Chinese Journal of Aeronautics, (2022), 35(9): 35–48

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

Large-scale real-world radio signal recognition with deep learning

Ya TU [a], Yun LIN [a,*], Haoran ZHA [a], Ju ZHANG [b], Yu WANG [c], Guan GUI [c], Shiwen MAO [d]

[a] College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China
[b] College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China
[c] College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[d] Department of Electrical and Computer Engineering, Auburn University, Auburn 36849, USA

Received 27 February 2021; revised 6 April 2021; accepted 24 May 2021
Available online 13 October 2021

KEYWORDS

Signal recognition;
Radio signal dataset;
Automatic Dependent Surveillance-Broadcast (ADS-B);
Deep learning;
Recognition benchmark

Abstract In the past ten years, many high-quality datasets have been released to support the rapid development of deep learning in the fields of computer vision, voice, and natural language processing. Nowadays, deep learning has become a key research component of the Sixth-Generation wireless systems (6G) with numerous regulatory and defense applications. In order to facilitate the application of deep learning in radio signal recognition, in this work, a large-scale real-world radio signal dataset is created based on a special aeronautical monitoring system - Automatic Dependent Surveillance-Broadcast (ADS-B). This paper makes two main contributions. First, an automatic data collection and labeling system is designed to capture over-the-air ADS-B signals in the open and real-world scenario without human participation. Through data cleaning and sorting, a high-quality dataset of ADS-B signals is created for radio signal recognition. Second, we conduct an in-depth study on the performance of deep learning models using the new dataset, as well as comparison with a recognition benchmark using machine learning and deep learning methods. Finally, we conclude this paper with a discussion of open problems in this area.

1. Introduction

In recent years, with the rapid development of algorithms, computation technologies, and the number of datasets, deep learning has made significant progress in computer vision, voice and natural language analysis.[1] In the implementation of deep learning, good-quality data collection plays a crucial role as the high performance of deep learning is mostly data-driven. For example, ImageNet was the first high-quality data-

* Corresponding author.
E-mail address: linyun@hrbeu.edu.cn (Y. LIN).
Peer review under responsibility of Editorial Committee of CJA.

Production and hosting by Elsevier

https://doi.org/10.1016/j.cja.2021.08.016
1000-9361

set including more than 14 million images in 20,000 categories, which drove the advances of deep learning in computer vision. After that, the general and robust image features in ImageNet have nurtured enormous successful computer vision deep learning models, such as AlexNet, VGG (Visual Geometry Group), and Deep Residual Network.[2] In the field of speech, the CSTR VCTK Corpus dataset consists of speech data collected from 110 English speakers with various accents. This dataset paved way for remarkable real-time Text-to-Speech (TTS) deep learning models such as WaveNet and Deep Voice.[3] The Bidirectional Encoder Representations from Transformers (BERT) has achieved great success in natural language processing with the help of BooksCorpus (including 800 M words) and English Wikipedia (including 2500 M words).[4] On some test datasets, BERT even outperforms average bilingual human translators.

Deep learning has become one of the most important research topics in the Sixth-Generation (6G) wireless communications,[5,6] which has been widely used in Multi-Input Multi-Output (MIMO),[7] resource management[8] and ultra-reliable and low-latency communications.[9] Automatic Modulation Classification (AMC) is considered as one of the most important techniques in 6G scenario. Hence, many researchers pay attention to get a better recognition performance based on deep learning.[5,6] Since the Convolutional Neural Network (CNN) is not good at learning temporal information in time series, researchers have proposed other deep learning methods to dip the signal persistent features. Rajendran et al.[10] proposed a new data-driven model for automatic modulation classification based on Long Short-Term Memory (LSTM), which is useful in classifying modulation signals with different symbol rates. The achieved accuracy of 75% on an input sample length of 64 for which it was not trained, substantiates the representation power of the model. Afan and Fan[11] demonstrated a novel method for the AMC based on autoencoder network, which is trained by a nonnegativity constraint algorithm. The results indicate that the Autoencoder with Nonnegativity Constraint (ANC) improves the sparsity and minimizes the reconstruction error in comparison with the conventional sparse autoencoder. Huang et al.[12] proposed a novel Gated recurrent residual neural Network (GrrNet) for feature-based AMC, where the amplitude and phase of the received signal are utilized as the inputs of GrrNet. Simulations are conducted to verify the classification performance and robustness of the proposed GrrNet and it is shown that GrrNet outperforms other recent deep learning based AMC methods. Zhang et al.[13] addressed the AMC based on CNN-LSTM which is a dual-stream structure by combining the advantages of CNN and LSTM. The experimental results not only demonstrate the superior performance of the proposed method compared with the existing state-of-the-art methods, but also reveal the potential of deep learning based approaches for AMC. Tu et al.[14] designed several key building blocks for Complex-Valued Neural Network (CVNN) for AMC. Additionally, signal recognition domain knowledge is also taken into consideration. Peng et al.[15] investigated several approaches to denote the complex signals into images with grid like topologies and utilized two CNN models (AlexNet and GoogLeNet) to learn features from these images for AMC. Similarly, Lin et al.[16] proposed contour stella image method, which can convey deep level statistical information by dot density in constellation diagram. This work bridges the gap between signal recognition and deep learning. Ma et al.[17] proposed a novel AMC method by using Cyclic Correntropy Spectrum (CCES) and deep Residual neural Network (ResNet). Experimental results confirm that the proposed algorithm outperforms the existing designs with much higher classification accuracy. Furthermore, some researchers are not only satisfied with accurate signal recognition but also consider few labeled data to achieve faster recognition. Tu et al.[18] applied Semi-Supervised Generative Adversarial Networks (SSGAN) to deal with unlabeled signal data and confirmed that the semi-supervised learning method is an effective way to exploit unlabeled data effectively to reduce over-fitting in deep learning. Wang et al.[19] proposed a novel deep learning based lightweight AMC (LightAMC) method with smaller model sizes and faster computational speed. The LightAMC method can effectively reduce model sizes and accelerate computation with the slight performance loss. Lin et al.[20] proposed a new filter-level pruning technique based on Activation Maximization (AM) that omits the less important convolutional filter. The proposed method can achieve equal or higher classification accuracy than conventional methods.

According to the above survey, various deep learning based AMC methods have been proposed to improve the signal recognition performance. But there still exist some problems. Firstly, the dataset is often generated from the simulation system, such as MATLAB, GNURadio, and Python. Secondly, the number of signal categories is limited. Thirdly, the real radio signal environment is not considered. Actually, all the defects are due to the lack of a large-scale and high-quality radio signal dataset in the real world. However, producing a good dataset requires a lot of time, money and manpower. Due to the unique characteristics of radio signal, it is very different to label a large-scale and high-quality dataset with the manual operation. Hence, in this paper, an automatic collection and labeling system is designed to capture radio signals in the real world without human participation. The main contributions of this paper are summarized as follows.

(1) The Automatic Dependent Surveillance-Broadcast (ADS-B) system is chosen as the signal source from the real world. An automatic collection and labeling system is designed to capture over-the-air ADS-B signals. By means of data cleaning, a large-scale real-world radio signal dataset is acquired, which includes 426613 pieces of long signals from 1661 categories of airplanes, and 167234 pieces of short signals from 1713 categories of airplanes.

(2) Numerous experiments are carried out to demonstrate the effectiveness and accuracy of the dataset under different scenarios. A rigorous benchmark is provided based on machining learning and deep learning techniques. The dataset will be released to the research community, which will catalyze the development of many new algorithms, models, and evaluation methods.

The rest of this paper is organized as follows. In Section 2, we introduce the work on ADS-B signals and the model of the radio channel used in this paper. In Section 3, we present the general framework of the ADS-B dataset generation process and highlight the key elements in this framework. In Section 4, we introduce the signal recognition models based on three different deep learning algorithms. In Section 5, we introduce a series of experiments conducted with the dataset. Finally, we conclude this paper in Section 6 with a discussion of future work.

