

## 面向射频指纹识别的高效 IQ 卷积网络结构\*

崔天舒<sup>1,2</sup>, 黄永辉<sup>1</sup>, 沈明<sup>3</sup>, 张晔<sup>1,2</sup>, 崔凯<sup>1,2</sup>, 赵文杰<sup>1</sup>, 安军社<sup>1,2</sup>

(1. 中国科学院国家空间科学中心 复杂航天系统电子信息技术重点实验室, 北京 100190;

2. 中国科学院大学 计算机科学与技术学院, 北京 100049; 3. 奥尔堡大学, 丹麦 奥尔堡 DK-9220)

**摘要:** 现有应用于射频指纹识别的卷积网络对时序同相正交(in-phase and quadrature-phase, IQ)信号的处理都是将其简单视为图像进行的, 存在识别准确率低和计算量大的问题。针对以上问题, 提出了一种基于 IQ 相关特征的卷积神经网络结构。该网络分步提取了 IQ 相关特征及时域特征, 通过自适应平均池化获得了各通道特征均值, 并用单个全连接层进行分类。实验结果表明, 较传统卷积网络结构, 所提网络在多种场景下的识别准确率更高, 并且计算量更小。

**关键词:** IQ 信号; 信号特征; 射频指纹; 卷积神经网络; 深度学习

**中图分类号:** TN92 **文献标志码:** A **文章编号:** 1001-2486(2022)04-180-10

## High-efficiency IQ convolutional network structure for radio frequency fingerprint identification

CUI Tianshu<sup>1,2</sup>, HUANG Yonghui<sup>1</sup>, SHEN Ming<sup>3</sup>, ZHANG Ye<sup>1,2</sup>, CUI Kai<sup>1,2</sup>, ZHAO Wenjie<sup>1</sup>, AN Junshe<sup>1,2</sup>

(1. Key Laboratory of Electronics and Information Technology for Complex Space Systems, National Space Science Center, Chinese Academy of Sciences, Beijing 100190; 2. School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China; 3. Aalborg University, Aalborg DK-9220, Denmark)

**Abstract:** Existing convolutional neural networks, which are used for radio frequency fingerprints recognition, process time-sequenced IQ (in-phase and quadrature) signals as images directly, resulting in low recognition accuracy and high computation complexity. IQCNet (convolutional neural network structure based on IQ correlation features), an efficient convolutional network structure, was proposed. IQCNet firstly extracted IQ correlation features and time domain features, then obtained the average value of each channel features through adaptive average pooling, and finally used only one fully connected layer for classification. Experimental results under a variety of channel conditions show that IQCNet improves recognition accuracy greatly with lower computation complexity compared with traditional convolutional neural networks.

**Keywords:** IQ signal; signal characteristics; radio frequency fingerprint; convolutional neural network; deep learning

当今社会无线网络无处不在, 在日常生活与工业生产中的应用越来越广泛, 尤其是随着物联网的发展, 无线设备的数量呈现指数级增长, 但无线网络在提供便利的同时, 也在隐私性和安全性方面带来了巨大的挑战。例如, 在低功耗广域网(low power wide area network, LPWAN)中, 由于成本、功耗和带宽等的限制, 终端向基站传输的数据往往是不加密或者弱加密的, 这就留下了安全漏洞。

电子元器件以及印刷电路板在制造和使用过程中存在制造容差与漂移容差, 容差效应导致即使是同一厂家同一型号同一系列甚至是同一批次的无线发射设备的实际硬件参数也会存在差

异<sup>[1]</sup>, 进而导致每个发射机个体都有其独特特性, 这一特性被称为射频指纹(radio frequency fingerprint, RFF)。射频指纹识别, 是利用无线信号识别特定发射设备的方法, 而信号指纹识别<sup>[2-4]</sup>和 RF-DNA<sup>[5-7]</sup>(RF distinct native attribute)等也可认为是与射频指纹识别不同名称的相同概念。射频指纹是无线设备的固有特征, 很难被篡改, 因此可通过射频指纹识别来确认无线设备身份, 增强网络安全。

此前, 比较流行的射频指纹识别方法是基于人工特征提取的方法, 先由本领域专家提取信号关键特征, 如同相正交(in-phase and quadrature-phase, IQ)不均衡<sup>[8]</sup>、幅度误差<sup>[9-10]</sup>和相位误

\* 收稿日期: 2020-11-16

基金项目: 中国科学院复杂航天系统电子信息技术重点实验室自主部署基金资助项目(Y42613A32S)

作者简介: 崔天舒(1986—), 男, 山东诸城人, 博士研究生, E-mail: cuitianshu@nssc.ac.cn;

黄永辉(通信作者), 男, 研究员, 博士, 博士生导师, E-mail: yonghui@nssc.ac.cn

差<sup>[11-12]</sup>等时域信号特征,或者频域变换<sup>[5,13]</sup>、Hilbert 变换<sup>[14-15]</sup>、小波变换<sup>[16-17]</sup>等域变换后再提取特征,然后采用机器学习方法进行分类。通常,需要选取几项甚至十几项特征,计算复杂度高,实际部署使用困难。同时,受通信协议的多样性与多径干扰效应的影响,该方式在特征选择和判决准则方面也缺乏普适性。

随着大数据和人工智能技术的发展,深度学习在图像识别<sup>[18-19]</sup>、语音识别<sup>[20-21]</sup>和自然语言处理与理解<sup>[22-23]</sup>等领域都取得了巨大的成功。射频指纹识别本质上也是一种模式识别,可以将深度学习与射频指纹识别技术深度融合,利用深度学习的方法,自动提取无线信号的模式特征,避免基于经验的人工特征提取,提高复杂电磁环境下发射机的身份识别能力。

在电磁信号识别领域,近年来采用卷积神经网络的方法越来越多。文献[24]提出了一种信噪比(signal noise ratio, SNR)自适应的兴趣区选择多分辨率卷积神经网络算法,在 SNR = 30 dB 的视距条件下,对 54 台 ZigBee 设备的分类精度高达 97%。文献[25]采用卷积神经网络对功放的非线性特征进行识别,实验结果表明,当信噪比大于 20 dB 时对 7 台通用软件无线电外设(universal software radio peripheral, USRP)设备的识别准确率能达到 80%。文献[26]以原始 IQ 采样数据为输入,经过离散傅里叶变换(discrete Fourier transform, DFT)将时域数据变换为频域表达,再经过数据图像化处理,将连续多次 DFT 的结果组合成一张二维频谱瀑布图,然后再输入卷积神经网络,像图像识别一样进行分类识别,虽然将 IQ 数据转换成时频图像进行处理的可读性更强,但相对于直接采用卷积网络处理原始 IQ 采样数据,这种方法在预处理过程中增加了额外计算量。但目前即使是直接处理 IQ 采样数据的卷积网络<sup>[27-31]</sup>,也存在信息利用率不高和计算量大等问题。

