SMoRFFI: A Large-Scale Same-Model 2.4 GHz Wi-Fi Dataset and Reproducible Framework for RF Fingerprinting

Zewei Guo[a], Zhen Jia[b], Jinxiao Zhu[c], Wenhao Huang[b] and Yin Chen[d,b,*]

[a]Future University Hakodate, Hakodate, 041-8655, Hokkaido, Japan
[b]Keio University, Fujisawa, 108-8345, Kanagawa, Japan
[c]Tokyo Denki University, Tokyo, 120-8551, Tokyo, Japan
[d]Reitaku University, Kashiwa, 277-8686, Chiba, Japan

ARTICLE INFO

Keywords:
Radio Frequency Fingerprinting
Physical Layer Identification
Internet of Things Devices
Software-Defined Radios

ABSTRACT

Radio frequency (RF) fingerprinting exploits hardware imperfections for device identification, but distinguishing between same-model devices remains challenging due to their minimal hardware variations. Existing datasets for RF fingerprinting are constrained by small device scales and heterogeneous models, which hinder robust training and fair evaluation of machine learning methods. To address this gap, we introduce a large-scale dataset of same-model devices along with an open-source experimental framework. The dataset is built using 123 same-model commercial IEEE 802.11g devices, which contain over 35.28 million raw I/Q samples from the preambles and corresponding 1.72 million RF features. The accompanying framework further provides a fully reproducible pipeline from data collection to performance evaluation. Within this framework, a Random Forest–based algorithm is implemented as a baseline to achieve 94.54% identification accuracy on this dataset.

Specifications table

Subject	Wireless communication
Specific subject area	Radio fingerprinting techniques based on machine learning algorithms.
Type of data	CSV files.
How data were acquired	This dataset was acquired using 1 USRP B210 receiver operated via GNU Radio, along with 123 M5Stack Core2 transmitters, 1 Huawei WS7100 V2 Wi-Fi access point, and 1 Minisforum NPB7 computer.
Data format	I/Q samples of preamble and corresponding radio frequency (RF) features. Specifically, each recorded frame mainly contains the following elements: (1) MAC address of the device, (2) Raw samples of preamble, (3) I/Q samples of long preamble, (4) Coarse Carrier frequency offset (CFO), (5) Fine CFO, (6) Combined CFO, (7) Phase error, (8) Magnitude error, (9) I/Q gain imbalance, (10) Fractal dimension.
Parameters for data collection	Up to 1,000 signal frames were collected from each transmitter. All transmitters and the USRP B210 receiver were set to work under the IEEE 802.11g standard with 20 MHz bandwidth. The receiver used the IEEE 802.11 a/g/p transceiver model [1], and the sampling rate was set as 20 MS/s.
Intended scope of dataset	This dataset was collected in a single controlled environment. The intended scope of the dataset is controlled benchmarking for device-feature-based RF fingerprinting methods in an indoor, static, short-distance, high-SNR scenario. It is not intended for deployment-ready or cross-environment evaluation, but rather for isolating and analyzing device-intrinsic characteristics under controlled conditions.

(continued on next page)

*Corresponding author
zwguo1996@gmail.com (Z. Guo); jiazhen0628@outlook.com (Z. Jia); jyuu@mail.dendai.ac.jp (J. Zhu); gerhua@sfc.keio.ac.jp (W. Huang); ychen@reitaku-u.ac.jp (Y. Chen)
ORCID(s):

(continued)

Description of data collection	To extract raw I/Q samples of the preamble, customized modifications were conducted on the IEEE 802.11 a/g/p transceiver model [1]. Specifically, the "WiFi Sync Short", "WiFi Sync Long", and "WiFi Frame Equalizer" blocks were rewritten, and the modified "RFtap Encapsulation" block [2] was integrated. These customized models were then used to receive Wi-Fi signals and record the raw preamble samples.
Data source location	Institution: Reitaku University City: Kashiwa, Chiba Country: Japan
Data accessibility	Repository name: RFFI-IQ_only-wifi-802.11g-2.4G-123-m5stack Direct URL to data: https://www.kaggle.com/datasets/yinchen1986/rffi-123-m5stack-iq-wifi-802-11g-2-4g Repository name: RFFI-kf_feature-IQ-wifi-802.11g-2.4G-123-m5stack Direct URL to data: https://www.kaggle.com/datasets/yinchen1986/rffi-kf-feature-iq-wifi-802-11g-2-4g-123-m5stack
Related code accessibility	Repository name: Dockerized-wifi-iq-preamble-capture Direct URL to code: https://github.com/aiot-lab-yin/Dockerized-wifi-iq-preamble-capture Repository name: RFF_Data_Calculate Direct URL to code: https://www.kaggle.com/code/zeweiguo/rff-data-calculate
Related research article(s)	Author's name: Zhen Jia, Zewei Guo, Wenhao Huang, Yin Chen, Jinxiao Zhu and Xiaohong Jiang Title: An Experimental Study on Radio Frequency Fingerprinting-based Authentication in IEEE 802.11g Journal: IEICE Tech. Rep., vol. 124, no. 122, BioX2024-70, pp. 344-349, Jul. 2024

Notation: Lower-case and lower-case bold-face letters represent scalars and vectors (e.g., a and $\mathbf{a}$), respectively. Let $||\mathbf{a}||$ and $\mathbf{a}^\dagger$ denote the Euclidean norm and conjugate transpose of vector $\mathbf{a}$, respectively. Denote $\Re\{c\}$, $\Im\{c\}$, $|c|$, and $\angle(c)$ as the real part, the imaginary part, the magnitude value, and the phase of the complex variable c.

1. Value of the Data

1.1. Why are these data useful?

RF fingerprinting has recently attracted increasing attention from both academia and industry [3, 4]. Recently, several datasets for RF fingerprinting have been published [5, 6, 7, 8, 9], as summarized in Table 1. While these datasets have enabled important progress in physical-layer identification under various scenarios, they differ from our dataset in several key aspects, including device heterogeneity, environmental settings, feature availability, and reproducibility. In particular, several needs remain insufficiently addressed: (1) a large-scale dataset of same-model devices is still lacking, which is essential for evaluating identification methods in the challenging same-model setting; (2) a dataset that provides pre-extracted RF features is highly desirable to facilitate the training and validation of lightweight, feature-based physical-layer identification methods; and (3) a fully open-source dataset with publicly available data-acquisition programs is needed to improve reproducibility and to enable more researchers to collect, extend, and release their own datasets.

To address these limitations, this paper presents a large-scale dataset collected from same-model devices, extracts RF features from the signal preambles, and releases a complete open-source experimental framework. The contributions of this paper are as follows:

- We provide a dataset containing over 35.28 million raw I/Q samples from 123 same-model devices, which represents a worst-case scenario for RF fingerprinting and serves as a more stringent and informative benchmark for future studies.
- We provide theoretical modeling of RF features, including CFO, magnitude error, phase error, I/Q gain imbalance, and fractal dimension, and then compute the corresponding RF features for each sample, enabling both signal-level and feature-level analysis.
- We introduce a random-forest-based RF fingerprinting algorithm as a baseline for the dataset and further provide a fully reproducible pipeline from data collection to performance evaluation.

1.2. Who can benefit from these data?

Developers who work on RF fingerprinting methods will find this dataset particularly valuable. In addition, data

Table 1
Comparison with Related Datasets

Dataset	Number of Devices	Type of Signals	Same-model Focus	Environmental Setting	Availability of Raw I/Q	Availability of Features	Reproducibility (Open-source Program)
Ref. [5]	25	LoRa	✓	Diverse	✓	✗	✗
Ref. [6]	27	Bluetooth	✗	Controlled	✓	✗	✗
Ref. [7]	10	Bluetooth, WiFi	✗	Controlled	✓	✗	✗
Ref. [8]	20	WiFi	✗	Diverse	✓	✗	✗
Ref. [9]	174	WiFi	✗	Controlled	✓	✗	✗
This Dataset	123	WiFi	✓	Controlled	✓	✓	✓

scientists who focus on the exploration and analysis of RF features can use this dataset for their research purposes.

1.3. How can these data be used for further insights and development of experiments?

These data can support further research and experimental development in several ways:

- The dataset serves as a large-scale benchmark that allows RF feature-based algorithm developers to train and evaluate machine learning and deep learning models.
- It provides extensive data for researchers to verify the effectiveness of newly proposed features.
- Preliminary analyses of the extracted RF features are also included, which help researchers compare their new features with those contained in this dataset.
- A complete experimental framework is provided to support reproducible research, which covers the entire process from data collection to performance evaluation. It allows researchers to develop and extend their own experiments based on this dataset.

1.4. What is the additional value of these data?

In addition to the I/Q samples and the corresponding RF features, this paper proposes a baseline physical layer identification algorithm based on the Random Forest model and provides related performance evaluations. This baseline allows developers to compare the performance of their RF feature-based algorithms with a standardized reference.