2. Related work

2.1. Related datasets

In recent years, several datasets have been proposed in the field of radio signal recognition, which are shown as follows:

(1) RadioML 2016.10A and RadioML 2016.04C. The RadioML 2016.10A[21] and RadioML 2016.04C datasets[22] provide 170163 and 220000 labeled In-Phase and Quadrature (I/Q) samples (two-dimensional data), respectively. Designed using GNU Radio, each synthetic dataset consists of 11 modulations (8 digital and 3 analog). The signals are synthesized for evaluating output under distinct signal and noise scenarios at various SNR levels with mild Local Oscillators (LO) drift, light fading, and multiple distinct labelled SNR increments.

(2) RadioML 2018.01A. RadioML 2018.01A[23] is a rich modulation classification dataset containing more than 2.5 million radio signals covering up to 24 analog and digital modulation formats in a wide range of SNR [−20: 2: 30] dB. This dataset contains both artificial virtual channel effects and over-the-air recordings of 24 tested forms of optical and analog modulation.

(3) ORACLE. The Oracle dataset[24] refers to a specific radio signal from a wide pool of machines that are bit-like (i.e. with the same hardware, protocol, physical address, and MAC ID) utilizing only physical layer IQ samples. ORACLE follows two approaches: (A) it trains a CNN to identify special hardware-centric signatures embedded in the transmitter radio chain (e.g., IQ imbalance, DC offsets, etc.); (B) it uses a receiver-feedback to insert modifications in the transmitter chain to conduct channel-independent fingerprinting of Radio Frequency (RF).

Datasets are an integral part of the field of machine learning. Major advances in this field can result from advances in learning algorithms (such as deep learning), computer hardware, and less intuitively, the availability of high-quality training datasets. Many major AI breakthroughs have actually been constrained by the availability of high-quality training datasets, but not by algorithmic advances. We have seen that numerous awesome datasets have been instrumental in advancing computer vision, NLPs and deep learning research. However, there are several datasets for radio signal recognition. But we believe that the high-quality and real-world labeled training datasets are still absent.

These existing datasets have been widely used in research, but they may not be sufficient for many practical scenarios. First, most categories of these datasets are generated by simulation, which could be very different from that captured from real applications. Second, the number of signal categories in these datasets may not be sufficient, which limits their applications in AMC. Third, these datasets are hard to extend for emerging wireless systems, and thus they could be out-of-date in the near future. The new radio signal dataset proposed in this paper will address all these issues.

2.2. Benchmark recognition approach

(1) Statistical features

In this paper, the bispectrum method[25] is used to extract the statistical features. The bispectral transform has the benefits of conservation of phase, variability of size, and invariance of time change. These benefits mean that during transformation, a lot of details in the signal are retained, which are particularly useful for the classification and recognition of radiation source devices. In addition, high-order spectral analysis can, technically, minimize the Gaussian noise. However, since the transformation of the bispectrum is not conducive to detection and identification, it generally absorbs a number of energies for two-dimensional picture recognition.

The two-dimensional signal image, then, will first be converted by the line integral bispectrum method to a one-dimensional signal. Axial Integral Bispectrum (AIB), Circumferential Integral Bispectrum (CIB), Radial Integral Bispectrum (RIB) and Rectangular Integral Bispectrum (RIB) are four types of well-integrated bispectrum algorithms. First, it is possible to express the signal bispectrum of $x(t)$ as follows:

$$B_x(\omega_1,\omega_2)=\sum_{\tau_1=-\infty}^{\infty}\sum_{\tau_2=-\infty}^{\infty}C_{3x}(\tau_1,\tau_2)\mathrm{e}^{-i(\omega_1\tau_1+\omega_2\tau_2)t} \tag{1}$$

where

$$\begin{aligned}C_{3x}(\tau_1,\tau_2)&=\int_{-\infty}^{\infty}x^*(t)x(t+\tau_1)x(t+\tau_2)\mathrm{d}t \\ &=E\{x^*(t)x(t+\tau_1)x(t+\tau_2)\}\end{aligned} \tag{2}$$

Among the four bispectrum algorithms, we choose AIB in this paper, which can be described as follows:

$$\begin{aligned}\mathrm{AIB}(\omega)&=\frac{1}{2\pi}\int_{-\infty}^{\infty}B(\omega_1,\omega_2)\mathrm{d}\omega_2 \\ &=\frac{1}{2\pi}\int_{-\infty}^{\infty}B(\omega_1,\omega_2)\mathrm{d}\omega_1\end{aligned} \tag{3}$$

From the Fourier transform projection, it can be shown that AIB can be considered as the axial portion Fourier transform of the signal's third-order correlation function.

(2) Machine learning based classifier

A variety of lightweight machine learning models or analytic decision processes can be used to classify the signal after mapping statistical features to a class name. Popular methods include Support Vector Machine (SVM),[26] Decision Trees (DTree),[27] and K-Nearest Neighbors (KNN),[28] which are briefly introduced in the following:

(A) SVM is based on the minimization of systemic risk and has strengths over generative learning. For example, where the ultimate aim is to obtain a classifier, not the distributions, the Vapnik-Chervonenkis theory[29] provides relevant objections against attempting to estimate probability distributions in generative learning.

(B) DTree is a decision support mechanism that utilizes a decision-making tree-like paradigm and its future effects, namely the implications of chance events, resource costs, and utility. It is an efficient way of expressing an algorithm that includes only conditional statements of power.

(C) KNN is a type of instance learning, or lazy learning, where the function is only locally approximated and all calculation is postponed before evaluation of the function. Since this algorithm relies on distance for classification, if the features come in vastly different scales, normalizing the training data can improve its accuracy dramatically.

2.3. Deep learning

To optimize wide parametric neural network models, deep learning depends entirely on Stochastic Gradient Descent (SGD). The key strategy over the years stays relatively unchanged. Neural networks consist of a set of layers that map the input h_0 of each layer to output h_1 utilizing dense parametric matrix operations accompanied by non-linearities. This can be represented clearly as follows:

$$h_1 = \max(0, h_0 W + b) \tag{4}$$

where the weights, W, have the dimension $|h_0 \times h_1|$, the bias, b, has the dimension $|h_1|$ (both constituting θ), and max is applied element-wise per output $|h_1|$ (i.e., as the Rectified Linear Unit (ReLU) activation function).

Traditionally, preparation leverages a loss function (L). In this case (for supervised classification), categorical cross-entropy of one-hot recognized class labels y_i (an all zero vector, except for a one value at the class index i of the correct class) and predicted class values $\widehat{y}_i$, is used.

$$L(y, \widehat{y}) = \frac{-1}{N} \sum_{i=0}^{N} [y_i \lg(\widehat{y}_i) + (1 - y_i) \lg(1 - \widehat{y}_i)] \tag{5}$$

For each epoch n within the network ($f(x, \theta_n)$), back propagation of loss gradients can be used to iteratively update network weights (θ) until validation loss no longer decreases.

Back propagation of loss gradients can be used to iteratively update network weights (θ) for each epoch n within the network ($f(x, \theta_n)$) until validation loss no longer decreases. The essential optimization method is as follows:

$$\theta_{n+1} = \theta_n - \eta \frac{\partial L(y, f(x, \theta))}{\partial \theta_n} \tag{6}$$

3. Dataset generation approach

3.1. Radio signal source

As mentioned above, the new radio signal dataset should have three important advantages. First, it should be collected in a real-world scenario with a large number of categories. Second, it should not be complicated to acquire and at a low expense. Finally, it should be easy to update in the future. Therefore, the source of radio signal is very important for the dataset. Fortunately, the ADS-B system is a good fit for the application. The ADS-B system is widely used to monitor the status of airplanes, which is very important for the safety of air traffic. The advantages of ADS-B include:

(1) Large Scale. The International Civil Aviation Organization (ICAO) requires that every airplane installs the ADS-B system, and every airplane should periodically transmit over-the-air ADS-B signal. Therefore, there are a lot of ADS-B signals from different airplane categories in the open radio environment.

(2) Easy labeling. The ADS-B system follows a standard and open protocol (DO-260B), which is designed by the ICAO. Therefore, it is easy to automatically collect and label the radio signal without human participation.

(3) Open Source. The ADS-B system works at the frequency of 1090 MHz, which is a passive receiving system and designed for open-source. Therefore, for scientific research, everyone can receive and collect ADS-B signals without causing security and privacy concerns.