图 1 为采用 IQNet 的射频指纹识别处理流程。面向射频指纹识别应用,本文提出了一种基于 IQ 相关特征的卷积神经网络结构:IQNet,此网络以原始 IQ 信号为输入,首先提取信号的 IQ 相关特征,然后再提取时域特征,最后采用全局平均池化获得各通道特征均值进行分类,具有信息利用率高、参数数量少和计算量小的特点。

### 1 IQ 卷积网络结构分析

在图像识别领域,网络输入数据格式是  $M \times$

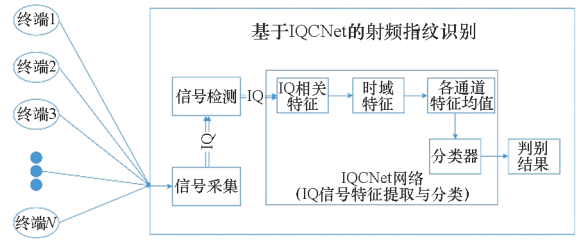


图 1 射频指纹识别处理流程

Fig. 1 RF fingerprint identification processing flow

$N$ ,数据点间存在空间关系,具有各向同性的性质,2 个维度间可进行相同操作。而在射频指纹识别领域,原始 IQ 采样数据格式为  $N \times 2$ ,其中  $N$  对应信号的时间长度,2 对应同相分量  $I$  与正交分量  $Q$ ,两个维度不具备相同性质,无法进行图像处理一样的 2 维度对称操作,所以目前卷积神经网络对 IQ 信号的处理都是采用一维卷积提取时间维度特征<sup>[27-31]</sup>,而忽略了 IQ 间相关特征,进而导致相位信息损失,降低了识别率。

在射频指纹识别领域,当前的卷积神经网络(convolutional neural networks, CNN)都是参照 AlexNet<sup>[32]</sup>的设计:首先采用大滤波器组的卷积层(convolutional layer, Conv)来提取特征,多个卷积层间采用最大池化(max-pooling)来减小数据维度,然后采用数据压平(flatten)操作将多个通道的特征值重新组合为一维,最后采用多个全连接层(fully connected layer, FC)进行分类。网络采用线性整流函数(rectified linear unit, ReLU)作为激活函数,此外为了提高网络的泛化能力,在训练时,全连接层间采用随机失活(dropout)操作进行参数随机舍弃,从而避免出现过拟合现象。

图 2 展示了目前卷积神经网络在提取时域 IQ 信号特征时所采用的 3 种操作方式,其中(a)是文献[27]中基准卷积网络采用的方式,(b)是文献[28-29]中卷积网络采用的方式,(c)是文献[30-31]中卷积网络采用的方式。(a)先采用一维时域滤波器提取时域特征,然后再采用最大池化的方式在 IQ 方向进行降维,这种在  $I$  与  $Q$  中取最大值的操作会损失幅度以及相位信息;

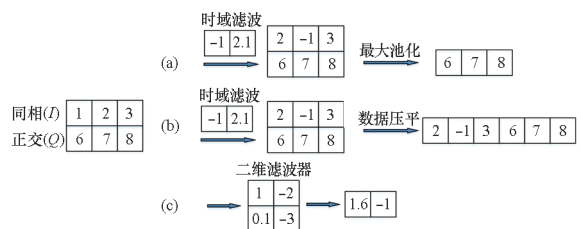


图 2 卷积神经网络的 3 种提取特征方式示意

Fig. 2 3 feature extraction methods of CNN

(b) 先采用一维时域滤波器提取时域特征, 然后采用 Flatten 的方式将数据展开, 再进行分类, 这种方式是在  $I$  与  $Q$  中提取相同的时域特征; (c) 是直接采用二维滤波器, 一次性直接提取时域与 IQ 相关特征, 但是, 这种操作只能在特征提取的最后一层进行, 难以多层连续展开进行。

在分类判别阶段, 如图 3 中所示, 网络<sup>[27-31]</sup>都采用多个全连接层, 虽然可以提高网络的分类能力, 但存在网络参数众多和计算量大的问题。

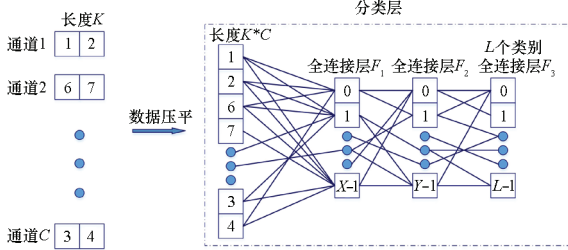


图 3 卷积神经网络分类层示意  
Fig. 3 CNN classification layer

## 2 IQ 卷积网络结构设计

### 2.1 IQCNet 结构

如图 4 所示, 输入数据是时间长度为  $N$  的 IQ 序列, 输出为分类类别数量  $L$ 。IQCNet 首先将输入数据扩展为  $C$  个通道 (卷积核), 然后在每个通道内进行 IQ 相关特征与时域特征的提取, 最后仅采用 1 个全连接层进行分类。

在特征提取层, 将特征提取分为 2 步, 分别在不同的维度上进行特征提取, 首先采用一维 IQ 滤波器提取 IQ 相关特征, 然后再使用多层一维时域滤波器提取时域特征, 由于充分利用了时序 IQ 信

号的 IQ 相关特征和时域特征, 信息利用率更高, 且由于第一个滤波器就完成了数据降维, 后续时域滤波的计算量更小。具体实现为: 在每个通道内部, 第 1 个卷积层采用 (1, 2) 的滤波器提取 IQ 相关特征, 输出数据维度由  $N \times 2$  变为了  $N \times 1$ , 后续处理的计算量降低了一半; 然后借鉴 VGG (visual geometry group) 网络<sup>[33]</sup>采用多层 (3, 1) 小滤波器提取时域特征, 在保证感受野的条件下减少卷积核参数数量; 接着, 每 2 个时域滤波器后采用最大池化降低数据维度, 减少时域方向计算量。

在分类阶段, 借鉴 ResNet 网络<sup>[34]</sup>采用的自适应平均池化 (adaptive average pooling) 替代 flatten, 首先采用自适应平均池化操作, 将每个特征通道的特征平均值作为新的特征值, 提高网络的泛化性能; 然后在分类时仅采用 1 个全连接层, 以降低参数数量与计算量。具体实现为: 首先使自适应平均池化的输出数量与通道数  $C$  一致, 然后与类别数量  $L$  形成 1 个输入为  $C$ 、输出为  $L$  的全连接层。

