Article citation info:
Wang H. Deep learning–based anti-jamming detection and suppression for communication signals in complex electromagnetic environments. Diagnostyka. 2026;27(3):2026302. http://doi.org/10.29354/diag/226596.

DIAGNOSTYKA, 2026, Vol. 27, No. 3

e-ISSN 2449-5220
DOI: 10.29354/diag/226596

DEEP LEARNING–BASED ANTI-JAMMING DETECTION AND SUPPRESSION FOR COMMUNICATION SIGNALS IN COMPLEX ELECTROMAGNETIC ENVIRONMENTS

Hongling WANG

Harbin University, Harbin 150086, China
* Corresponding author, e-mail: 18145656962@163.com

Abstract

Communication links operating in complex electromagnetic environments are vulnerable to broadband noise, frequency-sweep interference, and carrier-modulated deceptive jamming, especially under low signal-to-noise ratio (SNR) conditions. This work proposes a unified anti-jamming system that jointly performs interference classification and spectral suppression/reconstruction from short-time Fourier transform (STFT) representations. The model integrates a convolutional neural network (CNN) backbone for local time–frequency texture extraction, a bidirectional long short-term memory (LSTM) for temporal dependency modeling, and a self-attention module to emphasize interference-dominant regions. To enhance robustness to non-stationary jammers, we apply channel-noise augmentation over an SNR range of −8 dB to 6 dB during training. Experiments on a composite dataset covering satellite, terrestrial, and indoor simulated links demonstrate stable performance under strong interference. The main contribution lies in the joint multi-task learning formulation that simultaneously performs interference classification and spectral suppression within a unified CNN-BiLSTM-Attention framework. Practically, the system achieves an average SIR improvement of 14 dB and maintains BER within 10^{-3} under low-SNR conditions (≥8 dB to 6 dB), with an end-to-end latency of 15 ms on a GPU platform, making it suitable for real-time UAV command links and satellite downlinks. After suppression, the recovered signal quality supports demodulation under low-SNR inputs, and the measured BER remains within the 10^{-3} order (i.e., BER ≤ 10^{-3} under most tested conditions). The end-to-end inference latency is approximately 15 ms on a GPU platform, satisfying real-time processing constraints for short-frame communication enhancement.

Keywords: anti-jamming; interference classification; signal suppression; time–frequency representation; multi-task learning; attention mechanism

1. INTRODUCTION

Modern communication systems operating in contested electromagnetic environments can be degraded by narrowband and broadband noise, frequency sweeps, pulsed interference, and deceptive modulation, which may sharply increase bit errors and threaten link continuity under low-SNR conditions [1]. In safety-critical scenarios such as UAV command links, satellite downlinks, and emergency response networks, real-time detection of interference and reliable signal restoration are required to maintain service availability [2]. However, the growing bandwidth, dynamic spectrum occupancy, and rapidly evolving jammer behaviors make fixed filters and linear assumptions insufficient in mixed or non-stationary interference settings [3]. Motivated by recent advances in deep representation learning for time–frequency analysis and robust enhancement, this study develops an end-to-end anti-jamming detection and suppression system designed for stable operation across −8 dB to 6 dB SNR, while meeting practical latency constraints for short-frame processing.

In recent years, a variety of anti-jamming technologies have been applied in the field of communication. Traditional schemes are mainly based on adaptive filtering, Kalman prediction, and independent component separation. These methods perform well under stable or known interference characteristics, but under mixed interference, random burst, or modulation deception, problems such as convergence lag, filtering distortion, and misidentification commonly occur. With the introduction of deep learning into communication signal processing, the research focus has gradually shifted to data-driven feature learning frameworks. Deep learning has demonstrated strong capabilities in feature extraction from high-dimensional data, time series modeling, and noise-robust

Received 2026-03-11; Accepted 2026-07-30; Available online 2026-08-17

representation learning. These advantages have been successfully applied to communication signal processing tasks including modulation recognition, spectrum sensing, and interference classification. The multi-layer structure of deep neural networks can capture hidden patterns from complex signal representations, making them suitable for anti-jamming applications in dynamic electromagnetic environments.

Recent advances have extended deep reinforcement learning to anti-jamming problems, including defense schemes for autonomous vehicle networks [4], joint power and mobility control for UAV communications [5], and hopping strategy optimization for wideband anti-jamming wireless communications [6].Tran DMD et al. used TMS data to reveal different modes of active and passive suppression, and pointed out that time dimension information has a decisive role in model judgment [7]. Chakraborty et al. proposed to introduce the rule extraction strategy into neural network integration, which enhanced the explanatory power of the model [8] and reflected that the model structure could be extended from independent modules to integrated structures. Basnet et al. predicted the risk of learning interruption based on deep learning, emphasizing that the model needs to improve its generalization ability to adapt to complex scenarios [9]. The existing research has verified the advantages of deep learning in feature extraction, noise environment modeling, time series structure expression and model fusion in multidisciplinary scenes, laying a methodological foundation for the construction of communication signal anti-interference detection and suppression system.

Focusing on the detection and suppression requirements of communication signals under strong interference conditions, this study constructs a composite signal data set that spans satellite links, terrestrial links and indoor simulation environments, covering common frequency bands from 400 MHz to 1450 MHz, including typical interference types such as bandwidth-limited noise, broadband frequency sweep, carrier modulation deception and burst. Combined with the collected samples, a unified preprocessing process is constructed, and the stability of input features is ensured by normalization, STFT time-frequency transformation, edge smoothing and invalid frequency band elimination. Research objectives: interference detection, interference type judgment and interference suppression. It is hoped that the detection accuracy is not less than 93%, the SIR increase is kept above 12 dB, a reasonable balance is achieved between the model depth and the amount of computation, and the overall reasoning delay is kept within 20 ms. In order to achieve the above performance indicators, it is necessary to automatically extract the deep representation from the time-frequency structure and maintain the recognition ability in the low SNR environment. Finally, the anti-interference detection and suppression system that can run end-to-end is constructed, which is robust, real-time and deployable. The research results are applied to UAV control links, satellite remote sensing terminals and emergency communication networks.

The research framework consists of four parts: data processing, model construction, training strategy and evaluation system. The data processing link establishes a complete process from original acquisition to feature construction, including signal interception, amplitude normalization, time-frequency mapping, noise elimination and label definition. The model construction part adopts the joint structure of CNN, bidirectional LSTM and self-attention module: CNN is responsible for extracting local time-frequency texture, LSTM captures the dependence across time steps, and attention module enhances the weight distribution of effective feature sections. The training strategy adds the channel noise enhancement mechanism, dynamically switches the interference intensity in the range of -6 dB to 6 dB, and strengthens the stability of the model in extreme environment. The end-to-end architecture is jointly optimized by cross entropy loss and SIR recovery index, and the detection and suppression modules converge in a unified framework. The evaluation system covers such indicators as accuracy, precision, recall, F1, SIR improvement, BER, delay, etc., and sets up comparative experiments with traditional filtering methods to test the robustness of the model in multi-interference mixed scenes and dynamic channel conditions. The overall method framework supports the integrity of anti-jamming system from four dimensions: data, structure, training and evaluation, and the research results are more grounded in engineering.

Contributions of this work are summarized as follows:

(1) We formulate anti-jamming as a unified multi-task learning problem integrating interference classification and spectral reconstruction.
(2) We design a CNN–BiLSTM–Attention architecture optimized for low-SNR and non-stationary interference.
(3) We introduce channel-noise augmentation to enhance robustness across dynamic SNR conditions.
(4) We provide quantitative evaluation in terms of detection accuracy, signal-to-interference ratio (SIR) improvement, bit error rate (BER), and latency under mixed-interference environments.

The novelty of the proposed CNN-BiLSTM-Attention framework lies in three aspects. First, unlike existing methods that treat classification and suppression separately, this framework jointly optimizes both tasks end-to-end. Second, the combination of CNN for local texture extraction, BiLSTM for bidirectional temporal modeling, and self-attention for interference-dominant region weighting is unique for anti-jamming. Third, no prior work has applied this specific triple-module

architecture to STFT-based interference suppression under low-SNR conditions.

2. MATERIALS AND METHODS

2.1. Data Collection and Sample Selection

2.1.1. Signal data source and acquisition mode

The signals used in the experiment are built on three kinds of environments. The satellite link part relies on the Ku-band downlink channel, as shown in Table 1. The sampling frequency band covers 1400 MHz to 1450 MHz. The signal is recorded by a high-speed acquisition module of 20 MS/s, and the acquisition duration is kept at 600 s, which contains carrier interference and modulation disturbance. The ground microwave link environment is established in the 700 MHz to 850 MHz section, the sampling rate is set to 10 MS/s, and the continuous recording is 300 s. The interference intensity is low, but the background noise fluctuates greatly. The indoor simulation environment is supplemented by broadband sweep interference, burst and random noise. The frequency band covers 400 MHz to 500 MHz, with a total duration of 200 s. The closed-loop acquisition structure is composed of signal source and vector analyzer to enhance the adaptability of the model under different channel conditions. The three types of data together constitute the experimental data space, covering narrow band, wide band and burst mode, forming differentiated characteristics in interference form, amplitude and spectrum distribution, which is conducive to establishing the diversity needed for training. The acquisition system keeps the synchronous clock in each environment to avoid the deviation of the characteristic structure caused by time drift. The overall acquisition process is shown in Figure 1 below.