Two parts, including the preamble and the data block, consist of the ADS-B signal, which are shown in Fig. 1. The data block uses the pulse position modulation method: a high pulse following a low pulse represents "1", and a low pulse following a high pulse means "0".

In practice, the lengths of received ADS-B signals are different. In the preamble, there are 8 microseconds of signal header with a total of 4 pulses at fixed positions, which can be used to detect and synchronize the ADS-B signal.[30] In the data block, there are two different data formats. The long data format is 112 bits, and the short data format is 56 bits. From Fig. 1, it can be seen that the long and short data formats have the same structure in the first 32 bits. The ICAO code appears both in long and short signals, which is used to identify the airplane. Therefore, the ICAO code will be used as the unique identification of the airplane to labels the signals in the dataset. Long and short signals will both be labeled in the dataset, with increased amount of signal types.

3.2. Radio signal capture

3.2.1. Overall architecture of capture system

The overall system architecture of the proposed signal capture system is shown in Fig. 2. The Software Defined Radio (SDR) device is used to detect and capture the baseband I/Q data of ADS-B signal, and an automatic decoding algorithm is used to obtain the individual identity (ID) of the airplane. Then, an automatic clustering and labeling algorithm is used to label the baseband I/Q data with the corresponding airplane ID. In this way, a dataset will be obtained after continuous collection of ADS-B signals over a period of time. The entire system works well without requiring human participation, which will greatly save the cost, time, and human power for constructing a dataset.

3.2.2. Hardware setup of capture system

The hardware configuration of the proposed signal collecting system is shown as follows:

(1) Signal collecting device: SM200B is a SDR platform, which is produced by Signal Hound, Inc.. The collecting parameters are summarized in Table 1.

(2) Signal processing device: An HP Laptop computer is used for signal decoding, labeling, and storage. The laptop configuration is Intel(R) Core (TM) i7-10750H CPU @ 2.60 GHz, 32 GB RAM, and 256 GB SSD Hard Disk.

(3) Antennas device: A 1090 MHz omnidirectional antenna is used to collect over-the-air ADS-B signal.

As shown in Fig. 3, in the actual collecting environment, we select an open and unobstructed place in the acquisition environment to avoid the influence of the surrounding environment on the received signal fingerprint to the greatest extent.

3.2.3. Software algorithm of capture system

The software algorithm comprises five parts, including resampling, header search, information decoding, CRC check, and positioning and labeling.

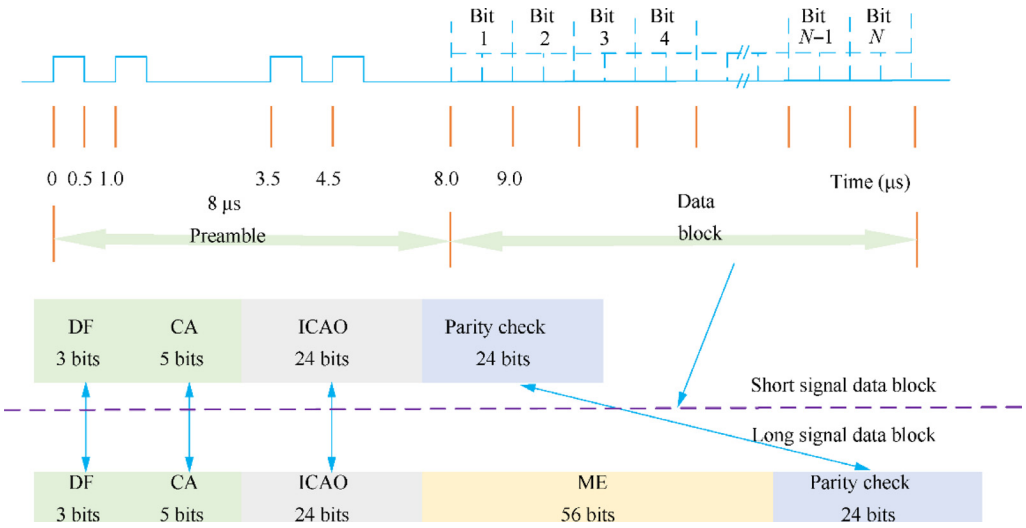


Fig. 1 Structure of ADS-B signal.

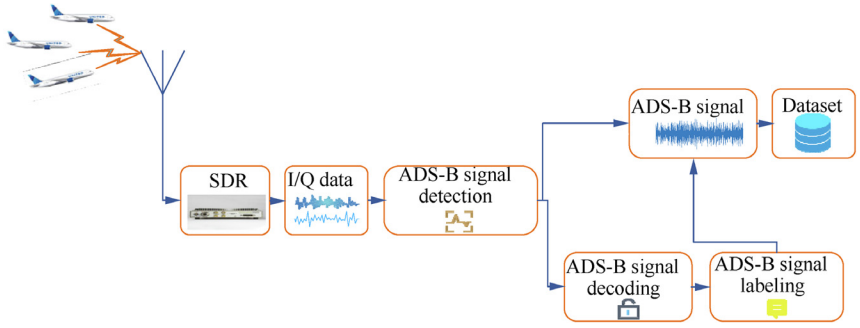


Fig. 2 Overall architecture of capture system.

Table 1 SM200B parameters during recording.

Parameter	Value
Sampling frequency (Mbps)	50
IF (Hz)	0 (baseband)
Center frequency (MHz)	1090
Bandwidth (MHz)	10
Gain (dB)	30

Fig. 3 Actual collecting environment.

(1) Resampling. This is used to reduce the amount of data, increase the rate of data processing, and extend the adaptability of different processing devices.

(2) Header Search. This is to detect the beginning of the ADS-B signal and synchronize the signal by finding the peak value of the cross-correlation output in a certain interval.

(3) Information Decoding. According to the DO-260B standard, the received ADS-B signal can be decoded to obtain specific information, which includes the airplane ID, status, position, and so on. Such information will be used for labeling the radio signal.

(4) CRC Check. Cyclic Redundancy Check (CRC) is an error-detection code that is widely used in digital networks and storage systems to identify unintentional data adjustments.

(5) Positioning and Labeling. The resampling factor D_{factor} and the position index L_D of the header are used to determine the starting position of the radio signal, i.e., $L_{\text{start}} = D_{\text{factor}} L_D$.

According to the signal length, the ending position of the radio signal can be easily calculated. Finally, a clustering method is used to label the ADS-B signal with the corresponding airplane ID. A high-quality radio signal dataset can be obtained without human operation in this way.

The overall dataset preprocessing procedure is summarized in Algorithm 1.

Algorithm 1. Software algorithm.

Input: I/Q signal $z_{\mathrm{BI}}(n)$, $z_{\mathrm{BQ}}(n)$, and the front-end sampling rate$1/T_{\mathrm{s}}$

1.Calculate the resampling factor D_{factor} and obtain the received I/Q signal $z_{\mathrm{BIR}}(n)$,$z_{\mathrm{BQR}}(n)$

2. Extract the envelope signal $a(n) = \sqrt{z_{\mathrm{BIR}}(n)^2 + z_{\mathrm{BQR}}(n)^2}$

3. Calculate the correlation coefficient $R_{s1,s2}(\tau)$ to find signal header position index L_D

4. Decode the received signal and get the airplane information

5. Check the effectiveness of decoding ADS-B information

6. Combine the airplane ID with the received I/Q signal, and store them to the database

Output: A high-quality radio signal dataset

3.3. Statistics and visualization of dataset

Data collection lasted for nearly one month. Tens of thousands of signals have been acquired, but we should choose the high-quality data. Therefore, after data cleansing, 26613 pieces of long signals in 1661 categories of airplanes, and 167234 pieces of short signals in 1713 categories of airplanes have been chosen for the dataset. The detailed quantity distribution of the captured ADS-B signals is shown in Fig. 4. In order to ensure the richness and balance of signal samples, 530 categories of long signals and 198 categories of short signals have been selected to build the dataset. Every category consists of 200 to 600 signal samples. In the following experiments, we will randomly select 80% of the data as the training set, 10% of the data as the validation set, and 10% of the data as the test set.

At a sampling rate of 50 MHz, the long signal length is 6000 sampling points, and the short signal is 3000 sampling points. Four ADS-B signals from different airplanes are shown in Fig. 5.