除此之外, 在卷积层的输出与激活函数之间, 添加批归一化 (Batch Normalization, BN)<sup>[35]</sup> 操作, 以增加模型的鲁棒性和训练速度, 并代替 dropout 防止过拟合; 网络中采用 ReLU 作为激活函数, 在分类层采用 Sigmoid 作为分类函数。

IQCNet 网络结构参数是可灵活配置的。如图 4 所示, IQCNet 网络由  $H$  个卷积层与 1 个全连接层构成, 其中每个卷积层包含  $C$  个特征提取通道。为表示不同结构参数的 IQCNet 网络, 本文用  $IQCNet(H, C, S)$  表示具体结构参数,  $H$  表示卷积层的层数,  $C$  表示每层卷积层的卷积核数量 (通道

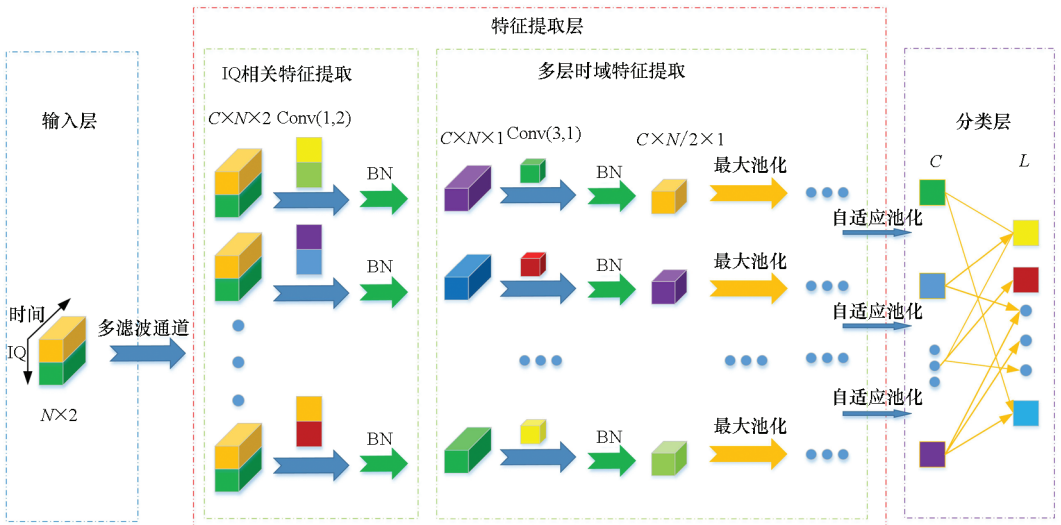


图 4 IQCNet 网络结构示意图  
Fig. 4 IQCNet network structure diagram

数),  $S$  表示时域卷积核大小。如 IQCNet(4, 32, 3) 表示具有 4 个卷积层, 且每层包含 32 个大小为 (3, 1) 时域卷积核的 IQCNet 网络, 时域卷积核的大小为 (3, 1)。表 1 是 IQCNet(4, 32, 3) 的网络结构, 其中数据输入格式为  $600 \times 2$ 。

表 1 IQCNet(4, 32, 3) 网络结构

Tab. 1 IQCNet(4, 32, 3) network structure

层名称	输入尺寸	尺寸	步进	卷积核数
Conv2d-1	$600 \times 2$	$1 \times 2$	$1 \times 1$	32
BatchNorm2d-1	$600 \times 1$			32
MaxPool2d-1	$600 \times 1$	$2 \times 1$	$2 \times 1$	32
Conv2d-2	$300 \times 1$	$1 \times 3$	$1 \times 1$	32
BatchNorm2d-2	$300 \times 1$			32
Conv2d-3	$300 \times 1$	$1 \times 3$	$1 \times 1$	32
BatchNorm2d-3	$300 \times 1$			32
MaxPool2d-2	$300 \times 1$	$2 \times 1$	$2 \times 1$	32
Conv2d-4	$150 \times 1$	$1 \times 3$	$1 \times 1$	32
BatchNorm2d-4	$150 \times 1$			32
AdaptiveAvgPool2d	$150 \times 1$			32
Linear-1	32			

## 2.2 对比网络结构

为对比验证 IQCNet 中提取 IQ 相关特征是否能有效提高识别准确率, 本文设计了除未提取 IQ 相关特征之外, 网络其他部分与 IQCNet 完全相同的网络, 并命名为 IQCNet-N( $H, C, S$ )。

将表 1 中的 IQCNet-N(4, 32, 3) 与表 2 中的 IQCNet(4, 32, 3) 对比发现, 两者的差别仅在于 IQCNet 第一个卷积层采用了 (1, 2) 的滤波器提取 IQ 相关特征, 而 IQCNet-N 未做此处理。

表 2 IQCNet-N(4, 32, 3) 网络结构

Tab. 2 IQCNet-N(4, 32, 3) network structure

层名称	输入尺寸	尺寸	步进	卷积核数
Conv2d-1	$600 \times 2$	$1 \times 1$	$1 \times 1$	32
BatchNorm2d-1	$600 \times 2$			32
MaxPool2d-1	$600 \times 2$	$2 \times 1$	$2 \times 1$	32
Conv2d-2	$300 \times 2$	$1 \times 3$	$1 \times 1$	32
BatchNorm2d-2	$300 \times 2$			32
Conv2d-3	$300 \times 2$	$1 \times 3$	$1 \times 1$	32
BatchNorm2d-3	$300 \times 2$			32
MaxPool2d-2	$300 \times 2$	$2 \times 1$	$2 \times 1$	32
Conv2d-4	$150 \times 2$	$1 \times 3$	$1 \times 1$	32
BatchNorm2d-4	$150 \times 2$			32
AdaptiveAvgPool2d	$150 \times 2$			32
Linear-1	32			

本文选取了网络深度、卷积核大小和通道数量不同的 CNN-DRID<sup>[28]</sup>、CNN-DLRF<sup>[29]</sup> 和 CNN-ORACLE<sup>[31]</sup> 3 种 CNN 网络结构与 IQCNet 进行实验对比, 表 3、表 4 和表 5 是以上三者的网络结构。