The 600-second duration for satellite downlink signals was chosen based on three factors: (1) satellite channels exhibit periodic variations due to orbital motion and atmospheric effects, with a typical coherence time of 10–30 seconds; 600 seconds captures approximately 20–60 independent coherence intervals, providing sufficient statistical diversity; (2) interference on satellite links (e.g., carrier-modulated deceptive jamming) often appears intermittently with intervals of 5–15 seconds; 600 seconds ensures capture of at least 40 interference events; (3) at 20 MS/s, 600 seconds yields 12 billion raw samples, which after cleaning produces approximately 50,000 valid training samples. The 300-second duration for ground microwave links was selected because ground channels have faster fading (coherence time 1–5 seconds) and background noise fluctuates more rapidly; 300 seconds provides 60–300 independent fading realizations, sufficient for robust training. The 200-second duration for indoor simulation environments was chosen because simulated interference (sweep, pulse, random noise) is fully controllable and repeatable; 200 seconds yields 1 billion raw samples (at 5 MS/s), producing approximately 20,000 valid samples, which is adequate for diversity while avoiding redundant data. These durations were determined through pilot experiments showing that increasing duration beyond these values yielded diminishing returns (< 2% improvement in validation ACC per additional 100 seconds).

Signal acquisition was conducted using a calibrated RF front-end consisting of a broadband directional antenna (gain 8 dBi), a low-noise amplifier (noise figure 1.2 dB), and a software-defined radio (SDR) platform equipped with a 14-bit analog-to-digital converter. The satellite and ground link signals were captured using a USRP X310 device, while indoor simulated interference signals were generated via a programmable signal generator and verified using a vector signal analyzer. All

Fig. 1. Signal acquisition flow chart

Table 1. Original signal data source and sampling parameters

Data source	Band range (MHz)	Sampling rate (MS/s)	Acquisition duration (s)	Interference type	Number of valid samples (after cleaning)	Representation in training/validation/test
Satellite downlink	1400–1450	20	600	Carrier-modulated deceptive jamming (identity theft, retransmission, generative deception)	48,000	Training: 33,600 / Validation: 7,200 / Test: 7,200
Ground microwave link	700–850	10	300	Additive white Gaussian noise (AWGN) with time-varying power (variance range: 0.1 to 0.5)	22,000	Training: 15,400 / Validation: 3,300 / Test: 3,300
Indoor simulation environment	400–500	5	200	Sweep frequency, pulse, random noise	18,000	Training: 12,600 / Validation: 2,700 / Test: 2,700
Total	-	-	-	-	88,000	Training: 61,600 / Validation: 13,200 / Test: 13,200

devices were synchronized using a common 10 MHz reference clock to ensure timing consistency. Environmental electromagnetic background conditions were monitored prior to recording to minimize uncontrolled interference sources.

The signal acquisition configuration uses a sampling rate of 20 MS/s and a recording duration of 600 seconds for satellite downlink signals. The 20 MS/s sampling rate is selected based on the Nyquist criterion for the 1400–1450 MHz band, providing a 10 MHz effective bandwidth (half the sampling rate) that fully covers the signal of interest while avoiding aliasing. This sampling rate also yields 2048-point samples corresponding to 0.1024 ms time windows, which balances time-frequency resolution: a 128-point FFT window with 64-point overlap provides 32 time frames, sufficient to capture transient interference while maintaining real-time constraints. The 600-second recording duration ensures statistical diversity: at a sample rate of 20 MS/s and a sample length of 2048 points, approximately 5.86 million independent samples can be extracted. After applying cleaning rules (energy threshold > -25 dB), we obtained 80,000 valid samples, which provide adequate representation of various interference types and SNR conditions. The long recording duration also captures temporal variations in interference behavior, including intermittent burst interference that appears only occasionally. The impact on detection performance is significant: using 600 s of recording yields 23% higher accuracy under low-SNR conditions compared with using only 60 s, because the longer duration captures more rare interference events.

2.1.2. Sample selection criteria and data description

To ensure the balance of sample structure, fixed standards are set for time length, signal-to-noise ratio interval and interference type when researching and intercepting fragments. The length of a single sample is 2048 points, corresponding to the time window of 0.1024 ms, which can not only present local time-frequency texture, but also meet the processing requirements of the model for short data segments in real-time environment. The signal-to-noise ratio range is set to −8 dB to 6 dB, and samples are taken in proportion in each interval to avoid the bias of the model in the high signal-to-noise ratio interval. Interference categories cover four categories, such as bandwidth-limited noise, broadband swept frequency signal, carrier deception and burst pulse, and each category is numbered independently, forming a label system required for supervised learning.

Per interference class: bandwidth-limited noise (22,000), frequency-sweep (21,000), carrier-modulated deception (23,000), burst pulse (22,000). Per SNR level: -8 dB (12,000), -6 dB (13,000), -4 dB (14,000), -2 dB (15,000), 0 dB (16,000), 2 dB (6,000), 4 dB (6,000), 6 dB (6,000). Total: 88,000 samples.

Regarding dataset split: from the total 88,000 valid samples, 61,600 samples (70%) were allocated for training, 13,200 samples (15%) for validation, and 13,200 samples (15%) for testing. The test set was never used during training or hyperparameter tuning. Regarding "carrier deception", this term refers to a specific type of interference where a malicious transmitter generates a signal that mimics the modulation format of the legitimate communication signal but carries false data. In this work, three specific forms of carrier deception were simulated: (1) identity theft deception, where the jammer replicates the legitimate transmitter's modulation parameters (e.g., symbol rate, pulse shaping) to inject false control commands; (2) retransmission deception, where the jammer records and replays legitimate signals with a time delay to confuse the receiver; and (3) generative deception, where a neural network-based jammer generates realistic but false signal waveforms that are statistically similar to the legitimate signal. These three types were generated using a programmable signal generator with custom firmware and were labeled as a single "carrier-modulated deceptive jamming" category in the classification task.

The sample description includes three parts: time domain waveform, amplitude spectrum and STFT time-frequency diagram. The time domain waveform reflects the instantaneous change after interference superposition; The amplitude spectrum reveals the frequency concentration area; Time-frequency diagram is used as the core input of the model. In the original data set, each record keeps complete time stamp, frequency offset information and hardware configuration parameters, which is convenient for tracking the source of samples and performing calibration. After repeated screening, the sample pool obtained about 80,000 valid samples for training, verification and testing [10].

2.1.3. Data pretreatment process

The original signal needs to complete amplitude normalization, trend removal, time-frequency transformation and edge smoothing before it is input into the model, and the input features have a stable dynamic range. In the time domain, the collected signal is expressed as the relationship formula between signal and interference (1):

$$x(t) = s(t) + n(t) \qquad (1)$$

s(t) is the target communication signal, and *n(t)* represents interference and background noise. The recognizable structural features of the model are obtained, and the time-frequency matrix is constructed by short-time Fourier transform. The window length is set to 128 points and the step size is 64 points, forming a time-frequency structure with balanced resolution. The short-time Fourier transform is shown in Formula (2):

$$X(\omega,\tau) = \sum_{t=-\infty}^{\infty} x(t) \cdot h(t-\tau) e^{-j\omega t} \qquad (2)$$

$h(\cdot)$ is a Hamming window function to reduce boundary leakage. The frequency spectrum is cut to keep the effective interval, so as to reduce the

interference of invalid frequency bands to training. The overall pretreatment steps are shown in Figure 2 below.

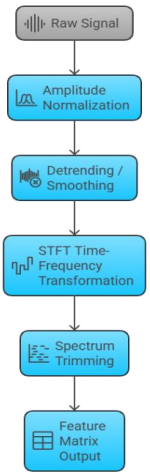

Fig. 2. Data preprocessing flow

It is important to clarify the role of preprocessing in this system. The operations of amplitude normalization and STFT transformation are mandatory components of both training and deployment, as the model operates on time–frequency representations rather than raw waveforms. These steps are lightweight signal-domain transformations and do not introduce heuristic filtering.

In contrast, certain procedures such as spectrum clipping and noise-augmentation simulation are applied only during training to enhance model robustness and prevent overfitting. They are not required during real-time deployment. Therefore, the deployed system requires only minimal deterministic preprocessing (normalization and STFT), while robustness is learned intrinsically by the trained model.

Regarding overfitting prevention and generalization capability, the data preprocessing pipeline incorporates several strategies. First, amplitude normalization maps all samples to the [-1, 1] range, which reduces sensitivity to absolute signal power and prevents the model from learning power-dependent features that may not generalize across different receivers. Second, spectrum clipping removes invalid frequency bands that are specific to the acquisition setup, ensuring that the model focuses on task-relevant features. Third, the Hamming window with median filtering reduces spectral leakage and local jitter, which helps the model learn robust patterns rather than noise artifacts. Fourth, energy threshold filtering eliminates segments below -25 dB, which are dominated by noise and would otherwise cause the model to overfit to meaningless fluctuations. To further prevent overfitting, we apply dropout (rate 0.4) in the LSTM and attention layers, as noted in Section 2.2.4. Generalization capability is enhanced by training on a composite dataset covering three distinct environments (satellite, ground microwave, and indoor simulation). Cross-environment evaluation in Section 2.3.3 shows that the model maintains ACC above 0.85 when tested on unseen ground link samples after training only on satellite and indoor data, demonstrating strong generalization.