2. Dataset

2.1. Data

This paper provides two types of datasets. Each dataset contains 123 CSV files, and each file includes up to 1,000 records collected from a single IoT device. The first dataset (named "RFFI-IQ_only-wifi-802.11g-2.4G-123-m5stack") includes only the MAC address and the preamble samples from each device. The second dataset (named "RFFI-kf_feature-IQ-wifi-802.11g-2.4G-123-m5stack") provides multi-dimensional RF features, where each record consists of eleven categories of information described as follows:

(1) **Device number (device_number)** serves as the unique label for each IoT device and is used to distinguish among different devices.

(2) **MAC address (mac_address)** is the MAC address of each device.

(3) **Raw samples of the preamble (preamble)** refer to the preamble data that have not undergone any processing steps. The structure of the preamble is defined in the IEEE 802.11g standard, and its location within the signal frame is shown in Figure 1. As shown in Figure 1, the preamble consists of a short training sequence (STS), a guard interval (GI2), and a long training sequence (LTS). Specifically, the STS consists of 10 identical training sequences, and each sequence has a length of 16 samples (0.8 μs at 20 MS/s). These sequences are used by the receiver for packet detection and coarse CFO correction. LTS includes two identical training sequences (i.e., L1 and L2), where each sequence consists of 64 samples. These sequences are used for fine carrier frequency offset (CFO) correction. GI2 is a cyclic prefix sequence consisting of 32 samples, which is used to prevent interference between the STS and LTS. To prevent STS truncation caused by erroneous packet detection, the first two training sequences of STS are dropped in this dataset. Consequently, the resulting preamble sequence $\mathbf{x}$ can be expressed as

$$\mathbf{x} = [\underbrace{x_1, \ldots, x_{128}}_{\text{STS}}, \ldots, \underbrace{x_{161}, \ldots, x_{288}}_{\text{LTS}}]. \tag{1}$$

(4) **Coarse CFO (short_freq)** is the CFO estimated using STS, which can be expressed as [11]

$$\alpha_S = \frac{1}{16} \angle \left(\sum_{i=1}^{112} \mathbf{x}[i]^{\dagger} \mathbf{x}[i+16] \right). \tag{2}$$

After obtaining α_S, the I/Q samples of LTS are corrected as

$$\hat{\mathbf{x}}[j] = \mathbf{x}[160+j] e^{-\jmath(j-1)\alpha_S}, j = 1, 2, \cdots, 128, \tag{3}$$

where $\jmath = \sqrt{-1}$.

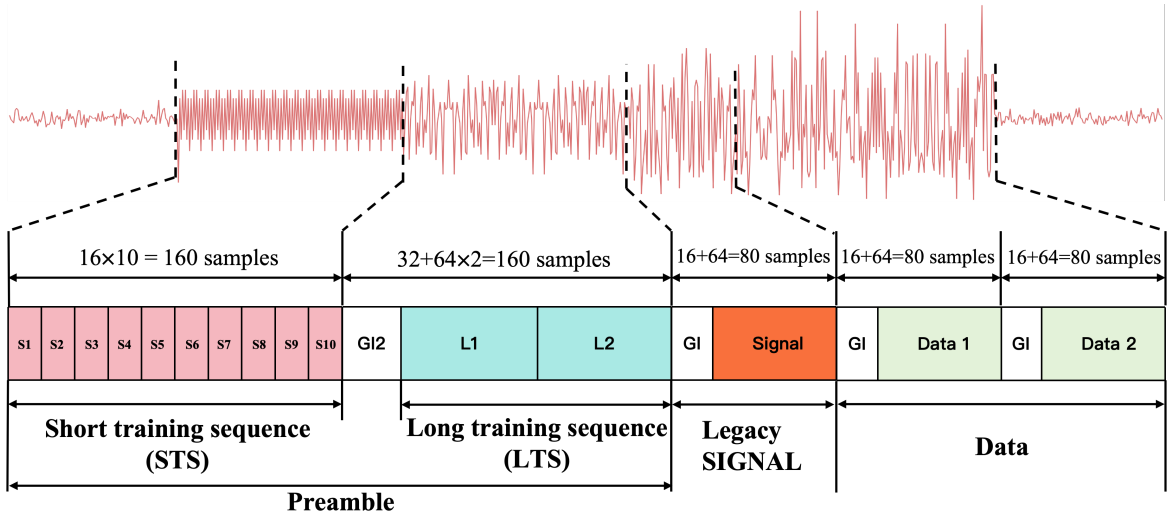


Figure 1: Structure of the IEEE 802.11g frame [10].

(5) **Fine CFO (long_freq)** is the CFO estimated from LTS after coarse CFO correction, which can be calculated as [11]

$$\alpha_L = \frac{1}{64}\angle\left(\sum_{i=1}^{64}\hat{\mathbf{x}}[i]^{\dagger}\hat{\mathbf{x}}[i+64]\right). \tag{4}$$

Then, the fine CFO correction can be expressed as

$$\check{\mathbf{x}}[j] = \hat{\mathbf{x}}[j]e^{-\jmath(j-1)\alpha_L}, j = 1, 2, \cdots 128. \tag{5}$$

(6) **Combined CFO (CFO)** denotes the total CFO of the signal frame. It is applied to compensate for the CFO of the remaining signal after the LTS. The combined CFO can be expressed as [11]

$$\alpha_F = \alpha_S + \alpha_L. \tag{6}$$

The coarse and fine offsets are stored in radians per sample. The combined CFO is stored in hertz as CFO $= (\alpha_S + \alpha_L) f_s/(2\pi)$, where $f_s = 20\,\mathrm{MS/s}$.

(7) **Equalized LTS (iq_preamble)** denotes the LTS samples after frequency offset correction, Fourier transform, and sub-carrier equalization [12], given by

$$\mathbf{l}[j] = \frac{\check{\mathbf{x}}_{\mathrm{FFT}}[j]}{\mathbf{h}[1 + ((j-1) \bmod 64)]}, \quad j = 1, 2, \ldots, 128. \tag{7}$$

Here, $\check{\mathbf{x}}_{\mathrm{FFT}}$ is the concatenation of the two shifted 64-point FFT outputs. At positions where the ideal LTS symbol is $+1$ or -1, the channel gain is estimated as

$$\mathbf{h}[k] = \frac{\check{\mathbf{x}}_{\mathrm{FFT}}[k] + \check{\mathbf{x}}_{\mathrm{FFT}}[64+k]}{2\,\mathbf{l}_{\mathrm{ideal}}[k]}, \tag{8}$$

where $\mathbf{l}_{\mathrm{ideal}}$ is the ideal LTS symbol defined in [10, Table L-5] and can be expressed as

$$\begin{aligned}\mathbf{l}_{\mathrm{ideal}} = [&0,0,0,0,0,0,1,1,-1,-1,1,1,-1,1,-1,1,1,\\ &1,1,1,1,-1,-1,1,1,-1,1,-1,1,1,1,1,0,1,\\ &-1,-1,1,1,-1,1,-1,1,-1,-1,-1,-1,-1,\\ &1,1,-1,-1,1,-1,1,-1,1,1,1,1,0,0,0,0,0].\end{aligned} \tag{9}$$

Finally, the two identical training sequences contained in the long preamble, denoted as L1 and L2, are represented by $\mathbf{l}_1$ and $\mathbf{l}_2$, respectively.

(8) **Phase error vector (phase_error_v1 and phase_error_v2)** denotes the angle between the measured and ideal phasors, which can be calculated as [13]

$$\mathbf{p}_n[k] = \arg\left(\mathbf{l}_n[k]\,\mathbf{l}_{\mathrm{ideal}}[k]^*\right), \quad n \in \{1,2\}, \tag{10}$$

where k indexes the active subcarriers and * denotes complex conjugation. The mean and variance of the phase error vector are further computed and recorded in the dataset as phase_error_mean and phase_error_var, respectively.

(9) **Magnitude error vector (mag_error_v1 and mag_error_v2)** is the difference in the magnitudes of the ideal and measured phasors, which can be expressed as [13]

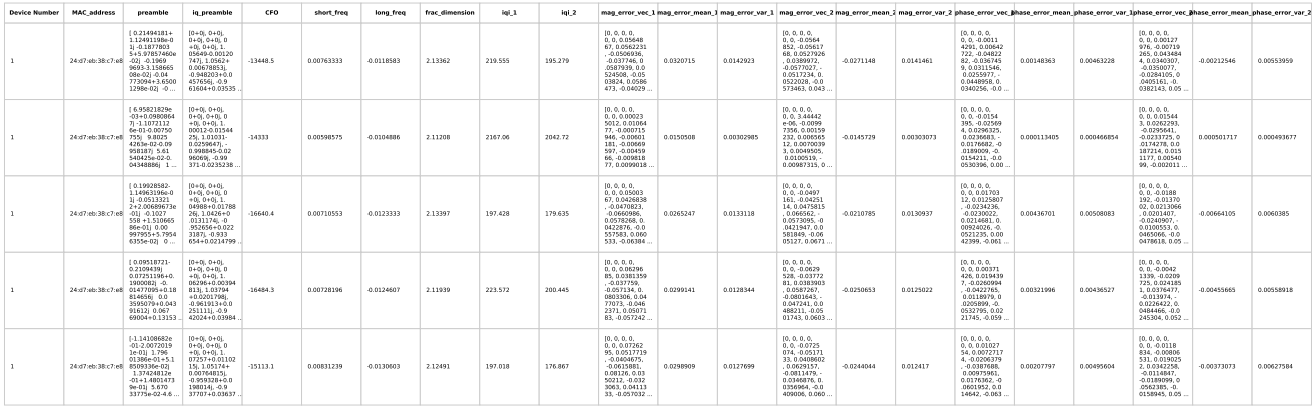
$$\mathbf{m}_n[k] = |\mathbf{l}_n[k]| - |\mathbf{l}_{\mathrm{ideal}}[k]|, n \in \{1,2\}, k = 1, 2, \cdots 64. \tag{11}$$

The mean and variance of each magnitude error vector are also computed and recorded in the dataset as `mag_error_mean` and `mag_error_var`, respectively.