表 3 CNN-DRID 网络结构

Tab. 3 CNN-DRID network structure

层名称	输入尺寸	尺寸	步进	卷积核数
Conv2d-1	$600 \times 2$	$7 \times 1$	$1 \times 1$	128
Conv2d-2	$594 \times 2$	$5 \times 1$	$1 \times 1$	128
MaxPool2d-1	$590 \times 2$	$2 \times 1$	$2 \times 1$	128
Conv2d-3	$295 \times 2$	$7 \times 1$	$1 \times 1$	128
Conv2d-4	$289 \times 2$	$5 \times 1$	$1 \times 1$	128
MaxPool2d-2	$285 \times 2$	$2 \times 1$	$2 \times 1$	128
Conv2d-5	$142 \times 2$	$7 \times 1$	$1 \times 1$	128
Conv2d-6	$136 \times 2$	$5 \times 1$	$1 \times 1$	128
MaxPool2d-3	$132 \times 2$	$2 \times 1$	$2 \times 1$	128
Conv2d-7	$66 \times 2$	$7 \times 1$	$1 \times 1$	128
Conv2d-8	$60 \times 2$	$5 \times 1$	$1 \times 1$	128
MaxPool2d-3	$56 \times 2$	$2 \times 1$	$2 \times 1$	128
flatten	$128 \times 28 \times 2$			
Linear-1	7 168			
Linear-2	256			
Linear-3	128			

表 4 CNN-DLRF 网络结构

Tab. 4 CNN-DLRF network structure

层名称	输入尺寸	尺寸	步进	卷积核数
Conv2d-1	$600 \times 2$	$19 \times 1$	$1 \times 1$	128
MaxPool2d-1	$582 \times 2$	$2 \times 1$	$2 \times 1$	128
Conv2d-3	$291 \times 2$	$15 \times 1$	$1 \times 1$	32
MaxPool2d-2	$277 \times 2$	$2 \times 1$	$2 \times 1$	32
Conv2d-5	$138 \times 2$	$11 \times 1$	$1 \times 1$	16
MaxPool2d-3	$128 \times 2$	$2 \times 1$	$2 \times 1$	16
flatten	$16 \times 64 \times 2$			
Linear-1	2 048			
Linear-2	128			
Linear-3	16			

从选取的 3 种典型卷积网络结构可以看出, CNN-ORACLE 的特征提取层最少, 只有 2 层;

CNN-DLRF 的特征提取层居中,有 3 层;而 CNN-DRID 的特征提取层最多,共 8 层。

表 5 CNN-ORACLE 网络结构

Tab.5 CNN-ORACLE network structure

层名称	输入尺寸	尺寸	步进	卷积核数
Conv2d - 1	600 × 2	7 × 1	1 × 1	50
Conv2d - 2	594 × 2	7 × 2	1 × 1	50
flatten	50 × 588 × 1			
Linear - 1	29 400			
Linear - 2	256			
Linear - 3	80			

从卷积核大小角度观察,CNN-DRID 使用了 (19,1)的大卷积核,而其余二者使用了相对较小的(7,1)的卷积核。从卷积核数量上观察,CNN-DRID 的卷积核最多,每层都使用了 128 个,CNN-ORACLE 卷积核最少,只有 50 个,而 CNN-DLRF 卷积核数量居中,分别使用了 128、32 和 16 个。另外,3 者都使用了 3 个全连接层进行分类。总体来看,CNN-DRID 的模型最大,特征提取能力最强,CNN-DLRF 模型居中,特征提取能力居中,而 CNN-ORACLE 模型最小,特征能力也最小。

### 3 实验验证

#### 3.1 数据集

本文采用 FIT/CorteXlab 射频指纹识别数据集进行算法验证。该数据集由 22 台 NI USRP N2932 生成,其中 21 台用作发射设备,1 台用作接收设备,通信频段为 433 MHz,接收机采样速率为 5 MSample/s,1 台接收机采集了 21 台发射设备的无线信号用于射频指纹识别<sup>[36]</sup>。

数据集中,共有 3 种发送数据类型与 2 种功率模式。

3 种发送数据类型分别为:①正交相移键控(quadrature phase shift keying, QPSK)调制固定数据序列,是经 QPSK 调制的固定数据,数据内容为 IEEE 802.15.4 前导头的位序列,以下称为固定包;②QPSK 调制随机数据序列,为 QPSK 调制的随机数据,以下称为随机包;③未调制噪声序列,为未经调制的随机噪声比特,以下称为噪声包。

2 类型功率模式为:普通模式,即所有发射机使用固定的功率,并且实验环境无干扰,此模式的传输信道最稳定。变幅度模式,也称为变功率模式,在发射设备位置不变的条件下,改变 IQ 信号

的幅值,用于模拟发射机功率变化,其功率变化范围为 20 dB。

由 3 种发送数据类型与 2 种功率模式组合成 6 种实验场景,分别为:固定功率发送固定包、固定功率发送随机包、固定功率发送噪声包、变功率发送固定包、变功率发送随机包和变功率发送噪声包。

实验控制 21 台发射设备分别周期性地发送信号,接收设备根据发送设备的编号记录并存储数据,发送周期为 600 个 IQ 数据点。

FIT/CorteXlab 数据集原始文件由带发射机标签和实验条件说明的 21 个同相信号与 21 个正交信号文件组成的,且 21 个发射机的数据样本数量完全相同<sup>[37]</sup>。本文首先将 42 个文件处理为 21 个带标签的  $M \times 2$  格式的时间连续 IQ 信号数据,然后再转换为  $(M/N) \times N \times 2$  的数据格式,文件格式化处理完毕后,每个场景下的数据包大小为 5.2 G,当时间长度  $N$  为 600 时,数据样本数为 1 091 175 个。

#### 3.2 评估方法

本文采用识别准确率对网络模型的泛化性能进行评估,对于分类问题,根据真实值和预测值的结果,可以将整个样本集分为真正例( $TP$ )、伪正例( $FP$ )、真反例( $TN$ )和伪反例( $FN$ ),识别准确率  $Acc$  定义为:

$$Acc = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

性能评估采用多次随机实验求平均值的方法获得测试结果,如图 5 所示,首先从数据集中随机抽取样本,并划分训练样本、验证样本与测试样本,使用随机参数初始化网络模型后进行模型训练、验证与测试,将  $N$  次测试结果的平均值作为最终的测试结果,本文所有实验都进行 3 次随机测试。

单次训练、验证与测试过程如图 6 所示,首先

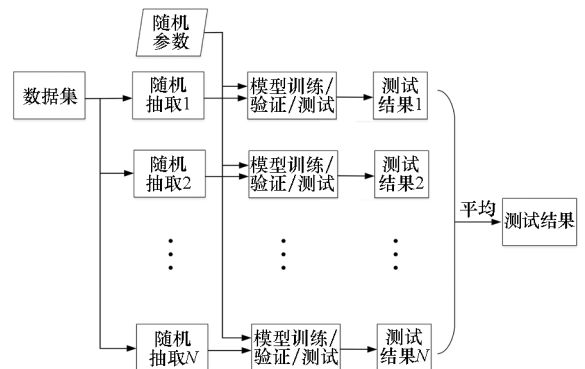


图 5 实验总体流程

Fig.5 Overall experiment process

采用无放回随机采样的方式,将数据集按 7 : 1 : 2 的比例生成训练样本、验证样本与测试样本,然后对训练样本进行训练,每训练 2 个轮次进行 1 次验证,选取在验证样本中获得最高识别准确率的参数作为最终模型参数,然后对测试样本进行测试,获得模型识别准确率。本文实验发现,训练 30 个轮次后 IQCNet 与对比网络损失函数的 loss 不再下降,因此确定后文每次训练执行 30 个轮次。