4. Signal recognition models

4.1. CNN-based model

It is well understood what deep neural networks actually learn as discriminating features in CV applications. For example, in CNN, the first layers are trained to detect small-scale "edges", which become increasingly complicated as the network deepens. CNN also has the benefit of being shift-invariant: converted weights detect patterns in arbitrary positions in the sequence in each layer, and the presence of a signal is passed to a higher layer by a max-pool layer. This is precisely the property that makes it excellent for these networks to detect.

The convolution operation can be written as

$$\mathrm{conv}(I,K)_{x,y} = \sum_{i=1}^{n_H}\sum_{j=1}^{n_W}\sum_{k=1}^{n_c} K_{i,j,k} I_{x+i-1,y+j-1,k} \tag{7}$$

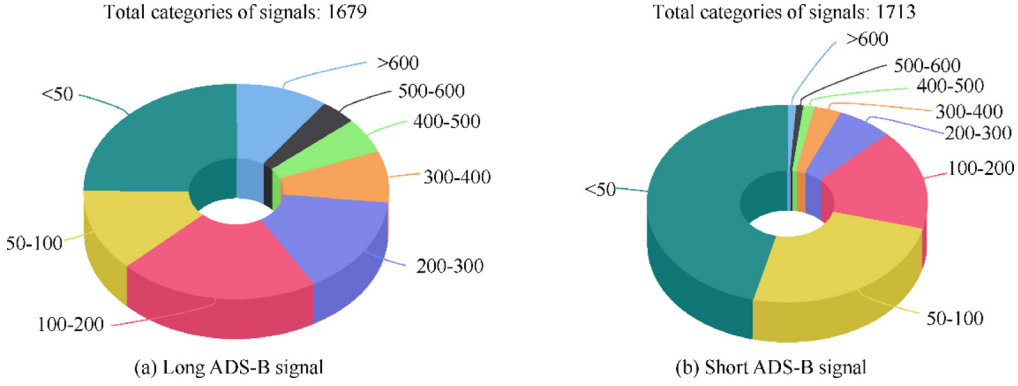
where I and K denote the input and convolutional layer weights, respectively, and $H, W,$ and$C$ denote the input's height, width, and channel, respectively.

However, in the wireless domain, CNN does not operate on images but on samples of I/Q. In the I/Q plane, various radio signal waveforms present various transition patterns. For example, transitions between (1, 0) and (–1, 0) are typical for BPSK signals, but they do not appear in QPSK signals, which have a constellation that is significantly different. This can constitute a unique "signature" of the signal that the CNN filters can ultimately learn.

The structure and parameters of the CNN used in this paper are provided in Table 2.

4.2. LSTM-based model

Recurrent Neural Network (RNN) is an advanced deep learning model for extracting temporal features, and it is generally applied to model language sequences and multimodal time series. Different from CNN, whose input dimensionality is fixed, RNN is suitable to analyze sequential data with variable lengths. In the wireless domain, RNN not only considers the current transition patterns in I/Q samples, but also looks back into historical transition patterns. In this way, RNN performs



(a) Long ADS-B signal (b) Short ADS-B signal

Fig. 4 Quantity distribution of the captured ADS-B signals.

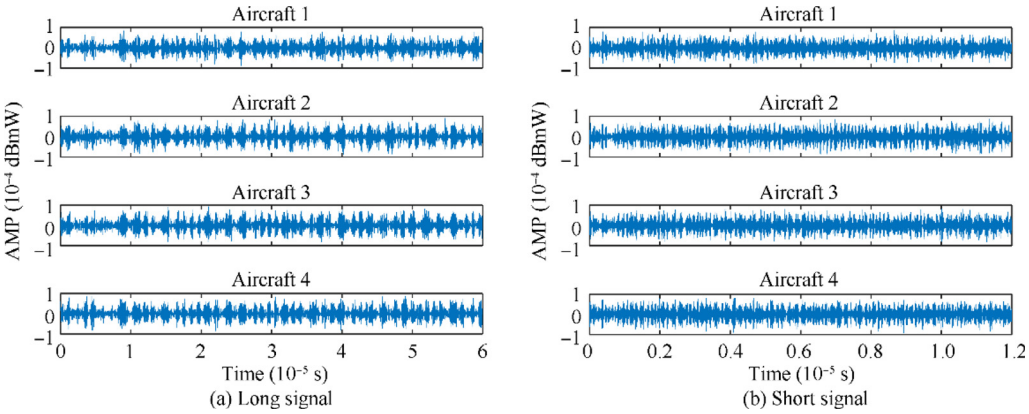


Fig. 5 Four types of ADS-B signal of different airplanes from dataset.

Table 2 Structure and parameters of convolution neural network.

Layer	Dimension	Activation
Input	1 × Signal length × 2	
Convolution	2 × Signal length × 50	ReLU
Batch normalization	2 × Signal length × 50	
Max pooling	2 × Signal length/2 × 50	
Convolution	2 × Signal length/2 × 75	ReLU
Batch normalization	2 × Signal length/2 × 75	
Max pooling	2 × Signal length/4 × 75	
Convolution	2 × Signal length/4 × 100	ReLU
Batch normalization	2 × Signal length/4 × 100	
Max pooling	2 × Signal length/8 × 100	
Convolution	2 × Signal length/8 × 150	ReLU
Batch normalization	2 × Signal length/8 × 150	
Max pooling	2 × Signal length/16 × 150	
Convolution	2 × Signal length/16 × 200	ReLU
Batch normalization	2 × Signal length/16 × 200	
Max pooling	2 × Signal length/32 × 200	
Convolution	2 × Signal length/32 × 300	ReLU
Batch normalization	2 × Signal length/32 × 300	
Average pooling	2 × Signal length/64 × 300	
Flatten	2 × Signal length/64 × 300	
Output	1 × classes number	Softmax

well by using more sequence features. Hence, the simple RNN can written as

$$h_t = \sigma_h(W_h x_t + U_h h_{t-1} + b_h) \tag{8}$$

$$y_t = \sigma_h\left(W_y h_t + b_y\right) \tag{9}$$

where x_t is the input vector, h_t is the hidden layer vector, W_h, W_y, U_h are RNN weight parameters, b b_h, $b_y$denotes RNN bias parameters, and σ_h denotes the activation function.

Usually, RNN models are trained with Back Propagation Through Time (BPTT), which may result in gradient explosion or gradient disappearance. In order to address the problem, several variants of RNN have been developed, such as LSTM, which is one of the most well-known and the most widely used models. However, LSTM has a high computing complexity, while Gated Recurrent Unit (GRU) greatly simplifies the LSTM structure, with a simple recurrent unit more effective than the RNN model.

The structure and parameters of the RNN used in this paper are provided in Table 3.

4.3. CVNN-based model

Complex-valued signals are observed in a wide variety of technologies, such as wireless networking, sensor array signal processing, as well as biomedical research and physics.[31] The outcome of the operation of Complex-Valued Neural Networks (CVNN) contains values that, as a consequence of the complex operation, carry features of both I/Q components. Let the complex kernel be $A + \mathrm{j}B$ and the complex input signal be $X + \mathrm{j}Y$. We can store the outcome as

(1) Separated channel: $XA - YB$

(2) Mixed channel: $XB + YA$

Mathematically, the outcome of this operation is still real-valued in one channel and imaginary in the other channel. To this end, the mixed channel will allow the CVNN to learn signal coherence information. It has been shown that the CVNN outperforms the Real-Valued Neural Network (RVNN) in high signal coherence regions (e.g., the high SNR region).[14]

The structure and parameters of the CVNN used in this paper are provided in Table 4.

5. Recognition performance analysis

Signal recognition efficiency of deep learning approaches would be greatly influenced by the architecture, deployment,

Table 3 Structure and parameters of LSTM.

Layer	Dimension	Activation
Input	1 × Signal length × 2	
LSTM	1 × Signal length × 128	ReLU
LSTM	1 × Signal length × 128	ReLU
LSTM	1 × Signal length × 128	ReLU
Flatten	1×(Signal length × 128)	
Dense	1 × 1024	ReLU
Dense	1 × 1024	ReLU
Output	1 × classes number	Softmax

Table 4 Structure and parameters of CVNN.