(10) **I/Q gain imbalance (iqi_1 and iqi_2)** refers to the gain imbalance in the in-phase (I) and quadrature (Q) components of a signal, which can be expressed as [14]

$$\mathrm{IQI}_n = \frac{\|\Re\{\mathbf{l}_n\}\|^2}{\|\Im\{\mathbf{l}_n\}\|^2}, n \in \{1,2\}. \tag{12}$$

(11) **Fractal dimension (frac_dimension)** quantifies the complexity and self-similarity of the I/Q signal distribution. It can serve as a statistical measure of the geometric structure in the sampled data. Following the procedure in [15], the fractal dimension is computed directly on the raw time-domain preamble $\mathbf{x}$ of 288 complex samples, without frequency offset correction or channel equalization. Consequently, the corresponding fractal dimension D is estimated



Device Number	MAC_address	preamble	iq_preamble	CFO	short_freq	long_freq	frac_dimension	iqi_1	iqi_2	mag_error_vec_1	mag_error_mean_1	mag_error_var_1	mag_error_vec_2	mag_error_mean_2	mag_error_var_2	phase_error_vec_1	phase_error_mean_1	phase_error_var_1	phase_error_vec_2	phase_error_mean_2	phase_error_var_2
1	24:d7:eb:38:c7:e8	[0.21494181+ 1.12491198e-0 1j -0.1877803 5+5.97857466e -01j -0.1969 9693-3.158665 08e-01j -0.04 773094+3.6500 1298e-02j -0 ...	[0+0j, 0+0j, 0+0j, 0+0j, 0+ =0j, 0+0j, 1. 05649-0.00120 743j, 1.0562+ 0.00678853j, -0.948203+0.0 457656j, -0.9 61604+0.03535 ...	-13448.5	0.00763333	-0.0118583	2.13362	219.555	195.279	[0, 0, 0, 0, 0, 0.05648 67, 0.0562231 , -0.0506936, -0.037746, 0 .0587939, 0.0 524508, -0.05 03824, 0.0586 473, -0.04029 ...	0.0320715	0.0142923	[0, 0, 0, 0, 0, 0, -0.0564 892, -0.05617 68, 0.0527926 , 0.0389972, -0.0577027, - 0.0517234, 0. 0522028, -0.0 573463, 0.043 ...	-0.0271148	0.0141461	[0, 0, 0, 0, 0, 0, -0.0011 4291, 0.00642 722, -0.04822 82, -0.036745 9, 0.0331546, 0.0255977, - 0.0448958, 0. 0340256, -0.0 ...	0.00148363	0.00463228	[0, 0, 0, 0, 0, 0, 0.01544 3, 0.0262293, -0.0295641, -0.0233725, 0 .0174278, 0.0 187214, -0.015 1177, 0.00540 99, -0.002011 ...	-0.00212546	0.00553959
1	24:d7:eb:38:c7:e8	[6.958212829e -03+0.0980864 7j -1.1072112 e-01j -0.10272 755j, 9.8025 4263e-02-0.09 958187j 5.61 540425e-02-0. 04348886j 1 ...	[0+0j, 0+0j, 0+0j, 0+0j, 0+ =0j, 0+0j, 1. 00012-0.01544 25j, 1.01031- 0.0259647j, - 0.998845-0.02 960969j, -0.99 371-0.0235238 ...	-14333	0.00598575	-0.0104886	2.11208	2167.06	2042.72	[0, 0, 0, 0, 0, 0, 0.00023 5012, 0.01064 77, -0.0007715 946, -0.00601 181, -0.00569 597, -0.00459 66, -0.009818 77, 0.0099018 ...	0.0150508	0.00302985	[0, 0, 0, 0, 0, 0, 3.44442 e-06, -0.0099 7356, 0.00159 232, 0.006565 12, 0.0070039 3, 0.0049505, 0.0100519, - 0.00987315, 0 ...	-0.0145729	0.00303073	[0, 0, 0, 0, 0, 0, -0.0154 395, -0.02569 4, 0.0296325, -0.0236683, - 0.0176692, -0 .0189009, -0 .0154211, -0.0 0530396, 0.00 ...	0.000113405	0.000466854	[0, 0, 0, 0, 0, 0, 0.01544 3, 0.0262293, -0.0295641, -0.0233725, 0 .0174278, 0.0 187214, -0.015 1177, 0.00540 99, -0.002011 ...	0.000501717	0.000493677
1	24:d7:eb:38:c7:e8	[0.19928582- 1.149631996e-0 1j -0.0531321 2+2.00689673e -01j -0.1027 558 +1.510665 86e-01j 0.00 997955+5.7954 6355e-02j 0 ...	[0+0j, 0+0j, 0+0j, 0+0j, 0+ =0j, 0+0j, 1. 04988+0.01788 26j, 1.0426+0 .0131174j, -0 .952656+0.022 3187j, -0.933 654+0.0214799 ...	-16640.4	0.00710553	-0.0123333	2.13397	197.428	179.635	[0, 0, 0, 0, 0, 0, 0.05003 67, 0.0426838 , -0.070823, -0.0660986, - 0.0578268, 0. 0422876, -0.0 557583, 0.060 533, -0.06384 ...	0.0265247	0.0133118	[0, 0, 0, 0, 0, 0, -0.0497 161, -0.04251 14, 0.0475815 , 0.066562, - 0.0573095, -0 .0413947, 0.0 581849, -0.06 05127, 0.0671 ...	-0.0210785	0.0130937	[0, 0, 0, 0, 0, 0, 0.01703 12, 0.0125807 , -0.0234236, -0.0230222, - 0.0214681, 0. 00924026, -0. 0521235, 0.00 42399, -0.001 ...	0.00436701	0.00508083	[0, 0, 0, 0, 0, 0, -0.0188 192, -0.01370 02, 0.0213066 , 0.0201407, - 0.0240907, - 0.0100553, 0. 0465066, -0.0 0478618, 0.05 ...	-0.00664105	0.0060385
1	24:d7:eb:38:c7:e8	[0.09518721- 0.2109419j 0.0725119 6+0.19900862j -0. 01477095+0.18 814656j 0.0 3595079+0.043 91612j 0.067 69004+0.13153 ...	[0+0j, 0+0j, 0+0j, 0+0j, 0+ =0j, 0+0j, 1. 08296+0.00394 813j, 1.03794 +0.0201798j, -0.961913+0.0 251111j, -0.9 42024+0.03984 ...	-16484.3	0.00728196	-0.0124607	2.11939	223.572	200.445	[0, 0, 0, 0, 0, 0, 0.08296 85, 0.0381359 , -0.037759, -0.057134, 0. 0803306, 0.04 77073, -0.046 2371, 0.05071 83, -0.057242 ...	0.0299141	0.0128344	[0, 0, 0, 0, 0, 0, -0.0629 528, -0.03772 81, 0.0383903 , 0.0587267, - 0.0801643, - 0.0472141, 0.0 4882111, -0.05 01743, 0.0603 ...	-0.0250653	0.0125022	[0, 0, 0, 0, 0, 0, 0.00371 426, 0.019439 7, -0.0260994 , -0.0422765, - 0.0118979, 0 .0205899, -0. 0532795, 0.02 21745, -0.059 ...	0.00321996	0.00436527	[0, 0, 0, 0, 0, 0, -0.0042 1339, -0.0209 725, 0.0214185 1, 0.0376477, - 0.0113974, - 0.0226422, 0. 0484466, -0. 0 245304, 0.052 ...	-0.00455665	0.00558918
1	24:d7:eb:38:c7:e8	[-1.14108682e -01-2.0072019 1e-01j 1.796 01386e-01+5.1 8599336e-02j 1 .37424812e -01+1.4801473 9e-01j 5.670 33779e-02-4.6 ...	[0+0j, 0+0j, 0+0j, 0+0j, 0+ =0j, 0+0j, 1. 07257+0.01102 15j, 1.05174+ 0.00764815j, -0.959328+0.0 198014j, -0.9 37707+0.03637 ...	-15113.1	0.00831239	-0.0130603	2.12491	197.018	176.867	[0, 0, 0, 0, 0, 0, 0.07262 95, 0.0517719 , -0.0404675, -0.0615881, - 0.08126, 0.03 50212, -0.032 3063, 0.04113 33, -0.057032 ...	0.0298909	0.0127699	[0, 0, 0, 0, 0, 0, -0.0725 074, -0.05171 33, 0.0408602 , 0.0629157, - 0.0811479, - 0.0346876, 0. 0356964, -0.0 409606, 0.060 ...	-0.0244044	0.012417	[0, 0, 0, 0, 0, 0, 0.01027 54, 0.0072717 4, -0.0206379 , -0.0387688, - 0.0097596 1, 0.0176362, -0 .0601952, 0.0 14642, -0.063 ...	0.00207797	0.00495604	[0, 0, 0, 0, 0, 0, -0.0118 834, -0.00806 531, 0.0190025 2, 0.0342258, - 0.0114847, - 0.0189099, 0 .0562385, -0. 0158945, 0.05 ...	-0.00373073	0.00627584