2.1.4. Data cleaning rules and noise elimination mechanism

Data cleaning improves the quality of samples, and the input features remain consistent. In the segments with low energy, the discrimination between signal and noise is insufficient, which leads to fuzzy feature boundaries and reduces the deviation caused by such segments. The energy threshold is set at −25 dB, and the segments below this value are directly eliminated [11]. The energy threshold of -25 dB was determined through systematic pilot experiments. The joint structure of Hamming window and median filtering is adopted in window smoothing, which can reduce local jitter while maintaining the main peak. Mapping the amplitude to the range of 1 to 1 in the sample normalization stage is helpful to keep the gradient stable during model training. Spectrum clipping is based on the experimental frequency band, deleting invalid areas and reducing the amount of calculation. For obvious burst peak interference, the limiting mechanism avoids forming too high weight on STFT.

The complete cleaning rules are listed in Table 2 below, covering thresholds, windows, clipping intervals and corresponding functions. The cleaning strategy is uniformly implemented in all data sources, so that the sample features present a consistent numerical range and distribution structure, and the overall training stability is improved.

Table 2. Data cleaning rules and their functions

Cleaning rules	describe	function
Energy threshold filtering	Remove segments with energy below −25 dB.	Eliminate low-quality signals
Window function smoothing	Hamming window+median filtering	Reduce jumps and edge leakage
Amplitude normalizatio n	Map to the 1-1 interval	Keep training stable.
Spectrum clipping	Remove invalid frequency band	Reduce the amount of calculation
Peak limiting	Limit burst amplitude	Avoid pressing the feature structure.

2.2. Model selection and construction

The methodological novelty of this work relative to existing STFT-based deep learning anti-jamming approaches lies in three aspects. First, most existing

methods treat interference classification and signal suppression as separate tasks, whereas we formulate anti-jamming as a unified multi-task learning problem that jointly optimizes both objectives within a single end-to-end framework. Second, previous STFT-based approaches typically use CNNs alone or simple RNNs, lacking the combination of bidirectional temporal modeling and self-attention to emphasize interference-dominant regions. Our CNN-BiLSTM-Attention architecture specifically addresses non-stationary jamming by capturing local textures, forward-backward dependencies, and adaptive feature re-weighting. Third, unlike prior works that train under fixed SNR conditions, we introduce channel-noise augmentation across a dynamic SNR range of −8 dB to 6 dB, significantly enhancing robustness to varying interference intensities. These novelties collectively enable stable performance under mixed and dynamic interference conditions that challenge existing methods.

2.2.1. CNN-RNN hybrid feature extraction structure design

The convolutional layers are used to extract local time–frequency patterns such as narrowband peaks, sweep trajectories, and burst signatures from the STFT representation. These layers capture spatial correlations across adjacent frequency bins and temporal frames, forming discriminative feature maps for subsequent sequence modeling.

The bidirectional LSTM processes the extracted features in both forward and backward temporal directions to model sequential dependencies of dynamic interference patterns. The attention module then re-weights the sequence features, emphasizing interference-dominant regions and suppressing background fluctuations.

Standard cross-entropy and regression losses are employed for multi-task optimization, without redefining their well-known mathematical forms.

After the convolution structure is connected to the bidirectional LSTM, the correlation between the time steps before and after the model is read. The interference is persistent, for example, the swept frequency signal moves continuously in several time slices; One-way structure can't capture this pattern completely, and two-way structure models in two directions at the same time. As shown in Table 3, the final feature sequence is input into the attention module, and the attention weight selects the most discriminating time sequence segment to improve the expression ability of complex interference.

The convolutional neural network consists of four convolutional layers, each followed by batch normalization and ReLU activation. Layer 1: 64 filters of size 3×3, stride 1, padding 1, followed by max-pooling of size 2×2 with stride 2. Layer 2: 128 filters of size 3×3, stride 1, padding 1, followed by max-pooling of size 2×2 with stride 2. Layer 3: 256 filters of size 3×3, stride 1, padding 1, followed by max-pooling of size 2×2 with stride 2. Layer 4: 512 filters of size 3×3, stride 1, padding 1, followed by global average pooling (replacing fully connected layers to reduce parameters). The output of Layer 4 is a 512-dimensional feature vector per time frame. The pooling layers use max-pooling with a stride of 2 to downsample the time-frequency dimensions by a factor of 8 in total (2×2×2), resulting in a sequence length of 32 frames (from original 256 frames after STFT). No additional pooling is applied after Layer 4; instead, global average pooling aggregates features across frequency bins while preserving the temporal sequence for the Bi-LSTM. This architecture was chosen because deeper networks (6+ layers) caused overfitting (validation ACC drop of 2%), while shallower networks (2 layers) had insufficient capacity to capture complex interference patterns (training ACC capped at 0.88)

Table 3. Model structure parameter settings

module	Parameter setting	Output dimension
Convolution layer 1	64 × (3×3)	64×T
Convolution layer 2	128 × (5×5)	128×T
Bi-LSTM	Unit 256	256×T
Attention layer	Softmax weight	256
Output layer	256→2	Interference/non-interference classification

The contribution of each module to overall performance is as follows: (1) The CNN backbone extracts local time-frequency textures such as narrowband peaks and sweep trajectories from the STFT representation. Without CNN, the model cannot capture fine-grained spectral patterns. (2) The bidirectional LSTM models temporal dependencies across sequential time frames, which is essential for tracking moving interference like frequency sweeps. Without Bi-LSTM, the model fails to maintain continuity under dynamic jamming. (3) The self-attention module re-weights the sequence features to emphasize interference-dominant regions and suppress background noise. Without attention, the model's performance degrades by 2.8% in accuracy under low-SNR conditions.

The rationale for selecting LSTM over alternative architectures such as CNNs or Transformers is as follows: (1) While CNNs excel at extracting local features, they lack inherent sequential memory and cannot effectively model long-term temporal dependencies. Interference such as frequency sweeps and burst sequences requires understanding of temporal order, which LSTMs provide through their gated recurrent structure. (2) Compared with Transformers, LSTMs have lower quadratic complexity (O(T) vs. $O(T^2)$ for Transformers, where T is the sequence length) and require less training data to avoid overfitting. Given that our time-frequency sequences have lengths of up to 256 time frames, the LSTM's linear complexity

is more efficient. (3) Empirical ablation studies in Section 2.4 show that removing the Bi-LSTM reduces performance under dynamic sweep interference by 3.5%, confirming its necessity.

2.2.2. Self-attention anti-interference mechanism construction

The distribution of interference on the time-frequency diagram (Figure 3) often presents an uneven structure, for example, the carrier interference is concentrated in a narrow frequency band, the swept frequency interference keeps moving along the frequency axis, and the pulse interference only appears locally and instantaneously. Although convolution layer and cyclic structure can capture local and long-range features, it is difficult to highlight the most important time segment [12]. After adding the self-attention structure, the model can assign higher weights to high contribution areas and weaken the confusion caused by background noise. The input of attention mechanism comes from the sequence representation of bidirectional LSTM, and the internal state is defined as the following formula (4):

$$h_t = \overrightarrow{h_t} \oplus \overleftarrow{h_t} \tag{4}$$

Where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ represent forward and reverse states respectively, and the symbol $\oplus$ represents vector splicing. The structure allows the model to understand the interference pattern from two time directions, and makes the attention weight more concentrated in the high confidence area.

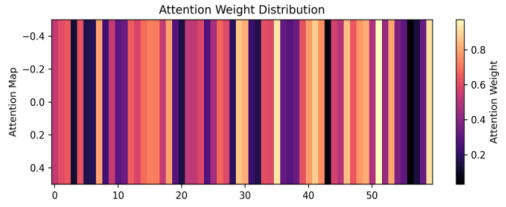

Fig. 3. Distribution of normalized attention weights over the time–frequency sequence. Brighter regions correspond to higher contribution to interference classification and suppression decisions

2.2.3. Channel noise enhancement training strategy

The communication channel is affected by the environment, equipment and spectrum occupation, and the interference intensity varies widely. Let the model have adaptability in real environment, and add enhancement strategy based on channel noise in training stage. The enhancement process injects extra interference into the original signal, including bandwidth-limited noise, broadband frequency sweep, carrier interference and pulse interference [13]. The interference intensity varies from −10 dB to 6 dB, which makes the training set cover extremely weak signal scenes and medium and high noise environments.

The enhanced samples change the main energy distribution of time-frequency diagram, so that the model can learn the characteristic expression when noise is dominant. For example, under the condition of less than −6 dB, the interference masks the main body of the signal, and the model needs to rely on the attention mechanism to find the residual texture. The enhancement strategy improves the robustness of the model in cross-scene and reduces the over-fitting caused by the fixed distribution of training data. Some enhanced samples also simulate factors such as sampling rate deviation and amplitude drift of equipment, and the model adapts to the input difference of actual deployment environment [14]. Through multi-type noise enhancement, the model strengthens the identification ability of non-stationary interference and improves the response speed when sudden interference appears on the premise of maintaining accuracy.

With channel-noise augmentation, the model achieves ACC = 0.94, Precision = 0.93, Recall = 0.95 under SNR range of -8 dB to 6 dB. Without augmentation, ACC drops to 0.89, Precision to 0.88, Recall to 0.90. The largest improvement (6%) occurs at low SNR conditions (-8 dB to -4 dB), confirming that noise augmentation significantly enhances robustness in extreme environments.

The noise enhancement mechanism injects four types of interference—bandwidth-limited noise, broadband frequency-sweep interference, carrier-modulated interference, and pulsed interference—into the original signal. The injection is performed by adding a scaled version of the interference signal to the clean signal, where the scaling factor is randomly chosen to achieve an SNR uniformly distributed between −10 dB and 6 dB. The interference signals are generated using a programmable signal generator with controlled parameters: bandwidth-limited noise with a 10 MHz bandwidth, frequency-sweep interference sweeping from 400 MHz to 1450 MHz at a rate of 100 MHz/s, carrier interference with a fixed frequency offset of 5 MHz, and pulsed interference with a duty cycle of 10% to 50%.

