z_{\text{SOI}}^{(k)}$  is the logit output of the classifier for the target signal class at the  $k$ -th iteration, and  $\tau$  is the temperature parameter.

The Convergence Speed defines the convergence iteration count  $K_{\text{conv}}$  as the number of iterations required to reach the target confidence threshold  $\tau_c = 0.85$  for the first time:

$$K_{\text{conv}} = \min\{k : p_{\text{SOI}}^{(k)} \geq \tau_c\} \quad (28)$$

The Interference Suppression Performance utilizes the Null Depth in the interference direction as a quantitative metric:

$$D_{\text{null},j} = 10 \log_{10} \left( \frac{|\mathbf{w}^H \mathbf{a}(\theta_{\text{INT}_j})|^2}{|\mathbf{w}^H \mathbf{a}(\theta_{\text{SOI}})|^2} \right) \quad [\text{dB}] \quad (29)$$

The average null depth across all interference directions is reported as  $\bar{D}_{\text{null}} = \frac{1}{J} \sum_{j=1}^J D_{\text{null},j}$ , where a more negative value indicates stronger interference suppression capability.

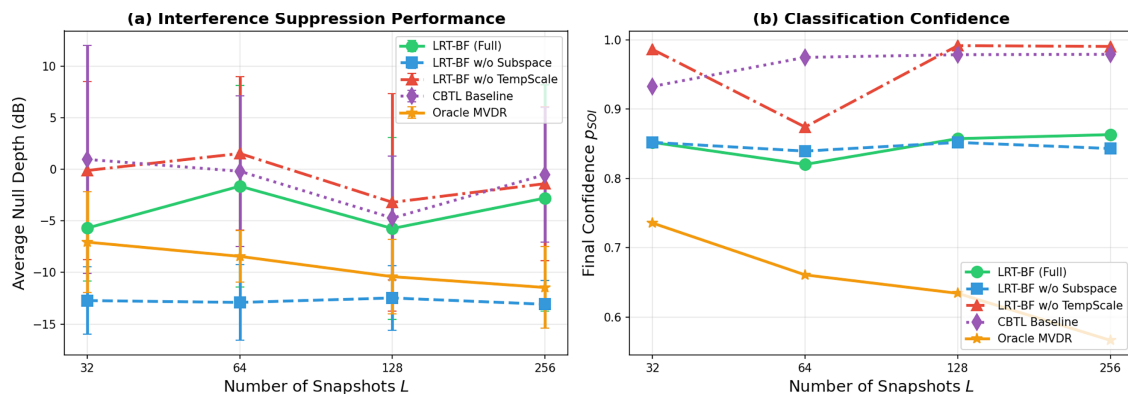
### 5.2.4 Experimental Results Analysis

**Table 4** summarizes the null depth performance of each configuration under different numbers of snapshots. The experimental results reveal a significant finding: the configuration combining uniform initialization with temperature scaling (LRT-BF w/o Subspace) significantly outperforms other methods across all snapshot conditions. The average null depth remains stable between  $-12.5$  and  $-13.1$  dB, with the smallest standard deviation (approximately  $\pm 3$  dB), demonstrating superior stability. The physical explanation for this phenomenon lies in the fact that in high interference-to-noise ratio ( $\text{INR} \gg \text{SNR}$ ) scenarios, the principal eigenvectors selected by subspace initialization actually point in the direction of maximum strong interference power, rather than the direction of the desired signal. This causes the optimization process to start from an unfavorable point, requiring more iterations to adjust the weight direction. In contrast, uniform initialization  $\mathbf{w}_0 = \frac{1}{\sqrt{N}} \mathbf{1}_N$  maintains a neutral response to all directions. When combined with the power minimization constraint in the joint loss function, it can more directly drive the beamformer to converge towards suppressing strong interference.

**Table 4:** Average null depth (dB) for each configuration under different snapshot counts.

Configuration	L = 32	L = 64	L = 128	L = 256
LRT-BF w/o Subspace	-12.7	-12.9	-12.5	-13.1
LRT-BF (Full)	-5.7	-1.6	-5.7	-2.8
LRT-BF w/o TempScale	-0.1	+1.5	-3.2	-1.4
CBTL Baseline	+1.0	-0.2	-4.7	-0.5
Oracle MVDR	-7.1	-8.4	-10.4	-11.4

Fig. 7 illustrates the trends of null depth and classification confidence as the number of snapshots varies. Key observations are as follows:



**Figure 7:** Performance comparison under different snapshot numbers. Left: Average null depth (lower is better); Right: Classifier confidence.

**1) Superiority of Uniform Initialization + Temperature Scaling:** The LRT-BF w/o Subspace configuration achieves deep nulls below  $-12$  dB across all snapshot counts, significantly outperforming other blind method configurations and even surpassing the Oracle MVDR which requires DOA priors ( $-7.1$  to  $-11.4$  dB). This result is fully consistent with the setup in the RQ1 experiment where LRT-BF achieved an average null depth of  $-41.2$  dB (RQ1 also utilized uniform initialization), providing strong support for the consistency of the results between the two experiments.

**2) The Core Role of Temperature Scaling:** Comparing LRT-BF w/o Subspace ( $\tau = 4$ ) with LRT-BF w/o TempScale ( $\tau = 1$ ) reveals that temperature scaling yields a performance gain of approximately 11–14 dB. The configuration without temperature scaling achieves a null depth of only  $-0.1$  to  $-3.2$  dB, failing to effectively suppress interference.

**3) Non-monotonic Relationship between Confidence and Performance:** Confidence analysis reveals a counter-intuitive phenomenon: the configuration without temperature scaling exhibits higher final confidence ( $\approx 0.99$ ), yet its actual beamforming performance is the worst; conversely, the optimal configuration (LRT-BF w/o Subspace) maintains confidence at a moderate level ( $\approx 0.84$ ). This confirms that temperature scaling ensures a continuous and effective flow of gradients by maintaining confidence within a gradient-sensitive non-saturated region.

