International Journal of Information Technologies and Systems Approach
Volume 16 • Issue 3

Radio Frequency Fingerprint Identification Based on Metric Learning

Danyao Shen, Xi'an Research Institute of High Technology, China*
Fengchao Zhu, Xi'an Research Institute of High Technology, China
Zhanpeng Zhang, Xi'an Research Institute of High Technology, China
Xiaodong Mu, Xi'an Research Institute of High Technology, China

ABSTRACT

With the popularization of the internet of things (IoT), its security has become increasingly prominent. Radio-frequency fingerprinting (RFF) is used as a physical-layer security method to provide security in wireless networks. However, the problems of poor performance in a highly noisy environment and less consideration of calculation resources are urgent to be resolved in a practical RFF application domain. The authors propose a new RFF identification method based on metric learning. They used power spectrum density (PSD) to extract the RFF from the nonlinearity of the RF front end. Then they adopted the large margin nearest neighbor (LMNN) classification algorithm to identify eight software-defined radio (SDR) devices. Different from existing RFF identification algorithms, the proposed LMNN method is more general and can learn the optimal metric from the wireless communication environment. Furthermore, they propose a new training and test strategy based on mixed SNR, which significantly improves the performance of conventional low-complexity RFF identification methods. Experimental results show that the proposed method can achieve 99.8% identification accuracy with 30dB SNR and 96.83% with 10dB SNR. In conclusion, the study demonstrates the effectiveness of the proposed method in recognition efficiency and computational complexity.

KEYWORDS

Large Margin Nearest Neighbor (LMNN), Metric Learning, Mixed SNR Strategy, Power Spectrum Density (PSD), Radio-Frequency Fingerprint (RFF)

INTRODUCTION

Radio-frequency fingerprint (RFF) is the intrinsic characteristics of wireless devices generated from hardware imperfection. Because hardware imperfection is unique for different wireless devices, RFF identification has become an emerging device authentication technique (Danev et al., 2012).

In general, RFF identification includes two steps: feature extraction and classification. Feature extraction determines the quality of RFF and directly affects classification accuracy. Many studies have explored the characteristics of different electronic components to extract effective RFF features—for example, in-phase and quadrature offset (Brik et al., 2008), phase offset (Nguyen et al., 2011), carrier

DOI: 10.4018/IJITSA.321194 *Corresponding Author

frequency offset (Wheeler et al., 2017), differential constellation trace figure (Peng et al., 2019), and signal spectrum (Rehman et al., 2014). Recently, Wang et al. (2016) built a theoretical model for the entire wireless communication link to analyze the effectiveness of different RFF features. The results show that power spectrum density (PSD) can characterize the nonlinearity of the RF front end, contributing to the most significant RFF feature.

On the other hand, classification algorithm design is another key part of RFF identification in which lots of machine learning algorithms have been used. Danev et al. (2009) successfully classified 50 radio-frequency identification (RFID) transponders using principal component analysis (PCA) and k-nearest neighbor (KNN). Baldini et al. (2017) compared the performance of KNN, support vector machine (SVM), and decision tree algorithm. Wang et al. (2017) used the Fisher linear discriminant analysis (LDA) based on the Mahalanobis distance metric to analyze the user capacity of wireless physical-layer identification. However, the performance of existing classification algorithms in RFF identification will severely degrade with the decrease of receive SNR. For example, six devices are classified with 98% accuracy under 30 dB and 90% accuracy under 10 dB (Patel et al., 2014). The results achieved only 51% identification accuracy for non-line-of-sight (NLOS) channel model (Wang et al., 2016), where the SNR is 15 dB. Peng et al. (2019) shared a method based on convolutional neural network (CNN) that can achieve 99.1% accuracy at high SNR but quickly dropped to 80% at 10 dB.

We propose a new RFF identification method based on metric learning (Weinberger et al., 2009) that can adapt to different SNRs. We used the large margin nearest neighbor (LMNN) to directly learn the optimal distance metric from training samples that have not been used in existing RFF identification works. Moreover, we propose a new training and test strategy based on mixed SNR that significantly improves the performance of conventional RFF identification methods with low complexity. We describe designing the real testbeds where eight devices are used for identification under different SNRs. The experiment results show that the LMNN algorithm achieves higher identification accuracy at low SNR than existing algorithms—for example, 96.83% accuracy at 10 dB.

The main contributions of this article are summarized as follows.

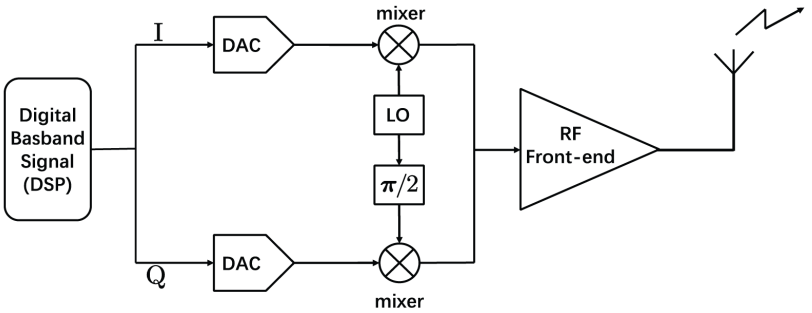
Methodologically, we propose a new RFF identification method based on metric learning. We adopted the LMNN classification algorithm to identify eight software-defined radio (SDR) devices. Different from existing RFF identification algorithms, the proposed LMNN method is more general and can learn the optimal metric from the wireless communication environment.