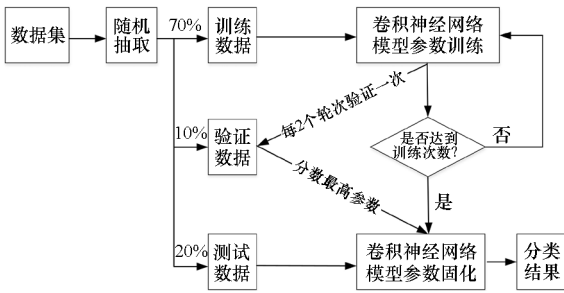


图 6 实验详细流程

Fig. 6 Detailed experiment process

### 3.3 实验条件

本文的实验条件:操作系统为 Ubuntu 18.04.4 LTS,内存为 64 GB,CPU 为 Intel i9 - 7920X CPU @ 2.90 GHz × 24,GPU 为 NVIDIA GeForce RTX 2080Ti,cuda 版本为 10.1,编程语言为 Python 3.7,深度学习框架为 Pytorch 1.4.0。

训练数据的批处理大小为 1 024,且每一批次的数数据会随机打乱;损失函数采用交叉熵,优化器采用 Adam 优化器,初始的学习率为 0.001,每 5 个轮次后衰减为原来的 0.8 倍,其余参数为 Pytorch 中的默认参数。

交叉熵损失函数定义如下:

$$C = -\frac{1}{N} \sum_x [y \ln \hat{y} + (1 - y) \ln (1 - \hat{y})] \quad (2)$$

式中, $x$  表示样本, $y$  表示实际的标签, $\hat{y}$  表示预测输出, $N$  表示样本总量。

在实验验证中,网络输入的时间长度  $N$  会采用 256,600,512,1 024,2 048 和 4 096 等数值,无特殊说明时,时间长度为 600,与发送周期一致。

文中涉及的参数数量、存储量以及计算量等数据,采用 Pytorch 的 torchstat 库计算获得。

### 3.4 识别准确率

相对于其他网络结构,IQCNet 的最大特点是将特征提取分为 IQ 相关特征与时域特征 2 步进行,为验证 IQ 相关特征是否有利于提高识别准确率,本文选取 256,512,600,1 024,2 048 和 4 096

等输入时间长度的数据对 IQCNet (8, 32, 3) 与 IQCNet-N (8, 32, 3) 进行比较验证。

如图 7 所示,在 6 种场景下,IQCNet 与 IQCNet-N 在不同输入长度下表现出来的趋势是一致的,即:功率稳定条件下,网络对发射机的识别率比较高,而当功率变化时,网络对发射机的识别准确率都发生下降;而在 3 种发送包类型中,发送固定包时发射机个体更容易被识别。

从识别准确率来看,在不同场景与不同输入长度下,IQCNet 的识别准确率都要比 IQCNet-N 要高,尤其是在变功率模式下。统计所有识别准确率平均值,IQCNet 与 IQCNet-N 分别为 90.39% 和 78.11%,前者比后者高 12.28%,证明 IQCNet 提取 IQ 相关特征确实有利于提高识别准确率。

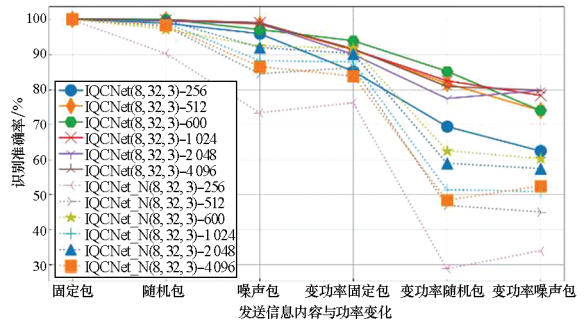


图 7 IQ 相关特征对识别准确率的影响

Fig. 7 Impact of IQ correlation features on recognition accuracy

图 8 展示了 IQCNet 网络中时域滤波器卷积核大小对识别准确率的影响。首先固定 IQCNet 网络的卷积深度与卷积核数量,然后比较时域滤波器中卷积核大小分别为 (3, 1)、(5, 1)、(7, 1) 与 (9, 1) 时的识别准确率。粗略来看,卷积核大小的变化对识别准确率影响不大,统计 6 种场景下的平均值,以上 4 者的平均准确率分别为 91.67%、91.69%、91.03% 和 91.11%,相差不超过 1%。但是,大卷积核会带来更多的模型参数与计算操作,以上模型的参数数量分别为 22 773、37 109、51 445 和 65 781,执行一次计算需要的浮点计算量分别为 3.52、5.58、7.49 和 9.27 的操作,所以在时域特征提取时采用 (3, 1) 的小卷积核是一种高性价比的选择。

图 9 展示了 IQCNet 网络中卷积层深度对识别准确率的影响。首先固定 IQCNet 网络中每层的卷积核数量与卷积核大小,然后比较网络深度分别为 4、6、8 和 10 时的识别准确率。粗略来看,增加卷积深度确实有利于提高识别准确率,统计 6 种场景下的平均值,以上 4 者的平均准确

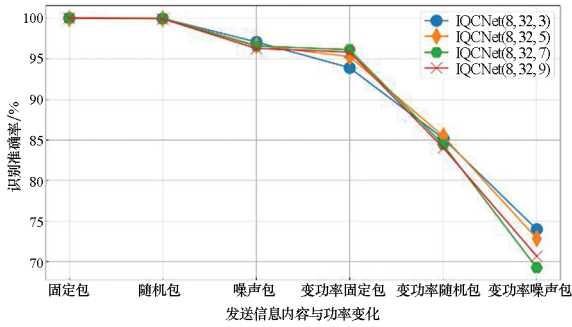


图 8 卷积核大小对识别准确率的影响

Fig. 8 Effect of convolution kernel size on recognition accuracy

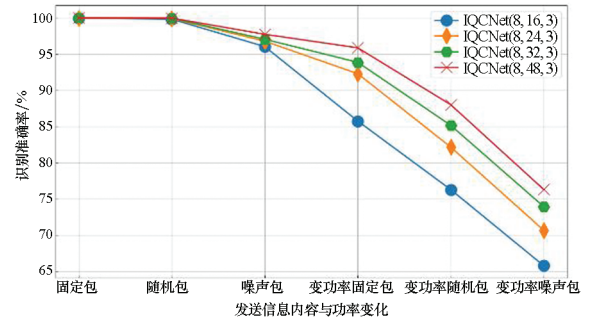


图 10 卷积核数量对识别准确率的影响

Fig. 10 Influence of the number of convolution kernels on the recognition accuracy