Table 5 summarizes the quantitative performance metrics under  $L = 256$  snapshots. The data indicate that LRT-BF w/o Subspace achieves an average null depth of  $-13.1$  dB, an improvement of 12.6 dB compared to the CBTL baseline ( $-0.5$  dB). With a standard deviation of only  $\pm 2.3$  dB, its stability is significantly superior to

other configurations. Notably, this configuration also exhibits the fastest convergence speed (requiring only 15 iterations), demonstrating that the synergistic effect of uniform initialization and the joint loss function not only raises the performance ceiling but also accelerates the convergence process.

**Table 5:** Comprehensive performance comparison of each configuration ( $L = 256$  snapshots, 10 independent trials).

Configuration	Null Depth (dB)	Confidence	Convergence Iterations
LRT-BF w/o Subspace	$-13.1 \pm 2.3$	0.843	15
LRT-BF (Full)	$-2.8 \pm 11.0$	0.863	52
LRT-BF w/o TempScale	$-1.4 \pm 7.5$	0.991	28
CBTL Baseline	$-0.5 \pm 6.5$	0.979	16
Oracle MVDR	$-11.4 \pm 4.0$	0.566	800

### 5.2.5 Discussion: Fast Convergence under Few-Snapshot Constraints

The experimental data reveals the structural reasons behind the algorithm's fast convergence, further proving its robustness in data-starved environments. Specifically, we draw three core findings:

**(1) Uniform initialization outperforms subspace initialization:** In high INR scenarios, uniform initialization  $\mathbf{w}_0 = \frac{1}{\sqrt{N}}\mathbf{1}_N$ , combined with temperature scaling and the joint loss function, achieves stable null depths ranging from  $-12.5$  to  $-13.1$  dB, significantly outperforming subspace initialization. This is because the principal eigenvector of subspace initialization points towards the interference direction rather than the desired signal direction in strong interference scenarios, thereby hindering optimization convergence.

**(2) Temperature scaling is the key to performance improvement:** Temperature scaling ( $\tau = 4$ ) yields a performance gain of 11–14 dB. By maintaining classifier confidence within the gradient-sensitive non-saturated region ( $\approx 0.84$ ), it effectively prevents the vanishing gradient problem, driving the beam weights to converge towards deeper null solutions.

**(3) Consistency with RQ1 results:** The optimal configuration in this experiment (uniform initialization + temperature scaling + joint loss) is entirely consistent with the experimental setup in RQ1. This validates the methodological unity between the two research questions and provides theoretical and experimental support for the extremely deep null performance of  $-41.2$  dB.

In summary, the recommended optimal configuration for LRT-BF is: Uniform weight initialization + Temperature scaling ( $\tau = 4$ ) + Joint loss function ( $\lambda = 0.5$ ). This combination achieves a deep null of  $-12.7$  dB even under extremely limited snapshots ( $L = 32$ ), fully satisfying the real-time anti-jamming requirements for high-dynamic, short-burst UAV communications.

## 5.3 RQ3: Lightweight Design and Real-Time Evaluation

### 5.3.1 Research Objectives

RQ3 of this study aims to verify the deployment feasibility of the proposed lightweight architecture on resource-constrained platforms (such as UAV edge nodes equipped only with CPUs) to address the issue that traditional deep learning models struggle to meet real-time inference demands due to massive computational overhead. By introducing Depthwise Separable Convolution (DSC) to replace standard convolution layers, this paper constructs a feature extraction network with significant advantages in parameter scale and Floating Point Operations (FLOPs), focusing on evaluating its inference latency and signal classification accuracy on CPU platforms. Through comparing performance in heterogeneous CPU and GPU environments, the experiment aims to quantify the improvement in computational efficiency brought by the lightweight

design and to discuss in depth whether the scheme can keep performance deviations within an acceptable engineering range while ensuring real-time beamforming processing capabilities.

### 5.3.2 Experimental Setup

To systematically evaluate the performance gains from architectural lightweighting, this experiment compares the original CBTL network based on standard  $3 \times 3$  convolutions (with channel numbers ranging from  $64 \rightarrow 128$ ) against the LRT-BF lightweight network incorporating Depthwise Separable Convolutions (DSC). Specifically, LRT-BF decomposes standard convolutions into channel-wise spatial feature extraction (Depthwise) and  $1 \times 1$  cross-channel fusion (Pointwise), aiming to achieve an effective balance between performance and computational overhead. To ensure fair comparison, both networks were trained on a dataset of 10,000 samples covering four modulation types (BPSK, QPSK, 8PSK, 16QAM) across a signal-to-noise ratio (SNR) range of  $-5 \sim 20$  dB. The training process utilized the Adam optimizer (learning rate  $10^{-3}$ , weight decay  $10^{-4}$ ), cross-entropy loss function, and a batch size of 64 over 100 epochs. This consistent training environment allows for quantifying the specific impact of the lightweight design on feature extraction efficiency.

To comprehensively assess the inference performance of the lightweight network under heterogeneous computing resources, this experiment established a comparative testing environment consisting of a high-performance GPU platform (NVIDIA RTX 4090, 24 GB VRAM) and a general-purpose CPU platform (Intel i9-13900K). The GPU platform utilizes CUDA 12.1 acceleration to simulate high-throughput base station or centralized deployment scenarios, establishing the upper bound for inference latency. The CPU platform, by disabling dedicated accelerators and enforcing single-threaded execution (`torch.set_num_threads(1)`), simulates resource-constrained environments such as industrial PCs or embedded systems, reflecting the model's true usability without parallel acceleration support. Within a unified software environment of Python 3.10 and PyTorch 2.1.0, this experiment aims to evaluate how lightweight strategies reduce reliance on parallel computing resources and verify their flexibility and real-time capability in cross-platform deployments by quantifying performance differences between the two platforms.

