Cognitive Risk Control for Physical-Layer RFID Counterfeit Tag Identification

Haifeng Wu, Chongrong Pu, Wei Gao, and Yu Zeng

***Abstract*— Physical-layer counterfeit tag identification is a widely adopted technology for detecting clone tags in radio-frequency identification (RFID) systems. It has gained significant attention in recent years owing to its affordability and ease of implementation. However, the effectiveness of physical-layer identification technology is highly dependent on the tag identification environment. When the reading distance is far or signal power is low, the signal-to-noise ratio (SNR) of the tag signal may be low, leading to a degradation in classification accuracy. To address the issue, this article explores the application of cognitive risk control (CRC) in tag classification, aiming to enhance the SNR and improve classification accuracy. Additionally, this article introduces more time- and frequency-domain features, resulting in a total of 104 features, to further enhance the classification performance. The experiment utilizes a software radio device to classify tags from three manufacturers across seven popular types. The results indicate that the implementation of CRC has, on average, improved the classification accuracy of the tags by approximately eight percentage points. Moreover, when compared to traditional 28 and seven features, the use of 104 features has shown an average increase in classification accuracy of about two and four percentage points, respectively.**

***Index Terms*— Anticounterfeiting, cognitive risk control (CRC), physical-layer identification, radio-frequency identification (RFID), tag.**

NOMENCLATURE

RFID	Radio frequency identification.
SNR	Signal-to-noise ratio.
IDs	Identities.
PUF	Physical unclonable function.
USRP	Universal Software Radio Peripheral.
UHF	Ultrahigh Frequency.
TX	Transmit power.
RX	Receive power.
CrossVal A	Cross-validation of different types or manufacturers.
CrossVal B	Cross-validation of the same type and manufacturer.
CRC	Cognitive risk control.
rms	Root mean square.
SVM	Support Vector Machine.
RF	Random forest.
KNN	K-nearest neighbors.
TP	True positives.
TN	True negatives.
FP	False positives.
FN	False negatives.

Manuscript received 17 July 2023; revised 20 September 2023; accepted 5 October 2023. Date of publication 27 October 2023; date of current version 9 November 2023. This work was supported in part by the Natural Science Foundation of China under Grant 62161052 and in part by the Yunnan Key Laboratory of Unmanned Autonomous System. The Associate Editor coordinating the review process was Dr. Wei Fan. *(Corresponding author: Haifeng Wu.)*

The authors are with the Department of Electrical and Information Engineering, Yunnan Minzu University, Kunming 650500, China (e-mail: whf5469@gmail.com; puchongrong@gmail.com; 434357606@qq.com; yv.zeng@gmail.com).

Digital Object Identifier 10.1109/TIM.2023.3328075

I. INTRODUCTION

RADIO-FREQUENCY identification (RFID) is a wireless automatic identification technology [1] that enables the identification of electronic tags with unique identities (IDs) without human intervention or physical contact. It has found widespread use in various sectors such as logistics supply chains, the retail industry, healthcare, and intelligent transportation. With the advent of the 5G era and the increasing popularity of smartphones, RFID technology is poised to have even broader applications in the future. However, the lack of effective security mechanisms for RFID tags poses a significant challenge. Without proper security measures, different readers can access tag information, making it susceptible to theft, and enabling criminals to counterfeit tags for unlawful gains [2]. Therefore, ensuring RFID security and anticounterfeiting measures is of utmost importance.

To enhance the security of RFID systems, various encryption algorithms and security protocols can be employed [3], [4]. However, the implementation of protocols requires the presence of a shared key between tags and readers, which adds complexity to key management. This poses challenges for passive tags with limited computational capabilities, making key management difficult to implement and encryption speeds slow. On the other hand, hash algorithms [5], [6] offer convenience as they do not rely on shared keys, but they have inherent security limitations. If the hash chain is too short, attackers can exploit brute-force methods for cracking. Access control mechanisms [7] can restrict unauthorized read and write operations, but their complex authentication and authorization processes increase system complexity, and cost, and reduce system response speed. Additionally, hardware-based technologies can be employed to prevent tag counterfeiting, such as incorporating special metal fibers or unique packaging during manufacturing [8], [9]. However, the technologies often come with increased manufacturing costs, reduced read and write performance, and limited versatility. In addition to various encryption algorithms and security protocols, there

are also some deep-learning methods [43], [44] that can also be used for authentic and counterfeiting signal recognition. However, the performance of deep learning often relies on massive amounts of data, and parameter adjustment is also a complex issue.

Recent research has revealed that RFID tag response signals possess a unique physical unclonable function (PUF) [10], [11], [12]. The function stems from the hardware design and physical characteristics of the tags, such as antenna structure, power, and frequency response. The hardware characteristics are inherent to the tag's manufacturing process and are typically difficult to counterfeit. As a result, physical-layer anticounterfeiting technology directly extracts PUF features from the tag response signal and employs signal processing and pattern recognition techniques to differentiate between authentic and counterfeit tags. In comparison to encryption and access control technologies, physical-layer identification requires fewer hardware modifications and incurs lower costs. Consequently, it is easily transferrable and suitable for large-scale, cost-effective RFID systems. However, the performance of physical-layer identification techniques relies heavily on the quality of the tag's signal and the resulting PUF features. In challenging communication environments with significant noise interference and low-signal power, the extracted features may not accurately reflect the tag type, leading to misclassification.

To address the aforementioned challenges, this article presents a novel physical-layer identification technique for distinguishing between authentic and counterfeit tags. The contributions of this research are as follows. First, the proposed method leverages the concept of cognitive risk control (CRC) [13] to establish a framework that can change the signal-to-noise ratio (SNR) of the tag signal to meet the tag classification requirement. This approach effectively overcomes the limitations of low tag classification accuracy in low SNR scenarios. Second, the method explores a broader range of time- and frequency-domain features in the tag response signals to enhance the classification accuracy and mitigate the model generalization associated with feature selection. In the experiments, we utilized a Universal Software Radio Peripheral (USRP) to classify response signals from ultrahigh frequency (UHF) RFID tags of seven different types, manufactured by three distinct companies. The experimental results demonstrate that the proposed method achieves an average improvement of approximately eight percentage points in classification accuracy by enhancing the tag signal SNR. Furthermore, the results reveal an average classification accuracy increase of about three percentage points compared to traditional features when incorporating the new time- and frequency-domain features.

II. Related Works

A. RFID Anticounterfeiting

To enhance the system security of RFID, the implementation of security protocols is a commonly adopted approach. Some RFID security protocols incorporate existing protocols, such as TLS/SSL [3] and IPSec [4]. However, the protocols have high power consumption and necessitate stable communication links, making them less suitable for certain RFID applications. Existing RFID standards also offer security protocols, such as the EPC global Class 1 Generation 2 standard [14], which employs a simple and practical password authentication method. However, if the password is compromised, it becomes susceptible to attacks. In contrast, the ISO/IEC 29167-10 standard [15] provides robust protocols with a variety of encryption algorithms and security levels to accommodate diverse security requirements. Nevertheless, due to its recent introduction, the adoption rate of the standard in RFID devices is comparatively lower than that of the EPC global standard. Additionally, there exist more complex RFID protocols, such as hashlock [16], robust security network [17], blocker tag [18], and verifiable anonymous RFID protocol [7], which utilize encryption algorithms, digital signatures, and key management techniques [19], [20], [21] to enhance RFID security. While the protocols offer higher levels of security, they entail significant computational and storage requirements, posing challenges for implementation in RFID systems. Furthermore, inadequate key management can compromise the security provided by the protocols. On the other hand, simpler implementation, and lower computational and storage requirements are characteristic of one-way hash-lock protocols [5], [22] and random number generation protocols [6], [23], albeit with slightly reduced security. In summary, security and complexity must be balanced, as algorithms with higher security often entail greater complexity, while those with lower complexity may offer reduced security. Consequently, it is crucial to select appropriate security algorithms or protocols based on specific requirements in practical applications.

In addition to security protocols, various hardware methods leverage the physical characteristics of RFID tags to enhance security. One such method is the use of a Faraday cage [8], [24], which is a physical device surrounded by a metal grid or conductor that shields external electromagnetic signals, preventing unauthorized reading of the tags. While the method is simple and effective, it requires enclosing the entire tag, leading to increased costs and complexity. Reflective shielding [25] is another technique that involves reflecting the reader signal back, causing interference and impeding attackers from reading the tag. The method offers flexibility in countering different attack methods but necessitates additional hardware and algorithm support. The physical damage approach prevents unauthorized access to the tag through physical damage, such as placing the tag in fragile packaging [9]. Once the packaging is damaged, the information becomes unreadable. While this approach is simple, it does not prevent attacks on the internal information of the tags. Temperature-sensitive tags [26] alter their signal response with temperature changes, enabling the detection of unauthorized tag reading. The tags do not require additional hardware, but careful adjustment of the tag's response to temperature changes is necessary to avoid misjudgments. Although the aforementioned hardware methods can provide high-security performance, particularly in resisting physical tampering, they require consideration during the system design and manufacturing stages. It is challenging to update them through software later, posing difficulties in terms of versatility and transplantability.

Classifying the authenticity of tags based on the features of physical-layer tag signals is a common technique used for anticounterfeiting. One approach involves directly extracting