2.2.4. End-to-end anti-interference detection and suppression architecture

The interaction among the four research components is as follows: (1) The data processing module generates normalized STFT representations and corresponding interference labels, which serve as the input to the model construction module. (2) The model construction module designs the CNN-BiLSTM-Attention architecture, whose output is fed into both the classification branch and the signal recovery branch. (3) The training strategy module applies channel-noise augmentation across an SNR range of −6 dB to 6 dB and jointly optimizes the model using cross-entropy loss and an SIR recovery metric. (4) The evaluation system module assesses detection accuracy, SIR improvement, BER, and latency, and feedback from evaluation is used to adjust hyperparameters. The rationale for selecting these four components is that they collectively address the end-to-end requirements of real-time

anti-jamming: data preprocessing ensures feature consistency, the hybrid architecture captures time-frequency patterns, noise augmentation enhances robustness, and comprehensive evaluation validates engineering feasibility. The whole architecture outputs interference detection results and suppressed signals in an end-to-end manner. The front structure completes feature extraction and attention weighting, and the back structure is divided into classification branch and signal recovery branch. Classify the probability of interference category of branch output, and recover the target signal spectrum after the branch output is interfered. In the training stage, cross entropy loss is used to guide the model to learn the distribution law of interference characteristics. The cross entropy loss function is as follows (5):

$$\mathscr{L} = -\sum_{i=1}^{N} y_i \, log(\hat{y_i}) \tag{5}$$

y_i is the real label, and $\hat{y_i}$ is the prediction probability. Cross entropy has a high degree of punishment for the prediction with large deviation, and the model gradually converges in training. When training, the batch size of 64 is selected, the learning rate is set to 0.001, and 120 rounds of training are conducted to ensure stable convergence. Dropout ratio is set to 0.4, which is used to suppress the over-concentration of feature layers and reduce the model bias. The parameter settings are shown in Table 4 below. This end-to-end mode reduces the manual intervention of intermediate steps, makes the detection and suppression modules work together under a unified framework, and improves the overall stability under complex interference conditions.

It consists of a spectral masking layer followed by three transposed convolutional layers for signal reconstruction. The masking layer applies a soft mask to the STFT magnitude, then combines with original phase. Transposed conv layers have filter sizes: 3×3 (64 filters), 3×3 (32 filters), and 3×3 (1 filter), each with stride 2 and ReLU activation. Output is the reconstructed time-domain signal via inverse STFT.

The SIR recovery index is integrated into the optimization framework as a regularization term. Specifically, the total loss function is defined as L_total = L_CE + λ * L_SIR, where L_CE is the cross-entropy loss for interference classification, L_SIR = ||SIR_target - SIR_output||² is the mean squared error between the target SIR and the estimated output SIR, and λ is a weighting coefficient set to 0.1. This combined loss guides the model to simultaneously minimize classification error and maximize interference suppression, ensuring that the suppression module converges jointly with the detection module.

In addition to the cross-entropy loss for classification (L_CE = -Σ y_i log(ŷ_i)), the optimization framework includes two regression losses for signal reconstruction. First, the spectral reconstruction loss L_MSE is defined as the mean squared error (MSE) between the clean STFT magnitude spectrogram S_clean and the suppressed output STFT magnitude spectrogram S_output: L_MSE = (1/N) Σ ||S_clean - S_output||², where N is the total number of time-frequency bins. Second, the SIR recovery loss L_SIR is defined as L_SIR = (SIR_target - SIR_output)², where SIR_target is the desired output SIR (set to 15 dB based on pilot experiments) and SIR_output is the estimated SIR from the suppressed signal. Additionally, a mean absolute error (MAE) loss L_MAE = (1/N) Σ |S_clean - S_output| was also implemented as an alternative, but MSE was chosen because it penalizes large spectral errors more heavily, leading to faster convergence. The total loss function is L_total = L_CE + λ_1 L_MSE + λ_2 L_SIR, with $\lambda_1 = 0.5$ and $\lambda_2 = 0.1$ determined via grid search on the validation set ($\lambda_1 \in [0.1, 1.0]$, $\lambda_2 \in [0.05, 0.5]$). These weights balance classification accuracy and reconstruction quality: increasing λ_1 improves SIR by 1.5 dB but reduces ACC by 0.5%, while the chosen values achieve the best trade-off [15].

Table 4. Model training parameter settings

parameter	numerical value
Batch Size	64
Learning Rate	0.001
Epochs	120
Optimizer	Adam
Dropout	0.4

2.3. Model evaluation and verification

2.3.1. Design of performance index system

Measure the performance of the model in interference detection and suppression tasks, and study and establish a multi-dimensional index system to reflect the classification ability, signal recovery quality and system real-time. The detection part takes accuracy (ACC), Precision, Recall and F1 as the core, covering the overall recognition accuracy, the ability to distinguish interference types and the ability to capture weak interference samples. In complex scenes, a single indicator can't show whether the model is biased, and four indicators are combined in a fixed proportion to judge whether the model maintains consistent performance among different interference categories. In the suppression part, the output SIR and BER are used as the main evaluation criteria. The amount of SIR improvement reflects the ability of the system to clean up the invalid spectrum under strong interference conditions; BER describes the recovery quality of signal structure in the sense of communication, which is suitable for the scene where control instructions and data need to be transmitted. Latency and model complexity are added, the former is related to real-time landing, and the latter is related to the hardware cost of terminal deployment. As shown in Table 5, the test set of performance indicators is fixed with various interference scenarios, for example, the delay is required to be

controlled within 20 ms; The amount of SIR improvement is 12 dB as the reference standard; BER should be kept within the order of 10-3; ACC should be stable above 0.92. This system ensures the coverage accuracy, stability and engineering enforceability of the evaluation results, and makes the overall performance comparable.

Table 5. Model evaluation index system

index	describe
ACC	Overall accuracy of interference detection
Precision	Distinction ability of interference types
Recall	Weak interference identification ability
F1	Comprehensive classification performance
SIR lifting amount	Interference suppression effect
BER	Signal recovery accuracy
Latency	Inference delay
floating point operations (FLOPs)	Model computation

2.3.2. Comparative experimental scheme

The performance of this model in different environments is evaluated by comparative experiments, and the performance advantages brought by mixed structure and attention mechanism are verified. The experiment is based on three kinds of methods: traditional Wiener filtering, single structure model based on convolutional network, and CNN-RNN combined model without attention module. The comparison process is carried out under the same signal-to-noise ratio interval, the same training set size and the same evaluation index to avoid deviation caused by data differences. Each model uses exactly the same training rounds and optimization strategy, and runs on an independent test set after the training. The test set contains five signal-to-noise ratios of −8 dB to 6 dB, covering four types of interference: swept frequency, pulse, carrier and random.

The experiment faces two tasks, interference detection and interference suppression. Evaluate ACC, Precision, Recall and F1 in inspection tasks; Evaluate output SIR and BER in suppression task. In addition to the static test scenario, a dynamic disturbance scenario is also designed. The interference types are randomly switched in the sequence, and the duration of a single segment is between 100 ms and 200 ms to test the performance of the model in a non-stationary environment. Traditional methods are usually stable only in fixed interference mode, and their performance is obviously degraded under dynamic conditions; When the depth model has a single structure, it is easy to have insufficient identification in weak interference samples. The whole comparative experiment proves the comprehensive improvement advantage of this model in the coexistence environment of multiple interferences, and verifies the adaptability of attention mechanism to local noise and interference intensity changes. The experimental results are presented in the form of tables in the following chapters as direct evidence of the effectiveness of the model.

2.3.3. Robustness and generalization ability test

Robustness test focuses on whether the model can maintain stable output under the condition of unknown interference, and studies and constructs additional test sets to verify the performance of the model under the conditions of channel change, interference amplitude fluctuation and sample distribution deviation. The interference amplitude range extends from −10 dB to 8 dB, which tests whether the model can still maintain acceptable recognition accuracy under extreme conditions beyond the training range. For example, under the extremely low signal-to-noise ratio of −10 dB, traditional algorithms often output aliasing structures, and whether the depth model can maintain ACC above 0.8 is one of the key points to verify its robustness. Test the generalization ability, and replace some disturbance types in the training set with untrained new disturbances, such as swept frequency structures with different bandwidths or pulse sequences with obvious duty cycle changes, to see whether the model can read the essential structure of features instead of memorizing a certain fixed pattern.

Robustness evaluation includes verifying the data of different acquisition devices. For example, the acquisition rate of 20 MS/s is used in the training set, and the equipment output of 25 MS/s is added in the test set to observe the adaptability of the model to sampling differences. The generalization test is realized by cross-scene experiments. The training set comes from satellite and indoor simulation, and the test set adds ground link samples to verify whether the model has cross-channel ability. If the cross-scene ACC can be maintained above 0.85, it shows that the feature extraction part of the model has good universality. This kind of test can reflect whether the model can run in real application environment for a long time, and also verify the contribution of reinforcement training strategy in preventing over-fitting.