Figure 2: Structure of the file 24:d7:eb:38:c7:e8_pre.csv.

as

$$D = \frac{1}{2} \times \left(3 - \frac{\ln \sum_{m=1}^{\lfloor 288/\tau \rfloor - 1} d_{\tau,m} - \ln \sum_{l=1}^{287} d_{1,l}}{\ln \tau} \right), \quad (13)$$

where

$$d_{\tau,m} = \| \mathbf{x}[m\tau + 1] - \mathbf{x}[(m-1)\tau + 1] \|, \quad (14)$$

$$d_{1,l} = \| \mathbf{x}[l + 1] - \mathbf{x}[l] \|, \quad (15)$$

Here, τ is a scale parameter, $d_{\tau,m}$ with $m = 1, 2, \cdots, \lfloor 288/\tau \rfloor - 1$ denotes the Euclidean distance between the $((m-1)\tau+1)$-th and $(m\tau + 1)$-th samples, and $d_{1,l}$ with $l = 1, 2, \cdots, 287$ denotes the Euclidean distance between the l-th and $(l+1)$-th samples. Following [15], the scale parameter is set to $\tau = 3$ in this dataset.

Based on the above definitions of RF features, the preamble and related RF features of each IoT device can be calculated and organized into CSV files. The structure of an example file is shown in Figure 2.

Remark 1: It is worth noting that this dataset mainly focuses on the device features. In existing RF fingerprinting datasets, signal collection is often conducted under less controlled or varying environmental conditions. However, inevitable environmental biases (e.g., thermal noise, scattering, and channel fading) may induce drift in device-related features, leading to the extracted RF features not purely capturing the underlying device features. Such entanglement may result in models unintentionally exploiting environment-dependent patterns rather than device-intrinsic features. In addition, inconsistent environmental biases across devices during data collection may change the separability among devices and thereby degrade the overall quality of this device-feature-oriented dataset. To keep the impact of environmental bias in this dataset relatively small and largely consistent across devices, we adopted the following two settings during data collection. First, the USRP receiver was placed close to the IoT transmitter to minimize the impact of environmental biases (e.g., scattering and channel fading) on the RF features. Second, all signals were collected within one day in a controlled environment, where both the device placement and the surrounding environment were kept unchanged throughout data collection, thereby making the impact of environmental bias largely consistent across devices. This controlled design helps reduce confounding factors and enables a more reliable analysis of device-dependent RF fingerprinting characteristics.

2.2. Intended use cases and evaluation approach

Since this dataset was collected in a single controlled environment, it is primarily recommended for training and evaluating device-feature-based RF fingerprinting methods intended for indoor, static, short-distance, high-SNR scenarios, and can serve as a conservative benchmark for such methods. The intended use cases of this dataset include (i) continuous single-frame authentication [16], (ii) one-shot authentication based on multiple frames [16], and (iii) unknown-device identification (i.e., open-set recognition). For these use cases, the performance of RF fingerprinting methods on this dataset can be evaluated using standard metrics such as accuracy (ACC), recall, F1-score, false acceptance rate (FAR), false rejection rate (FRR), and receiver operating characteristic (ROC) curves. It should be noted that, due to the controlled acquisition conditions, this dataset is not intended for evaluating robustness under varying environments or for direct deployment in real-world dynamic scenarios. Care should be taken when interpreting results, as performance on this dataset primarily reflects the ability to capture device-intrinsic features rather than generalization across environments. Future evaluations may incorporate multi-environment or domain-adaptive settings to complement this benchmark.

3. Experimental Design, Materials and Methods

3.1. Dataset construction workflow

This section presents the overall workflow for constructing the dataset. As shown in Figure 3, the framework is composed of three modules: (1) data collection, (2) feature extraction, and (3) benchmark evaluation. The data collection module collects I/Q samples of preambles from 123

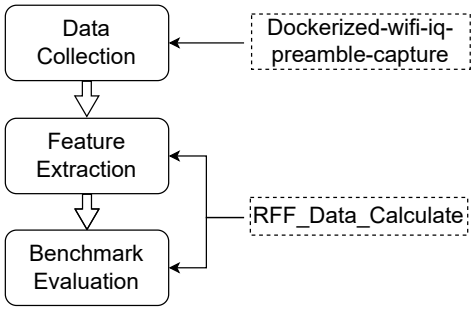


Figure 3: The workflow for constructing the dataset.

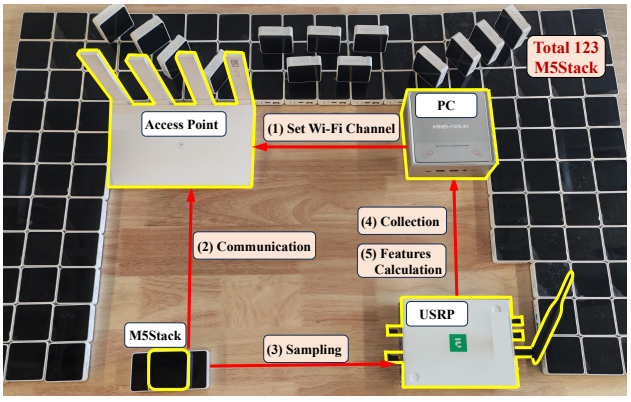


Figure 4: Experimental platform.

same-model devices. Then, the feature extraction module calculates the RF features from the collected I/Q samples. By utilizing these features, the benchmark evaluation module provides reproducible baseline results.

3.2. Data collection

As shown in Figure 4, the collection platform consists of a USRP B210 software-defined radio acting as a receiver, 123 M5Stack Core2 devices serving as transmitters, a Huawei WS7100 V2 Wi-Fi access point (AP), and a Minisforum NPB7 computer. AP is configured according to the IEEE 802.11g standard on Channel 6 (20 MHz bandwidth). The computer runs the GNU Radio 802.11 a/g/p transceiver model [1], which controls the USRP and manages baseband signal processing. All experiments are conducted in an indoor office, with a fixed 25 cm separation between each transmitter and the receiver.

The data collection procedure can be summarized as follows. First, the AP is configured to operate on the target Wi-Fi channel, ensuring that all transmissions from the transmitters are routed through this channel. A Python control script then triggers a transmitter to repeatedly transmit packets toward the AP. During transmissions, the USRP continuously listens and samples the preambles of signals. The captured samples are then stored on the computer as a CSV file. This process is repeated until up to 1,000 preambles have been collected for each device.

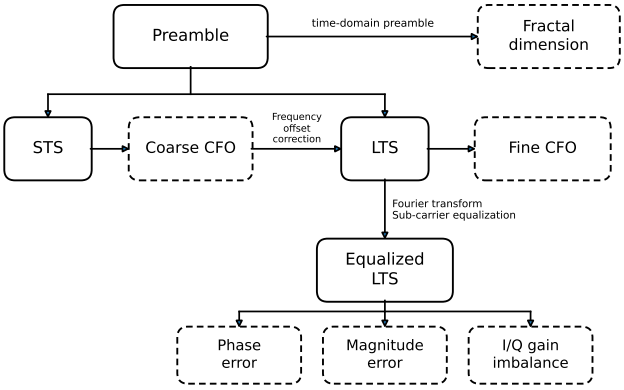


Figure 5: Workflow of the feature extraction model.

3.3. Feature extraction

Based on the raw I/Q preambles collected in the first module, this module extracts multi-dimensional feature representations from the sampled data. The workflow of the feature extraction model is illustrated in Figure 5. Specifically, we first extract the coarse CFO and fine CFO from the STS and LTS, respectively. Then, frequency offset correction, Fourier transform, and subcarrier equalization are applied to the LTS to obtain the equalized LTS. Finally, the phase error, magnitude error, and I/Q gain imbalance are calculated based on the equalized LTS, whereas the fractal dimension is calculated directly from the raw time-domain preamble.