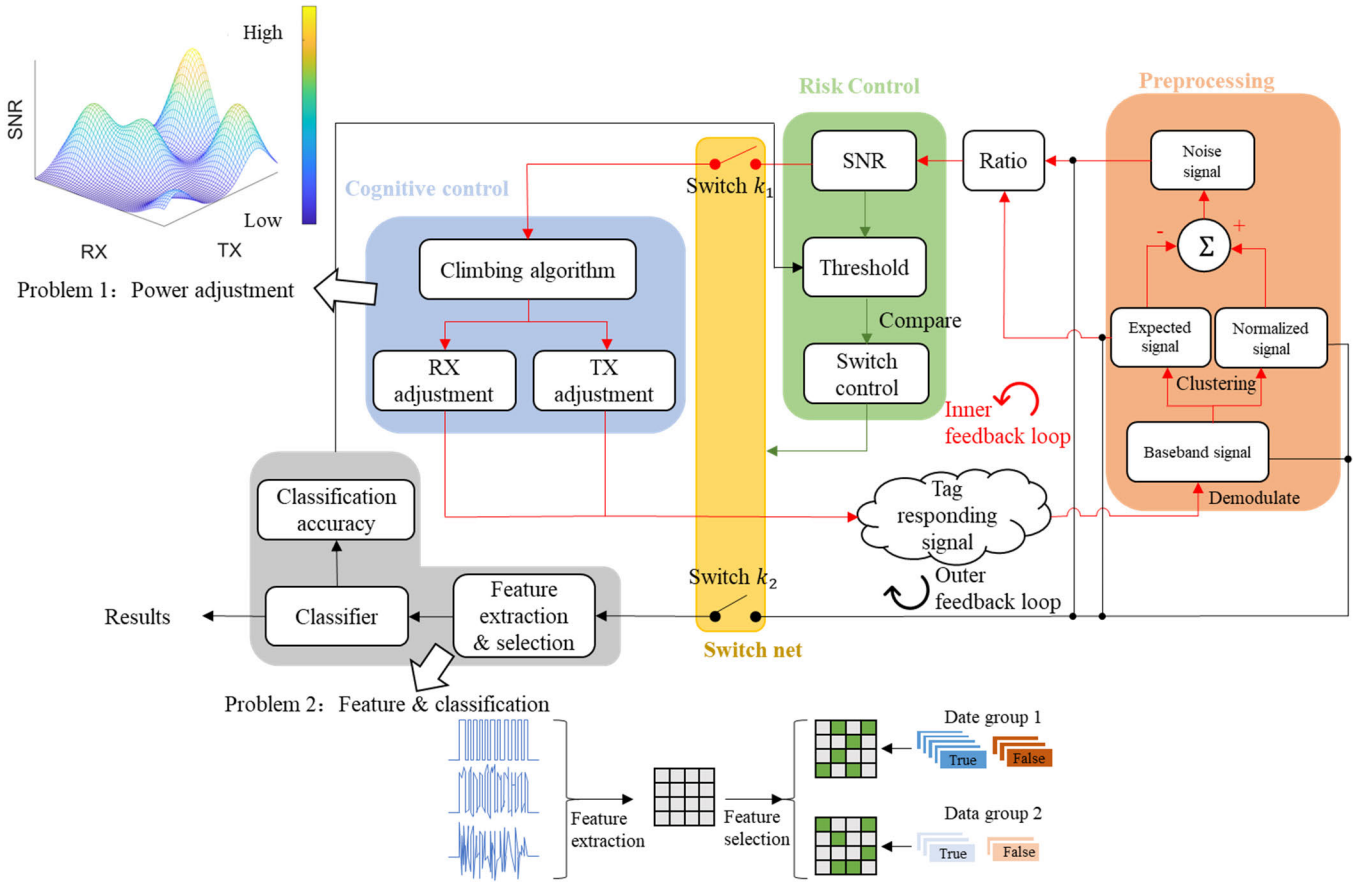


Fig. 1. CRC for tag classification.

physical features from the signal, such as the reflection coefficient [27] and measured distance [28]. Another approach focuses on extracting time- and frequency-domain features, including signal fingerprints [10], [11], phase offsets [12], frequency offsets [29], and higher order statistics [30]. Whether extracting physical features or time- and frequency-domain features, the classification is based on distinguishing the differences between authentic and counterfeit tags. However, if the tag response signals are heavily contaminated with noise, the extracted features may become inaccurate, thereby degrading the classification performance.

B. CRC and Feature Selection

CRC, initially applied in cognitive radio, cognitive radar, and autonomous driving [13], proves to be an effective method for enhancing system robustness and communication quality. In this article, we aim to apply CRC to enhance RFID communication security. In the CRC of cognitive radar, it uses Kalman filtering to estimate the hidden signal, calculates entropy through the selected waveform parameters, obtains rewards, and then maximizes the rewards. The maximization problem is Bellman optimization, and Q-learning is used to solve it. The purpose of the CRC is to make the error of the estimated signal smaller, that is, to improve the tracking accuracy. The CRC in this article is used for tag anticounterfeiting. It first uses IQ demodulation and clustering decoding algorithms to obtain the transmitted signal and noise signal. The SNR is calculated through the obtained noise signal, so that the SNR reaches or exceeds the threshold, and a search method is used to achieve it. Since the CRC in this article is to obtain better anticounterfeiting, it also involves classification, feature extraction, as well as training and testing modules. Additionally, traditional physical-layer identification methods typically consider only a limited number of features for tag classification. However, the characteristics exhibited by tag signals are often diverse, and limited features can only capture a subset of the signal characteristics. The findings from feature selection studies [30], [35], [36], [37] indicate that utilizing only a few fixed features for tag classification results in suboptimal accuracy. Therefore, this article aims to incorporate as many features as possible, encompassing over 100 time- and frequency-domain features, in order to better capture the variations and distinctions between tags.

III. Problem Description

This article employs CRC to address the challenge of low tag classification accuracy under low SNR, as illustrated in Fig. 1. The fundamental concept is to initiate the classification process only when the SNR surpasses a predefined threshold. Otherwise, reader's transmit or receive power (TX or RX) is adjusted to enhance the SNR. The workflow begins with preprocessing the tag's responding signal to obtain the baseband signal and the expected signal. By subtracting the expected signal from the baseband signal, the noise signal

is derived. The SNR is then calculated as the ratio of the noise signal power to the expected signal power. Next, the risk control module comes into play. The obtained SNR is compared with the threshold. If it falls below the threshold, the switch is directed to the CRC module. Conversely, once the SNR reaches the threshold, the switch shifts to the feature and classification module. In the CRC module, a climbing search algorithm is employed to adjust the TX or RX. The CRC and preprocessing modules constitute an inner feedback loop, which terminates once the SNR exceeds the threshold. Within the classification module, feature extraction and feature selection are conducted on the baseband, expected, noise, and normalized signals. The selected features are then fed into the classifier for authentic-false classification. Furthermore, the threshold in the risk control module is adjusted based on the training results obtained during classification. The preprocessing, classification, and risk control modules together form an outer feedback loop, which concludes when the adjusted threshold achieves a high classification accuracy.

In the CRC discussed above, two key problems need to be addressed. The first problem pertains to the design of the CRC module. Various factors influence the SNR of tags, including the electromagnetic environment, tag hardware, and the distance between the tag and the reader. However, since the CRC is implemented at the RFID reader, it primarily relies on adjusting the TX and RX of the reader to modify the SNR [31]. Setting a high TX or RX power is not always feasible as it can lead to increased power consumption, waveform distortion in the amplifier, and decoding issues. Moreover, the sensitivity of tags may differ due to manufacturing variations, meaning that a fixed power level does not result in the same SNR scale for each tag. Therefore, dynamic power adjustment is a more practical approach. However, different combinations of TX and RX powers yield different SNR levels, as depicted in the top left corner of Fig. 1. Identifying a strategy to quickly determine the optimal power parameters to achieve a suitable SNR requires detailed exploration. This article will propose a fast, simple, and efficient search algorithm to tackle the problem.

The second problem revolves around the design of the feature and classification module, as shown at the bottom of Fig. 1. Traditional methods often rely on a limited number of fixed features to classify all types of tags, which may not be practical. The diversity among tags necessitates the consideration of their distinct characteristics, where certain features may be effective for specific tag groups but not for others. Therefore, maximizing the extraction of features may prove more effective. This article not only extracts time-domain features from the tags but also incorporates frequency-domain features. Additionally, feature selection is employed to evaluate the newly extracted features and determine their effectiveness.

IV. Cognitive Risk Control for Tag Classification

A. Preprocessing

The framework of CRC applied to RFID tag classification is shown in Fig. 1, and this section will introduce each module in detail. The tag-responding signal received by the reader first passes through the preprocessing module. The first function of the module is to calculate the SNR of the signal to control the switch network. The second one is to preprocess the tag signal for subsequent feature extraction.

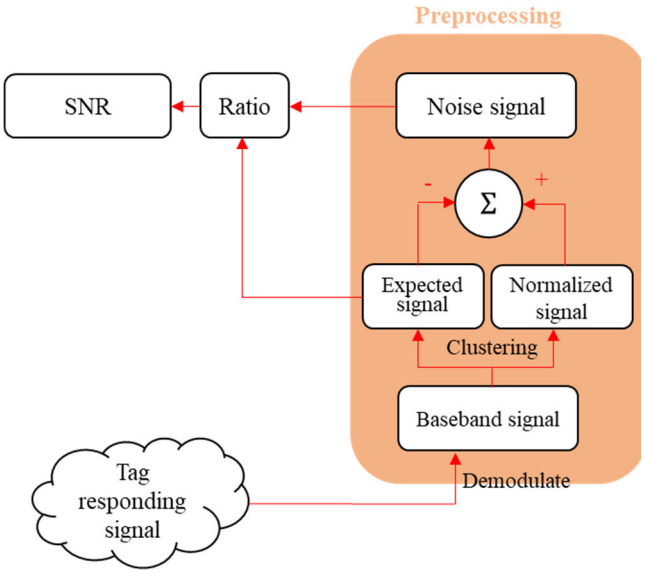


Fig. 2. Preprocessing.

The preprocessing module is shown in Fig. 2. First, the responding signal is IQ demodulated [32] to obtain the I- and Q-channel signal, and further obtains the baseband signal $a(n)$ after modulo, where $n = 1, 2, \ldots, N$ are the sampling points. Then, the expected signal will be

$$a_e(n) = \text{dec}[a(n)] \tag{1}$$

through a decision for the baseband signals. The decision is denoted as follows:

$$\text{dec}(x) = \begin{cases} 0, & \text{if } |x - v_0| \le |x - v_1| \\ 1, & \text{if } |x - v_0| > |x - v_1| \end{cases} \tag{2}$$

where v_0 and v_1 are two clustering centers of the baseband signal $a(n)$, corresponding to bits 0 and 1. Note that v_0 is determined via the point close to the clustering center of the silent period signal [30], also shown in Fig. 7. To remove the difference between the signal powers, the baseband signal will be normalized to

$$a_n(n) = \frac{a(n) - v_1}{v_1 - v_0}. \tag{3}$$

Subtracting the expected signal from the normalized signal will have the noise signal

$$a_\eta(n) = a_n(n) - a_e(n). \tag{4}$$

After the above processing, we will get four groups of signals, expected signal $a_e(n)$, normalized signal $a_n(n)$, noise signal $a_\eta(n)$, and baseband signal $a(n)$,and features will be extracted from the signals. Finally, SNR can be calculated via

$$\text{SNR} = 10\lg\frac{P_e}{P_\eta} \tag{5}$$

where P_e and P_η are the average powers calculated from (1) and (4), respectively.

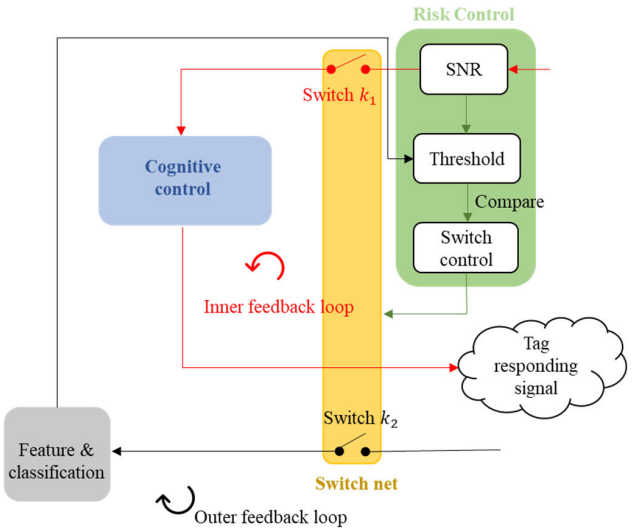


Fig. 3. Risk control and switch network.