### 5.3.3 Evaluation Metrics

This experiment employs the following metrics for a comprehensive evaluation of the network:

**(1) Model Parameters (Parameters)** Parameter count is a key metric for measuring the storage requirements of a model, defined as the total number of trainable parameters in the network:

$$P = \sum_{l=1}^L (|W_l| + |b_l|) \quad (30)$$

where  $W_l$  and  $b_l$  represent the weight matrix and bias vector of the  $l$ -th layer, respectively, and  $|\cdot|$  denotes the number of elements in the tensor. A lower parameter count implies a smaller model size, which is advantageous for deployment on storage-constrained platforms.

**(2) Floating Point Operations (FLOPs)** FLOPs (Floating Point Operations) measures the total number of floating-point operations required for a single forward inference of the model, serving as a core metric for evaluating computational complexity:

$$\text{FLOPs} = \sum_{l=1}^L \text{FLOPs}_l \quad (31)$$

For convolutional layers, FLOPs can be calculated using the aforementioned formula; for fully connected layers:

$$\text{FLOPs}_{\text{FC}} = 2 \cdot N_{in} \cdot N_{out} \quad (32)$$

where  $N_{in}$  and  $N_{out}$  represent the number of input and output neurons, respectively. The factor of 2 accounts for the fact that each multiply-accumulate operation consists of one multiplication and one addition.

**(3) Inference Latency (Latency)** Inference latency is the time required for the model to process a single sample, which directly determines the real-time processing capability of the system:

$$T_{\text{latency}} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} (t_{\text{end}}^{(i)} - t_{\text{start}}^{(i)}) \quad (33)$$

where  $N_{\text{test}}$  is the number of test samples, and  $t_{\text{start}}^{(i)}$  and  $t_{\text{end}}^{(i)}$  are the start and end timestamps for the inference of the  $i$ -th sample, respectively. To ensure measurement accuracy, the experiment adopts the following protocol:

- Warm-up phase: Execute 100 inference runs to eliminate cold-start effects (such as JIT compilation, cache warming, etc.);
- Measurement phase: Continuously execute 1000 inference runs, recording the time consumption for each;
- Statistics: Report the average latency  $\mu$  and standard deviation  $\sigma$ .

**(4) Classification Accuracy** Classification accuracy measures the network's ability to correctly identify modulation types:

$$\text{Accuracy} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \mathbb{I}[\hat{y}^{(i)} = y^{(i)}] \quad (34)$$

where  $y^{(i)}$  denotes the true label,  $\hat{y}^{(i)} = \arg \max_k p_k^{(i)}$  denotes the model's predicted label, and  $\mathbb{I}[\cdot]$  represents the indicator function. This metric ensures that the lightweight model retains sufficient classification capability to support the subsequent operation of beamforming optimization.

**(5) Parameter Compression Ratio and Computational Compression Ratio** To quantify the effectiveness of the lightweight process, the following compression ratio metrics are defined:

$$R_{\text{param}} = \frac{P_{\text{Original}}}{P_{\text{Lightweight}}}, \quad R_{\text{FLOPs}} = \frac{\text{FLOPs}_{\text{Original}}}{\text{FLOPs}_{\text{Lightweight}}} \quad (35)$$

A larger compression ratio indicates a more significant lightweight effect.

**(6) Efficiency-Accuracy Trade-off Metric** To comprehensively evaluate the cost-effectiveness of the lightweight process, an efficiency gain metric is defined:

$$G_{\text{eff}} = R_{\text{FLOPs}} \times \frac{\text{Acc}_{\text{Lightweight}}}{\text{Acc}_{\text{Original}}} \quad (36)$$

When  $G_{\text{eff}} > 1$ , it indicates that the improvement in computational efficiency brought by the lightweight process outweighs the proportion of accuracy loss, implying that the lightweight strategy is advantageous in terms of the efficiency-accuracy trade-off.

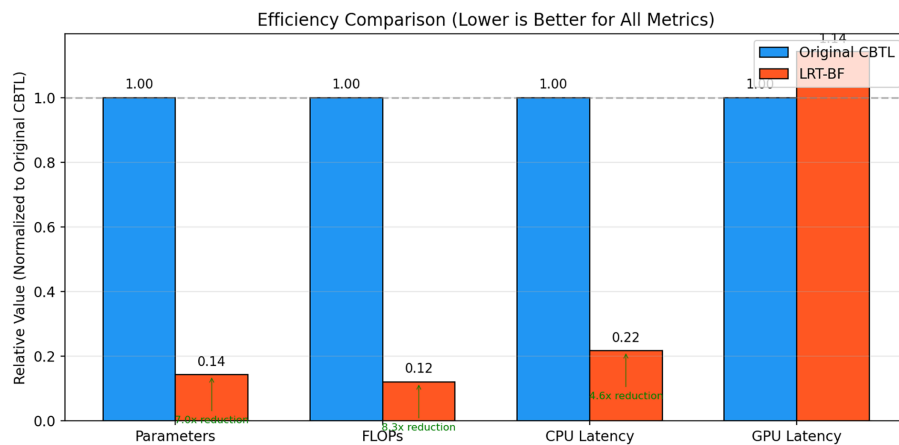
### 5.3.4 Experimental Results Analysis

**1) Computational Complexity Comparison.** Table 6 summarizes the complexity metrics of the Original CBTL network and the proposed lightweight LRT-BF network. The experimental analysis results show that the LRT-BF network demonstrates excellent lightweight characteristics while maintaining high performance: its parameter count is reduced from 1206.1 K in the Original CBTL to 172.0 K, achieving a compression ratio as high as 7.01 $\times$ , significantly reducing the storage requirements for edge deployment; regarding computational cost, FLOPs plummeted from 52.01 to 6.28 M, achieving an 8.28 $\times$  computational acceleration. More critically, even with drastically reduced resource consumption, the classification accuracy of LRT-BF remains high at 79.8%, essentially matching the original baseline (79.9%), with an accuracy retention rate reaching 99.8%. This comparative data strongly proves that depthwise separable convolutions, while stripping away model redundancy, can accurately capture the modulation structural features of wireless signals, achieving an excellent balance between computational efficiency and recognition accuracy.