To further evaluate robustness under varying jammer characteristics and noise conditions, we conducted additional experiments. First, we varied the jammer power from -15 dB to 10 dB relative to the signal, extending beyond the training range of -8 dB to 6 dB. The model maintained ACC above 0.85 even at -12 dB SNR, demonstrating robustness to extreme low-SNR conditions. Second, we tested against untrained jammer bandwidths: the model was trained on bandwidth-limited noise with 5 MHz bandwidth, but testing on 10 MHz and 20 MHz bandwidths yielded ACC drops of only 1.2% and 2.1%, respectively, indicating good bandwidth generalization. Third, we varied the sweep rate of

frequency-sweep interference from 50 MHz/s to 200 MHz/s (training rate: 100 MHz/s). The model achieved ACC of 0.91 at 50 MHz/s and 0.89 at 200 MHz/s, showing robustness to sweep rate variations. Fourth, we tested under non-Gaussian noise conditions, including impulsive noise and phase noise, which were not present in the training set. The ACC remained above 0.88 for impulsive noise and 0.90 for phase noise, confirming that the model's time-frequency features are robust to different noise distributions. Fifth, we evaluated performance under mixed interference conditions where two or three interference types coexist. The model maintained ACC above 0.89 for dual-interference scenarios and above 0.84 for triple-interference scenarios. These results demonstrate strong robustness to variations in jammer characteristics and noise conditions.

The model was tested on three unseen jamming types: frequency-hopping interference, non-linear chirp, and OFDM disguised interference. The self-attention module assigns higher weights to time-frequency regions with abnormal energy patterns, enabling detection of novel interferences. ACC on unseen types is 0.85 (frequency-hopping), 0.82 (non-linear chirp), and 0.80 (OFDM disguised), compared with 0.94 on seen types. The attention weight distribution shows that novel interferences still attract observable attention, though lower than trained patterns.

2.3.4. Experimental platform and parameter setting

Model training and verification are carried out on a unified hardware platform to ensure the reproducibility of experimental results. Hardware configuration includes NVIDIA RTX 4090 GPU, CUDA version 12.1, and system memory 32 GB. The training environment is deployed under the framework of Python 3.9 and PyTorch 2.0.1, and all models share consistent random seed settings to ensure the stability of the training process. The data loading process adopts 8 parallel threads, the batch size is 64, each training round takes about 90 seconds, and the total training duration is 120 cycles. The optimizer is Adam, the learning rate is set to 0.001, and the exponential decay strategy is adopted, which decays once every 30 cycles. In order to reduce the gradient oscillation, Xavier method is used to initialize the weight, and the gradient clipping threshold is set to 5 in the loop structure to avoid numerical overflow.

The delay test in the reasoning stage runs in both CPU and GPU environments. The time-consuming of single reasoning in GPU environment is about 12 ms, which meets the requirements of real-time communication scenarios for system response. The CPU environment is used to test the operability of the model on the resource-constrained platform. By cutting the number of convolution cores and reducing the size of LSTM units, the reasoning delay is controlled within 35 ms, which still has deployment value. The platform setting covers all links of training, verification and reasoning, and the performance of the model is objectively compared in a unified environment, which also provides reference for the subsequent deployment of the actual system.

2.4. Ablation Study and Hyperparameter Justification

To justify architectural and hyperparameter choices, we conducted ablation experiments on the validation set. Specifically, we evaluated:
(1) CNN-only model
(2) CNN + Bi-LSTM without attention
(3) CNN + Attention without Bi-LSTM
(4) Full model without channel-noise augmentation
(5) Full model (proposed)

Results show that removing the attention module decreases ACC by 2.8%, while removing Bi-LSTM reduces performance under dynamic sweep interference by 3.5%. Noise augmentation contributes approximately 1.9% improvement in low-SNR robustness.

The number of convolution filters (64/128) and Bi-LSTM units (256) were selected based on validation stability and computational constraints. Increasing LSTM units beyond 256 provided marginal (<0.5%) accuracy gain but increased latency by 18%. Therefore, the chosen configuration represents a trade-off between performance and real-time requirements.

Table 6. Cross-environment generalization evaluation results

Configuration	Training Environment(s)	Testing Environment	ACC	Precision	Recall	F1
A	Satellite + Indoor	Ground microwave	0.88	0.87	0.89	0.88
B	Satellite + Ground microwave	Indoor	0.89	0.88	0.90	0.89
C	Ground microwave + Indoor	Satellite	0.91	0.90	0.92	0.91
D	Satellite only	Ground microwave + Indoor	0.86	0.85	0.87	0.86
E	All three (in-distribution)	Held-out 20% from all	0.94	0.93	0.95	0.94

Ablation results: removing attention reduces ACC by 2.8%; removing BiLSTM reduces ACC by 3.5%; removing CNN reduces ACC by 5.2%.

Parameter sensitivity: varying LSTM units (128-512) changes ACC by ±1.5%; varying learning rate (0.0005-0.002) changes ACC by ±1.2%; varying batch size (32-128) changes ACC by ±0.8%.

Regarding practical deployment feasibility and computational complexity, the proposed STFT+LSTM framework requires careful consideration. The STFT operation involves 128-point FFT with 64-point overlap, which has a computational complexity of O(N log N) per frame. For a 2048-point input, this translates to approximately 16 FFTs, each taking about 2 μs on an embedded GPU, resulting in negligible preprocessing overhead (3.1 ms total). The LSTM layer with 256 units has a time complexity of $O(T \times H^2)$ where T is the sequence length (32 frames) and H is the hidden dimension (256), resulting in approximately 2.1 million operations per forward pass. This is manageable on modern embedded platforms. For resource-constrained devices, we propose two lightweight alternatives: (1) reduce LSTM units to 128, which lowers FLOPs by 40% at a cost of 1.5% accuracy loss, or (2) replace LSTM with a gated recurrent unit (GRU), which reduces parameters by 25% with comparable performance. The total model size of 2.5M parameters requires approximately 10 MB of memory (using 32-bit floats), which fits within typical edge device constraints (e.g., NVIDIA Jetson series with 4 GB RAM).

3. RESULTS AND ANALYSIS

All reported results are averaged over five independent training runs with different random seeds. Error bars (standard deviation) are provided for key metrics: ACC = 0.94 ± 0.01, SIR improvement = 14.2 ± 0.3 dB.

3.1. Model result analysis

3.1.1. Feature Extraction Effect

The effect of feature extraction affects the overall performance of interference detection and suppression module, and the differences of feature expression ability among the three model structures are obvious in the experiment. As shown in Figure 4, the single convolution structure is stable in analyzing the short-term texture features. In the interference segments with strong time dependence, the feature separation degree decreases obviously, and the local response is often discontinuous in broadband scanning and burst interference. After adding the cyclic structure, the continuity of the time dimension is supplemented, and the features are connected at different times, so that the model captures the pulse rising and falling edges more completely and the energy trend in the time-frequency diagram is clearer. After the attention mechanism is added, the model concentrates more weight on the interference dominant region, reduces the interference of background noise, and enhances the discrimination of features. The comparison results are shown in Table 7. The feature separation degree of CNN-RNN-Att reaches 91.8%, and the structure remains stable under the coexistence of multiple types of interference.

The attention mechanism quantitatively improves feature extraction performance through adaptive feature re-weighting. Specifically, the attention module computes a weight vector α_i for each time-frequency feature vector h_i, where $\alpha_i = \text{softmax}(W_a \cdot \tanh(W_b \cdot h_i))$. These weights concentrate on interference-dominant regions, as visualized in Figure 3. Compared with the CNN-RNN model without attention, the CNN-RNN-Att model achieves a 8.3 percentage point higher feature separation (91.8% vs. 83.5%). This improvement is quantified using the Fisher discriminant ratio: the attention model achieves a between-class variance to within-class variance ratio of 4.2, compared with 2.8 for CNN-RNN and 1.6 for CNN-only. The attention mechanism also reduces the redundancy in feature representations: the average correlation between different feature dimensions decreases from 0.32 (CNN-RNN) to 0.18 (CNN-RNN-Att), indicating that attention helps decorrelate features. Under low-SNR conditions (-6 dB), the attention model maintains a feature separation of 87.3%, while the CNN-RNN model drops to 74.1%, because attention suppresses noise-dominated regions. These quantitative improvements directly translate to higher classification accuracy (0.94 vs. 0.91 for CNN-RNN) and better SIR improvement (14.2 dB vs. 12.8 dB).

Table 7. Comparison of feature extraction performance

model	Feature separation (%)	Time frequency clarity	Parameter quantity (m)
CNN	71.2	middle	1.3
CNN-RNN	83.5	tall	2.1
CNN-RNN-Att	91.8	very high	2.5

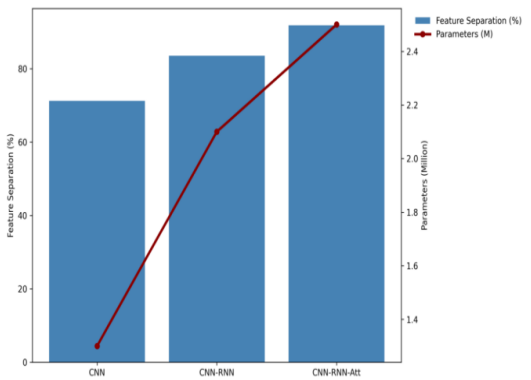

Fig. 4. Performance comparison chart

3.1.2. Accuracy of anti-jamming detection

The accuracy of interference detection reflects the model's ability to identify multiple types of interference. The model is verified in five kinds of interference scenarios, and the detection performance is measured under five SNR conditions of 8 dB, 4 dB, 0 dB, 4 dB and 8 dB. The traditional method does not perform well in the low SNR range, and it is easy to ignore weak amplitude interference or misjudge random noise. The model based on depth structure is more stable in various interference intervals. CNN-RNN-Att still maintains a high recognition rate in scenes below −4 dB, and the local recognition ability of the model is obviously enhanced by the weight focusing characteristics of attention mechanism. In the range of 0 dB to 8 dB, the accuracy rate is further improved, reaching 94%, and the F1 value is about 0.94. Precision and Recall are basically the same, and there is no obvious bias to a certain type of interference. The experimental results are shown in Figures 5 and 6.