B. Risk Control and Switch Network

The function of the risk control module is to control the switch network, as shown in Fig. 3. If the SNR does not reach the threshold, the system turns to the cognitive control module. In this case, it is an inner feedback loop, where the SNR is adjusted until it is greater than or equal to the threshold. If the SNR does, the system turns to the feature and classification. The above process is expressed as follows:

$$\langle k_1, k_2 \rangle = \begin{cases} \langle \text{on, off} \rangle, & \text{if SNR} < V_{\text{th}} \\ \langle \text{off, on} \rangle, & \text{if SNR} \geq V_{\text{th}} \end{cases} \tag{6}$$

where k_1 and k_2 are switches for the system to switch to the cognitive control and feature and classification module, respectively, "ON" means closed and "OFF" means open. V_{th} represents the SNR threshold, which is determined by the optimal tag classification accuracy of the classification module, expressed as follows:

$$V_{\text{th}} = \arg\max_{\text{SNR}} f_c(\text{SNR}) \tag{7}$$

where $f_c(\text{SNR})$ is expressed as a classification accuracy function determined by SNR, which means that if the classification accuracy is optimal, the corresponding SNR is the required threshold. Equation (7) is the purpose of the outer feedback loop, which can be completed through training data.

C. Cognitive Control

If SNR does not reach the threshold, it turns to cognitive control, and switch k_1 is closed, as shown in Fig. 4. The purpose of cognitive control is to make SNR reach or exceed the threshold by adjusting the TX and RX when the SNR does not meet the requirement. Fig. 5 shows the schematic of the target SNR search. The grids represent SNR values by the TX and RX coordinates. The green grid represents the coordinates that meet the required SNR, the white grids represent the coordinates that do not meet the requirements, and the blue grids represent the current coordinate. According to the increase, decrease, or no change of the TX and RX, the actions have eight directions, up, down, left, right, upper left, lower left, upper right, and lower right. From the CRC,

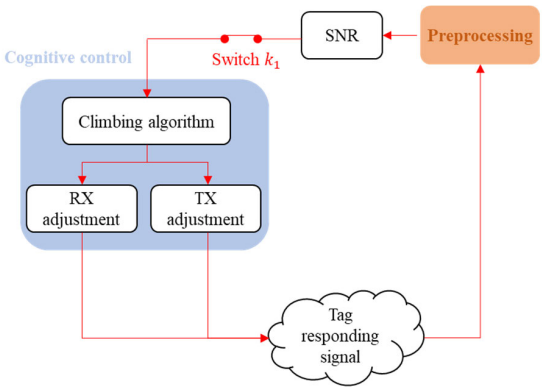


Fig. 4. Cognitive control.

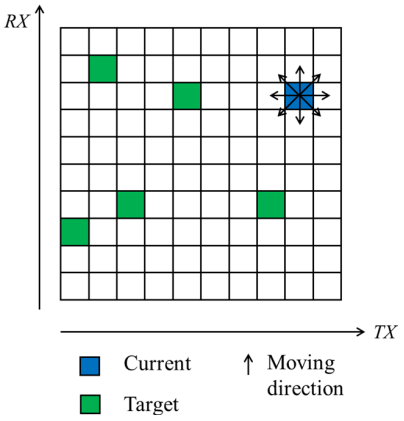


Fig. 5. Target SNR search.

which action to use to find the target TX and RX is a Bellman dynamic programming [13], and it is similar to path planning. However, if Bellman programming is used, multiple actions need to be tried repeatedly to find the optimal value. The reader needs to set TX and RX multiple times, which will increase the adjustments and is not efficient. For this, we change the Bellman programming into a target search, trying to find the target SNR with as few TXs and RXs as possible. As shown in problem 1 in Fig. 1, the initial state may be in the "valley" of SNR. Finding the target SNR is like climbing from the "valley" to "the top of hill." Therefore, the search is called the climbing algorithm. Note that there is no case in which the local maximum has not yet reached the threshold in this problem, so there is no need to consider falling into a local optimum. As long as the value is gradually increased at each step, the target can be reached in the end. Below, we elaborate on the algorithm.

When the $t+1$th action a_{t+1} is applied, the original TX and RX state vector $\mathbf{S}_t = [\text{TX}_t, \text{RX}_t]^{\text{T}}$ will be updated to $\mathbf{S}_{t+1}$. Because the process is a Markov process, i.e. $\mathbf{S}_{t+1}$ determined only by $\mathbf{S}_t$, the process can be expressed as follows:

$$\mathbf{S}_{t+1} = \mathbf{S}_t + \mathbf{a}_{t+1}, \mathbf{a}_{t+1} \in \mathcal{A} \tag{8}$$

where $\mathcal{A}$ is an action set, including eight actions such as increasing, decreasing, or unchanging TX_t and RX_t, expressed as follows:

$$\mathcal{A} = \{[\delta, 0]^{\text{T}}, [-\delta, 0]^{\text{T}}, [0, -\delta]^{\text{T}}, [0, \delta]^{\text{T}}, [\delta, -\delta]^{\text{T}}, [-\delta, \delta]^{\text{T}}, [\delta, \delta]^{\text{T}}, [-\delta, -\delta]^{\text{T}}\} \tag{9}$$

in which δ represents the step size of the power adjustment.

Let SNR be updated by $\mathrm{SNR}_{t+1} = Q(\mathbf{S}_t, \Im_t)$,that is, the SNR is related to not only the power $\mathbf{S}_t$, but also the environmental factors $\Im_t$, such as reading distance, and electromagnetic environment. If the time is short during the adjustment of $t = 1, 2, \ldots, T$, it can be assumed that $\Im_t$ is time-invariant. After omitting subscript t, SNR_{t+1} is

$$\mathrm{SNR}_{t+1} = Q(\mathbf{S}_t, \Im). \tag{10}$$

We specify that this climbing algorithm performs the following actions. If the updated SNR is greater than the previous one and the threshold is not reached, then the search direction is correct and is maintained. If the updated one is not, then the direction is incorrect and another action needs to be randomly selected from the action set. If the updated one is greater than or equal to the threshold, no actions need to be taken and the search stops. The above search is expressed as follows:

$$\mathbf{a}_{t+1} = \begin{cases} \mathbf{a}_t, & \text{if } \mathrm{SNR}_t \le \mathrm{SNR}_{t+1} < V_{\mathrm{th}} \\ \mathbf{a}_{\mathrm{rnd}}, & \text{if } \mathrm{SNR}_{t+1} < \mathrm{SNR}_t \\ \emptyset, & \text{if } \mathrm{SNR}_{t+1} \ge V_{\mathrm{th}} \end{cases} \tag{11}$$

where $\mathbf{a}_{\mathrm{rnd}} \in \mathcal{A}$ is a random action and $\mathbf{a}_{\mathrm{rnd}} \neq \mathbf{a}_t$,which means that the selected action cannot be the same as the action such that $\mathrm{SNR}_{t+1} < \mathrm{SNR}_t$.

In addition, as shown in Fig. 5, the number of climbing searches is related to the initial point, and an initial point close to the target can reduce the number of searches. This algorithm adopts cross-validation to determine the initial point. Accumulates the SNRs from the possible TX and RX combination in the training set, and the combination for the maximum accumulated value will be used as the initial point in the test set, expressed as follows:

$$\mathbf{S}_0 = \arg\max_{\mathbf{S}} \sum_i Q_i(\mathbf{S}, \Im) \tag{12}$$

where i represents the ith tag in the test set. In addition, the environmental factor $\Im$ will also have an impact on the final SNR, and it should be ensured that the test set is consistent with the training set; otherwise, there will be model generalization problems. Therefore, training can be carried out in various environments $\Im$. For example, train the initial points of far, medium, and near reading distances respectively, and select the corresponding training initial points according to the actual distance during testing.

Finally, Table I gives the steps of climbing algorithms.

TABLE I
STEPS OF CLIMBING ALGORITHM

Input:
The tth transmit and receive power vector $\mathbf{S}_t$, SNR_t and action $\mathbf{a}_t$
Output:
Updated SNR_{t+1}
Initialized:
$\mathbf{S}_0$ from (12), SNR_0 from (5) and $\mathbf{a}_0$ randomly from (9)
Known:
V_{th} is from (7)
Steps:
① Update SNR_{t+1} via (10)
②From (11)
if $SNR_t \le SNR_{t+1} < V_{\mathrm{th}}$, $\mathbf{a}_{t+1} = \mathbf{a}_t$, go to ③
if $SNR_{t+1} < SNR_t$, $\mathbf{a}_{t+1} = \mathbf{a}_{\mathrm{rnd}}$, go to ③
if $SNR_{t+1} \ge V_{\mathrm{th}}$, end
③Update $\mathbf{S}_{t+1}$ via (8), $t = t + 1$, and go to ①

D. Feature and Classification

The raw tag responding signal is preprocessed to obtain four groups of signals, i.e., baseband, expected, normalized, and noise signals. From them, we can extract the features, such as mean, variance, second center distance, maximum autocorrelation, Shannon entropy, skewness, kurtosis, and other traditional time domain statistics [11], [30]. Besides, some frequency-domain features can be extracted, such as center of gravity frequency, mean square frequency, root-mean-square frequency, frequency standard deviation, and spectral kurtosis [33]. Finally, we also extract other time-domain features such as peak-to-peak, rectified mean, root mean square, form factor, crest factor, pulse factor, and margin factor. Tables II and III gives the details for the above features. Through the features, 26 features are extracted for each group of signals. For four groups of signals, a total of 104 features are extracted.

TABLE II
EXTRACTED TIME-DOMAIN FEATURES

	Features	Description
1	Mean	$\mu = \Sigma_{n=1}^{N} z(n)/N$
2	Variance	$\sigma^2 = \Sigma_{n=1}^{N}[z(n)-\mu]^2/(N-1)$
3	Standard deviation	$\sigma = \Sigma_{n=1}^{N}\lvert z(n)-\mu\rvert/(N-1)$
4	Second center distance	$D = \Sigma_{n=1}^{N}[z(n)-\mu]^2/N$
5	Maximum autocorrelation	$Ra = \max_{\tau} \Sigma_{n=1}^{N} z(n)z(n+\tau)$
6	Shannon entropy	$H = -P(Z_i)\Sigma_i \log_2 P(Z_i)$
7	Skewness	$S = \Sigma_{n=1}^{N}[z(n)-\mu]^3/\sigma^3 N$
8	Kurtosis	$K = \Sigma_{n=1}^{N}[z(n)-\mu]^4/\sigma^4 N$
9	Maximum value	$MAX = \max\{z(n)\}$
10	minimum value	$MIN = \min\{z(n)\}$
11	Peak-to-peak	$PPV = MAX - MIN$
12	Rectified average	$ARV = \Sigma_{n=1}^{N}\lvert z(n)\rvert/N$
13	RMS	$RMS = [\Sigma_{n=1}^{N} z^2(n)/N]^{1/2}$
14	Form factor	$FF = RMS/ARV$
15	Crest factor	$PAR = PPV/ARV$
16	Impulse factor	$XI = PPV/ARV$
17	Margin factor	$XCL = PPV/ARV^2$

Note:
1. $z(n)$ is denoted as time-domain signals;
2. Z_i represents the i-th value after discretizing the time-domain signal $z(n)$.

After the features are extracted, it is necessary to verify which feature is effective for the classification of which group, and the feature selection method can be used. Feature selection can usually be divided into filtering, embedded, and wrapping methods. Since the performance of the filtering methods does not depend on the classifier, this article adopts the filtering feature selection.