率分别为 82.50%、88.95%、91.67% 和 92.41%，增加卷积深度明显提高了识别准确率。以上模型的模型参数分别为：10 229、16 501、22 773、29 045，浮点计算量分别为 2.46、3.17、3.52、3.7 的操作，模型参数与计算量也随卷积层深度的增加而增加。

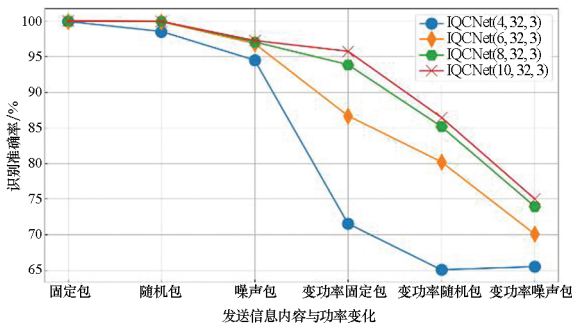


图 9 卷积层深度对识别准确率的影响

Fig. 9 Effect of convolutional layer depth on recognition accuracy

图 10 展示了 IQNet 网络中卷积核数量对识别准确率的影响。首先固定 IQNet 网络中卷积层深度与卷积核大小，然后比较卷积核数量分别为 16、24、32 和 48 时的识别准确率。粗略来看，增加卷积和数量有利于提高识别准确率，统计 6 种场景下的平均值，以上 4 者的平均准确率分别为 87.29%、90.31%、91.67% 和 92.99%，增加卷积数量明显提高了识别准确率。以上模型的模型参数分别为：6 021、13 053、22 773、50 277，浮点计算量分别为 0.926、2.02、3.52、7.9 的操作，模型参数与计算量也随卷积核数量的增加而增加。

图 11 展示了 IQNet 网络中不同算子对识别准确率的影响。

将 IQNet(8,32,3) 网络中的自适应平均池化层替换为自适应最大池化层的网络称为 IQNet\_AdpMaxP，在 6 种场景下其平均识别准确

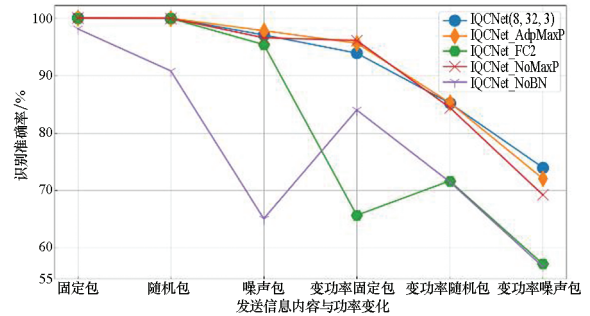


图 11 算子对识别准确率的影响

Fig. 11 Influence of operators on recognition accuracy

率为 91.78%，比 IQNet(8,32,3) 高 0.11%，考虑到众多随机因素的影响，这点差距微不足道。可认为，对 IQNet 网络而言，自适应最大池化与自适应平均池化是可替换使用的。

将 IQNet(8,32,3) 网络中的最大池化层去掉的网络称为 IQNet\_NoMaxP，在 6 种场景下其平均识别率为 91.04%，虽然其网络参数与 IQNet(8,32,3) 都为 22 773 个，但是浮点计算量为 13.25 是 IQNet(8,32,3) 的 3.76 倍，说明 IQNet 网络中的最大池化可以在不降低识别准确率的情况下大幅减少计算量。

在 IQNet(8,32,3) 网络的基础上，在分类阶段增加 2 个每层 32 个神经元的全连接层，称为 IQNet\_FC2，并且在训练时全连接层间增加随机失活率为 50% 的 dropout 防止过拟合，虽然其参数数量和浮点计算量分别为 24 885 与 3.52，与 IQNet(8,32,3) 相比增加不多，但在 6 种场景下其平均识别率仅为 81.62%，比 IQNet(8,32,3) 减少了整整 10%，说明在 IQNet 网络中仅使用 1 个全连接层是比较好的选择。

将 IQNet(8,32,3) 中的批归一化操作去除的网络称为 IQNet\_NoBN，在 100 个轮次的训练后，在 6 种场景其平均识别准确率仅为 77.68%，比 IQNet(8,32,3) 在 30 个轮次训练后的结果相

比相差了 14%。图 12 为在变功率固定包模型下,100 个轮次训练过程中 IQCNet(8,32,3) 与 IQCNet\_NoBN 分别在训练样本、验证样本与测试样本下的识别准确率,由图 12 可知,在整个训练过程中,批归一化都起到了加快收敛的作用,有利于提高模型的识别准确率。不过,批归一化会带来额外的计算量,在 100 个轮次的训练中,IQCNet(8,32,3) 与 IQCNet\_NoBN 的训练时间分别为 9 390 s 与 7 200 s,可认为批归一化操作增加了 30% 的计算时间。

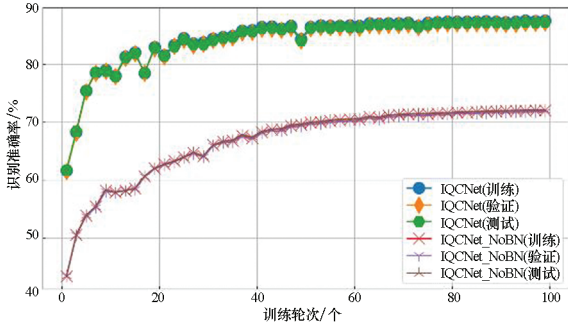


图 12 批归一化对识别准确率的影响  
Fig. 12 Effect of batch normalization on recognition accuracy

图 13 展示了 IQCNet(8,32,3) 与 CNN-DRID、CNN-DLRF、CNN-ORACLE 等对比网络在 6 种场景下的识别准确率,由此结果可知:

1) 识别准确率会受发射机发送数据内容的影响。当发送固定数据内容时,不管发射功率是否变化,发射机个体都更容易被识别,所以提取固定数据内容(例如前导头)更有利于识别发射机身份。