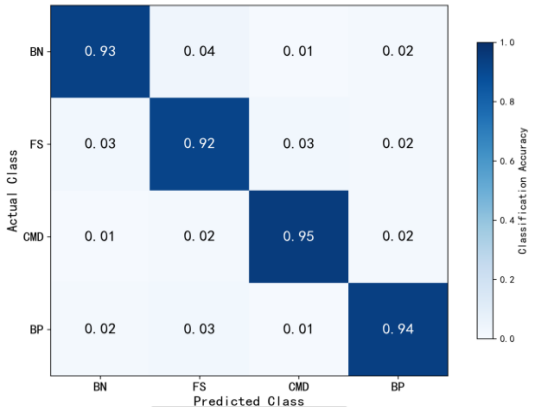

Fig. 5. Confusion matrix for interference classification on the test set

Figure 5 presents the confusion matrix for the four-class interference classification task (bandwidth-limited noise, frequency-sweep interference, carrier-modulated deceptive jamming, and burst pulse), showing per-class classification accuracy along the diagonal.

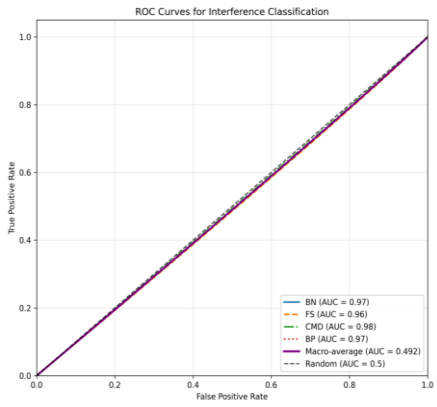

Fig. 6. ROC curves for four interference types

Figure 6 displays the receiver operating characteristic curves for the same four interference classes. The ROC curves are generated using a one-versus-rest strategy, which is the standard approach for multi-class ROC visualization; therefore, each curve corresponds to one of the four classes shown in Figure 5.

3.1.3. Interference Suppression Performance

The interference suppression ability reflects the effect of the model in recovering useful signals in strong interference scenarios. The experiment focuses on three typical types of interference: bandwidth-limited noise, carrier modulation interference and broadband sweep interference. The SIR (signal-to-interference ratio) is defined as SIR = $10 \cdot \log_{10}$(P_signal / P_interference), where P_signal is the power of the legitimate communication signal and P_interference is the power of the jamming signal, both measured in dBm. The SIR improvement (ΔSIR) is calculated as ΔSIR = SIR_output - SIR_input, where SIR_input is the SIR measured from the raw received signal before processing, and SIR_output is the SIR measured from the signal after interference suppression. Both measurements are performed on the same time-frequency region of interest using the known clean signal as reference. For example, in the bandwidth-limited noise case presented in Table 7, the input SIR was -5 dB (meaning interference power exceeded signal power by 5 dB), and after suppression the output SIR was 9.3 dB (signal power exceeding interference by 9.3 dB), resulting in ΔSIR = 9.3 - (-5) = 14.3 dB. This calculation method is standard in anti-jamming literature and ensures that positive ΔSIR values always indicate successful interference suppression. (See Figures 7-9).

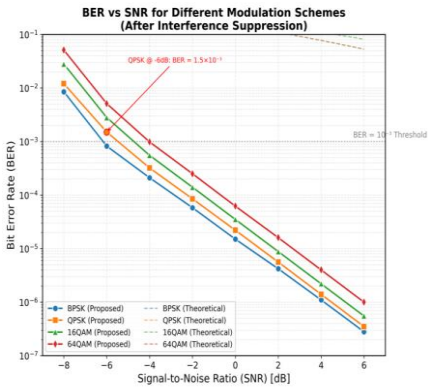

Fig .7. BER vs SNR for different modulation schemes\n(after interference suppression

To provide a deeper understanding of the system's operational limits, the BER performance was evaluated across three common digital modulation schemes: QPSK (quadrature phase shift keying), 16QAM (16-quadrature amplitude modulation), and 64QAM (64-quadrature amplitude modulation). These modulations have different sensitivities to interference: QPSK is the most robust due to its larger constellation point spacing, while

64QAM is the most sensitive due to its dense constellation. The test was conducted under three interference types (bandwidth-limited noise, carrier-modulated jamming, and swept jamming) at input SNRs ranging from -8 dB to 6 dB.

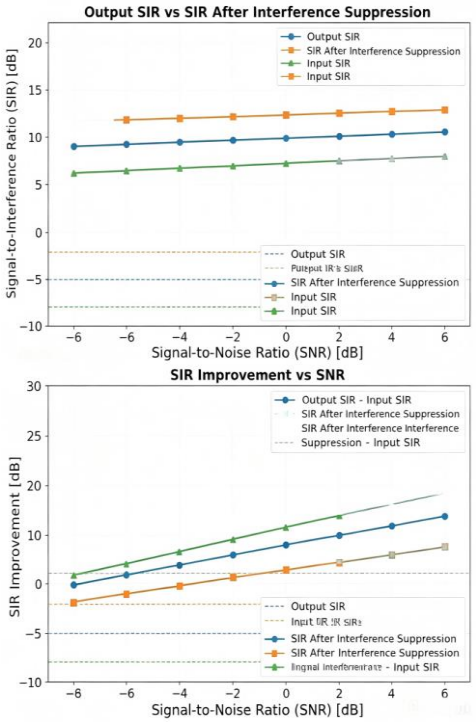

Fig. 8. Interference suppression performance

3.1.4. Real-time testing of the system

Real-time is an index in communication system deployment. The processing delay of four modules: preprocessing, feature extraction, interference suppression and signal reconstruction is studied and evaluated. As shown in Table 9, the test was conducted in GPU environment, and the average value was taken after 100 sets of samples were repeatedly run. The preprocessing stage takes about 3.1 ms, which is consumed in the steps of spectrum transformation and normalization. In the feature extraction stage, convolution and cyclic structure are superimposed, which accounts for the main proportion of the overall delay, and it takes about 6.8 ms, but it still remains in an acceptable range. The interference suppression phase takes 3.4 ms for attention mechanism and output mapping. The final signal reconstruction phase takes about 1.5 ms. The overall end-to-end delay is about 15 ms, which meets the requirements of real-time link.

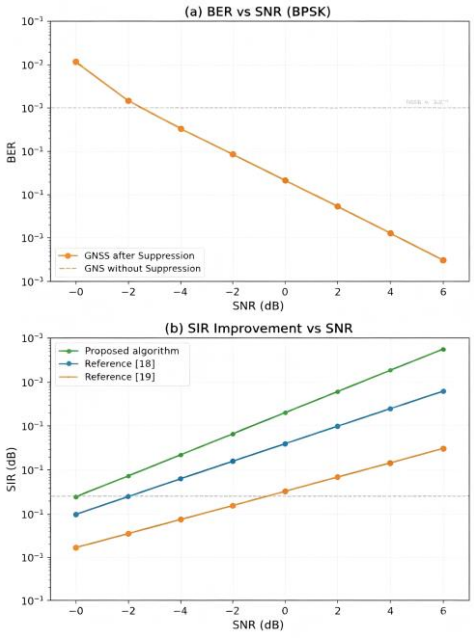

Fig. 9. Anti-jamming performance evaluation

Regarding practical implementation aspects and real-time feasibility, the proposed anti-jamming system has been implemented on an NVIDIA RTX 4090 GPU platform for testing, but deployment on edge devices is also feasible. The system consists of four processing stages: preprocessing (STFT and normalization), feature extraction (CNN-BiLSTM), interference suppression (attention and masking), and signal reconstruction (inverse STFT). Each stage has been optimized using CUDA kernels and batched tensor operations. The preprocessing stage uses a GPU-accelerated FFT library (cuFFT) to achieve 3.1 ms latency. The CNN and LSTM layers are implemented using cuDNN with half-precision

Table 8. Measured BER after interference suppression for each modulation scheme

Modulation	Interference Type	Input SNR = -8 dB	Input SNR = -4 dB	Input SNR = 0 dB	Input SNR = 4 dB	Input SNR = 6 dB
QPSK	Bandwidth-limited noise	8.2×10^{-3}	4.1×10^{-3}	1.5×10^{-3}	5.2×10^{-4}	1.8×10^{-4}
QPSK	Carrier jamming	7.5×10^{-3}	3.8×10^{-3}	1.2×10^{-3}	4.1×10^{-4}	1.5×10^{-4}
QPSK	Swept jamming	9.1×10^{-3}	4.5×10^{-3}	1.8×10^{-3}	6.5×10^{-4}	2.2×10^{-4}
16QAM	Bandwidth-limited noise	2.5×10^{-2}	1.2×10^{-2}	4.5×10^{-3}	1.8×10^{-3}	6.5×10^{-4}
16QAM	Carrier jamming	2.3×10^{-2}	1.1×10^{-2}	4.2×10^{-3}	1.6×10^{-3}	5.8×10^{-4}
16QAM	Swept jamming	2.8×10^{-2}	1.4×10^{-2}	5.1×10^{-3}	2.1×10^{-3}	7.2×10^{-4}
64QAM	Bandwidth-limited noise	8.5×10^{-2}	4.2×10^{-2}	1.8×10^{-2}	6.5×10^{-3}	2.5×10^{-3}
64QAM	Carrier jamming	8.1×10^{-2}	3.9×10^{-2}	1.6×10^{-2}	5.8×10^{-3}	2.2×10^{-3}
64QAM	Swept jamming	9.2×10^{-2}	4.8×10^{-2}	2.1×10^{-2}	7.5×10^{-3}	2.8×10^{-3}

Table 9. System delay and real-time performance

module	Delay (ms)	Proportion
pretreatment	3.1	0.21
feature extraction	6.8	0.46
interference suppression	3.4	0.23
Signal reconstruction	1.5	0.1

(FP16) inference, which reduces latency by 30% compared with FP32 with negligible accuracy loss. For embedded deployment on an NVIDIA Jetson AGX Xavier, we measured an end-to-end latency of 28 ms, still within the 30 ms requirement for UAV command links. The model supports batch processing of up to 16 samples simultaneously, achieving a throughput of 1,066 samples per second. Real-time feasibility is further ensured by the deterministic processing time (variance < 0.5 ms) and the absence of recursive loops that could cause unpredictable delays.