Let a cell variable in a training set be $\chi = \langle \mathbf{X}, y \rangle$, which consists of the signal feature vector $\mathbf{X} = [x^{(1)}, x^{(2)}, \ldots, x^{(M)}]$ and its classification label y. Calculate the weight $\omega^{(m)}$ of each feature $x^{(m)}$ in the training set, and sort them. Select the W features with the largest weights, that is

$$\langle p_1, p_2, \ldots, p_W \rangle = \underset{m}{\mathrm{argmax}}\ \omega^{(m)}. \tag{13}$$

TABLE III
EXTRACTED FREQUENCY-DOMAIN FEATURES

	Features	Description
1	Gravity frequency	$FC = \Sigma_{n=1}^{N} f_n P(n) / \Sigma_{n=1}^{N} P(n)$
2	Frequency variance	$VF = \Sigma_{n=1}^{N} (f_n - FC)^2 P(n) / \Sigma_{n=1}^{N} P(n)$
3	Mean square standard deviation	$RVF = [\Sigma_{n=1}^{N} (f_n - FC)^2 P / \Sigma_{n=1}^{N} P(n)]^{1/2}$
4	Mean square frequency	$MSF = \Sigma_{n=1}^{N} f_n^2 P(n) / \Sigma_{n=1}^{N} P(n)$
5	RMS frequency	$PMSF = [\Sigma_{n=1}^{N} f_n^2 P(n) / \Sigma_{n=1}^{N} P(n)]^{1/2}$
6	Mean spectral kurtosis	$SKM = \Sigma_{f=1}^{M} SK(f)/M$
7	Spectral kurtosis standard deviation	$SKS = [\Sigma_{f=1}^{M} [SK(f) - SKM]/M]^{1/2}$
8	Spectral kurtosis skewness	$SKSK = \Sigma_{f=1}^{M} [SK(f) - SKM]^3 / SKS^3 M$
9	kurtosis of spectral kurtosis	$SKSU = \Sigma_{f=1}^{M} [SK(f) - SKM]^4 / SKS^4 M$

Note:
1. $P(n)$ represents the power spectrum of the n-th sampling point;
2. f_n represents the frequency at sampling point n;
3. $SK(f) = [\langle |X(n,f)|^4 \rangle_n / \langle |X(kP,f)|^2 \rangle_n] - 2$ is spectral kurtosis;
4. $X(n,f)$ is the power spectral density of the time-domain signal in the nth window and frequency f after short-time Fourier transform;
5. $\langle \cdot \rangle_n$ is the time averaging operator on the n-th window.

Constitute the selected features into a new cell $\chi^S = \langle \mathbf{X}^S, y \rangle$ and get a new training set S, so that it makes

$$\chi^S = \langle \mathbf{X}^S, y \rangle \in \mathcal{S} \tag{14}$$

satisfied, where $\mathbf{X}^S = [x^{(p_1)}, x^{(p_2)}, \ldots, x^{(p_W)}]$. Similarly, the cell in a test set $\chi^T = \langle \mathbf{X}^T, y \rangle$ can be obtained, and the test set $\mathcal{T}$ should make

$$\chi^T = \langle \mathbf{X}^T, y \rangle \in \mathcal{T} \tag{15}$$

satisfied, where $\mathbf{X}^T$ is a vector of W features with the largest weights. After feature selection, cross-validation can be performed. If weight w of the classifier $f_{\text{clas}}(\cdot)$ make

$$y = f_{\text{clas}}(w, \chi^{\text{S}}), \quad \chi^{\text{S}} \in \mathcal{S} \tag{16}$$

satisfied, the training is completed, and the test result is obtained from

$$\hat{y} = f_{\text{clas}}(w, \chi^{\text{T}}), \quad \chi^{\text{T}} \in \mathcal{T}. \tag{17}$$

Comparing test label $\hat{y}$ with expected label y, the classification accuracy can be obtained.

Finally, Table IV presents the steps for the CRC algorithm.

TABLE IV
STEPS FOR THIS CRC ALGORITHM

Input:
Baseband signal $a(n)$
Output:
Classification label $\hat{y}$
Known:
V_{th} determined via (8)
Steps:
① Calculate SNR from (1-6)
② From(7)
if $SNR < V_{\text{th}}$, go to ③
if $SNR \geq V_{\text{th}}$, go to ④
③ Perform climbing algorithm in Tabel1, and go to ④
④ Perform feature and classification from (16-20)

V. EXPERIMENTAL SETUP

A. Data Generation

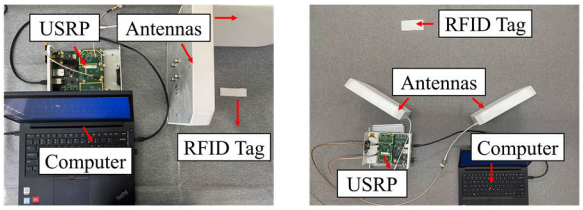
The experimental tags adopt the passive UHF tags specified by EPC C1 Gen2, and 140 tags of seven types that are common in the market are used. The seven types of tags are made by three manufacturers, and the details are shown in Table V. Write the same EPC code to all 140 tags before collecting data. The writer used is the UHF100U from Guangzhou Wangyuan Electronics Manufacturer. The writer parameters are shown in Table VI. Data collection is completed through a UHF RFID system [31] by a USRP software-defined radio. The system complies with the EPC C1 Gen2 standard, and the software is realized by GNU Radio. Table VII gives the detailed parameters. The code download address is https://github.com/nkargas/Gen2-UHF-RFID-Reader.

TABLE V
TAG TYPE AND MANUFACTURER

	Type	Manufacturer
1	Alien9640	Guangzhou Wang yuan Electronics
2	Alien9662	Shenzhen Qibao Technology
3	Alien9654	Shenzhen Qibao Technology
4	Alien9662	Guangzhou Wang yuan Electronics
5	Alien7017	Guangzhou Wang yuan Electronics
6	Alien9662	Nanjing Lejay Technology
7	Alien9654	Nanjing Lejay Technology

TABLE VI
PARAMETERS FOR RFID WRITER

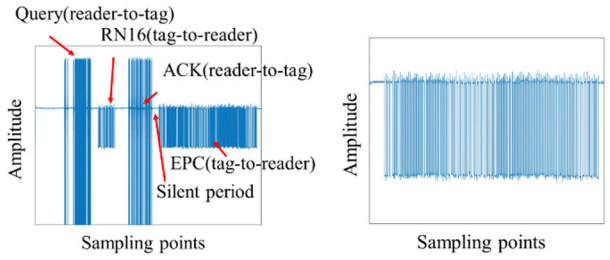
Parameters	Description
Model	UHF 100U
Frequency	865~868MHZ 902~928MHZ
Standard	EPC C1 Gen2
Distance	0-0.1m
Communication port	USB
Voltage	DC+5V
Power	4W



(a) (b)

Fig. 6. Devices for RFID tag signal collection: (a) scene I, 26 × 26 cm, which means that the distance from the RFID tag to both antennas is 26 cm and (b) scene II, 30 × 80 cm, which means that the distance from the RFID tag to two antennas is 30 and 80 cm, respectively.

For each data collection, only one tag is within the magnetic field of the reader, ensuring that no other tags are present to reduce the risk of collisions. All data collection was not performed in an isolated environment, which may include thermal noise, cell phone noise, wireless network, and radio-frequency (RF) noise and so on. The tags are randomly placed in an area formed by two antennas, shown in Fig. 6. Each tag records 10 s of data, and randomly segments an EPC signal with a silent period, shown in Fig. 7.



(a) (b)

Fig. 7. RFID tag responding signal: (a) raw baseband signal and (b) segmented EPC signal with a silent period.

TABLE VII
PARAMETERS FOR USRP SYSTEM

Parameter	Description
Motherboard	USRP N200
Daughter board	RXF900
Antenna	
Quantity	2
Type	Circularly polarized
Gain	7dBic
Distance	0.5-1.5m
Link frequency	40kHz
Maximum queries	1000
Encoding	FM0
Transmission power	17.8dBm
Emission amplitude	0.1
Sampling frequency	1000kHz

B. Algorithms and Classification

In this experiment, the CRC module is used to adjust the SNR of the received signal to meet the classification requirements. The parameters of the module are shown in Table VII. The following three algorithms are used in the module to search for the target SNR. The number of searches for the algorithms is compared, and fewer searches means less time is required for the reader to reach the target.

1) *Q-Learning:* A model-free Q-learning algorithm [34] where the learning rate is 0.1, the discount factor is 0.8, the greedy coefficient is 0.1, the initial reward is set to 100, the reward is -1 for each step, and it is 0 when the target value is reached.
2) *Random Search:* The values of TX and RX are randomly determined each time until the SNR $\geq V_{\text{th}}$.
3) *Climbing Search:* The proposed algorithm in Table I.

This experiment uses two kinds of cross-validations to obtain classification results. One is that the tags come from different types or different manufacturers, and the other is that the tags are all from the same type and the same manufacturer. The details are as follows.

1) Cross-validation of different types or different manufacturers (CrossVal A): Fivefold cross-validation is adopted, as shown in Fig. 8(a), where each type or manufacturer has 20 tags, and there are a total of $L = 7$ tag types or manufacturers. Let the lth type be true and the mth type be false where $l \neq m$, and form training sets $\mathcal{S}_l$ and $\mathcal{S}_m$ and test sets $\mathcal{T}_l$ and $\mathcal{T}_m$, respectively. Then, the classification accuracy of the lth type will be the average of the binary classification results for test set $\mathcal{T}_l$ and each $\mathcal{T}_m$, $m = 1, 2, \ldots, L$.
2) Cross-validation of the same type and manufacturer (CrossVal B): Fivefold cross-validation is used, as shown in Fig. 8(b), where each type has 20 tags. Randomly select a tag from the lth type, and independently collect 19 data from the tags as a true set, denoted by "1." Collect 19 data from the remaining 19 tags of the type, and use them as a false set, denoted by "0." The above two sets will be divided into training sets $\mathcal{S}_l^1$, $\mathcal{S}_l^0$ and test sets $\mathcal{T}_l^1$, $\mathcal{T}_l^0$. The classification accuracy of the l-type tag will be the result of the binary classification of test sets $\mathcal{T}_l^1$ and $\mathcal{T}_l^0$.

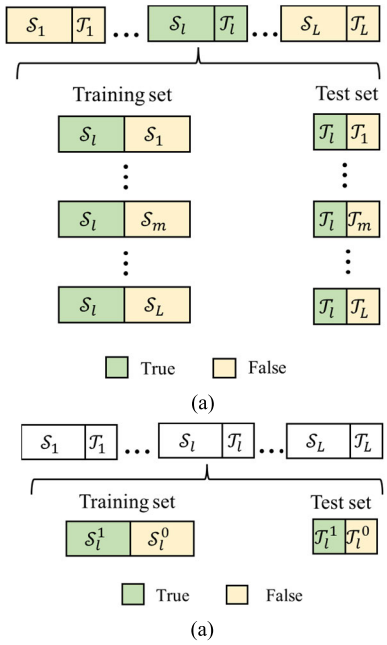


Fig. 8. Two kinds of cross-validation: (a) CrossVal A of cross-validation for tags from different types or different manufacturers and (b) Cross-Val B of cross-validation for tags from the same type and manufacturer.