Experimentally, we propose a new training and test strategy based on mixed SNR that significantly improves the performance of conventional low-complexity RFF identification methods that produce poor datasets. Specifically, with our method, the signal-to-noise ratio has dropped to 0 dB. Experimental results within such poor datasets also show the high applicability of our methods in IoT scenarios where physical and computational resources are extremely scarce.

RFF GENERATION

The key idea behind RFF identification is exploiting the unique characteristics of hardware to identify wireless devices. A simplified transmitter circuit and the signal transmit procedure are shown in Figure 1. Based on digital modulation, the baseband binary bits are first mapped to the in-phase and quadrature (I/Q) channels. A digital-to-analog converter (DAC) is then used to convert the I/Q signal into a time-continuous analog signal. After the DAC is finished, the mixer and RF front end move the analog signal to the passband. Note that almost all elements of the transmitter circuit are not perfect, and the imperfection is mainly generalized from the following aspects. First, the imperfection of the local oscillator (LO) will affect the carrier frequency of devises (i.e., frequency offset and phase offset). Second, the quadrature mixer is

Figure 1. A simplified transmitter circuit



often impaired by gain and phase mismatches that result in I/Q imbalance and random phase noise. Finally, the passband signals go through the RF front-end amplifier and filter to gain enough power for radiation, which generates the most significant RFF (i.e., the nonlinearity of the RF front end).

Figure 1 shows how the digital-to-analog converter (DAC) introduces quantization error and integral nonlinearity. The local oscillators (LOs) introduce frequency offset, the quadrature mixers introduce I/Q imbalance, and the front end introduces nonlinearity distortions.

Moreover, almost all modern digital communication systems contain a stable preamble signal at the front of the data packets; this signal does not change from one transmission to the next. Consequently, the received preamble signal can be used to analyze the impairment of transmit circuits. Note that the impairment of transmit circuits is unique for different devices, and this impairment data can be used for RFF identification. It's paramount to note that the impairment of transmit circuits are unique for diverse devices, which can be utilized for RFF identification.

For example, if we assume the received signal sequence is $s^{M\times 1}$, then s can be expressed as shown in equation (1):

$$s = \left[s_w; s_p; s_d\right] \tag{1}$$

In this equation, s_w is the noise signal, s_p is the preamble signal, and s_d is the data signal. Using the I/Q modulation, we can further express the m th sample of s as shown in equation (2):

$$s\left(m\right) = s_I\left(m\right) + js_Q\left(m\right) \tag{2}$$

In equation (2), $s_I\left(m\right)$ and $s_Q\left(m\right)$ are the in-phase and quadrature components, respectively. The amplitude of the m th sample of s is formulated as shown in equation (3):

$$a_s\left(m\right) = \sqrt{s_I^2\left(m\right) + s_Q^2\left(m\right)} \tag{3}$$

Then the amplitude changes of $s\left(m\right)$ can be detected by using the variance of $s_a\left(m\right)$, which can be expressed using the formula shown in equation (4):

$$v(m)=\begin{cases}\alpha\dfrac{1}{L-1}\displaystyle\sum_{l=1}^{L}\left(a_s(m-l)-\bar{a}_L\right)^2, & m>L\\ 0, & m\le L\end{cases} \tag{4}$$

In equation (4), α is a scaling factor that can be determined by experiment; L is the length of the sliding window, and $\bar{a}_L$ is the mean value of the received signal sequence $a_s(m-L),\ldots,a_s(m-1)$. When the discrete variance vector v is derived, the threshold detection approach (Rasmussen et al., 2007) and the cumulative (Pignatiello et al., 1990) sum (CUSUM) algorithm can be used to find out the preamble signal s_p.

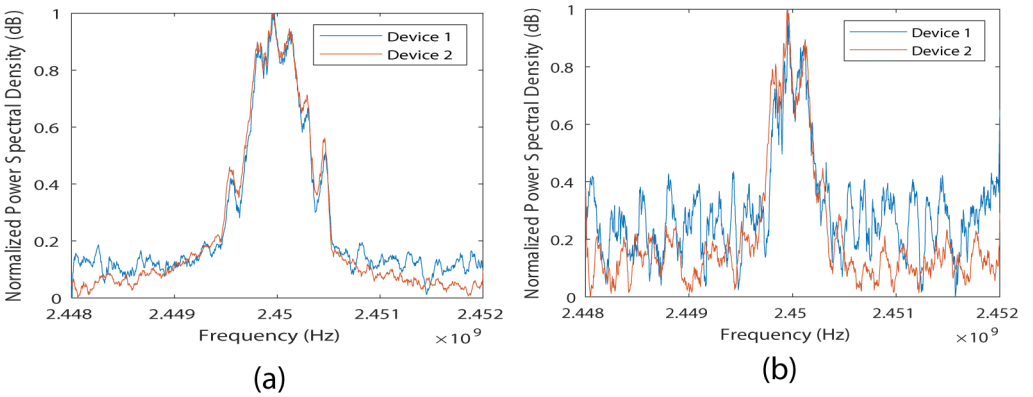
Because time-domain I/Q signals are sensitive to noise, directly using the preamble s_p to identify different wireless devices will result in poor performance. Consequently, most existing RFF identification algorithms contain the process of feature extraction in which the nonlinearity of the RF front end has been extracted as the most significant RFFs. The PSD of preamble s_p can be calculated as shown in equation (5):