Layer	Dimension	Activation
Input	1 × Signal length × 2	
Complex convolution	1 × Signal length × 50	CReLU
Complex batch normalization	1 × Signal length × 50	
Max pooling	1 × Signal length/2 × 50	
Complex convolution	1 × Signal length/2 × 75	CReLU
Complex batch normalization	1 × Signal length/2 × 75	
Max pooling	1 × Signal length/4 × 75	
Complex convolution	1 × Signal length/4 × 100	CReLU
Complex batch normalization	1 × Signal length/4 × 100	
Max pooling	1 × Signal length/8 × 100	
Complex convolution	1 × Signal length/8 × 150	CReLU
Complex batch normalization	1 × Signal length/8 × 150	
Max pooling	1 × Signal length/16 × 150	
Complex convolution	1 × Signal length/16 × 200	CReLU
Batch normalization	1 × Signal length/16 × 200	
Max pooling	1 × Signal length/32 × 200	
Complex convolution	1 × Signal length/32 × 300	CReLU
Complex batch normalization	1 × Signal length/32 × 300	
Average pooling	1 × Signal length/64 × 300	
Flatten	1 × Signal length/64 × 300	
Output	1 × classes number	Softmax

training, and data consideration. Therefore, in this section, several experiments are conducted to evaluate the most common design parameters, which will greatly impact the recognition performance, including different classifier, signal categories, sampling rate, and signal impairments.

5.1. Dataset and deep learning configuration

In the following experiments, long and short signals are separated to evaluate the recognition performance. The entire dataset will be separated into three non-overlapping parts, which include the training set (80% of the dataset), the validation set (10% of the dataset), and the testing set (10% of the dataset). Therefore, there are 144550 and 50484 samples in the long and short signal training sets, respectively; 18068 and 6311 samples in the long and short signal validation sets, respectively; 18068 and 6311 samples in the long and short signal test sets, respectively.

One epoch means that a whole dataset is transferred forward and backward through the neural network once, and batch size is the total number of training examples present in a single batch. To balance the performance and time consumption, the maximum training epoch is 1000 and the batch size is 256 for all the deep learning models in the following experiments.

Early stopping is a way to define an unspecified large number of training epochs and avoid training until the output of the model finishes improving on a validation dataset holdout. Early stopping is scheduled to end the training phase in this experiment when there is no change across 20 epochs.

The rate of learning controls how easily the model is adapted to the problem. When smaller adjustments are made to the weights in each update, smaller learning rates need more training epochs, whereas larger learning rates contribute to faster improvements which need less training epochs. The initial learning rate is 0.001 for all the models. The decreased learning rate enables our models to descend into areas of the loss landscape that are "more optimal." The learning rate decreasing scheme in the experiments is as follows:

$$\alpha = \begin{cases} 10^{-3}, & \text{epoch} \le 15 \\ 10^{-4}, & 15 < \text{epoch} \le 40 \\ 10^{-5}, & \text{Otherwise} \end{cases} \tag{10}$$

During the training process, the optimizer will tweak and change the parameters (i.e., the weights) of the model to try and minimize the loss function and make predictions as correct and optimized as possible. In our experiments, (stochastic gradient descent) SGD with momentum is chosen for all the deep learning models.

Considering the effectiveness of the imbalanced sample categories, we choose mean Average Precision (mAP) to reduce the influence from imbalance dataset. The mAP metrics we used is defined as

$$\text{mAP} = \frac{\sum_{q=1}^{Q} P(q)}{Q} \tag{11}$$

where Q is the number of queries and $P(q)$ is precis for each query.

5.2. Recognition performance of different classifiers

In this part, three different machine learning and three different deep learning methods are selected to compare their recognition performance using the ADS-B signal dataset, and the comparison results are presented in Fig. 6.

(1) BiSpec + KNN. The KNN classifier combined with BiSpectrum features. The number of neighbors is set to 5, the uniform weight function is applied, and minkowski distance is adopted as the distance metric.

(2) BiSpec + Dtree. The decision tree classifier combined with BiSpectrum features. The Gini coefficient is chosen to measure the quality of a split, the minimum number of samples required to split an internal node is two, and the minimum number of samples required to be at a leaf node is one.

(3) BiSpec + SVM. The SVM classifier combined with BiSpectrum features. The radial basis function is chosen as the kernel, and regularization is ℓ_2 regularization, whose strength is 1.0.

(4) CNN. The CNN-based deep learning algorithm introduced in Section 4.

(5) LSTM. The LSTM-based deep learning algorithm introduced in Section 4.

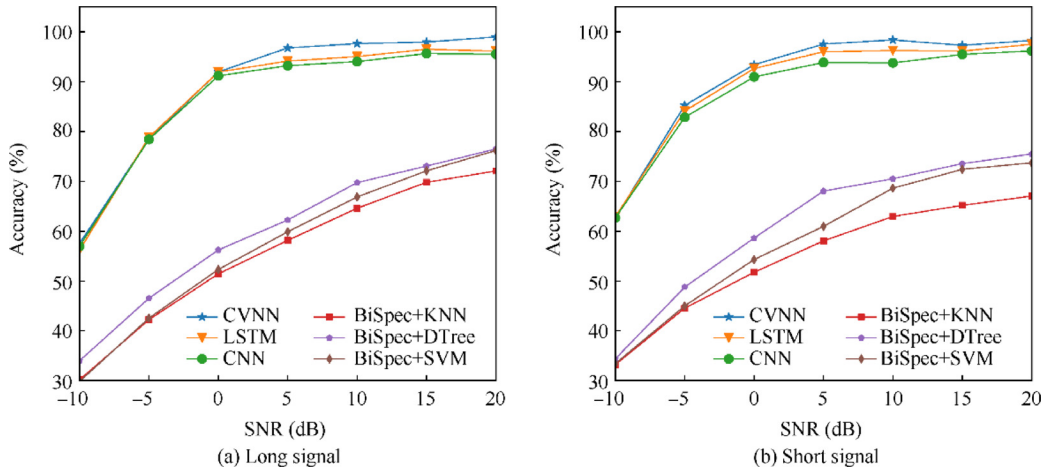


Fig. 6 Recognition performance at different SNR levels.

(6) CVNN. The CVNN-based deep learning algorithm introduced in Section 4.

According to Fig. 6, several observations can be made as follows:

(1) The recognition performance of deep learning is much better than that of machine learning. This is due to different feature extraction mechanisms. With the help of extra layers, deep learning can extract robust and accurate features automatically without human participation. However, machine learning should use the handcrafted features, which have a limited capacity to obtain correct real-world signal features.

(2) The recognition rate of deep learning will be improved with increased SNR, but it will become stable in the high SNR region. According to Fig. 6, when the SNR is higher than 0 dB, the performance rates of all the deep learning models will exceed 90%.

(3) The CVNN achieves the highest recognition rate among all the models at high SNR. The reason is that the correlation information from the I/Q channels is not polluted by noise at high SNR levels. Therefore, according to the correlation information from the I/Q channels, the CVNN can acquire more information to extract features, which lead to better recognition performance at high SNR.

(4) We compared three different machine learning and three different deep learning methods. The accuracy that we achieved from CVNN, RNN, and CNN is approximate to 99.8%, 97.2% and 96% respectively at 20 dB for long and short ADS-B signal, which implies that this dataset is high-quality labeled and the noise in the signal is very low. In this way, the reader can consider that this dataset is clean and add any influence on it.

5.3. Recognition performance under different numbers of categories

It is common to receive hundreds to thousands of categories of ADS-B signals. Therefore, it is crucial to explore how the number of categories of ADS-B signals influences the recognition performance. In this experiment, the numbers of categories are $N_c = 530$, 424, 318, 212, and 106 (corresponding to 100%, 80%, 60%, 40%, and 20% of long signals) and $N_c = 198$, 158, 119, 79, and 40 (corresponding to 100%, 80%, 60%, 40%, and 20% of short signals), which are randomly selected from the ADS-B signal dataset. To eliminate the randomness of selecting categories, 20 experiments are conducted and the average results are presented. Then the deep learning models mentioned in Section 4 are applied to test their recognition performance under different SNR levels, and the result is shown in Fig. 7.