In this experiment, it is also necessary to evaluate various factors that may affect the classification results, such as the feature selection, the number of selected features, the newly added time- and frequency-domain features, and different classifiers, as follows.

1) *CRC:* The method proposed in this article.
2) *Seven With SVM:* Extract the traditional seven time-domain features [35] from the EPC baseband signal of tags, without feature selection, and directly use a support vector machine (SVM) classifier.
3) *Twenty-Eighty With SVM:* Extract a total of 28 features from the baseband, normalized, expected and noise signal of the tag EPC [30], without feature selection, and directly use SVM classifier.
4) *One-Hundred-and-Four With SVM:* Add new time- and frequency-domain features, increase the number of features to 104, shown in Tables II and III, and directly use SVM classifier without feature selection;
5) *One-Hundred-and-Four With Relieff:* Use ReliefF feature selection [36] on the 104 features, select the features with a weight greater than 0, and use the SVM classifier.

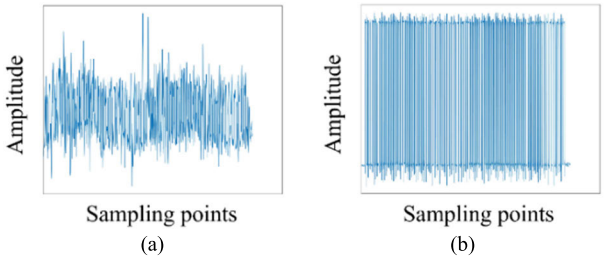


Fig. 9. Responding signals of the same tag before and after CRC adjustment: (a) TX and RX set randomly and (b) TX and RX set by CRC.

6) *Twenty-One/Forty-One/Sixty-One/Eighty-One/One-Hundred-and-One With Chi2:* Use chi-square test feature selection [37] on the 104 features, the number of features selected as $W = 21$, 41, 61, 81, and 101, and use SVM classifier.
7) *Twenty-One/Forty-One/Sixty-One/Eighty-One/ One-Hundred-and-One With Fsulaplacian:* Use Laplacian feature selection [38] on the 104 features, the number of features selected as $W = 21$, 41, 61, 81, and 101, and use SVM classifier.
8) *Seven/Twenty-Eight/One-Hundred-and-Four With RF:* Use the 104 features without feature selection and a random forest classifier [39] where the number of trees is selected as 50.
9) *Seven/Twenty-Eight/One-Hundred-and-Four With KNN:* Use the 104 features without feature selection and the K-nearest neighbors [40] where the number of neighbors is selected as 3.
10) *VGG16:* A deep network where the input uses the wavelet transform time-frequency distribution of the tag signal as the input image [11], [43], [44].

This experiment uses classification accuracy, acc to evaluate the classification performance, which is defined as follows:

$$\text{acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{18}$$

where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

VI. Experimental Results

A. Preprocessing Results

Fig. 9 shows the comparison of the responding signal of the same tag before and after being adjusted by CRC, in Scene I. (Note that Figs. 9–30 are all the results of Scene I. At the request of the reviewer, the results of Scene II, i.e., the long-distance scene are added and are given in Table IX.) It can be seen from the figure that when TX and RX are randomly selected, the upper envelope of the signal is not neat, showing a lower SNR. On the contrary, after the adjustment of CRC, the envelope of the received signal becomes tidy, showing the shape of a higher SNR. Fig. 10 shows the SNR heatmaps of three different tags. It can be seen from the figure that the SNR results from different TX and RX are different, and the heatmaps of different tags are also different. The purpose of CRC is to find the target SNR with as few TXs and RXs as possible, where the SNR is calculated from

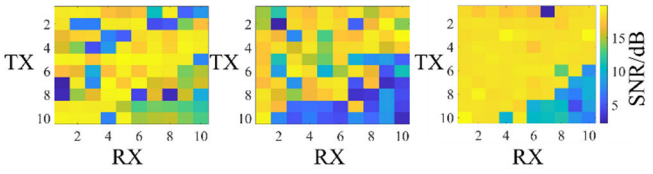


Fig. 10. SNR heatmaps of three different tags: (a) raw baseband signal and (b) segmented EPC signal with a silent period.

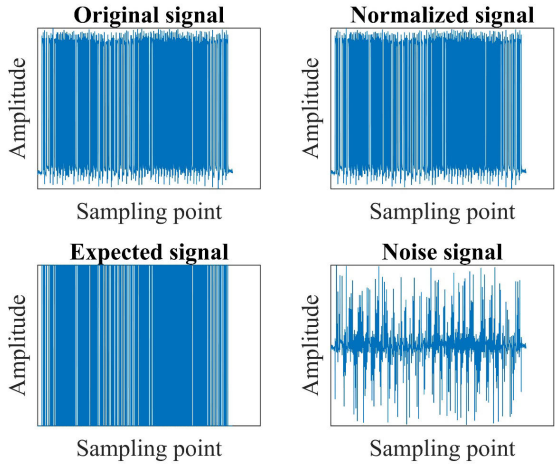


Fig. 11. Baseband, standard, expected, and noise signal of a tag after preprocessed.

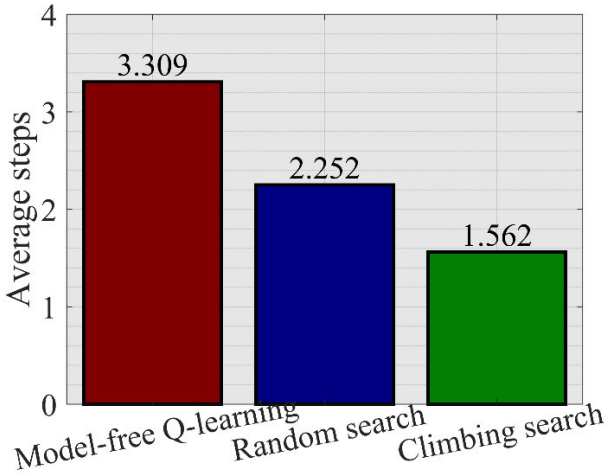


Fig. 12. Average number of searches of the three algorithms.

the expected signal power and the noise signal power, seen in (5). Fig. 11 shows the expected signal and noise signal after preprocessed.

Fig. 12 shows the number of searches for the three algorithms of Q-learning, random search, and climbing search, that is, the number of adjustment for TX and RX to reach the target SNR. It can be seen from the figure that the proposed climbing search takes the least number of searches, less than 2. It should be noted that the results in Fig. 12 are average results, and different tags will have different search results because of their different sensitivities. For example, if the tag in the third figure in Fig. 10 is used, the number of searches will be less because more grids meet the requirement; if the tag in the second figure in Fig. 10 is used, the number of searches will increase.

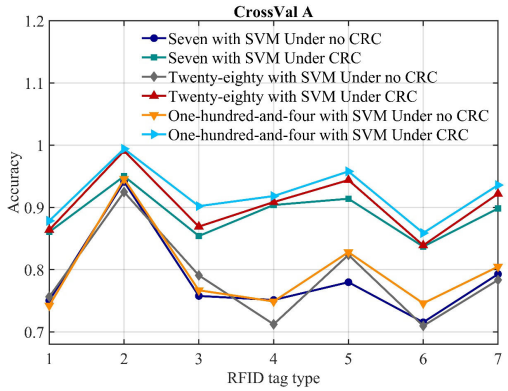


Fig. 13. Classification curves of SVM with and without CRC under CrossVal A.

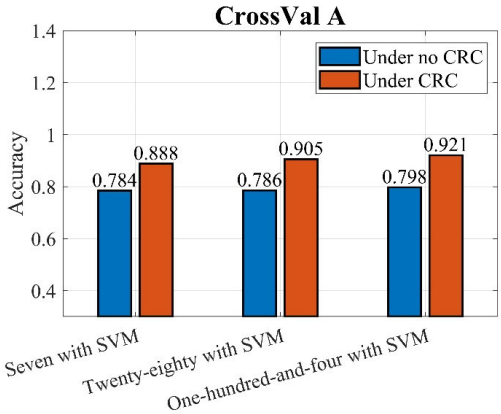


Fig. 14. Average classification accuracy of SVM with and without CRC under CrossVal A.

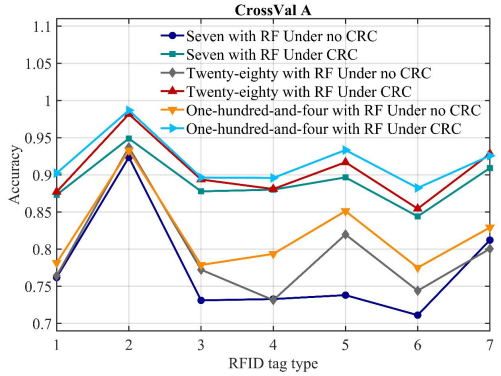


Fig. 15. Classification accuracy curves of RF with and without CRC under CrossVal A.

B. CrossVal A

This section presents the cross-validation results for CrossVal A. In this cross-validation, it mainly evaluates the classified tags from different types or different manufacturers. Fig. 13 shows the SVM classification accuracy under CrossVal A with and without CRC, and the number of the extracted features is 7, 28, and 104, respectively. From the figure, after adopting CRC, the classification accuracy has been improved, regardless of the number of features. Moreover, 104 features are better than 28 features, and 28 features are better than seven features, that is, the more the number of features, the higher the classification accuracy. Fig. 14 shows the average classification accuracy. Similar to Fig. 13, the accuracy with CRC is higher than that without it.

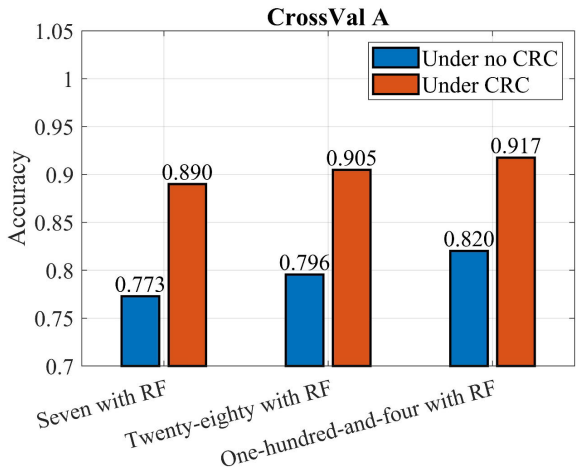


Fig. 16. Average classification accuracy of RF with and without CRC under CrossVal A.

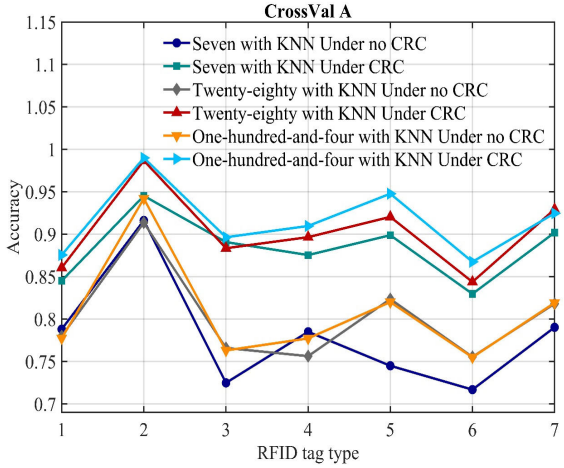


Fig. 17. Classification accuracy curves of KNN with and without CRC under CrossVal A.