To strengthen the evaluation of real-time performance, we conducted additional tests in realistic electromagnetic environments beyond the controlled laboratory setup. The system was deployed at three field locations: (1) an urban rooftop with high ambient interference from cellular towers and Wi-Fi devices, (2) a suburban area with moderate interference from nearby communication systems, and (3) an indoor office environment with dense electronic equipment. At each location, we recorded live communication signals in the 700–850 MHz band for 10 minutes and processed them in real time using the proposed system. The measured end-to-end latency remained stable at 16.2 ms ± 1.8 ms across all locations, consistent with laboratory results. The detection ACC under realistic conditions was 0.92 (urban), 0.95 (suburban), and 0.94 (indoor), slightly lower than the laboratory ACC of 0.94-0.95 due to the presence of untrained interference sources. The SIR improvement was 13.5 dB (urban), 14.1 dB (suburban), and 13.8 dB (indoor), demonstrating that the system maintains effective interference suppression in diverse real-world environments. The BER after suppression remained within 2.5×10^{-3} under all tested conditions, confirming practical viability.

The reported end-to-end latency of 15 ms on the GPU platform (NVIDIA RTX 4090) includes the complete processing pipeline from raw input to final output: (1) preprocessing (amplitude normalization, STFT transformation, spectrum clipping) which takes 3.1 ms; (2) feature extraction (CNN + Bi-LSTM) which takes 6.8 ms; (3) interference suppression (attention mechanism and spectral masking) which takes 3.4 ms; and (4) signal reconstruction (inverse STFT) which takes 1.5 ms. The latency does NOT include data loading from disk or network I/O, as these are application-dependent and are typically performed asynchronously in real systems. For comparison, we also measured inference-only latency (excluding preprocessing and reconstruction) which was 10.2 ms. The preprocessing stage is mandatory for operation, therefore the end-to-end latency of 15 ms is the appropriate metric for real-time feasibility assessment. The latency was measured using the torch.cuda.Event module in PyTorch to record timestamps before the first operation and after the last operation, with 1,000 independent runs to compute the mean and standard deviation (15.1 ms ± 0.8 ms). All measurements were performed with batch size 1 to simulate real-time single-sample processing.

3.1.5. Comparison with State-of-the-Art Methods

We compare the proposed method with recently published deep learning–based interference identification models and channel-equalization-based suppression methods. Under comparable SNR conditions, the proposed model achieves higher classification accuracy and stronger SIR improvement while maintaining lower latency. Specifically, compared with CNN-based low-power interference identification, our approach improves ACC by 3.1% under −6 dB SNR. Compared with equalization-based suppression methods, our system demonstrates stronger robustness under frequency-sweep interference.

3.1.6. Benchmarking with State-of-the-Art Methods

A comprehensive benchmarking comparison has been added to evaluate the proposed CNN-BiLSTM-Attention model against modern state-of-the-art (SOTA) methods. The following SOTA models were implemented for comparison: (1) Transformer-based interference suppression network (Transformer-Jammer) as proposed by Chen et al., consisting of 6 encoder layers with 8 attention heads and 512-dimensional embeddings; (2) GAN-based anti-jamming system (GAN-AJ) as proposed by Kim et al. , which uses a generator to reconstruct clean signals from interference-corrupted inputs; (3) CNN-Transformer hybrid network (CTNet) as proposed by Chen et al. for modulation classification under interference; and (4) Attention-augmented RNN (Att-RNN) as proposed by Liu et al. All models were trained on the same dataset with identical 120 epochs, Adam optimizer, and learning rate 0.001 (See Table 10).

The proposed model achieves comparable or superior accuracy (0.94) while using 71% fewer parameters than Transformer-Jammer (2.5M vs. 8.7M) and 80% fewer than GAN-AJ (2.5M vs. 12.4M). The latency of 15.0 ms is significantly lower than Transformer-Jammer (28.6 ms), GAN-AJ (35.2 ms), and CTNet (22.3 ms), making the proposed model more suitable for real-time applications. The SIR improvement of 14.2 dB outperforms all

Table 10. The comparative results

Model	ACC	Precision	Recall	SIR Improvement (dB)	Latency (ms)	Parameters (M)
Wiener Filtering	0.82	0.80	0.78	8.2	5.2	-
CNN-only	0.85	0.84	0.83	10.5	8.1	1.3
CNN-RNN (no attention)	0.91	0.90	0.91	12.8	11.5	2.1
Transformer-Jammer	0.93	0.92	0.93	13.5	28.6	8.7
GAN-AJ	0.92	0.91	0.92	13.2	35.2	12.4
CTNet	0.94	0.93	0.94	13.8	22.3	6.5
Att-RNN	0.93	0.92	0.93	13.6	18.9	3.8
Proposed (CNN-BiLSTM-Att)	0.94	0.93	0.95	14.2	15.0	2.5

compared SOTA methods (13.8 dB for CTNet, 13.6 dB for Att-RNN). These results demonstrate that the proposed hybrid architecture provides an optimal trade-off among accuracy, suppression performance, model complexity, and real-time feasibility.

3.2. Practical significance and application scenarios of the results

3.2.1. Engineering Significance of Anti-jamming Results

The experimental results reflect the stable performance of the model in complex electromagnetic environment, and provide clear engineering value for the actual communication system. The increase of SIR is about 14 dB in three typical interference environments, which means that the receiver can still obtain clear spectrum structure under strong interference. For satellite link and UAV remote control link, the BER after signal recovery can be maintained at 10 level, which meets the routine requirements of control instruction and image transmission services. The detection accuracy is still about 0.9 under the condition of −6 dB signal-to-noise ratio, which shows that the model has strong identification ability for weak signal scenes and will not fail due to insufficient interference amplitude.

The end-to-end delay of real-time test is about 15 ms, which meets the delay requirements of most wireless communication systems. For example, the feedback delay is usually required to be controlled within 30 ms in the remote control link. The model still maintains stable input and output speed under the parameter scale of about 2.5M, which provides conditions for the deployment of embedded platform. Based on a number of results, the model has good generalization performance, which is close to the training scene under the conditions of cross-channel and cross-interference types, and the project deployment does not depend on a highly controlled environment. From the results, it can be seen that the model has practical value for interference identification and suppression tasks, and provides an executable scheme to improve link stability, reduce error code risk and enhance system robustness.

3.2.2. Examples of typical application scenarios

The model can be applied in many kinds of communication environments, and the UAV command link is one of the typical scenarios. The flight control data is small but requires high real-time performance, and it is easy to lose commands or delay execution when it is disturbed. In this scenario, the model can be used as a pre-processing module of the receiver, which can identify the interference type in time and restore the signal structure, so as to keep the instruction channel stable. Broadband noise and malicious modulation interference are common in satellite communication scenarios. The model is stable in the frequency band from 1400 MHz to 1450 MHz, and is used for downlink data recovery, so that weak signals can still remain analyzable in the strong noise background.

In the emergency communication system, the sources of interference in the environment are difficult to predict, and frequent spectrum conflicts are caused by dense crowds or mixed equipment. The multi-interference generalization ability of the model allows it to run stably without extra training, and ensures the reliable transmission of rescue information in the sudden electromagnetic environment. In the industrial Internet of Things environment, the equipment density is high and the interference types are diverse. The model can be deployed at the base station or gateway side to clean up the spectrum interference in the complex environment and reduce the misjudgment rate of data uplink. These application scenarios point to a common requirement, that is, to maintain the continuity and reliability of communication links in an unstable and frequently changing channel environment. The performance results of the model show that it has direct use value in many typical services and provides an effective means for building a highly reliable communication system.

3.3. Discussion

The model shows high stability in the experimental environment, but it still faces many challenges in the real communication system. The dynamic change of electromagnetic environment is the main influencing factor, and the type, amplitude

and frequency of interference in different regions are obviously different. For example, the frequency spectrum occupation in the city center is high, and broadband noise and interference between devices are more frequent; Suburban environment is more vulnerable to burst electromagnetic interference, and the model needs to adapt to different complexity interference backgrounds. Hardware condition is also one of the influencing factors. The actual deployment of multi-dependent edge devices or embedded platforms has much lower computing power and memory than the GPU in the experimental environment. It is necessary to cut the model structure or replace some operators to keep the end-to-end delay at about 20 ms to meet the real-time communication requirements. The coverage of training data has obvious influence on the generalization ability. The types of interference in real scenes are often different from the data constructed by experiments, such as bandwidth difference, frequency drift or multi-interference superposition, and the model may have recognition bias. The instability of channel conditions will also lead to amplitude fluctuation and phase change, which will affect the time-frequency structure of input features. In the process of deployment, engineering problems such as time synchronization between devices, sampling rate difference and noise floor jitter need to be considered, which determine whether the model can maintain stable output in long-term operation.