2) 功率变化会降低识别率,而且功率变化对固定包的影响相对较小,对随机包和噪声包影响更加明显。

3) IQCNet 的识别准确率最高。在 6 种场景下,CNN-DRID、CNN-DLRF、CNN-ORACLE 与 IQCNet(8,32,3) 的平均识别准确率分别为 82.61%、72.09%、61.71% 和 91.67%,IQCNet(8,32,3) 相对前 3 者的识别率高 9.06%、19.58% 和 29.96%。

### 3.5 参数与计算量

本文采用 Pytorch 的 torchstat 库计算网络结构的参数数量、存储量以及计算量,表 6 是 IQCNet(8,32,3)、IQCNet-N(8,32,3) 与 CNN-DRID、CNN-DLRF、CNN-ORACLE 参数数量、存储量以及计算量等信息。

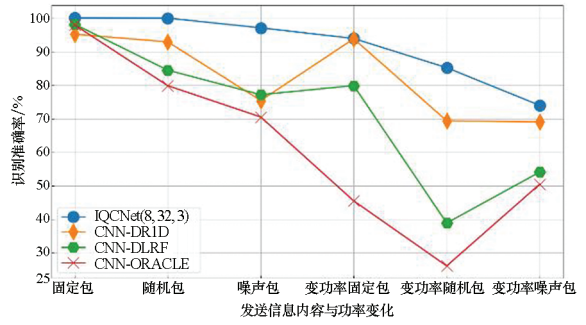


图 13 IQCNet 与对比网络识别准确率的比较  
Fig. 13 Recognition accuracy comparison between IQCNet and contrast network

表 6 网络模型参数及计算量

Tab. 6 Network model parameters and calculations

网络	参数数量	浮点计算量	内存读写量	训练时间
DRID	2 544 533	288.91	13.46	39 200
DLRF	334 373	38.75	2.28	10 924
ORACLE	7 584 367	28.63	29.62	9 095
IQCNet-N	<b>22 741</b>	7.01	2.00	4 759
IQCNet	22 773	<b>3.52</b>	<b>1.05</b>	<b>2 838</b>

在参数数量方面,CNN-DRID、CNN-DLRF、CNN-ORACLE 和 IQCNet-N(8,32,3) 的参数数量大约分别为 IQCNet(8,32,3) 的 111 倍、14 倍、333 倍和 1 倍,所以在参数数量方面 IQCNet(8,32,3) 有非常明显的优势。

在计算量操作方面,CNN-DRID、CNN-DLRF、CNN-ORACLE 和 IQCNet-N(8,32,3) 的浮点计算量大约分别为 IQCNet(8,32,3) 的 82 倍、11 倍、8 倍和 2 倍,所以 IQCNet(8,32,3) 在计算量方面也具有非常明显的优势。

在存储量方面,CNN-DRID、CNN-DLRF、CNN-ORACLE 和 IQCNet-N(8,32,3) 的内存读写使用量大约分别为 IQCNet(8,32,3) 的 13 倍、2 倍、28 倍和 2 倍,IQCNet(8,32,3) 在存储量方面优势同样明显。

在训练时间方面,使用 NVIDIA RTX2080Ti GPU 平台,在 6 个场景共 31.2 GB 数据训练 30 个轮次,CNN-DRID、CNN-DLRF、CNN-ORACLE 和 IQCNet-N(8,32,3) 的总时间分别为 IQCNet(8,32,3) 的 14 倍、4 倍、3 倍和 2 倍,IQCNet(8,32,3) 在计算时间方面具有明显的优势。

所以,IQCNet(8,32,3) 在参数数量、计算量和存储量以及计算时间等 4 个方面,相对于 CNN-DRID、CNN-DLRF、CNN-ORACLE 和 IQCNet-N(8,



32,3)都具有明显的优势。

## 4 结论

本文针对射频指纹识别的应用,对直接处理时序 IQ 信号的端到端卷积神经网络结构进行了研究。

首先,对当前处理时序 IQ 信号的几种典型卷积神经网络操作进行了分析,指出了其存在信息利用率低和计算量大等不足,进而提出了基于 IQ 相关特征的卷积神经网络结构:IQCNet,通过分步提取 IQ 域和时域等不同维度的特征、多层小滤波器组合和自适应平均池化等手段,以解决以上问题。

然后,本文在 FIT/CorteXlab 射频指纹识别数据集下,采用多次随机试验对网络的泛化性能进行评估。实验结果表明:在发送内容与功率变化的 6 种场景下,本文提出的 IQCNet 相较于未利用 IQ 相关特征的 IQCNet-N 平均识别准确率至少提高了 12%,而计算量、内存使用量和计算时间仅为后者的 50%,说明提取 IQ 相关特征既能提高识别准确率,又有利于提高计算效率。此外,本文还对 IQCNet 结构中网络深度、卷积核大小与数量,以及批归一化、最大池化层等算子的作用进行了验证,进一步证明了 IQCNet 采用的算子的有效性。

最后,采用 CNN-DRID、CNN-DLRF 和 CNN-ORACLE 等 3 种卷积网络结构与 IQCNet 进行比较,相对于前 3 者,IQCNet 的平均识别率总体提高近 20%,而仅使用了 7.5% 的计算量与 19% 的存储量,证明 IQCNet 网络识别准确率更高,而对存储与计算资源的要求更低,是一种高效的 IQ 卷积网络结构。