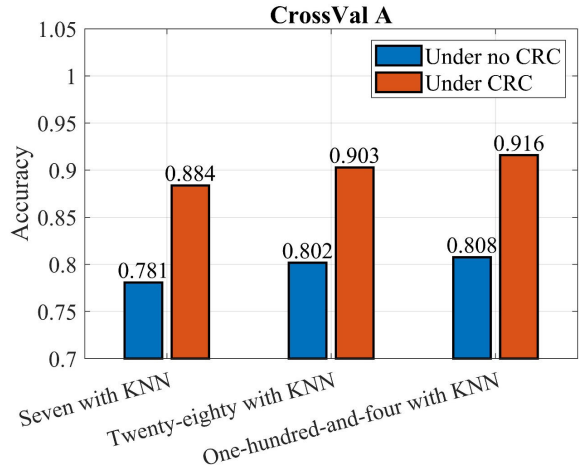


Fig. 18. Average classification accuracy of KNN with and without CRC under CrossVal A.

Figs. 15–18 show the classification results using RF and KNN, respectively. It can be seen from the results that after adopting CRC, although different classifiers are used, the classification accuracy has been significantly improved. Therefore, the improvement of classification performance by the proposed CRC method is not because of the classifier. Besides, for any classifier, more features have higher classification accuracy.

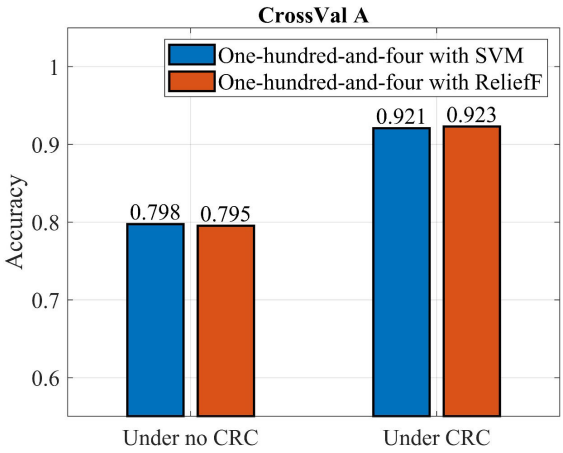


Fig. 19. Average classification accuracy of SVM with ReliefF and without feature selection under CrossVal A, where CRC and no CRC are considered, respectively.

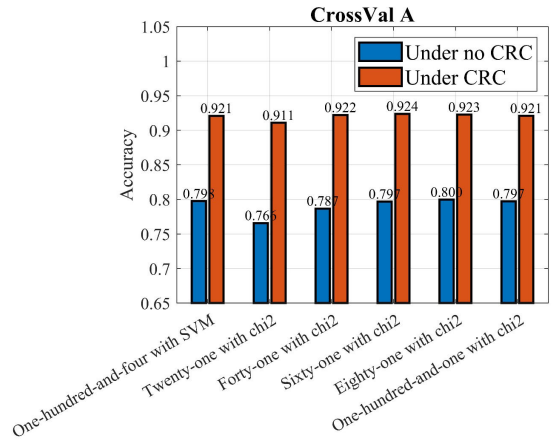


Fig. 20. Average classification accuracy of SVM with chi2 and without feature selection under CrossVal A, where CRC and no CRC are considered, respectively.

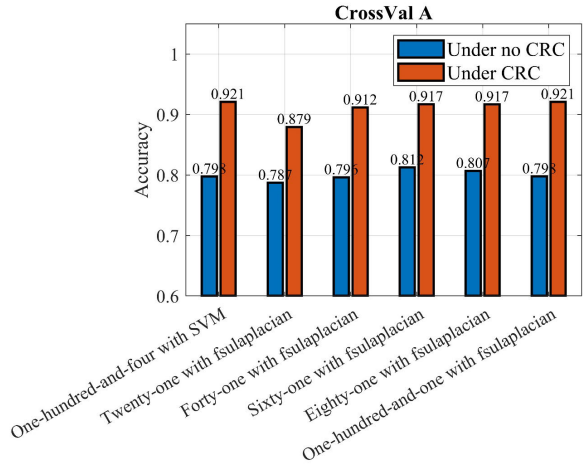


Fig. 21. Average classification accuracy of SVM with fsulaplacian and without feature selection under CrossVal A, where CRC and no CRC are considered, respectively.

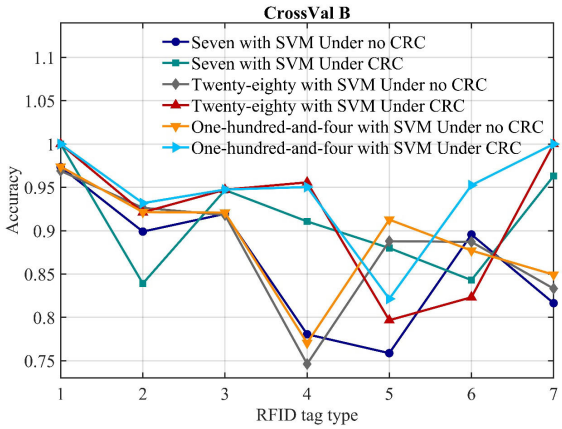


Fig. 22. Classification accuracy curves of SVM with and without CRC under CrossVal B, where the number of the extracted features is 7, 28, and 104, respectively.

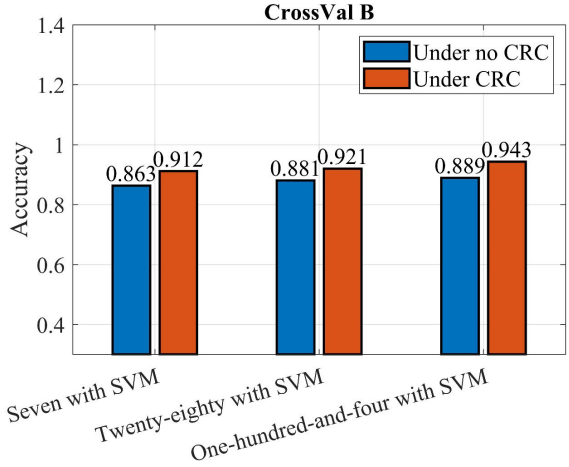


Fig. 23. Average classification accuracy of SVM with and without CRC under CrossVal B, where the number of the extracted features is 7, 28, and 104, respectively.

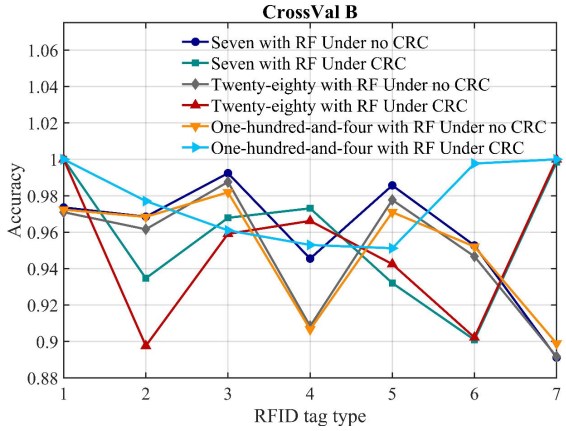


Fig. 24. Classification accuracy curves of RF with and without CRC under CrossVal B, where the number of the extracted features is 7, 28, and 104, respectively.

Figs. 19–21 shows the comparison between the classification accuracy with and without feature selection when 104 features are used as SVM input. The feature selection methods are ReliefF, chi2, and fuslaplasian, respectively. It can be seen from the figure that no matter whether feature selection is used or what kind of feature selection is used, there is no significant impact on the classification accuracy of the 104 features. Moreover, no matter whether CRC is used or not, it can be observed that the feature selection has no significant impact on 104 features. The result is not the same as that of [27], where when 28 features are used in this article, the performance with the feature selection will be better than that

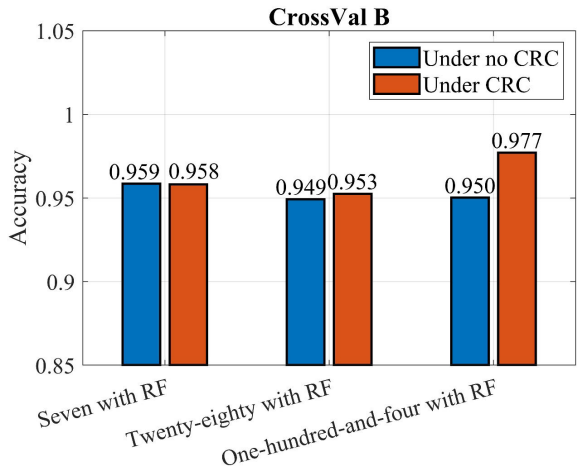


Fig. 25. Average classification accuracy of RF with and without CRC under CrossVal B, where the number of the extracted features is 7, 28, and 104, respectively.

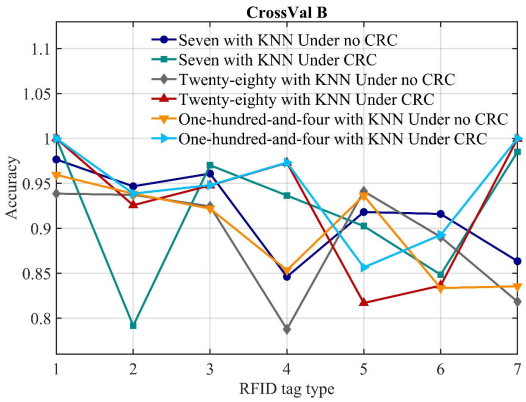


Fig. 26. Classification accuracy curves of KNN with and without CRC under CrossVal B, where the number of the extracted features is 7, 28, and 104, respectively.

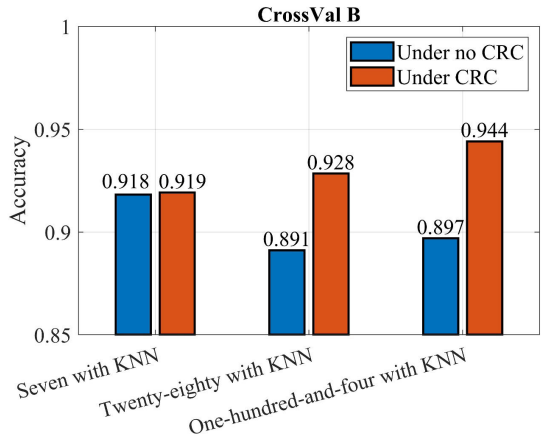


Fig. 27. Average classification accuracy of KNN with and without CRC under CrossVal B, where the number of the extracted features is 7, 28, and 104, respectively.

without it, and the difference in the number of the selected features will also produce different performance.

C. CrossVal B

This section presents the cross-validation results for CrossVal B. In this cross-validation, it mainly evaluates the classified tags from the same type and the same manufacturer.