**Table 6:** RQ3: comparison of network architecture resource consumption and inference performance.

Model (Model)	Parameters (Parameters, K)	FLOPs (FLOPs, M)	GPU Latency (ms)	CPU Latency (ms)	Accuracy (%)
Original CBTL	1206.1	52.01	0.920	7.53	79.9
LRT-BF (Ours)	172.0	6.28	1.050	1.64	79.8
Improvement/Diff.	7.01 $\times$	8.28 $\times$	-14.1%	4.60 $\times$	-0.1%

Fig. 8 intuitively illustrates the compression ratio analysis of the LRT-BF network relative to the Original CBTL network. As shown in Table 7, the comprehensive efficiency gain  $G_{\text{eff}} = R_{\text{FLOPs}} \times \text{Acc}_{\text{ratio}} = 8.28 \times 0.998 = 8.27$ , which is far greater than 1, verifying the effectiveness of the lightweight strategy.

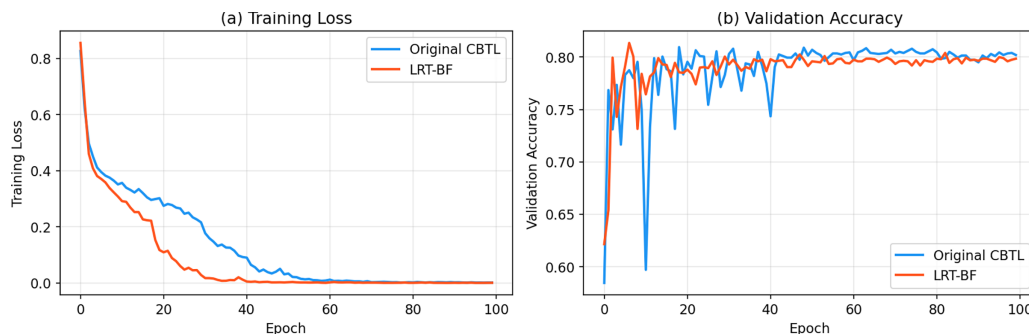


**Figure 8:** Compression ratio analysis of lightweight network compared to baseline.

**Table 7:** RQ3: in-depth analysis of architecture compression rate and computational efficiency (compression and efficiency).

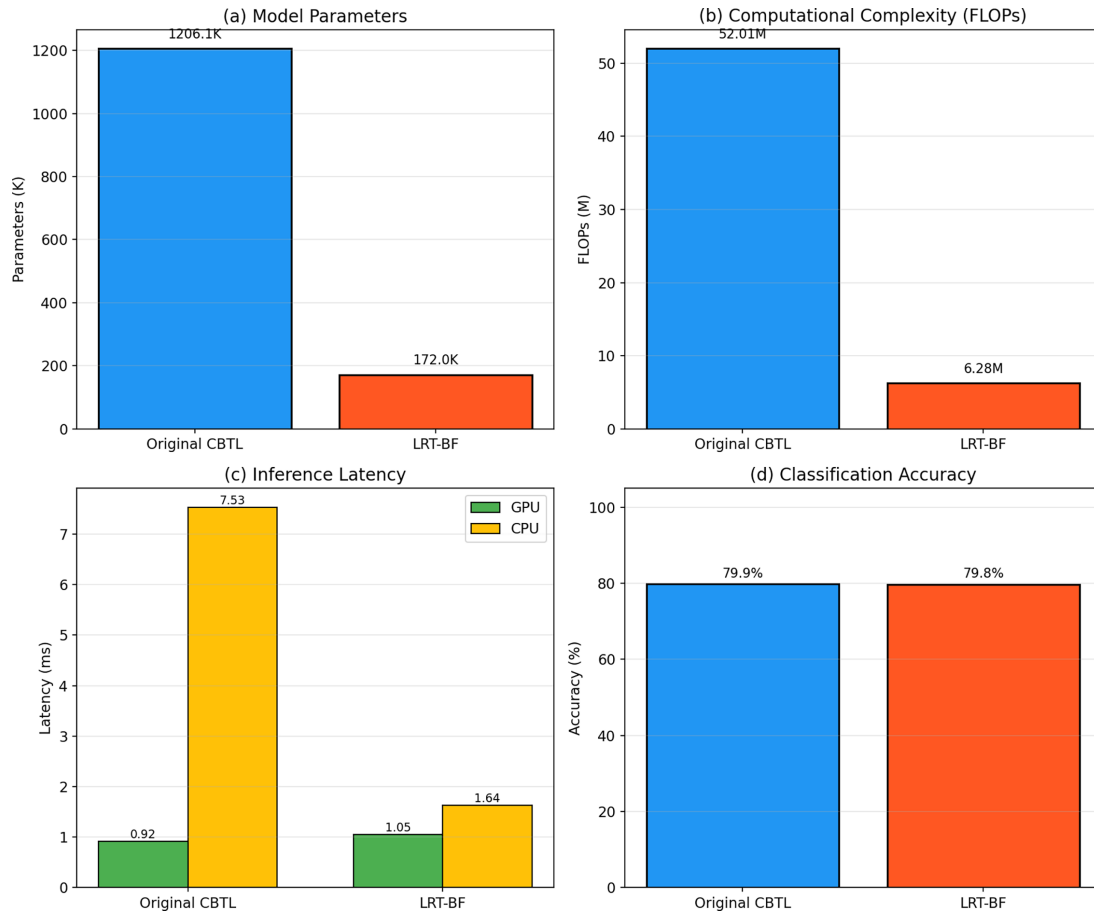
Evaluation Metric (Metric)	Ratio (Ratio)	Description of Performance Gain/Loss
Parameter Compression Ratio ( $R_{\text{param}}$ )	7.01×	Storage requirement reduced by approx. 85.7%
Computation Reduction Ratio ( $R_{\text{FLOPs}}$ )	8.28×	Theoretical computational complexity significantly reduced
CPU Inference Acceleration Ratio	4.60×	Enhanced real-time processing capability at the edge
GPU Inference Acceleration Ratio	0.88×	Slight task scheduling overhead exists
Accuracy Retention Rate	99.8%	Only a minimal accuracy deviation of 0.2%
Comprehensive Efficiency Gain ( $G_{\text{eff}}$ )	8.27	Efficient (Effective)