According to Fig. 7, some conclusions can be drawn:

(1) With the increase of signal categories, the recognition rate will decrease in low SNR. However, the recognition rate will also improve with the increase of SNR. Therefore, the recognition rate of different category numbers will become stable at the same level in high SNR. It indicates that deep learning method has a better capacity of feature extraction and finds a better decision boundary in high SNR, even at a large category number.

(2) CVNN model is proved to be the best recognition method for both long and short ADS-B signal at any category number. It indicates that we should fully use I/Q information to improve the recognition rate in the future.

5.4. Recognition performance at various sampling rates

A higher sampling rate requires a higher performance receiver. However, a higher performance receiver may not always be available. An experiment is conducted to find out how sampling rate affects the ADS-B recognition performance. In the experiment, MATLAB is used to resample the original signal at 50 MHz, and then resample the long and short signals with sampling rates of 50 MHz, 40 MHz, 30 MHz, 20 MHz, and 10 MHz. To guarantee that we obtain the interested band, we used an Anti-Aliasing Filter (AAF) which is a filter used before a signal sampler restricts the bandwidth of a signal.

The recognition performance under different sampling rates are presented in Fig. 8. It indicates that the sample rate has a great influence on recognition rate for both long and short signal. Because the information of original signals will be lost in sample acquisition with a low sampling rate, according to Fig. 8, the good recognition rate can be obtained with the sample rate higher than 40 MHz, and we know that the signal bandwidth of ADS-B signal is 10 MHz. Therefore, we can draw a conclusion that the sample rate should be four times higher than the bandwidth of original signal.

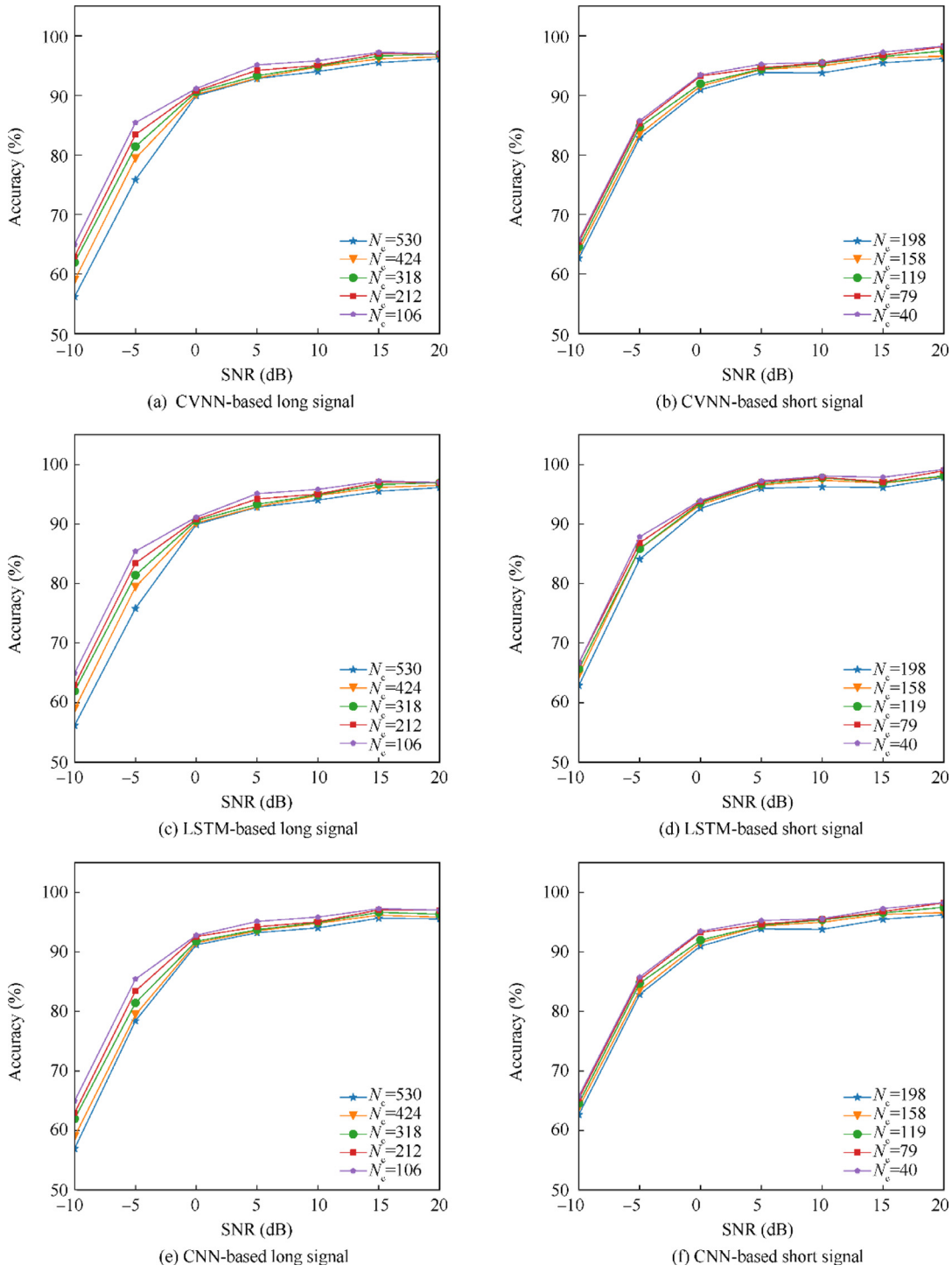


Fig. 7 Recognition performance at different categories number for different models.

5.5. Recognition performance under various signal impairments

Complex and various channels lead to the impairments of signal in real world, whereas the Additive White Gaussian Noise (AWGN) is commonly utilized in simulation and modeling. It is important to analyze how well-trained classifiers do and compare their performance under those transmission impairments.

One of the several non-ideal situations that may influence the baseband receiver design is Carrier Frequency Offset