3.4. Benchmark evaluation

This module provides a unified environment for analyzing the extracted features and establishing a reproducible baseline for RF fingerprinting. It operates on the features produced by Section 3.3 and supports both dataset enhancement and benchmark evaluation.

Feature visualization: To help users examine the characteristics of the extracted features, the framework includes t-SNE–based embedding tools and correlation-analysis utilities. These visualizations offer an intuitive view of feature separability of transmitters.

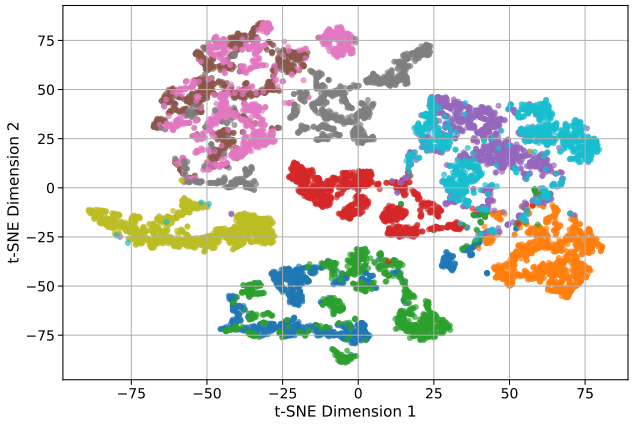
Feature enhancement: This tool provides an optional Kalman Filtering–based smoothing mechanism that can be applied to reduce noise in the computed features.

Benchmark algorithm: A Random Forest classifier is implemented as the default identification baseline. This benchmark provides a standardized entry point for evaluating new RF fingerprinting methods on the dataset.

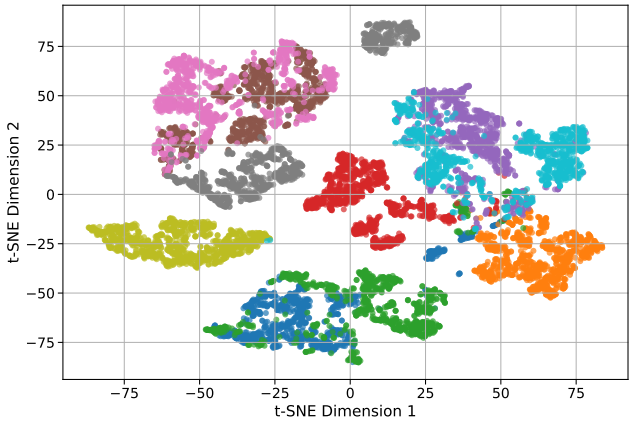
3.5. Implementation resources

To support full reproducibility of the proposed framework, we provide open-source implementations that correspond directly to each module described above.

Data collection environment. A complete data collection module is fully encapsulated in a Docker container. This containerized setup can be deployed to reproduce the



(a) Before Kalman Filtering



(b) After Kalman Filtering

Figure 6: RF fingerprint features for 10 randomly selected devices, where each color represents a device.

procedure in Section 3.2. Repository: `https://github.com/aiot-lab-yin/Dockerized-wifi-iq-preamble-capture`

Feature extraction and benchmark evaluation. Both modules are encapsulated in a Jupyter Notebook using the Python programming language, which provides online implementation from feature extraction (Section 3.3) to benchmark evaluation (Section 3.4). Repository: `https://www.kaggle.com/code/zeweiguo/rff-data-calculate`

Table 2
Evaluation Results for Random Forest Model

Dataset & Setting	Accuracy (ACC)
80/20 Split **without** KF	88.65
80/20 Split **with** KF	94.54

4. Experimental Results and Analysis

4.1. Feature enhancement via Kalman Filtering

The extracted RF features exhibit temporal variations caused by hardware imperfections, measurement noise, and minor environmental fluctuations. These variations blur the device-dependent patterns and reduce inter-device separability. To obtain a more stable representation, we apply Kalman Filtering (KF) to smooth each feature dimension.

To demonstrate the effect of KF on this dataset, we visualize the high-dimensional feature vectors using t-distributed Stochastic Neighbor Embedding (t-SNE). For clarity, the visualization is generated from ten randomly selected devices, with each color representing one device. As shown in Figure 6a, the unfiltered features form clusters with noticeable overlap, indicating weakened device distinguishability. After applying KF, the clusters in Figure 6b become compact within each device and more clearly separated across devices, which confirms that KF improves the discriminability of the feature space.

4.2. Benchmark results

In this subsection, we adopt the Random Forest model to achieve baseline identification performance on the proposed dataset. In particular, the Random Forest classifier is configured with 128 decision trees, trained on 80% of the data, and tested on the remaining 20% to evaluate the performance. Based on the established baseline model, we further analyze the relative importance and inter-feature correlations among the extracted RF features.

As summarized in Table 2, the results of a single 80/20 train-test split of the Random Forest model are reported for both cases with and without KF. Results show that applying KF increases the accuracy from 88.65% to 94.54%. This quantitative improvement is consistent with the enhancement in feature separation visualized in Figure 6b, which validates the effectiveness of KF in enhancing identification accuracy.

Remark 2: It is worth noting that the RF features extracted in this dataset have been widely adopted in prior physical-layer identification studies [3, 4, 13, 15, 17], and several works have shown that these features can reflect representative hardware impairments (e.g., oscillator mismatch [18], power-amplifier nonlinearity [19], and I/Q-branch gain mismatch [20]). Thus, these features are well-suited for characterizing device-intrinsic properties from a physical-layer perspective. Moreover, existing studies have reported high identification accuracy using these features on relatively small-scale and heterogeneous-device datasets when employing more sophisticated machine- or deep-learning models [3, 4, 15]. However, such high performance may partially benefit from easier identification settings, where device heterogeneity or environmental variations introduce additional discriminative cues. Therefore, we believe that the accuracy reported in this paper can be reasonably attributed to the following two reasons: (i) the increased identification difficulty of our dataset, (ii) the simplicity of the baseline learning model. More specifically, (i) this dataset is collected from a large number of the same-model devices, such that their RF features are more similar than those in many commonly used datasets involving heterogeneous device models, which makes the device identification more challenging; furthermore, under controlled conditions,