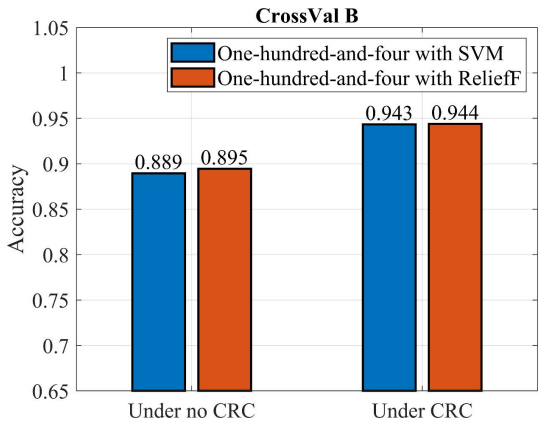


Fig. 28. Average classification accuracy of SVMs without feature selection and with ReliefF feature selection under CrossVal B, where CRC and no CRC are considered, respectively.

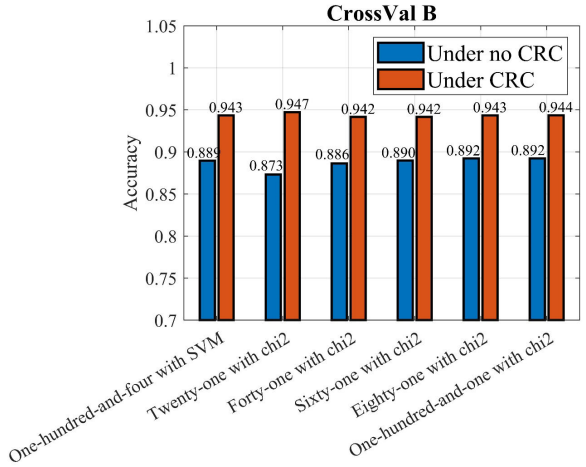


Fig. 29. Average classification accuracy of SVM without feature selection and with chi2 feature selection under CrossVal B, where CRC and no CRC are considered, respectively.

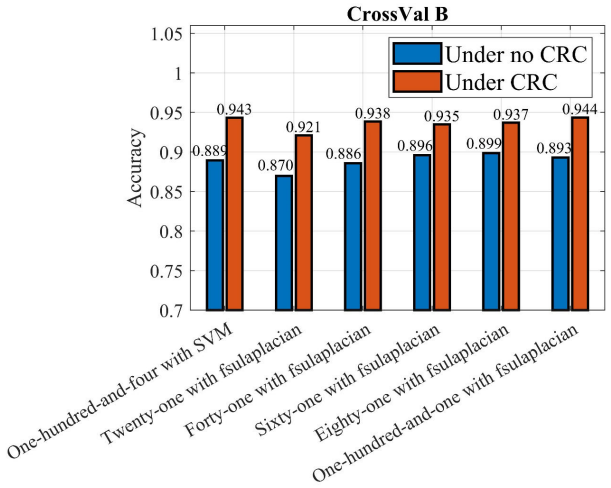


Fig. 30. Average classification accuracy of SVM without and with fsulaplacian feature selection under CrossVal B, where CRC and no CRC are considered, respectively.

Figs. 22–27 shows the classification accuracy results using SVM, RF, and KNN. Similar to CrossVal A, no matter what kind of classifier is used, after adopting CRC, the average classification accuracy is improved, although the improvement is not as large as that of CrossVal A. In addition, the classification accuracy is still related to the number of features. When

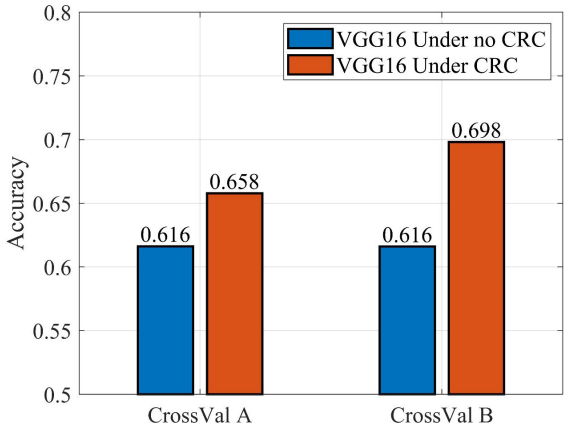


Fig. 31. Average classification accuracy of VGG16 under CrossVal A and CrossVal B, where CRC and no CRC are considered, respectively.

TABLE VIII
PARAMETERS FOR CRC

Parameters	Value
transmit power gain, TX	$1 \leq TX \leq 10$
receive power gain, RX	$1 \leq RX \leq 10$
SNR threshold, V_{th}	17.5dB
Step size, δ	1

104 features are used as the classifier input, it has the highest classification accuracy, which is higher than seven features and 28 features. When RF is used as the classifier, the average classification accuracy reaches 97.7%. Besides, note that the classification accuracy in tag types 5 or 6 under no CRC is even better than that with CRC in Fig. 22. The reason will be further discussed in Section VII.

Figs. 28–30 shows the comparison between the classification accuracy with and without feature selection when 104 features are used as SVM input under CrossVal B. The feature selection methods are ReliefF, chi2, and fuslaplasian, respectively. It can be seen from the figure that no matter whether feature selection is used or what kind of feature selection is used, there is no significant impact on the classification accuracy of the 104 features. Like Figs. 19–21, moreover, the results that feature selection has no significant effect on 104 features can be observed in the results with and without CRC. Therefore, it is unnecessary for the feature selection, and we can circumvent the model generalization.

In addition, we give the results of deep learning, VGG16 as shown in Fig. 31, but the classification accuracy is not high. The possible reason is that deep learning generally requires large-scale data, and the tag data in this article are not large. However, it should be seen that the classification accuracy of VGG16 using CRC is still higher than that of no CRC. Finally, to evaluate the impact of the number of features on computing complexity, we tested the algorithm running time on a Legion 7000P2021 laptop, and the program code was run in MATLAB2021A loaded on the Windows 11 system. The results are shown in Table X. As can be seen from the table, although the running time of the algorithm with 104 features is the longest, the added time is only about 10^{-1} s.

VII. DISCUSSION AND CONCLUSION

As a technology for anticounterfeiting RFID tags, physical-layer identification has gained wide popularity due to its low cost and ease of implementation. However, existing physical-layer identification techniques typically operate in an environment with high SNR, enabling accurate classification of different tags based on extracted features. In contrast, in some environments where the reading distance is far or signal power is low, the SNR of tags is no longer regulated. In the presence of high-noise power, the extracted features may become contaminated, thereby negatively impacting tag classification performance. To address the challenge, this article proposes the use of CRC in tag classification to enhance the SNR. By adjusting the reader's TX and RX, the CRC approach aims to improve the classification accuracy of tags under varying environmental conditions.

TABLE IX
CLASSIFICATION ACCURACY IN TWO DIFFERENT SCENES

	Algorithms	Accuracy
Scene I	Seven with SVM Under CRC	88.8%
	Twenty-eighty with SVM Under CRC	90.5%
	One-hundred-and-four with SVM Under CRC	92.1%
Scene II	Seven with SVM Under CRC	86.5%
	Twenty-eighty with SVM Under CRC	89.1%
	One-hundred-and-four with SVM Under CRC	90.7%

TABLE X
RUNNING TIME FOR DIFFERENT FEATURES

Algorithm	Average time
Seven with SVM	24.4336s
Twenty-eighty with SVM	24.4756s
One-hundred-and-four with SVM	24.3201s

In this experiment, we use a USRP to test the scheme proposed in this article. We first evaluate the number of adjustments, i.e., the number of searches to reach the target SNR. Calculate the SNR of the tag responding signal through the USRP device, and adjust the TX and RX if the SNR does not reach the target. During the adjustment, it is necessary to consider how to quickly reach the target. If the adjustment time is too long, the efficiency will be low. Traditional CRC regards the adjustment as a Bellman dynamic program and uses the Q-algorithm for it. However, the essence of the Q-algorithm is to find the action with the largest reward from an action set. For this software radio system, it means that more receiving and transmitting powers will be tried, which increases the number of searches. In this article, the adjustment is regarded as an optimization problem, and the target is searched by the climbing algorithm. Of course, the algorithm may fall into a local optimum, but in this system, the local maximum can meet the target, which can be seen from the heatmaps of the SNR and power. In addition, to further reduce the number of searches, we set an initial power by pretraining, which ensures that the initial value can be closer to the target and thereby reduce the number of searches. The experimental results show that the average number of searches of the proposed climbing algorithm is about 1.6, which is less than 2.3 of random searches and 3.3 of Q-algorithm.

In the CRC scheme, another important aspect to evaluate is whether tag classification accuracy is enhanced after CRC. Given the enhanced SNR and more accurate feature extraction, the answer appears to be evident. The classification results for tags from different types or different manufacturers align

with expectations, with a classification accuracy approximately 12% higher when CRC is utilized. The improvement is observed in SVM, RF, and KNN classifiers. However, when classified tags are from the same type and the same manufacturer, the results differ. On average, the classification accuracy with CRC is only increased by approximately 4%. The limited improvement may be attributed to the already relatively high classification accuracy for tags of the same type and manufacturer, as reported in [35], [41], and [42]. In this experiment, it is evident that even without CRC, the average classification accuracy has exceeded 90%. Therefore, the adoption of CRC does not lead to significant improvements. For instance, when using the random forest classifier, the classification accuracy before and after CRC is only increased from 95.0% to 97.7%. Consequently, the impact of CRC on the classification of tags of the same type is not as substantial as that on the identification of different types.

In addition to addressing the lower SNR environment of tag classification, this article delves into the issue of tag signal feature extraction. We extend the feature extraction to more frequency-domain features, such as spectral kurtosis. Moreover, to further increase the features, time-domain features like peaks and pulses are also extracted. The added time- and frequency-domain features are extracted from the raw baseband signal, expected signal, noise signal, and normalized signal, resulting in a total of 104 features. Theoretically, extracting a greater number of features enables a more comprehensive representation of the tag signal's characteristics, thereby facilitating the classification between different tags. The experimental results demonstrate that in the classification of different types of tags, the average classification accuracy with the proposed 104 features is approximately 2% higher than that with the traditional 28 features and approximately 4% higher than that with the traditional seven features. Likewise, in the classification of tags from the same type, the average classification accuracy with 104 features is also two percentage points higher than the traditional 28 and seven features.

Another interesting result is that when the number of features increases to 104, the sensitivity of the tag classification to feature selection will decrease. In the results of the different types of tag classification, no matter whether feature selection is used or the number of selected features is used, the classification accuracy does not change much. The possible explanation is that when the features are greatly increased from 7 or 28 to 104, there may be some correlation or overlap between different features, and the information they provide may be similar. Therefore, even if some features are not selected, the remaining features still contain similar information, resulting in no significant change in classification accuracy. The benefit of the result is that the proposed method can circumvent the optimal number of feature selections. As we know, different numbers of selected features will bring different classification accuracy. In particular, some seemingly optimal number of selected features may also cause over-fitting or under-fitting when the model is generalized. Therefore, since the experimental results have shown that feature selection does not affect the classification accuracy of 104 features, a higher classification accuracy can be achieved by directly using 104 features for classification, thereby circumventing the model generalization caused by feature selection.