## 参考文献 (References)

- [1] 袁红林,胡爱群. 射频指纹的产生机理与惟(唯)一性[J]. 东南大学学报(自然科学版), 2009, 39(2): 230-233.  
YUAN H L, HU A Q. Fountainhead and uniqueness of RF fingerprint[J]. Journal of Southeast University (Natural Science Edition), 2009, 39(2): 230-233. (in Chinese)
- [2] 徐书华. 基于信号指纹的通信辐射源个体识别技术研究[D]. 武汉:华中科技大学, 2007.  
XU S H. On the identification technique of individual transmitter based on signalprints [D]. Wuhan: Huazhong University of Science and Technology, 2007. (in Chinese)
- [3] 赵蔚宸. 基于深度学习技术的信号指纹提取[D]. 北京:北京邮电大学, 2017.  
ZHAO Y C. Extraction signal fingerprint based on deep learning technology[D]. Beijing: Beijing University of Posts and Telecommunications, 2017. (in Chinese)
- [4] 严丹莉. 基于通信信号指纹的无人机个体识别[D]. 成都:电子科技大学, 2020.  
YAN D L. A drone classification method based on RF fingerprints[D]. Chengdu: University of Electronic Science and Technology of China, 2020. (in Chinese)
- [5] COBB W E, GARCIA E W, TEMPLE M A, et al. Physical layer identification of embedded devices using RF-DNA fingerprinting [C]//Proceedings of the 2010 Military Communications Conference, 2010: 2168-2173.
- [6] 王超. 基于 RF-DNA 的无线设备个体识别技术研究[D]. 哈尔滨:哈尔滨工程大学, 2019.  
WANG C. Research on individual identification technology of wireless devices based on RF-DNA fingerprinting [D]. Harbin: Harbin Engineering University, 2019. (in Chinese)
- [7] 陈翔,郝晓军,许雄,等. 基于时域 RF-DNA 的功率放大器射频指纹识别[J]. 太赫兹科学与电子信息学报, 2020, 18(1): 129-135.  
CHEN X, HAO X J, XU X, et al. RF fingerprinting extraction of power amplifier based on time-domain RF-DNA fingerprint[J]. Journal of Terahertz Science and Electronic Information Technology, 2020, 18(1): 129-135. (in Chinese)
- [8] DING G S, HUANG Z T, WANG X. Radio frequency fingerprint extraction based on singular values and singular vectors of time-frequency spectrum[C]//Proceedings of IEEE International Conference on Signal Processing, Communications and Computing, 2018: 1-6.
- [9] EZUMA M, ERDEN F, ANJINAPPA C K, et al. Micro-UAV detection and classification from RF fingerprints using machine learning techniques [C]//Proceedings of IEEE Aerospace Conference, 2019: 1-13.
- [10] BRIK V, BANERJEE S, GRUTESER M, et al. Wireless device identification with radiometric signatures [C]//Proceedings of the 14th ACM International Conference on Mobile Computing and Networking, 2008: 116-127.
- [11] KNOX D A, KUNZ T. Wireless fingerprints inside a wireless sensor network[J]. ACM Transactions on Sensor Networks, 2015, 11(2): 1-30.
- [12] ZHUO F, HUANG Y L, CHEN J. Radio frequency fingerprint extraction of radio emitter based on I/Q imbalance[J]. Procedia Computer Science, 2017, 107: 472-477.
- [13] DANEV B, CAPKUN S. Transient-based identification of wireless sensor nodes [C]//Proceedings of International Conference on Information Processing in Sensor Networks, 2009: 25-36.
- [14] PATEL H J, TEMPLE M A, BALDWIN R O. Improving ZigBee device network authentication using ensemble decision tree classifiers with radio frequency distinct native attribute fingerprinting[J]. IEEE Transactions on Reliability, 2015, 64(1): 221-233.
- [15] REISING D R, TEMPLE M A, JACKSON J A. Authorized and rogue device discrimination using dimensionally reduced RF-DNA fingerprints[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(6): 1180-1192.
- [16] KLEIN R W, TEMPLE M A, MENDENHALL M J. Application of wavelet denoising to improve OFDM-based signal detection and classification [J]. Security and Communication Networks, 2010, 3(1): 71-82.
- [17] BERTONCINI C, RUDD K, NOUSAIN B, et al. Wavelet

- fingerprinting of radio-frequency identification (RFID) tags[J]. *IEEE Transactions on Industrial Electronics*, 2012, 59(12): 4843–4850.
- [18] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition [C]//*Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016: 770–778.
- [19] SZEGEDY C, LIU W, JIA Y Q, et al. Going deeper with convolutions [C]//*Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015: 1–9.
- [20] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [C]//*Proceedings of Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017.
- [21] HANSEN J H L, JOGLEKAR A, SHEKHAR M C, et al. The 2019 inaugural fearless steps challenge: a giant leap for naturalistic audio [C]//*Proceedings of Interspeech 2019*, 2019.
- [22] SAHBA R, EBADI N, JAMSHIDI M, et al. Automatic text summarization using customizable fuzzy features and attention on the context and vocabulary [C]//*Proceedings of World Automation Congress*, 2018: 1–5.
- [23] EBADI N, LWOWSKI B, JALOLI M, et al. Implicit life event discovery from call transcripts using temporal input transformation network [C]//*Proceedings of IEEE Access*, 2019: 172178–172189.
- [24] YU J B, HU A Q, LI G Y, et al. A robust RF fingerprinting approach using multisampling convolutional neural network[J]. *IEEE Internet of Things Journal*, 2019, 6(4): 6786–6799.
- [25] HANNA S S, CABRIC D. Deep learning based transmitter identification using power amplifier nonlinearity [C]//*Proceedings of International Conference on Computing, Networking and Communications (ICNC)*, 2019: 674–680.
- [26] 周鑫, 何晓新, 郑昌文. 基于图像深度学习的无线电信号识别[J]. *通信学报*, 2019, 40(7): 114–125.  
ZHOU X, HE X X, ZHENG C W. Radio signal recognition based on image deep learning [J]. *Journal on Communications*, 2019, 40(7): 114–125. (in Chinese)
- [27] AL-SHAWABKA A, RESTUCCIA F, D'ORO S, et al. Exposing the fingerprint: dissecting the impact of the wireless channel on radio fingerprinting [C]//*Proceedings of IEEE Conference on Computer Communications*, 2020: 646–655.
- [28] RESTUCCIA F, D'ORO S, AL-SHAWABKA A, et al. DeepRadioID: real-time channel-resilient optimization of deep learning-based radio fingerprinting algorithms [C]//*Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2019: 51–60.
- [29] MERCHANT K, REVAY S, STANTCHEV G, et al. Deep learning for RF device fingerprinting in cognitive communication networks[J]. *IEEE Journal of Selected Topics in Signal Processing*, 2018, 12(1): 160–167.
- [30] RIYAZ S, SANKHE K, IOANNIDIS S, et al. Deep learning convolutional neural networks for radio identification [J]. *IEEE Communications Magazine*, 2018, 56(9): 146–152.
- [31] SANKHE K, BELGIOVINE M, ZHOU F, et al. ORACLE: optimized radio classification through convolutional neural networks [C]//*Proceedings of IEEE Conference on Computer Communications*, 2019: 370–378.
- [32] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [C]//*Advances in Neural Information Processing Systems 25 (NIPS 2012)*, 2012: 1097–1105.
- [33] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[EB/OL]. (2014–09–04) [2020–10–18]. <https://arxiv.org/abs/1409.1556v6>.
- [34] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition [C]//*Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016: 770–778.
- [35] IOFFE S, SZEGEDY C. Batch normalization: accelerating deep network training by reducing internal covariate shift[EB/OL]. (2015–02–11) [2020–10–18]. <https://arxiv.org/abs/1502.03167>.
- [36] MORIN C, CARDOSO L S, HOYDIS J, et al. Transmitter classification with supervised deep learning[EB/OL]. (2019–05–20) [2020–10–18]. <https://arxiv.org/abs/1905.07923>.
- [37] CorteXlab Wiki. Transmitter identification[EB/OL]. (2019–07–03) [2020–10–18]. <https://wiki.cortexlab.fr/doku.php?id=tx-id>.