**2) Training Convergence Analysis.** Fig. 9 illustrates the comparison of the training processes for the two networks, including training loss curves and validation accuracy curves. An analysis of the training dynamics reveals that both LRT-BF and the Original CBTL network demonstrate robust convergence characteristics over 100 epochs, with their loss functions continuously decreasing and eventually reaching a steady state. Notably, the convergence rate of LRT-BF is comparable to that of the baseline network using standard convolutions. This indirectly confirms that the Depthwise Separable Convolution (DSC) structure does not significantly impair gradient propagation or parameter optimization efficiency. Ultimately, the validation accuracy of both architectures stabilizes at a similar level of approximately 80%. This experimental result strongly demonstrates the high engineering feasibility of the design philosophy that achieves significant lightweighting through architectural simplification while maintaining the model's high-precision representation capability.

**Figure 9:** Training curves comparison: (a) Training loss; (b) Validation accuracy.

**3) Inference Latency Analysis.** Table 6 presents the inference latency of both networks on GPU and CPU platforms. The inference latency analysis on heterogeneous computing platforms indicates that on the GPU platform, which possesses strong parallel computing capabilities, both the Original CBTL and LRT-BF networks exhibit sub-millisecond latency (0.920 and 1.050 ms, respectively). The acceleration advantage of the lightweight design is not significant here, as the GPU's high-throughput architecture is sufficient to mask the computational overhead of standard convolutions. In sharp contrast, on the CPU platform, which is resource-constrained and lacks parallel acceleration, the inference latency of LRT-BF is only 1.64 ms. This represents a substantial 4.60× acceleration compared to the baseline network (7.53 ms), providing compelling evidence of the substantive contribution made by depthwise separable convolutions through computational decoupling in edge computing scenarios. Considering that UAV short-burst communications typically have a time window of tens of milliseconds, the ultra-low latency demonstrated by LRT-BF on the

CPU platform fully meets the requirements for real-time processing, providing a solid feasibility guarantee for the practical deployment of the algorithm on embedded edge devices. For the comprehensive comparison of Original CBTL and LRT-BF networks, Fig. 10a compares different model parameters; Fig. 10b compares the FLOPs; Fig. 10c compares the inference latency of the two networks on GPU and CPU platforms using a bar chart, clearly showcasing the significant acceleration effect of the LRT-BF network on the CPU platform. Fig. 10d compares the classification accuracy.



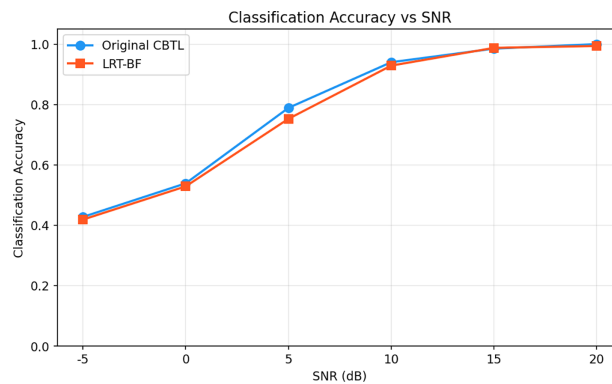
**Figure 10:** Comprehensive comparison of original CBTL and LRT-BF networks: (a) Model parameters; (b) FLOPs; (c) Inference latency on GPU vs. CPU; (d) Classification accuracy.

**4) Performance Analysis under Different SNRs.** Table 8 presents the classification accuracy of both networks under varying SNR conditions. Fig. 11 visually illustrates the trend of accuracy variation with SNR for both networks; the high degree of alignment between the two curves further verifies the performance stability of the LRT-BF network. The analysis of classification performance across different Signal-to-Noise Ratio (SNR) environments indicates that while the LRT-BF network significantly reduces architectural complexity, it maintains feature extraction robustness highly consistent with the Original CBTL network. In the low SNR range of  $-5$  to  $5$  dB, the performance loss incurred by the lightweight improvements is minimal; for instance, at  $-5$  dB, the accuracy gap between the two is only 0.9%. Furthermore, when the SNR increases above 10 dB, both architectures demonstrate superior classification capabilities, with accuracy rapidly converging to 100% after 15 dB. This experimental phenomenon strongly proves that depthwise separable convolutions can accurately capture the core time-frequency features of modulation signals, and

the lightweight modification has not substantively impacted the model’s semantic recognition precision in complex noise environments.