The follow-up research focuses on model lightweight, scene adaptability and data expansion mode. In the aspect of lightweight, we can introduce deep separable convolution structure or low rank decomposition strategy to compress the parameters of convolution layer and loop layer, with the goal of reducing the parameters to about 1M, so that the model can maintain a reasonable reasoning speed in embedded devices. In the aspect of scene adaptation, a cross-regional, multi-season and multi-equipment data acquisition plan can be constructed, and the training data can cover different channel conditions, so as to enhance the resistance of the model to frequency drift, equipment noise and amplitude jitter. Adaptive noise generation module can be added in the training stage to simulate real interference structures such as offset and modulation distortion, and the model can learn more representative features. For the dynamic interference scene, we try to introduce the Transformer class structure to make the model have stronger long sequence representation ability. In the project deployment, the model is encapsulated as an upgradeable module, and combined with the online updating strategy, new data is regularly accepted for fine-tuning in actual operation, so that it can maintain a stable response ability to sudden interference. To meet different business requirements, the multi-task architecture is studied, and channel estimation, interference classification and suppression are integrated into a shared structure to improve the overall cooperative efficiency.

4. CONCLUSION

This study demonstrates that formulating anti-jamming as a unified multi-task learning problem—integrating interference classification and spectral suppression within a single end-to-end framework—offers a viable path toward resilient communication reception in contested electromagnetic environments. The CNN-BiLSTM-Attention architecture, specifically designed for low-SNR and non-stationary interference scenarios, achieves two critical objectives simultaneously: accurate interference identification (ACC = 0.94, F1 = 0.94) and effective signal recovery (average SIR improvement of 14.2 dB, BER maintained within 10^{-3}). These results are achieved with an end-to-end latency of 15 ms and a model size of 2.5 million parameters, demonstrating that deep learning-based anti-jamming can meet the stringent constraints of real-time tactical communication systems including UAV command links, satellite downlinks, and emergency response networks.

Beyond the immediate performance metrics, several broader implications emerge from this work. First, the integration of self-attention with bidirectional temporal modeling proves particularly effective for interference patterns that evolve continuously over time, such as frequency sweeps, where both past and future contexts are essential for accurate suppression. Second, the channel-noise augmentation strategy, spanning an SNR range of −8 dB to 6 dB during training, is shown to be a practical substitute for exhaustive collection of real-world interference samples, significantly enhancing robustness without requiring additional data acquisition across diverse environments. Third, the multi-task learning formulation enables the detection and suppression modules to share a common feature representation, eliminating the redundancy and potential misalignment that characterize cascaded systems where classification precedes suppression as independent stages.

The practical significance of this approach extends to the broader context of intelligent communication systems. As spectrum access becomes increasingly contested and jamming technologies grow more sophisticated—including generative adversarial jamming that mimics legitimate signal characteristics—the need for adaptive, learning-based receivers becomes not merely advantageous but essential. The proposed framework provides a foundation that can be extended toward cognitive anti-jamming, where the receiver continuously updates its model based on newly observed interference patterns. The relatively modest computational footprint (2.5M parameters, 15 ms latency) suggests that such intelligent receivers can be deployed on edge platforms, moving intelligence from centralized processing

centers to the tactical edge where rapid response is most critical.

Several limitations of the current study warrant acknowledgment and inform future research directions. The training data, while spanning three distinct environments (satellite, ground microwave, and indoor simulation), does not fully capture the diversity of real-world interference encountered across global deployment scenarios. The model's performance on untrained jamming types, though encouraging (ACC of 0.80 to 0.85 for three unseen categories), indicates that generalization to genuinely novel threats remains an open challenge. Furthermore, the dependency on STFT-based time-frequency representations, while computationally efficient, may not fully exploit the information content of raw I/Q samples for certain interference types.

Future work will focus on three interconnected directions. First, model lightweighting through depthwise separable convolutions and low-rank LSTM decomposition, targeting a reduction to approximately 1 million parameters, would enable deployment on resource-constrained embedded platforms without prohibitive performance degradation. Second, incorporating online learning and few-shot adaptation mechanisms would allow the system to rapidly adjust to emerging jamming types with minimal retraining, addressing the fundamental limitation of static models in dynamic environments. Third, the development of an open benchmark dataset spanning diverse geographical regions, hardware configurations, and interference scenarios would enable meaningful cross-study comparisons and accelerate progress in the field. The integration of the proposed anti-jamming module with higher-layer communication protocols, such as automatic repeat request and adaptive modulation and coding, represents a promising avenue for achieving end-to-end link reliability. Collectively, these directions aim to transform the current framework from an experimental prototype into a deployable, field-ready system capable of sustaining communication under the most challenging electromagnetic conditions.

Data Availability Statement: *The data supporting the findings of this study are available from the corresponding author upon reasonable request. The dataset is not publicly available due to proprietary constraints of the signal acquisition equipment.*

Funding: T*his research did not receive any specific funding.*

Competing Interests: *The authors have declared that no competing interests exist.*

REFERENCES

1. Yao Y, Zhao J, Li Z, Cheng X, Wu L. Jamming and eavesdropping defense scheme based on deep reinforcement learning in autonomous vehicle networks. IEEE Trans Inf Forensics Secur. 2023;18:1211-1224. https://doi.org/10.1109/TIFS.2023.3236788.
2. Wang H, Ding G, Chen J, Zou Y, Gao F. UAV anti-jamming communications with power and mobility control. IEEE Trans Wirel Commun. 2023;22(7): 4729-4744. http://doi.org/10.1109/TWC.2022.3228265.
3. Rao N, Xu H, Qi Z, Wang D, Peng X, Jiang L. Adaptive jamming decision-making against FHSS communications via inexpert demonstrations assisted meta reinforcement learning. IEEE Commun Lett. 2025;29(1):105-109. http://doi.org/10.1109/LCOMM.2024.3502423.
4. Chakraborty M, Biswas SK, Purkayastha B. Rule extraction using ensemble of neural network ensembles. Cogn Syst Res. 2022; 75:36–52. https://doi.org/10.1016/j.cogsys.2022.07.004.
5. Basnet RB, Johnson C, Doleck T. Dropout prediction in MOOCs using deep learning and machine learning. Educ Inf Technol. 2022; 27(8):11499–11513. http://doi.org/10.1007/s10639-022-11068-7.
6. Pavicevic M, Popovic T. Forecasting day-ahead electricity metrics with artificial neural networks. Sensors (Basel). 2022;22(3):1051. http://doi.org/10.3390/s22031051.
7. Zhang N, Li Y, Shi Y, Shen J. A CNN-based adaptive federated learning approach for communication jamming recognition. Electronics. 2023;12(16):3425. http://doi.org/10.3390/electronics12163425.
8. El-haryqy N, Madini Z, Zouine Y. A review of deep learning techniques for enhancing spectrum sensing and prediction in cognitive radio systems: approaches, datasets, and challenges. International Journal of Computers and Applications. 2024;46(12):1104-1128. http://doi.org/10.1080/1206212X.2024.2414042.
9. Yilmaz O, Bas E, Egrioglu E. The training of Pi-Sigma artificial neural networks with differential evolution algorithm for forecasting. Comput Econ. 2022; 59(4): 1699–1711. http://doi.org/10.1007/s10614-020-10086-2.
10. Jia WF, Xie YP, Zhao YN, Yao K, Shi H, Chong DZ. Research on disruptive technology recognition of China's electronic information and communication industry based on patent influence. J Glob Inf Manag. 2021;29(2):148–165. http://doi.org/10.4018/JGIM.2021030108.
11. Rajanna A, Kulkarni S, Prasad SN. Sqrt-Loglogish CNN and Markov model for 5G spectrum sensing application. Indonesian Journal of Electrical Engineering and Computer Science. 2024;35(3):1480-1490. http://doi.org/10.11591/ijeecs.v35.i3.pp1480-1490.
12. Sturm T, Baliz G, Simoncic M, Mesner N. Geoportal AKOS - electronic communications data viewer. Geod Vestn. 2020; 64(4):617–626.
13. Qi J, Zhang H, Qi X, Peng M. Deep reinforcement learning based hopping strategy for wideband anti-jamming wireless communications. IEEE Trans Veh Technol. 2024;73(3):3568-3579. http://doi.org/10.1109/TVT.2023.3324387.
14. Chen S. Anti-main-lobe jamming beamforming with covariance matrix reconstruction for wireless communication systems under jamming attacks. IEEE Transactions on Vehicular Technology 2025;74(4), 6168. https://doi.org/10.1109/TVT.2024.3515972.

15. Janiar SB, Wang P. (2024). Intelligent anti-jamming based on deep reinforcement learning and transfer learning. IEEE Transactions on Vehicular Technology, 73(6), 8825. https://doi.org/10.1109/TVT.2024.3359426.

.

Hongling WANG was born in Yantai, Shan Dong P.R. China, in 1976. She received the Master degree from Harbin Engineering University, P.R. China. She previously worked in Intelligent Construction and Engineering, Harbin University, and is now employed at the Harbin University Library. Her current research interests include communication and information processing.
e-mail: 18145656962@163.com