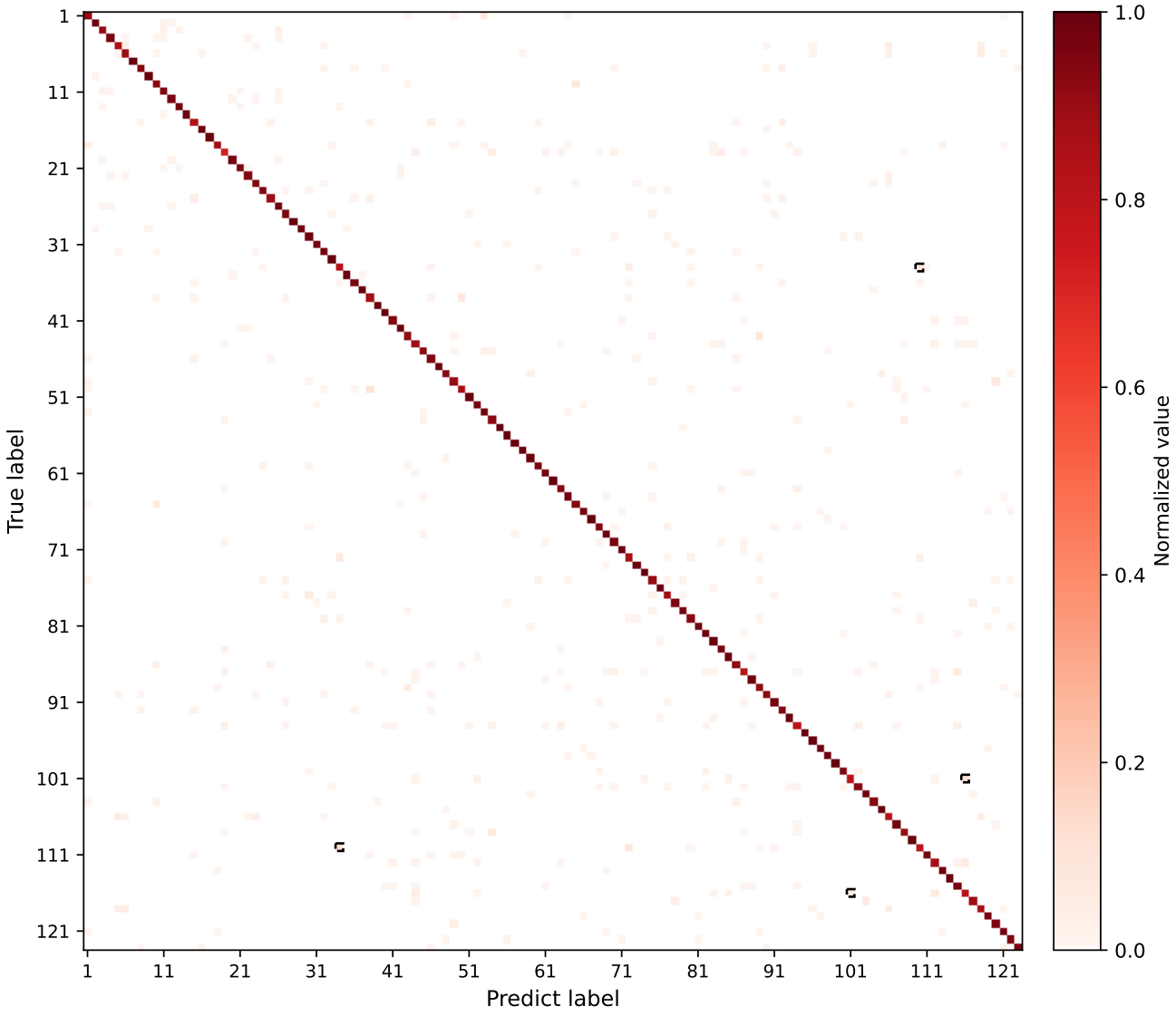
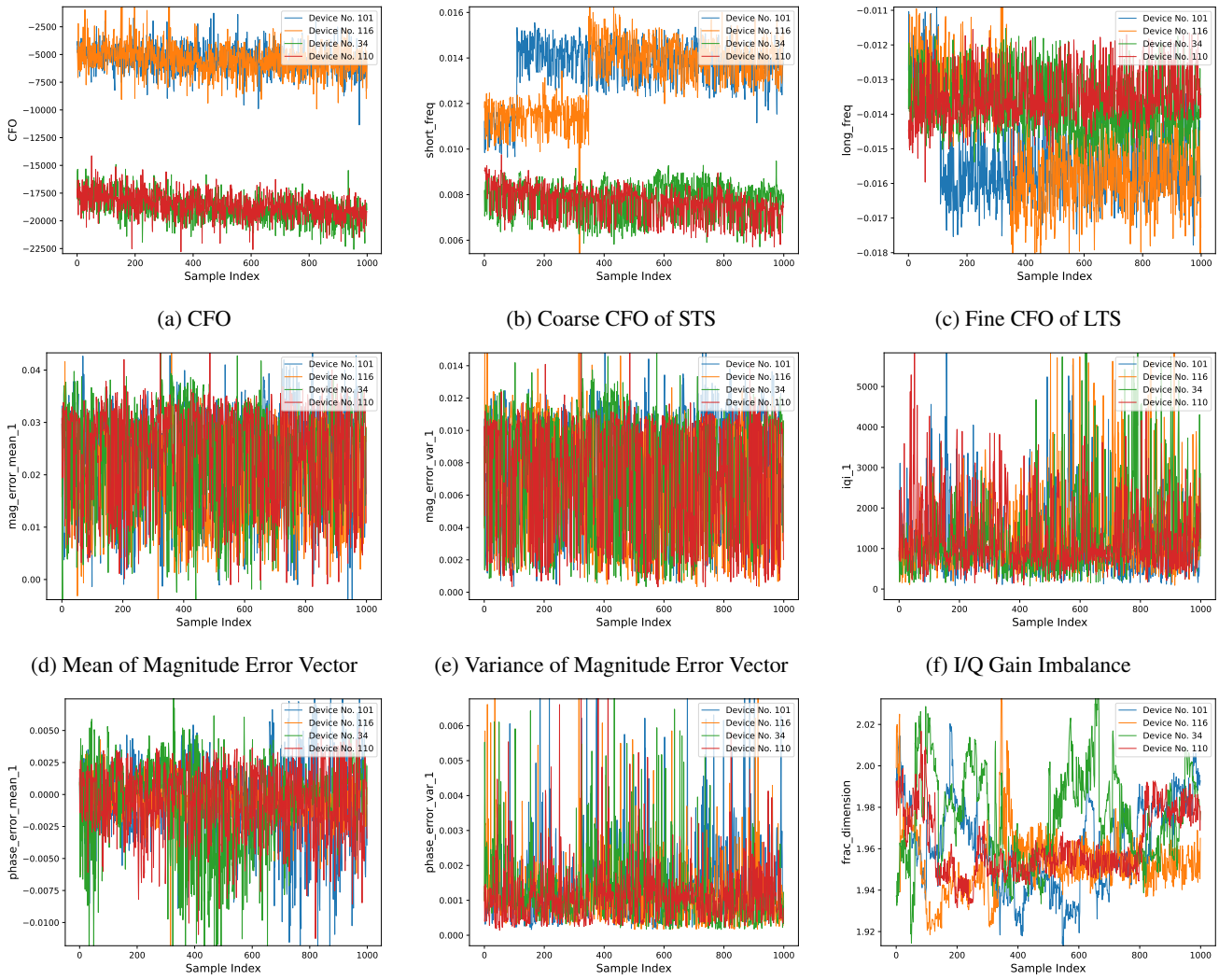


Figure 7: Confusion matrix among devices.

Table 3
Identification Difficulty of Per-device

No.	ACC	Recall	F1 score	No.	ACC	Recall	F1 score	No.	ACC	Recall	F1 score
1	0.91	0.90	0.91	**42**	0.97	0.99	0.98	**83**	0.96	1.00	0.98
2	0.99	0.99	0.99	**43**	0.92	0.91	0.91	**84**	0.90	0.96	0.93
3	0.96	0.91	0.93	**44**	0.85	0.88	0.87	**85**	0.96	0.99	0.97
4	0.98	0.96	0.97	**45**	0.95	0.92	0.94	**86**	0.93	0.92	0.92
5	0.88	0.84	0.86	**46**	0.90	0.95	0.93	**87**	0.92	0.80	0.86
6	0.87	0.89	0.88	**47**	1.00	1.00	1.00	**88**	0.98	0.98	0.98
7	1.00	1.00	1.00	**48**	0.97	0.93	0.95	**89**	0.89	0.90	0.90
8	0.95	0.93	0.94	**49**	0.90	0.90	0.90	**90**	0.95	0.92	0.93
9	0.99	1.00	0.99	**50**	0.88	0.86	0.87	**91**	0.98	0.97	0.98
10	0.88	0.93	0.91	**51**	0.98	0.99	0.99	**92**	0.90	0.93	0.92
11	0.94	0.95	0.94	**52**	0.95	0.97	0.96	**93**	0.99	0.99	0.99
12	0.97	0.96	0.96	**53**	0.88	0.97	0.92	**94**	0.91	0.81	0.85
13	0.97	0.98	0.97	**54**	0.87	0.93	0.90	**95**	1.00	1.00	1.00
14	1.00	0.99	1.00	**55**	1.00	1.00	1.00	**96**	0.98	1.00	0.99
15	0.90	0.84	0.87	**56**	0.96	1.00	0.98	**97**	0.98	0.98	0.98
16	0.99	1.00	0.99	**57**	0.99	1.00	1.00	**98**	0.98	0.98	0.98
17	1.00	1.00	1.00	**58**	1.00	1.00	1.00	**99**	1.00	1.00	1.00
18	0.97	0.88	0.92	**59**	0.94	0.99	0.96	**100**	0.97	0.94	0.95
19	0.79	0.73	0.76	**60**	0.96	0.95	0.95	**101**	0.92	0.80	0.86
20	0.97	0.97	0.97	**61**	0.96	0.97	0.96	**102**	0.95	0.93	0.94
21	0.98	0.96	0.97	**62**	1.00	1.00	1.00	**103**	0.94	0.97	0.95
22	0.97	0.95	0.96	**63**	0.93	0.93	0.93	**104**	0.98	0.94	0.96
23	0.93	0.95	0.94	**64**	0.96	0.98	0.97	**105**	0.97	0.99	0.98
24	0.96	0.95	0.95	**65**	0.92	0.94	0.93	**106**	0.80	0.82	0.81
25	0.94	0.89	0.92	**66**	0.98	0.97	0.98	**107**	0.97	0.97	0.97
26	0.92	0.98	0.95	**67**	0.99	1.00	0.99	**108**	0.91	0.90	0.90
27	0.95	0.96	0.96	**68**	0.96	0.94	0.95	**109**	1.00	1.00	1.00
28	1.00	1.00	1.00	**69**	0.97	0.98	0.97	**110**	0.86	0.80	0.83
29	1.00	0.97	0.99	**70**	0.92	0.97	0.94	**111**	0.98	0.96	0.97
30	0.91	0.97	0.94	**71**	0.97	0.99	0.98	**112**	0.86	0.87	0.87
31	0.99	0.97	0.98	**72**	0.85	0.86	0.85	**113**	0.97	0.98	0.98
32	0.93	0.97	0.95	**73**	0.98	1.00	0.99	**114**	1.00	1.00	1.00
33	0.97	1.00	0.98	**74**	0.99	1.00	0.99	**115**	0.86	0.96	0.91
34	0.80	0.82	0.81	**75**	0.90	0.90	0.90	**116**	0.79	0.80	0.79
35	0.99	0.98	0.99	**76**	0.99	0.99	0.99	**117**	0.95	0.89	0.92
36	0.96	0.96	0.96	**77**	0.94	0.88	0.91	**118**	0.88	0.86	0.87
37	0.99	1.00	0.99	**78**	0.94	0.95	0.94	**119**	0.97	0.97	0.97
38	0.86	0.88	0.87	**79**	0.98	0.97	0.97	**120**	0.90	0.96	0.93
39	0.99	1.00	1.00	**80**	0.94	0.94	0.94	**121**	0.94	0.97	0.95
40	0.98	1.00	0.99	**81**	0.98	0.98	0.98	**122**	0.97	0.95	0.96
41	0.96	0.94	0.95	**82**	0.94	0.98	0.96	**123**	0.99	0.96	0.98