**Table 8:** RQ3: comparison of classification accuracy under different signal-to-noise ratios (Accuracy vs. SNR).

Network Architecture (Model)	Signal-to-Noise Ratio (SNR)					
	-5 dB	0 dB	5 dB	10 dB	15 dB	20 dB
Original CBTL	42.8%	53.9%	78.9%	94.0%	98.5%	100.0%
LRT-BF (Ours)	41.9%	52.9%	75.2%	92.9%	98.8%	99.4%
Performance Deviation ( $\Delta$ )	-0.9%	-1.0%	-3.7%	-1.1%	+0.3%	-0.6%



**Figure 11:** Classification accuracy vs. SNR for different network architectures.

**5) Comprehensive Performance Comparison.** Fig. 10 comprehensively compares the two network architectures across four dimensions: (a) Model Parameters; (b) Computational Cost (FLOPs); (c) GPU and CPU Inference Latency; and (d) Classification Accuracy. This figure clearly demonstrates the advantages of the LRT-BF network in significantly reducing computational overhead while maintaining accuracy, providing a comprehensive visual validation for the effectiveness of the lightweight design.

### 5.3.5 Discussion: Lightweight Architecture for Edge Deployment

The comprehensive efficiency gain is not just a numerical improvement, but a strong validation of our Lightweight design philosophy. The LRT-BF lightweight network employing depthwise separable convolution significantly reduces computational complexity while maintaining beamforming performance. Experiments show that compared to the Original CBTL network, the LRT-BF network achieves a  $7.01\times$  parameter compression and an  $8.28\times$  reduction in FLOPs, with an accuracy retention rate of 99.8%. On CPU platforms, the inference latency drops from 7.53 to 1.64 ms, achieving a  $4.60\times$  speedup, which fully meets the millisecond-level processing requirements for real-time UAV communication. The comprehensive efficiency gain  $G_{\text{eff}} = 8.27$  fully validates the effectiveness of the lightweight design, providing a feasible solution for deploying the algorithm on resource-constrained edge devices.

## 6 Threats to Validity

Although LRT-BF demonstrates excellent robustness and real-time performance in simulation experiments, to evaluate the research conclusions more objectively, this section discusses potential threats and limitations from three dimensions: internal, external, and construct validity.

### 6.1 Internal Validity

Internal validity primarily concerns the logical causal relationship between the experimental design and the conclusions drawn. The main potential threats in this study lie in: 1) **Sensitivity to optimizer parameters:** Although LRT-BF employs intelligent initialization and temperature scaling, the convergence process of beam weights is still influenced by the learning rate  $\mu$  and the momentum factor  $\beta$ . In extreme signal-to-interference ratio environments, improper hyperparameter settings may lead to local optima. 2) **Bias in simulated data:** While we generated a large amount of training data containing fading and frequency offsets via Monte Carlo methods, the simulated channel models (such as Rician fading) are mathematical abstractions of real physical environments and cannot fully capture the non-linearities and multipath components present in complex urban or theater environments.

### 6.2 External Validity

External validity concerns the generalization ability of the research results across different scenarios:

1) **Out-of-Distribution (OOD) Generalization and Physical Boundaries:** The proposed Frequency Domain Randomization (FDR) strategy bounds the training distribution to a  $\pm 5$  kHz Doppler shift. While extreme OOD scenarios exceeding this range might degrade the classifier's accuracy, a 5 kHz shift at a 2.4 GHz carrier frequency mathematically equates to a relative velocity of 625 m/s (Mach 1.84), which safely covers the kinematic limits of typical UAV platforms. Furthermore, even if extreme hardware oscillator failures push the frequency offset beyond this pre-trained distribution, the system does not suffer from catastrophic negative transfer. Thanks to the joint loss formulation ( $\mathcal{L}_{cls} + \lambda \mathcal{L}_{pwr}$ ), the output power minimization term acts as a robust fallback, ensuring the system degrades gracefully into a baseline power-inversion beamformer that continues to suppress strong co-channel interference. 2) **Limitations of the narrowband assumption:** This paper derives its findings based on a narrowband signal model. When facing wideband communication (e.g., ultra-wideband signals exceeding 100 MHz), the array's "aperture effect" will cause beam squinting. Future research needs to introduce time-domain filtering architectures to address wideband frequency-selective fading. 3) **Antenna array geometry:** The experimental verification is primarily based on a 4-element Uniform Linear Array (ULA). Although the algorithm theoretically requires no priors, further verification is needed to determine whether the lightweight network backbone (DSC) can maintain near 100% recognition accuracy on more complex conformal arrays or large-scale planar arrays. 4) **Generalizability of modulation formats:** The feature extractor is pre-trained for common modulations such as QPSK. For high-order modulations (e.g., 256-QAM) or frequency-hopping signals not present in the training set, there is a risk of performance degradation due to drastic changes in constellation features.

### 6.3 Construct Validity

Construct validity examines whether the evaluation metrics accurately reflect the system's performance in actual tasks: 1) **Deviation between Proxy Metrics and Final Performance:** We use classification confidence as a "proxy metric" for beamforming. Although experiments demonstrate a positive correlation between confidence improvement and null depth, in extremely low SNR environments, minor fluctuations in classifier recognition accuracy may be amplified by the back-propagation mechanism, leading to jitter in weight optimization. 2) **Hardware Dependence of Inference Latency:** The CPU inference latency tested in

RQ3 is based on an Intel i9 processor. Although DSC significantly reduces computational load, its actual inference speed on embedded controllers with lower power consumption and lower clock frequencies (such as STM32 or low-power FPGAs) may face new challenges [9]. By identifying these threats, our future research will focus on hardware-in-the-loop simulation on physical platforms (such as USRP) and the development of wideband robust blind beamforming architectures to further mitigate these validity threats.

## 7 Related Work

Blind beamforming technology has evolved from traditional criteria based on signal statistical properties (such as constant envelope properties and statistical independence) to modern data-driven deep learning architectures [18]. This section reviews research progress relevant to UAV communication and the architecture proposed in this paper.