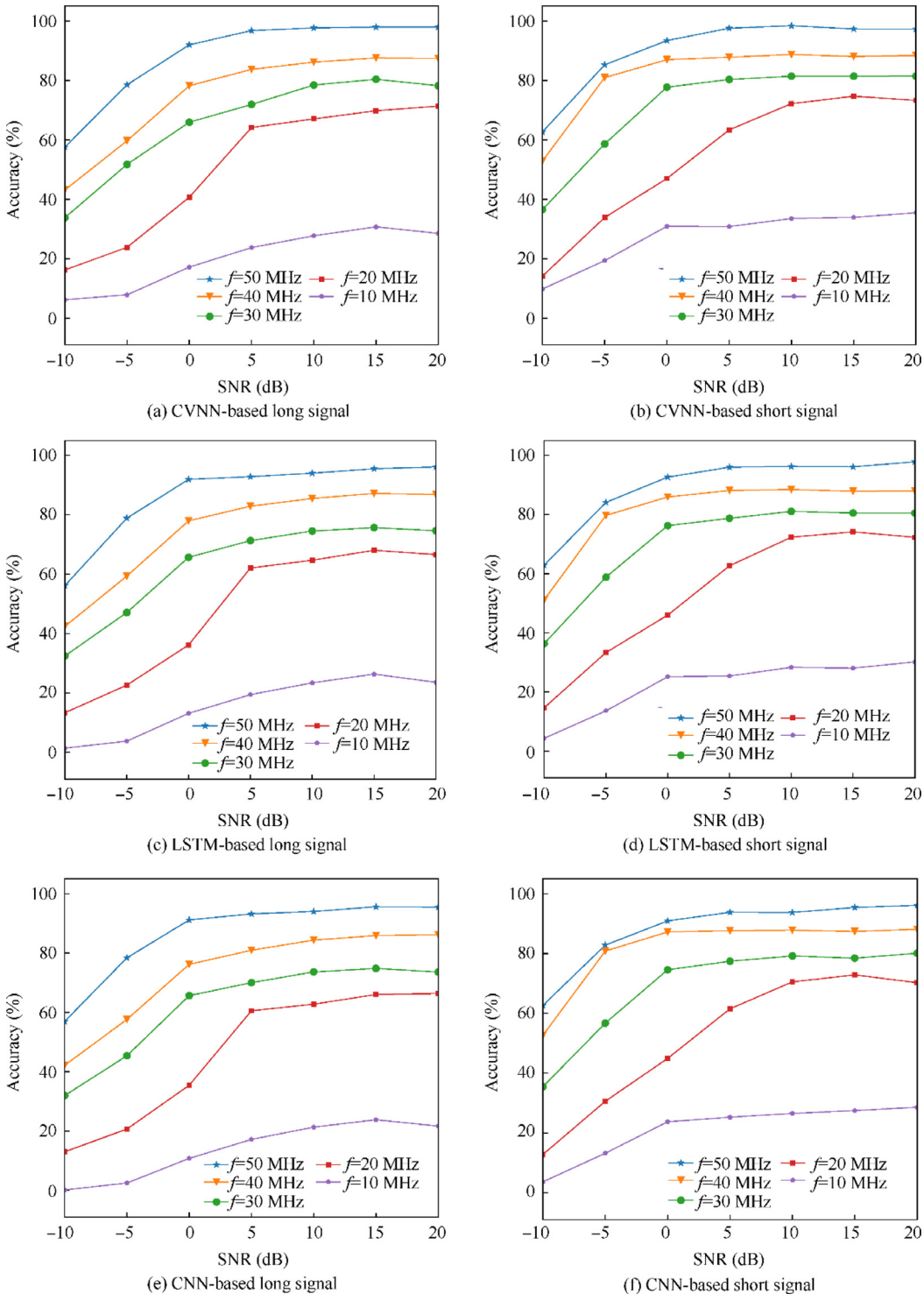


Fig. 8 Recognition performance at different sampling rates with different models.

(CFO). CFO usually happens where the local oscillator signal in the receiver is not compatible with the carrier in the obtained signal for down-conversion. Two significant reasons may be related to this phenomenon: the frequency mis-match between the transmitter and the oscillators of the receiver; the Doppler effect when the transmitter and/or receiver pass.

Like the carrier frequencies, transmitter and receiver sampling frequencies used by the Digital to Analog Converter (DAC) and Analog-to-Digital Converter (ADC) are generally slightly mismatched. Such impairment is known as the Sampling Rate Offset (SRO) and will also degrade the system performance.

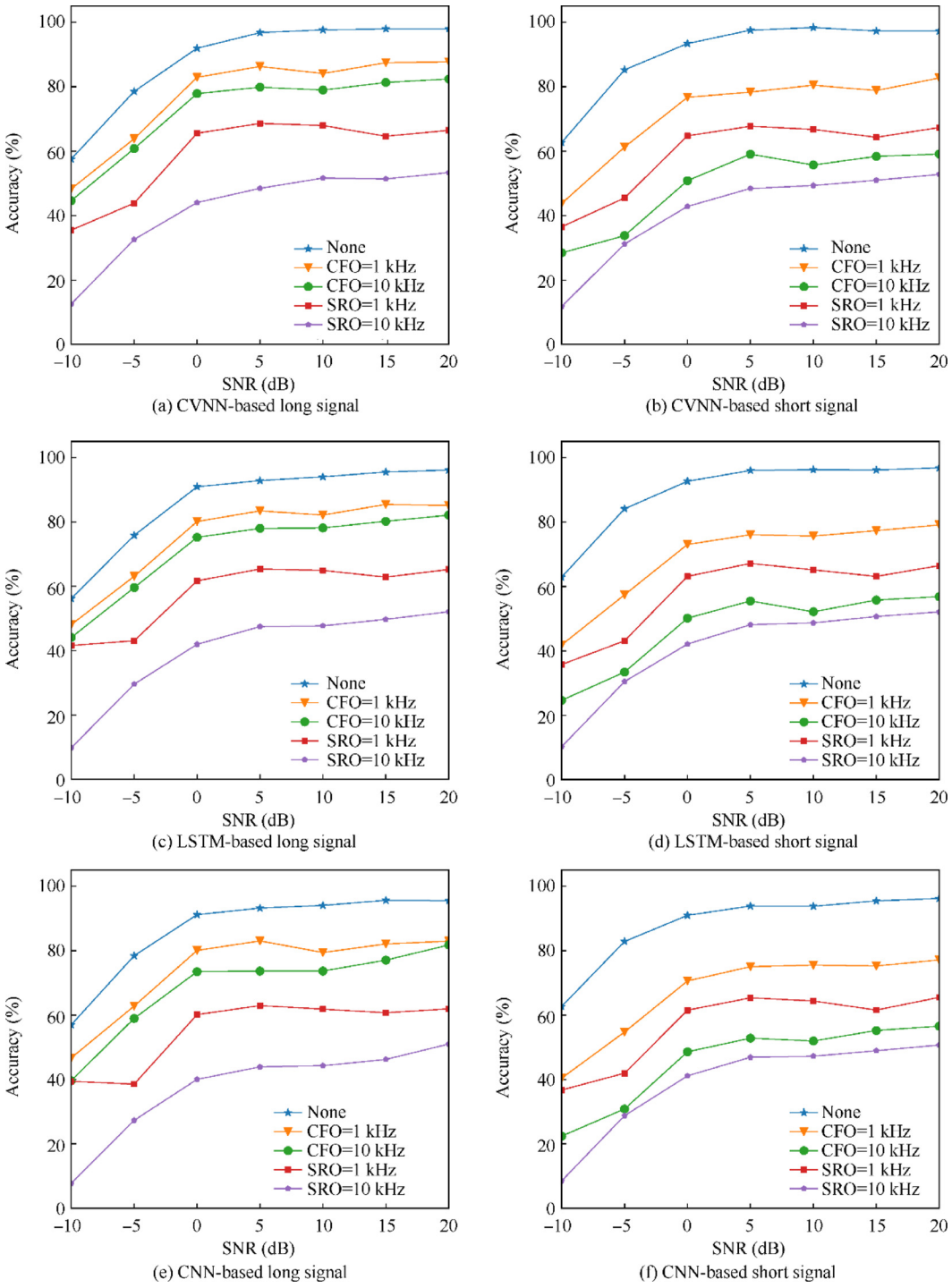


Fig. 9 Recognition performance under different signal impairments with different models.

In this experiment, common signal impairments, such as SRO and CFO, are considered. For long and short ADS-B signals, SRO is set to 1 KHz or 10 KHz, and CFO to 1 KHz or 10 KHz. The experimental results are presented in Fig. 9.

The following observations can be made from the results in Fig. 9:

(1) It is obvious that signal impairments have a great impact on the recognition performance, and a higher offset leads to a larger decline of the recognition performance. In this way, a high-performance receiver is needed to mitigate such influence.

(2) Due to CFO, we suppose that the received frame at right frequency f_c is down-converted with a local carrier frequency $(1+\varepsilon)f_c$. The resulting baseband samples are subject to phase rotation. The waveform of ADS-B signal will be distorted nonlinearly and make it hard to recognize when facing worse CFO.

(3) Due to SRO, it is supposed that the sampling duration is $T_s^{'} = (1-\varepsilon)T_s$. Hence, if the sampling ε offset is positive (or

negative), we obtain the history or future sample in the frame, which is taken at time $\tau_n = n\varepsilon T_s$ earlier (or later) than it should. The ideal feature like pike and high frequency changing in the waveform will be lost. It will be more difficult for deep learning method to learn the correct features in ADS-B signal.

(4) The impairments of signal cannot be avoided in real world, and they will make a great influence on the recognition performance. Therefore, first, we should try to collect much larger data from real world, which includes different signal impairments. Deep learning can learn to deal with the complex conditions. Second, we should try to find better signal processing method to reduce the impairments before inputting into deep learning models.

6. Conclusions

In this paper, a large-scale and real-world dataset was presented, which was designed to advance the deep learning application in radio signal recognition. We also presented the key part about how to collect the dataset without human participation. Benchmark results were also presented using the dataset under different radio signal recognition scenarios, such as different classifiers, different numbers of categories, different sample rates, and different signal impairments. As shown in the results, the deep learning methods have a great potential in signal recognition. Finally, this dataset will be released to the research community, which will catalyze the development of new algorithms, models, and evaluation studies.

In the future, this dataset would become a primary reference for a wide variety of academic studies on signal processing, including the following potential applications:

(1) A benchmark dataset. The current benchmark datasets in radio signal processing, such as RadioML 2016.10A and RadioML 2018.01A, have played a critical role in advancing signal recognition research. The proposed high-quality, rich-diversity, large-scale, and real-world radio signal dataset will become a new and valuable benchmark dataset for future research in the field.