(a) CFO (b) Coarse CFO of STS (c) Fine CFO of LTS

(d) Mean of Magnitude Error Vector (e) Variance of Magnitude Error Vector (f) I/Q Gain Imbalance

(g) Mean of Phase Error Vector (h) Variance of Phase Error Vector (i) Fractal Dimension

Figure 8: Impact of sessions on different RF features.

the influence of environmental factors is reduced, which limits the possibility for models to exploit environment-dependent patterns, thereby increasing the intrinsic difficulty of the task; (ii) the proposed random forest classifier is a lightweight baseline to facilitate reproducible benchmarking. Its representational capacity is typically lower than that of more advanced machine learning or deep learning models, which further increases the performance gap.

To further analyze the identification difficulty of individual devices and the confusion patterns among them, we report the per-device identification accuracies in Table 3 and present the confusion matrix in Figure 7. It can be seen from Table 3 that the lowest per-device accuracies are observed for devices No. 116, 19, 106, and 34. As highlighted by the dashed boxes in Figure 7, device No. 101 is frequently confused with No. 116, and device No. 110 is frequently confused with No. 34. To explore the reason behind this phenomenon, we further compared RF features for devices No. 101, 116, 34, and 110 in Figure 8.

Figure 8 shows the values of RF features across sessions. It can be seen that the combined CFO exhibits a slight decreasing trend as the sample index increases, while the coarse CFO and the fine CFO shift in opposite directions at some point of the capture, so that their sum stays continuous. The fractal dimension drifts slowly over the capture, whereas the other features remain relatively stable. Notably, the observed decreasing trend of the CFO is consistent with the CFO variation reported in [21]. However, since the data collection duration of this dataset is relatively short, it is insufficient to draw definitive conclusions on how time (or session evolution) affects RF features. How to rigorously characterize time-induced effects in RF features remains an important open problem. Accordingly, a dedicated dataset with longer-term measurements is warranted to support studies of temporal effects on RF features.

Following the accuracy evaluation, Table 4 presents the feature importance evaluation for device identification based on the Random Forest model. As shown in Table 4,

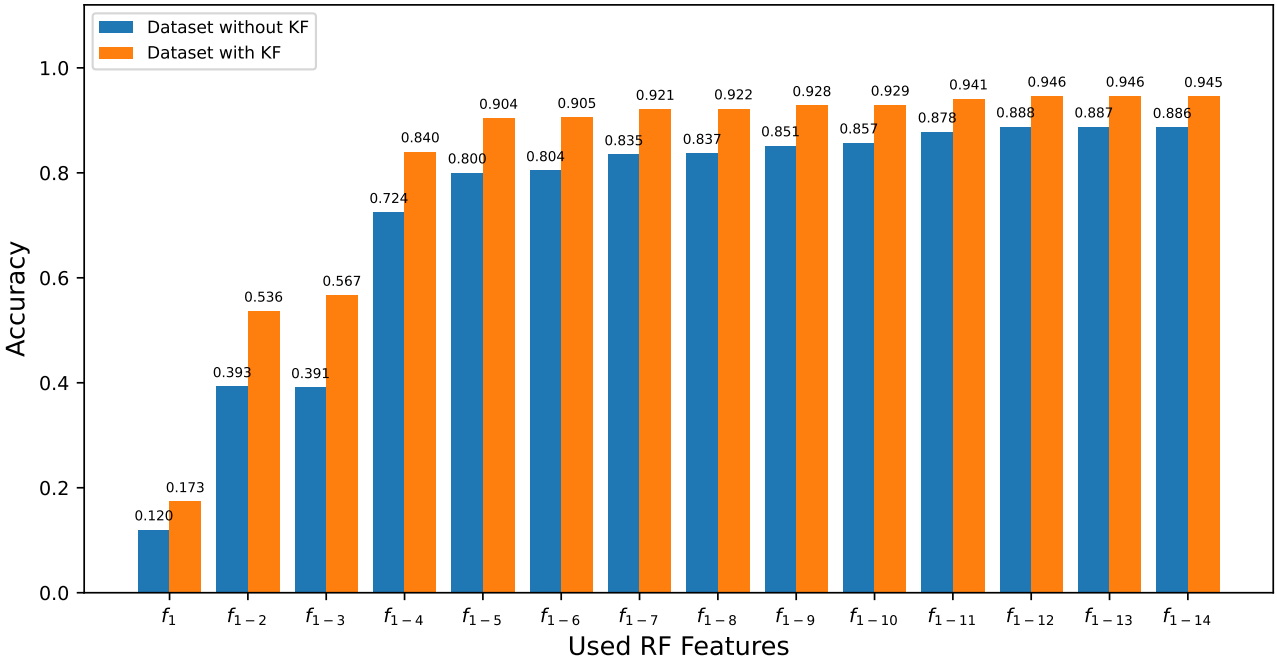


Figure 9: Identification accuracy versus feature groups.

Table 4
Feature Importance for Random Forest Model

Notation	Feature	Importance
f_1	CFO	0.2286
f_2	Coarse CFO of STS (short_freq)	0.1780
f_3	Fractal Dimension (frac_dimension)	0.1496
f_4	Fine CFO of LTS (long_freq)	0.1402
f_5	Mean of Phase Error Vector of L1 (phase_error_mean_1)	0.0378
f_6	I/Q Gain Imbalance of L1 (iqi_1)	0.0376
f_7	Mean of Phase Error Vector of L2 (phase_error_mean_2)	0.0365
f_8	I/Q Gain Imbalance of L2 (iqi_2)	0.0346
f_9	Mean of Magnitude Error Vector of L2 (mag_error_mean_2)	0.0300
f_{10}	Mean of Magnitude Error Vector of L1 (mag_error_mean_1)	0.0293
f_{11}	Variance of Magnitude Error Vector of L2 (mag_error_var_2)	0.0254
f_{12}	Variance of Magnitude Error Vector of L1 (mag_error_var_1)	0.0246
f_{13}	Variance of Phase Error Vector of L1 (phase_error_var_1)	0.0241
f_{14}	Variance of Phase Error Vector of L2 (phase_error_var_2)	0.0235