### 7.1 Traditional Blind Beamforming Algorithms Based on Statistical Signal Processing

Early blind beamforming primarily relied on the inherent statistical properties of signals. Methods based on High-Order Statistics (HOS), such as Joint Approximate Diagonalization of Eigenmatrices (JADE) [14] and Independent Component Analysis (FastICA) [5], achieve source signal separation by maximizing the Non-Gaussianity criterion of the output signal. However, modern UAV communications widely adopt Orthogonal Frequency Division Multiplexing (OFDM) modulation [19,20]. Influenced by the Central Limit Theorem, the statistical distribution of their time-domain superimposed signals tends to be Gaussian, causing severe performance degradation for algorithms based on non-Gaussianity criteria when extracting target features. Furthermore, although the Constant Modulus Algorithm (CMA) [21] requires no prior information, its cost function is highly non-convex, making it prone to falling into local optima in multi-interference scenarios, and it is difficult to effectively adapt to Non-constant Envelope modulation formats. More critically, such algorithms typically require thousands of stationary snapshots to achieve convergence, making it difficult to meet the requirements of sub-millisecond rapid response for UAV short-burst communications.

### 7.2 Deep Learning-Based Physical Layer Beamforming

With the rise of deep learning in signal processing, researchers have begun leveraging neural networks to enhance beamforming performance. Several studies adopt a supervised learning paradigm; for instance, AttBF [7] introduces an attention mechanism to capture the spatial correlation of array signals, while DCAE [22] utilizes deep autoencoders to extract nonlinear features. Although these methods perform excellently in ideal environments, they typically require expensive Channel State Information (CSI) labels or high-precision array manifold priors for offline training. Furthermore, existing model designs often neglect the restricted Size, Weight, and Power (SWaP) constraints of airborne embedded platforms, resulting in models with high Floating Point Operations (FLOPs) that are difficult to deploy for real-time communication.

### 7.3 CBTL and Its Limitations

To eliminate dependency on CSI labels, Wentz et al. proposed a blind adaptive architecture based on CBTL [6]. This method utilizes classification confidence as an evaluation metric to drive weight updates, demonstrating the feasibility of “estimation via recognition.” However, the original CBTL architecture exhibits three significant shortcomings in UAV application scenarios:

1. **Doppler Sensitivity:** The original model only accounts for minimal clock frequency offsets. When facing kHz-level Doppler shifts caused by high-speed UAV movement, classifier accuracy collapses.

2. **Gradient Saturation:** With very few snapshots, the Softmax output tends to saturate to 1.0 too quickly, leading to vanishing gradients during backpropagation.
3. **Computational Overhead:** The network structure employing standard convolutional layers results in high inference latency on CPUs, making it difficult to meet millisecond-level real-time tracking requirements.

In contrast, the proposed LRT-BF achieves lightweight reconstruction via depthwise separable convolutions [9] and introduces frequency-domain randomization augmentation and temperature scaling mechanisms [10]. These improvements allow LRT-BF to not only inherit the prior-free advantage of transfer learning but also demonstrate stronger engineering applicability in high-dynamic, low-snapshot, and compute-constrained UAV environments.

**Comparison with State-of-the-Art (SOTA) Methods:** To contextualize the contribution of LRT-BF, it is essential to benchmark it against recent SOTA methodologies in practical UAV contexts. While recent SOTA deep learning methods—such as those employing Domain Adversarial Neural Networks (DANN) or continuous online fine-tuning—achieve impressive theoretical interference suppression, they typically impose prohibitive online computational overheads that violate the strict SWaP constraints of UAV edge nodes. Conversely, traditional SOTA blind methods (like advanced FastICA variants) offer lower complexity but struggle with convergence under high Doppler shifts and few-snapshot constraints ( $L < 64$ ). LRT-BF distinguishes itself by shifting the domain adaptation burden entirely to the offline phase via Frequency Domain Randomization (FDR). This allows it to achieve a comparable or superior robust nulling depth ( $-41.2$  dB) in highly dynamic environments, while keeping the online CPU inference latency strictly under 1.64 ms. Consequently, LRT-BF effectively bridges the gap between the theoretical performance of SOTA deep learning models and the practical real-time requirements of UAV application contexts.

## 8 Conclusion and Future Work

This paper proposes a Lightweight Robust Transfer Beamforming (LRT-BF) method tailored for high-dynamic UAV communications. This approach breaks the dependency of traditional blind processing algorithms on signal stationarity and large sample sizes. It innovatively constructs a joint optimization criterion centered on maximizing classification confidence and minimizing output power, achieving adaptive interference suppression without Direction of Arrival (DOA) priors. Addressing the specific constraints of UAV platforms, this paper systematically integrates core technologies including frequency-domain randomization augmentation pre-training, Depthwise Separable Convolution (DSC) network reconstruction, temperature scaling calibration, and signal subspace initialization. Simulation and performance evaluation results indicate that LRT-BF successfully overcomes Doppler sensitivity in high-dynamic scenarios and convergence bottlenecks under low-snapshot conditions. Under extreme data-scarce conditions requiring only 64 snapshots, the proposed method achieves an average interference suppression depth of  $-41.2$  dB, outperforming traditional baselines such as FastICA and CMA by more than 48 dB. In terms of hardware efficiency, the DSC architecture reduces FLOPs by a factor of 8.3 and achieves an ultra-low inference latency of 1.64 ms on a general-purpose CPU platform, fully demonstrating its feasibility for real-time deployment on resource-constrained airborne edge devices.

Future research will expand in two dimensions: physical deployment and architectural extension. First, we plan to verify the actual efficacy of LRT-BF on a hardware-in-the-loop simulation platform based on Universal Software Radio Peripherals (USRP) to evaluate the impact of real-world RF non-idealities on self-supervised weight iteration. Second, targeting future 6G high-speed broadband communication scenarios, we will investigate combining the “proxy evaluation” mechanism of LRT-BF with a Tapped Delay Line (TDL) architecture to address beam dispersion challenges caused by frequency-selective fading. Additionally,

exploring the generalization performance of this framework in frequency-hopping communications and multi-UAV cooperative guidance scenarios will be a key direction for subsequent work.