(2) Automatic data labeling. In this paper, an automatic data labeling framework for radio signals is proposed. Additional ADS-B signal data could be added in the dataset without much cost. The proposed system will be a useful tool to collect other types of signal in the future.

(3) Training resources for other ADS-B tasks. This dataset could serve as a pretrain dataset for other tasks of the ADS-B system, such as ADS-B demodulation and separation. This is mainly due to the rich, diverse, and robust features captured in this dataset. One interesting research direction could be: transferring the deep learning models to conduct few-shot learning about ADS-B signals in other scenarios.

(4) This paper's dataset and codes have been open in "https://gitee.com/heu-linyun". We hope it can be used to attract more researchers to join the fields, and better deep learning method can be provided to improve the recognition performance.

However, there are also some improvements that should be continued with more effort. In wireless communications, fading is variation of the attenuation of a signal with common various variables. To give a comprehensive benchmark result about our dataset, we will consider channel fading factor in the future. First, the category and number of signals are still not enough, which should be further replenished. Second, we only capture the ADS-B signal in one city, the location should change to several different cities, and we can even design a distributed capturing system. Third, we only label the airplane ID with the ADS-B signal, and more tag should be used, such as SNR, speed, and altitude. Fourth, the evaluation method is not perfect, a more comprehensive target should be proposed. Finally, a lot of deep learning methods should be included in future recognition task. Fifth, the long tail concept has found some ground for signal classification, such as minor class ADS-B signal recognition, zero-shot learning, and supervised learning. We will consider long-tail application in this dataset, especially for those ADS-B categories that have less than 50 samples.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 61771154) and the Fundamental Research Funds for the Central Universities, China (No. 3072021CF0815). This work was also supported by the Key Laboratory of Advanced Marine Communication and Information Technology, Ministry of Industry and Information Technology, Harbin Engineering University, Harbin, China.

References

1. Pouyanfar S, Sadiq S, Yan Y, et al. A survey on deep learning. *ACM Comput Surv* 2019;**51**(5):1–36.
2. Guo J, He H, He T, et al. GluonCV and GluonNLP: Deep learning in computer vision and natural language processing. arXiv preprint: 1907.04433, 2019.
3. Alam M, Samad MD, Vidyaratne L, et al. Survey on deep neural networks in speech and vision systems. *Neurocomputing* 2020;**417**:302–21.
4. Otter DW, Medina JR, Kalita JK. A survey of the usages of deep learning for natural language processing. *IEEE Trans Neural Netw Learn Syst* 2021;**32**(2):604–24.
5. Kato N, Mao B, Tang F, et al. Ten challenges in advancing machine learning technologies toward 6G. *IEEE Wirel Commun* 2020;**27**(3):96–103.
6. Gui G, Liu M, Tang F, et al. 6G: Opening new horizons for integration of comfort, security, and intelligence. *IEEE Wirel Commun* 2020;**27**(5):126–32.
7. Dong P, Zhang H, Li GY, et al. Deep CNN-based channel estimation for mmWave massive MIMO systems. *IEEE J Sel Top Signal Process* 2019;**13**(5):989–1000.
8. Liang F, Shen C, Yu W, et al. Towards optimal power control via ensembling deep neural networks. *IEEE Trans Commun* 2020;**68**(3):1760–76.
9. She C, Dong R, Gu Z, et al. Deep learning for ultra-reliable and low-latency communications in 6G networks. *IEEE Netw* 2020;**34**(5):219–25.
10. Rajendran S, Meert W, Giustiniano D, et al. Deep learning models for wireless signal classification with distributed low-cost spectrum sensors. *IEEE Trans Cogn Commun Netw* 2018;**4**(3):433–45.

11. Afan A, Fan YY. Automatic modulation classification using deep learning based on sparse autoencoders with nonnegativity constraints. *IEEE Signal Process Lett* 2017;**24**(11):1626–30.
12. Huang S, Dai R, Huang J, et al. Automatic modulation classification using gated recurrent residual network. *IEEE Internet Things J* 2020;**7**(8):7795–807.
13. Zhang Z, Luo H, Wang C, et al. Automatic modulation classification using CNN-LSTM based dual-stream structure. *IEEE Trans Veh Technol* 2020;**69**(11):13521–31.
14. Tu Ya, Lin Y, Hou C, et al. Complex-valued networks for automatic modulation classification. *IEEE Trans Veh Technol* 2020;**69**(9):10085–9.
15. Peng S, Jiang H, Wang H, et al. Modulation classification based on signal constellation diagrams and deep learning. *IEEE Trans Neural Netw Learn Syst* 2019;**30**(3):718–27.
16. Lin Y, Tu Ya, Dou Z, et al. Contour stella image and deep learning for signal recognition in the physical layer. *IEEE Trans Cogn Commun Netw* 2021;**7**(1):34–46.
17. Ma JT, Lin SC, Gao HJ, et al. Automatic modulation classification under non-Gaussian noise: A deep residual learning approach. ICC 2019 - 2019 IEEE International Conference on Communications (ICC); 2019 May 20-24; Shanghai, China. Piscataway: IEEE Press; 2019.p.1–6
18. Tu Y, Lin Y, Wang J. Semi-supervised learning with generative adversarial networks on digital signal modulation recognition. *Comp, Mater Continua* 2018;**55**(2):243–54.
19. Wang Yu, Yang J, Liu M, et al. LightAMC: Lightweight automatic modulation classification via deep learning and compressive sensing. *IEEE Trans Veh Technol* 2020;**69**(3):3491–5.
20. Lin Y, Tu Y, Dou Z. An improved neural network pruning technology for automatic modulation classification in edge devices. *IEEE Trans Veh Technol* 2020;**69**(5):5703–6.
21. O'Shea TJ, Corgan J, Clancy TC. *Convolutional radio modulation recognition networks. Engineering applications of neural networks*. Cham: Springer International Publishing; 2016. p. 213–26.
22. O'Shea TJ, Corgan J, Clancy TC. Unsupervised representation learning of structured radio communication signals. *2016 First International Workshop on Sensing, Processing and Learning for Intelligent Machines* (SPLINE); 2016 Jul 6-8; Aalborg, Denmark; Piscataway: IEEE Press; 2016: p. 1–5.
23. O'Shea TJ, Roy T, Clancy TC. Over-the-air deep learning based radio signal classification. *IEEE J Sel Top Signal Process* 2018;**12**(1):168–79.
24. Sankhe K, Belgiovine M, Zhou F, et al. No radio left behind: radio fingerprinting through deep learning of physical-layer hardware impairments. *IEEE Trans Cogn Commun Netw* 2020;**6**(1):165–78.
25. Lin Y, Jia JC, Wang S, et al. Wireless device identification based on radio frequency fingerprint features. *ICC 2020–2020 IEEE International Conference on Communications (*ICC); 2020 Jun 7-11; Dublin, Ireland. Piscataway: IEEE Press; 2020. p.1–6.
26. Muller FCBF, Cardoso C, Klautau A. A front end for discriminative learning in automatic modulation classification. *IEEE Commun Lett* 2011;**15**(4):443–5.
27. Meng F, Chen P, Wu L, et al. Automatic modulation classification: a deep learning enabled approach. *IEEE Trans Veh Technol* 2018;**67**(11):10760–72.
28. Aslam MW, Zhu ZC, Nandi AK. Automatic modulation classification using combination of genetic programming and KNN. *IEEE Trans Wirel Commun* 2012;**11**(8):2742–50.
29. Editorial G. Vapnik-Chervonenkis (VC) learning theory and its applications. *IEEE Trans Neural Netw* 1999;**10**(5):985–7.
30. Haoran Z, Sen W, Shihao W, et al. A method using blind source separation to improve the decoding efficiency of space-based ADS-B receiver. *Int J Performability Eng* 2020;**16**(6):950.
31. Gopalakrishnan S, Cekic M, Madhow U. Robust wireless fingerprinting via complex-valued neural networks. *2019 IEEE Global Communications Conference (GLOBECOM)*; 2019 Dec 9-13; Waikoloa, USA. Piscataway: IEEE Press; 2019. p. 1–6.