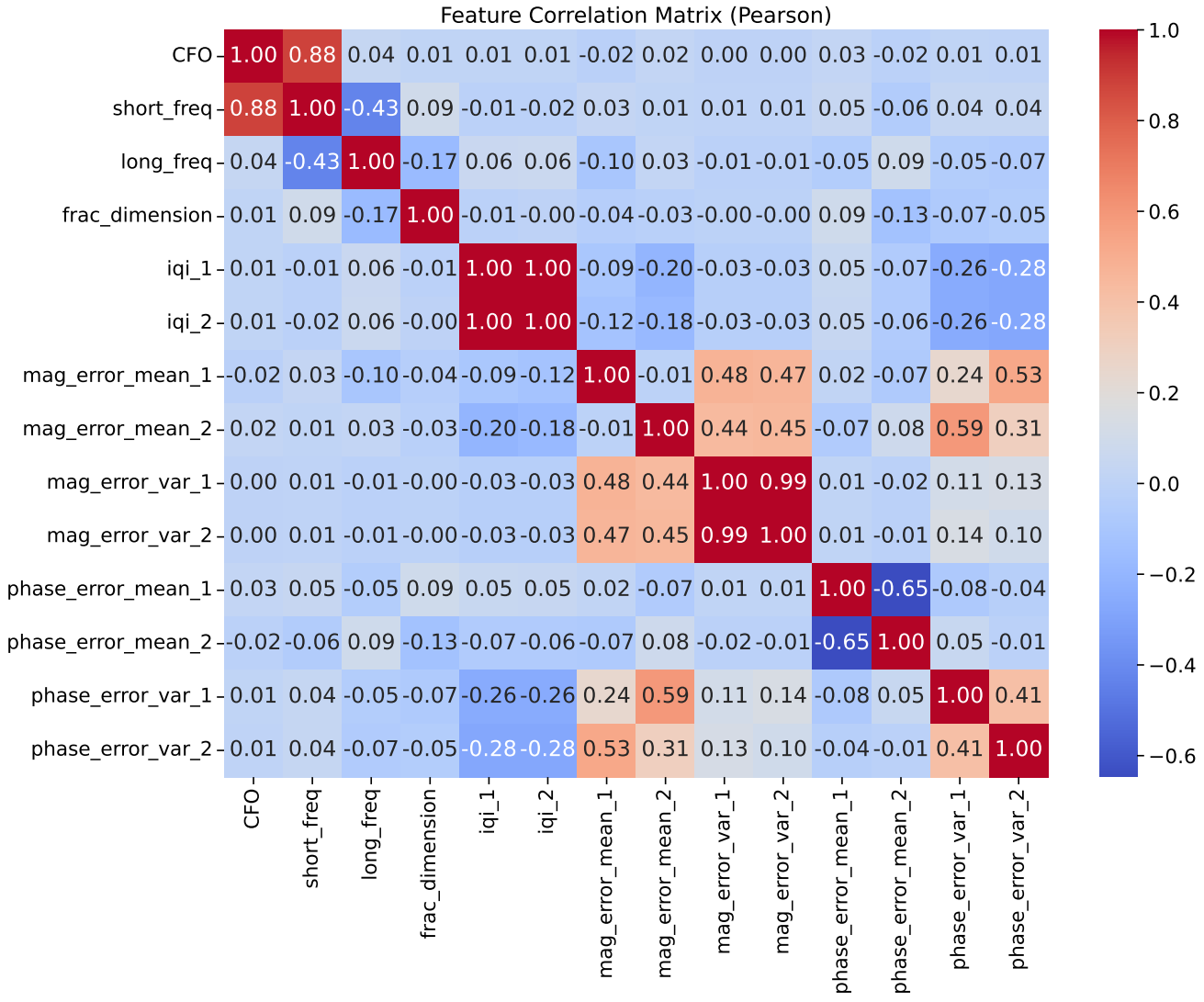


Figure 10: Correlation matrix of RF fingerprint features.

frequency-related features (i.e., CFO and short_freq) are the most discriminative, reflecting both their robustness and device-specific characteristics. Constellation-related features such as phase and magnitude-based features, together with I/Q gain imbalance, exhibit moderate importance. The fractal dimension ranks third, which indicates that the time-domain preamble carries substantial device-dependent information.

Figure 9 illustrates how the identification accuracy of the Random Forest method varies with the selected feature groups, where f_{1-a} denotes the feature group consisting of RF features 1 through $a \in \{2, 3, \cdots, 14\}$, following the order of the feature list used in the benchmark code. It can be observed that the accuracy first increases as more features are included, and then remains almost constant once features beyond f_{12} are added. This trend suggests that features f_{1-11} capture most of the RF features in this dataset, whereas the additional features f_{12-14} provide limited incremental discriminative information.

To explore the relationships among the extracted features, the Pearson correlation matrix is illustrated in Figure 10. We can observe a strong positive correlation ($r = 0.88$) between the CFO and the coarse CFO of the STS (short_freq), which indicates that they capture similar characteristics. The fine CFO (`long_freq`) has a weak Pearson correlation with the combined CFO ($r = 0.04$) and a negative correlation with the coarse CFO ($r = -0.43$). Furthermore, the two I/Q gain imbalance features (iqi_1 and iqi_2) exhibit an extremely high correlation ($r = 0.999$), implying that they have consistent structural patterns and potential redundancy. The fractal dimension shows low correlation with all other features, with a maximum absolute value of 0.17, which indicates that it provides complementary information. Meanwhile, within the constellation-related group, the magnitude-error means of the two training symbols are almost uncorrelated ($r = -0.01$), whereas the corresponding variances ($r = 0.99$) and the phase-error means ($r = -0.65$) are strongly dependent. Overall, the prevalence of

low inter-feature correlations confirms the diversity of the features, which contributes to the discriminative ability of the identification model.

5. Conclusions

This paper introduces a large-scale RF feature dataset that is constructed from 123 same-model devices and contains over 35.28 million raw I/Q samples together with 1.72 million extracted RF features. To establish a reproducible benchmark, we further provide a complete experimental framework that integrates data collection, feature extraction, and performance evaluation. A Random Forest–based identification model is implemented as the benchmark, which achieves an accuracy of 94.54% on the proposed dataset. In addition, our feature analysis reveals that frequency-related features contribute most significantly to device identification. Overall, the dataset and the framework provide a unified basis for reproducible RF fingerprinting research. They are expected to promote standardized evaluation and support future algorithmic advancements.

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.

Acknowledgments

This work was partly supported by JST Moonshot R&D Grant Number JPMJMS2215, and JSPS KAKENHI Grant Number JP24K07482.

References

[1] B. Bloessl, M. Segata, C. Sommer, F. Dressler, An IEEE 802.11a/g/p OFDM receiver for GNU radio, in: Proc. ACM 2nd Workshop Softw. Radio Implement. Forum, SRIF '13, New York, NY, USA, 2013, p. 9–16.

[2] RFtap, accessed on 2025-06-24 (2016). [link]. URL https://rftap.github.io/

[3] S. Zeng, Y. Chen, X. Li, J. Zhu, Y. Shen, N. Shiratori, Visibility graph entropy based radiometric feature for physical layer identification, Ad Hoc Networks 127 (2022) 102780.

[4] S. Zeng, Y. Chen, X. Li, J. Zhu, Y. Shen, N. Shiratori, Time–frequency fusion for enhancement of deep learning-based physical layer identification, Ad Hoc Networks 142 (2023) 103099.

[5] A. Elmaghbub, B. Hamdaoui, LoRa device fingerprinting in the wild: Disclosing RF data-driven fingerprint sensitivity to deployment variability, IEEE Access 9 (2021) 142893–142909.

[6] E. Uzundurukan, Y. Dalveren, A. Kara, A database for the radio frequency fingerprinting of bluetooth devices, MDPI Data 5 (2) (2020) 55.

[7] A. Jagannath, Z. Kane, J. Jagannath, Bluetooth and WiFi dataset for real world RF fingerprinting of commercial devices, IEEE Commun. Mag. 62 (5) (2024) 50–56.

[8] A. Al-Shawabka, F. Restuccia, S. D'Oro, T. Melodia, Massive-scale I/Q datasets for WiFi radio fingerprinting, Computer Networks 182 (2020) 107566.

[9] S. Hanna, S. Karunaratne, D. Cabric, Wisig: A large-scale wifi signal dataset for receiver and channel agnostic RF fingerprinting, IEEE Access 10 (2022) 22808–22818.

[10] IEEE 802 LAN/MAN Standards Committee and others, IEEE standard for information technology-telecommunication and information exchange between systems-local and metropolitan area networks-specific requirements part11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment1: Radio resource measurement of wireless LANs, Standard 802.11 (2012).

[11] E. Sourour, H. El-Ghoroury, D. McNeill, Frequency offset estimation and correction in the IEEE 802.11a WLAN, in: Proc. IEEE 60th Veh. Technol. Conf., Vol. 7, 2004, pp. 4923–4927.

[12] OpenOFDM, accessed on 2025-07-04 (2017). [link]. URL https://openofdm.readthedocs.io/en/latest/

[13] V. Brik, S. Banerjee, M. Gruteser, S. Oh, Wireless device identification with radiometric signatures, in: Proc. ACM 14th Int. Conf. Mobile Comput. Netw., MobiCom '08, Association for Computing Machinery, New York, NY, USA, 2008, p. 116–127.

[14] H. Arslan, IQ gain imbalance measurement for OFDM based wireless communication systems, in: Proc. IEEE Mil. Commun. Conf., 2006, pp. 1–5.

[15] X. Li, Y. Chen, J. Zhu, S. Zeng, Y. Shen, X. Jiang, D. Zhang, Fractal dimension of DSSS frame preamble: Radiometric feature for wireless device identification, IEEE Trans. Mobile Comput. 23 (2) (2024) 1416–1430.

[16] T. Kikuchi, K. Shibata, K. Yasumoto, J. Zhu, Y. Chen, Evaluating effectiveness of temporal features and DTW distance for radio frequency fingerprinting, in: Proc. the 31st Annu. Int. Conf. Mob. Comput. Netw. (MobiCom), 2025, p. 1296–1298.

[17] Y. Shi, M. A. Jensen, Improved radiometric identification of wireless devices using MIMO transmission, IEEE Trans. Inf. Forensics Security 6 (4) (2011) 1346–1354.

[18] W. Hou, X. Wang, J.-Y. Chouinard, A. Refaey, Physical layer authentication for mobile systems with time-varying carrier frequency offsets, IEEE Trans. Commun. 62 (5) (2014) 1658–1667.

[19] S. S. Hanna, D. Cabric, Deep learning based transmitter identification using power amplifier nonlinearity, in: Proc. Int. Conf. Comput., Netw. Commun. (ICNC), 2019, pp. 674–680.

[20] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, K. Chowdhury, ORACLE: Optimized radio classification through convolutional neural networks, in: Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM), 2019, pp. 370–378.

[21] G. Shen, J. Zhang, A. Marshall, L. Peng, X. Wang, Radio frequency fingerprint identification for LoRa using deep learning, IEEE J. Sel. Areas Commun. 39 (8) (2021) 2604–2616.