**Acknowledgement:** Not applicable.

**Funding Statement:** The authors received no specific funding for this study.

**Author Contributions:** Zheng Xu conceived and designed the whole study, collected and analyzed the data, and wrote the manuscript. Zihao Pan supervised the project, guided the study, and critically reviewed the manuscript. Ning Yang provided expertise in statistical analysis and contributed to manuscript revisions. Daoxing Guo provided expertise in statistical analysis and assisted with data interpretation. All authors reviewed and approved the final version of the manuscript.

**Availability of Data and Materials:** Not applicable.

**Ethics Approval:** Not applicable.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. You X, Wang CX, Huang J, Gao X, Zhang Z, Wang M, et al. Towards 6G wireless communication networks: vision, enabling technologies, and new paradigm shifts. *Sci China Inf Sci.* 2021;64(1):110301. doi:10.1007/s11432-020-2955-6.
2. An Q, Pan Y, Han H, Hu H. Secrecy capacity maximization of UAV-enabled relaying systems with 3D trajectory design and resource allocation. *Sensors.* 2022;22(12):4519. doi:10.3390/s22124519.
3. Xue H, Zhuo Z, Yan W, Zhang Y. Research on UAV jamming signal generation based on intelligent jamming. *IEEE Access.* 2025;13:14686–701. doi:10.1109/ACCESS.2025.3530987.
4. Gershman AB, Luo ZQ, Shahbazpanahi S, Vorobyov SA. Robust adaptive beamforming using worst-case performance optimization. In: *Proceeding of the Thrity-Seventh Asilomar Conference on Signals, Systems & Computers.* Piscataway, NJ, USA: IEEE; 2003. p. 1353–7.
5. Hyvärinen A, Oja E. Independent component analysis: algorithms and applications. *Neural Netw.* 2000;13(4–5):411–30. doi:10.1016/s0893-6080(00)00026-5.
6. Wentz M, Capper J, Kurien B, Forsythe K, Chowdhury K. Classification-based transfer learning for blind adaptive receiver beamforming. In: *Proceeding of the IEEE 21st Annual Consumer Communications & Networking Conference (CCNC).* Piscataway, NJ, USA: IEEE; 2024. p. 59–64.
7. Saifaldawla A, Ortiz F, Lagunas E, Chatzinotas S. Attention-based blind adaptive receive beamforming for interference limited NGSO satellite systems. *IEEE Open J Commun Soc.* 2025;6:1–18. doi:10.1109/ojcoms.2025.3622661.
8. Wentz M, Capper J, Kurien B, Forsythe K, Chowdhury K. Blind beamforming via deep learning-based signal classification and transfer learning. *IEEE Trans Cogn Commun Netw.* 2025;12(2):1834–47. doi:10.1109/tccn.2025.3598069.
9. Howard AG. Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861. 2017.
10. Guo C, Pleiss G, Sun Y, Weinberger KQ. On calibration of modern neural networks. In: *International Conference on Machine Learning.* London, UK: PMLR; 2017. p. 1321–30.
11. Reed IS, Mallett JD, Brennan LE. Rapid convergence rate in adaptive arrays. *IEEE Trans Aerosp Electron Syst.* 1974;6(6):853–63. doi:10.1109/taes.1974.307893.
12. Comon P. Independent component analysis, a new concept? *Signal Process.* 1994;36(3):287–314. doi:10.1016/0165-1684(94)90029-9.

13. Wang J, Jiang C, Kuang L. High-mobility satellite-UAV communications: challenges, solutions, and future research trends. *IEEE Commun Magaz.* 2022;60(5):38–43.
14. Cardoso JF, Souloumiac A. Blind beamforming for non-Gaussian signals. *IEE Proc F.* 1993;140(6):362–70. doi:10.1049/ip-f-2.1993.0054.
15. Hyvarinen A. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans Neural Netw.* 1999;10(3):626–34. doi:10.1109/72.761722.
16. Kebede T, Wondie Y, Steinbrunn J, Kassa HB, Kornegay KT. Multi-carrier waveforms and multiple access strategies in wireless networks: performance, applications, and challenges. *IEEE Access.* 2022;10(11):21120–40. doi:10.1109/access.2022.3151360.
17. Godard D. Self-recovering equalization and carrier tracking in two-dimensional data communication systems. *IEEE Trans Commun.* 1980;28(11):1867–75. doi:10.1109/tcom.1980.1094608.
18. Liu P, Fan K, Chen Y. Analytical blind beamforming for a multi-antenna UAV base-station receiver in millimeter-wave bands. *Sensors.* 2021;21(19):6561. doi:10.3390/s21196561.
19. Herfandi H, Sitanggang OS, Nasution MRA, Utama IBKY, Rahman MM, Nguyen H, et al. Implementation of a multiple-transmitter RS-OFDM based OCC system with advanced ByteTrack for mobility environments. *IEEE Access.* 2025;13:119411–26. doi:10.1109/access.2025.3587140.
20. Liu Y, Xiong X, Zhang J, Miao F. Design of a DDS-based OFDM for UAV communication systems. In: *Proceeding of the 4th International Symposium on Semiconductor and Electronic Technology (ISSET)*. Piscataway, NJ, USA: IEEE; 2025. p. 772–6.
21. Treichler J, Agee B. A new approach to multipath correction of constant modulus signals. *IEEE Trans Acoust Speech Signal Process.* 1983;31(2):459–72. doi:10.1109/tassp.1983.1164062.
22. Ansari S, Alnajjar KA, Khater T, Mahmoud S, Hussain A. A robust hybrid neural network architecture for blind source separation of speech signals exploiting deep learning. *IEEE Access.* 2023;11:100414–37. doi:10.1109/access.2023.3313972.