Of course, there are still some uncertainties in the experimental results. First, we assume that the magnetic field of the tags and the reader is static, without considering the changes in the external environment on the SNR, so we only consider changing the SNR by the transmitting and receiving power. If the actual electromagnetic environment changes, the heatmaps of the tags may change, and the number of searches may also change. In addition, the current experiments only considered Alien-type tags commonly used in the market. To make the physical-layer technology applicable to more classification scenarios, more tag types, and manufacturers need to be tested. Future work will be to build a larger tag training set to improve the applicability and versatility of the algorithm.

REFERENCES

[1] R. Nayak, *Radio Frequency Identification (RFID): Technology and Application in Garment Manufacturing and Supply Chain*. London, U.K.: Chapman & Hall, 2019.

[2] S. A. Ahson and M. Ilyas, *RFID Handbook: Applications, Technology, Security, and Privacy*. Boca Raton, FL, USA: CRC Press, Dec. 2017.

[3] H. Suo et al., "Security in the Internet of Things: A review," in *Proc. Int. Conf. Comput. Sci. Electron. Eng.*, vol. 3. pp. 648–651, Mar. 2012.

[4] A. Alamr et al., "A secure ECC-based RFID mutual authentication protocol for Internet of Things," *J. Supercomputing*, vol. 74, pp. 4281–4294, Sep. 2018.

[5] B. Woo-Sik, "Formal verification of an RFID authentication protocol based on hash function and secret code," *Wireless Pers. Commun.*, vol. 79, no. 4. pp. 2595–2609, Mar. 2014.

[6] W. Yu and Y. Jiang, "Mobile RFID mutual authentication protocol based on hash function," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery (CyberC)*, Jan. 2017, pp. 358–361.

[7] M. R. Rieback, B. Crispo, and A. S. Tanenbaum, "RFID Guardian: A battery-powered mobile device for RFID privacy management," in *Proc. ACM Symp. Inf., Comput. Commun. Secur.*, 2006, pp. 220–230.

[8] T. Wang et al., "Transformer fault diagnosis using self-powered RFID sensor and deep learning approach," *IEEE Sensors J.*, vol. 18, no. 15, pp. 6399–6411, Jun. 2018.

[9] Z. Y. Lopez, Z. Akhter, and A. Shamim, "3D Printed RFID tag antenna miniaturized through volumetric folding and slow-wave structures," *IEEE J. Radio Freq. Identificat.*, vol. 6, pp. 164–175, Feb. 2022.

[10] H. Huang and H. Zhu, "RFID security and privacy: A research survey," *J. Netw. Comput. Appl.*, vol. 52, pp. 1–10, 2015.

[11] C. Bertoncini, K. Rudd, B. Nousain, and M. Hinders, "Wavelet fingerprinting of radio-frequency identification (RFID) tags," *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4843–4850, Dec. 2012.

[12] X. Chen, J. Liu, X. Wang, X. Zhang, Y. Wang, and L. Chen, "Combating tag cloning with COTS RFID devices," in *Proc. IEEE SECON*, Jun. 2018, pp. 1–9.

[13] S. Feng and S. Haykin, "Cognitive risk control for transmit-waveform selection in vehicular radar systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9542–9556, Oct. 2018.

[14] *EPC Radio Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz-960 MHz, Version 2.0. 1*, EPCglobal, G. S., Brussels, Belgium, 2015.

[15] *Technology—Automatic Identification and Data Capture Techniques—Part 10: Crypto Suite AES-128 Security Services for Air Interface Communications*, Standard ISO/IEC 29167-10:2017, Sep. 2017.

[16] B. Woo-Sik, "Formal verification of an RFID authentication protocol based on hash function and secret code," *Wireless Pers. Commun.*, vol. 79, no. 4, pp. 2595–2609, Mar. 2014,

[17] R. K. Pateriya and S. Sharma, "The evolution of RFID security and privacy: A research survey," in *Proc. Int. Conf. Commun. Syst. Netw. Technol.*, Katra, India, Jul. 2011, pp. 115–119.

[18] X. Liu et al., "RFID estimation with blocker tags," *IEEE/ACM Trans. Netw.*, vol. 25, no. 1, pp. 224–237, Nov. 2016.

[19] Y. Guo, J. Yang, and B. Liu, "Application of chaotic encryption algorithm based on variable parameters in RFID security," *EURASIP J. Wireless Commun. Netw.*, vol. 2021, no. 1, pp. 1–17, Dec. 2021.
[20] S. El Abkari, S. Kaissari, J. El Mhamdi, A. Jilbab, and E. H. El Abkari, "RFID system for hospital monitoring and medication tracking using digital signature," in *Proc. Digit. Technol. Appl. (ICDTA)*, Fez, Morocco. Cham, Switzerland: Springer, 2021, pp. 1051–1060.
[21] A. A. Elngar, K. M. Sagayam, and A. A. Elngar, "Augmenting security for electronic patient health record (ePHR) monitoring system using cryptographic key management schemes," *Fusion, Pract. Appl.*, vol. 5, no. 2, pp. 42–52, 2021.
[22] S.-S. Yang, Y.-H. Jang, M.-H. Park, S.-C. Park, and H.-J. Kim, "Design and implementation of active access control system by using NFC-based EAP-AKA protocol," *Wireless Pers. Commun.*, vol. 118, no. 4, pp. 2487–2503, Jun. 2021.
[23] M. Gao and Y. Lu, "URAP: A new ultra-lightweight RFID authentication protocol in passive RFID system," *J. Supercomput.*, vol. 78, no. 8, pp. 10493–10905, 2022.
[24] D. Dobrykh, D. Filonov, A. Slobozhanyuk, and P. Ginzburg, "Hardware RFID security for preventing far-field attacks," *IEEE Trans. Antennas Propag.*, vol. 70, no. 3, pp. 2199–2204, Mar. 2022.
[25] F. Lu, C. XiaoSheng, and T. Ye Terry, "Performance analysis of stacked RFID tags," in *Proc. IEEE Int. Conf. RFID*, May 2009, pp. 330–337.
[26] N. Javed, M. A. Azam, and Y. Amin, "Chipless RFID multisensor for temperature sensing and crack monitoring in an IoT environment," *IEEE Sensors Lett.*, vol. 5, no. 6, pp. 1–4, Jun. 2021, Art. no. 6001404.
[27] F. Costa et al., "A robust differential-amplitude codification for chipless RFID," *IEEE Microwave Wireless Compon. Lett.*, vol. 25, no. 12, pp. 832–834, Dec. 2015.
[28] M. Omer and G. Y. Tian, "Indoor distance estimation for passive UHF RFID tag based on RSSI and RCS," *Measurement*, vol. 127, pp. 425–430, Oct. 2018.
[29] P. Li, Z. An, L. Yang, P. Yang, and Q. Lin, "RFID harmonic for vibration sensing," *IEEE Trans. Mobile Comput.*, vol. 20, no. 4, pp. 1614–1626, Apr. 2021.
[30] H. Wu, W. Gao, C. Pu, and Z. Yu, "Feature selection and cross validation for physical-layer RFID counterfeit tag identification," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, 2022.
[31] N. Kargas, F. Mavromatis, and A. Bletsas, "Fully-coherent reader with commodity SDR for Gen2 FM0 and computational RFID," *IEEE Wireless Commun. Lett.*, vol. 4, no. 6, pp. 617–620, Dec. 2015, doi: 10.1109/LWC.2015.2475749.
[32] H. Wu, X. Wu, Y. Li, and Y. Zeng, "Collision resolution with FM0 signal separation for short-range random multi-access wireless network," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 7, pp. 438–450, 2021, doi: 10.1109/TSIPN.2021.3093000.
[33] J. Antoni, "The spectral kurtosis: A useful tool for characterising non-stationary signals," *Mech. Syst. Signal Process.*, vol. 20, no. 2, pp. 282–307, 2006.
[34] S.-W. Lin, C.-C. Chu, and C.-F. Tung, "Distributed Q-learning droop control for frequency synchronization and voltage restoration in isolated AC micro-grids," in *Proc. IEEE Ind. Appl. Soc. Annu. Meeting (IAS)*, Detroit, MI, USA, Oct. 2022, pp. 1–8, doi: 10.1109/IAS54023.2022.9939855.
[35] C. Bertoncini, K. Rudd, B. Nousain, and M. Hinders, "Wavelet fingerprinting of radio-frequency identification (RFID) tags," *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4843–4850, Dec. 2012, doi: 10.1109/TIE.2011.2179276.
[36] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF and RReliefF," *Mach. Learn.*, vol. 53, nos. 1–2, pp. 23–69, Oct. 2003.
[37] A. Satorra and P. M. Bentler, "A scaled difference chi-square test statistic for moment structure analysis," *Psychometrika*, vol. 66, no. 4, pp. 507–514, Dec. 2001.
[38] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 507–514.
[39] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, pp. 5–32, Oct. 2001.
[40] H. Zhang, Z. Wang, W. Xia, Y. Ni, and H. Zhao, "Weighted adaptive KNN algorithm with historical information fusion for fingerprint positioning," *IEEE Wireless Commun. Lett.*, vol. 11, no. 5, pp. 1002–1006, May 2022, doi: 10.1109/LWC.2022.3152610.
[41] H. P. Romero, K. A. Remley, D. F. Williams, and C. Wang, "Electromagnetic measurements for counterfeit detection of radio frequency identification cards," *IEEE Trans. Microw. Theory Techn.*, vol. 57, no. 5, pp. 1383–1387, May 2009.
[42] B. Danev, T. S. Heydt-Benjamin, and S. Capkun, "Physical-layer identification of RFID devices," in *Proc. 18th USENIX Secur. Symp.*, San Jose, CA, USA, 2009, pp. 199–214.
[43] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.
[44] S. Wei, Q. Qu, X. Zeng, J. Liang, J. Shi, and X. Zhang, "Self-attention bi-LSTM networks for radar signal modulation recognition," *IEEE Trans. Microw. Theory Techn.*, vol. 69, no. 11, pp. 5160–5172, Nov. 2021.

Haifeng Wu was born in Kunming, Yunnan, China, in 1977. He received the M.S. degree in electrical engineering from Yunnan University, Kunming, in 2004, and the Ph.D. degree in electrical engineering from Sun Yat-sen University, Guangzhou, China, in 2007.

He is currently a Professor with the Department of Information Engineering, Yunnan Minzu University, Kunming. His research interests include neural signal processing, machine learning, and mobile communications.

Chongrong Pu was born in Chuxiong, Yunnan, China, in 1996. He received the B.E. degree in electronic information engineering from Yunnan Minzu University, Kunming, Yunnan, in 2020, where he is currently pursuing the M.S. degree with the Department of Information Engineering.

His research interests include ultrahigh frequency (UHF) radio frequency identification (RFID) communication.

Wei Gao was born in Heze, Shandong, China, in 1998. He received the B.E. degree in communication engineering from the Inner Mongolia University of Science and Technology, Baotou, China, in 2016. He is currently pursuing the M.S. degree with the Department of Information Engineering, Yunnan Minzu University, Kunming, China.

His research interests include mobile communications and wireless near-field communication.

Yu Zeng was born in Kunming, Yunnan, China, in 1981. She received the M.S. degree in electrical engineering from Yunnan University, Kunming, in 2006.

She is currently an Assistant Professor with the Department of Information Engineering, Yunnan Minzu University, Kunming. Her research interests include wireless networks and mobile communications.