$$x(k)=\frac{1}{N_{FFT}}\left|\sum_{n=0}^{N_{FFT}-1}s_p(n)\exp\left(-j\frac{2\pi}{N_{FFT}}nk\right)\right|^2 \tag{5}$$

In equation (5), N_{FFT} is the length of FFT transformation points, and $N_{FFT}\ge N$ should be satisfied. The PSDs of two devices under different SNRs are shown in Figure 2. It is obvious in Figure 2(a) that the difference of PSDs is big at high frequency and small at low frequency. Consequently, the PSDs at high frequency can be used to identify different devices. However, Figure 2(b) shows that the PSDs at low SNR (15 dB) will be severely affected by noise.

Figure 2(a) shows the normalized PSD of devices 1 and 2 under 30dB SNR. Figure 2(b) shows the normalized PSD of devices 1 and 2 under 15dB SNR. These two figures use dB as the PSD unit, and the curves in them use smoothing.

Figure 2. PSDs of two devices under different SNRs



RFF IDENTIFICATION

Existing works (Wang et al., 2016; Peng et al., 2019) have tried various algorithms to classify different PSDs. However, the performance of most algorithms is weak at low SNR (from 0 dB to 10 dB). In this section, we propose an RFF identification framework based on metric learning in which the LMNN is used to learn more effective distance metric at low SNR.

Distance Metric

Similarity measurement (i.e., the calculation of the distance metric) is the critical procedure for most RFF identification algorithms; it can affect the overall identification accuracy. For example, the well-known supervised classification algorithm k -nearest neighbor (KNN) is affected profoundly by the distance metric and should be carefully selected through experience.

Define the dataset of the PSD as $\{(x_i, y_i)\}_{i=1}^{m}$, $y_i = \mathbb{R}$, where $x_i \in \mathbb{R}^{d\times 1}$ represents the PSD of the i th preamble in the entire dataset, d is the dimension of vector, and x_i, y_i is the label for different transmitters. The Euclidean distance between two samples x_i, x_j can be expressed as shown in equation (6):

$$\begin{aligned} &D_e\left(x_i, x_j\right) = \| \left(x_i - x_j\right) \|^2 \\ &= dist_{ij,1}^2 + dist_{if,2}^2 + \cdots + dist_{ij,d}^2 \end{aligned} \tag{6}$$

In equation (6), $d_{ij,k}^2$ represents the distance in the k th dimension of x_i and x_j. However, it has been shown in Figure 2 that high-frequency components are easily noised under low SNR. Therefore, we can assign weights for different spectrum components. A simple weighted distance metric between x_i and x_j can be expressed as shown in equation (7):

$$\begin{aligned} &D_w\left(x_i, x_j\right) = w_1 dist_{ij,1}^2 + w_2 dist_{if,2}^2 + \cdots + w_d dist_{ij,d}^2 \\ &= \left(x_i - x_j\right)^T W \left(x_i - x_j\right) \end{aligned} \tag{7}$$

In equation (7), W is a diagonal matrix, $W\left(i,i\right) = w_i \geq 0$. Note that the nondiagonal elements of W are all zeros, which implies that different spectral components are irrelevant. However, there are usually correlations between different spectral components in practical datasets. Thus, a more general semi-define symmetric matrix M will outperform the diagonal matrix W. Matrix M can be used to construct a new distance metric using the formula shown in equation (8):

$$\begin{aligned} &D_M\left(x_i, x_j\right) = \left(x_i - x_j\right)^T M \left(x_i - x_j\right) \\ &= \| x_i - x_j \|_M^2 \end{aligned} \tag{8}$$

In equation (8), the distance D_M is exactly the parameterized Mahalanobis distance. It is easily known that D_M contains the Euclidean distance D_e and weighted distance metric D_w as special cases.

Robust Classification

Using the parameterized Mahalanobis distance D_M, we now introduce a new RFF classification algorithm based on LMNN (Weinberger et al., 2009). The LMNN implements a more robust KNN classification based on two simple intuitions. First, each training sample x_i should share the same label y_i with its k nearest neighbors; second, the training samples with different labels should be widely separated. The formulation of LMNN constraints can be expressed as shown in equation (9):

$$\| x_i - x_l \|_M^2 - \| x_i - x_j \|_M^2 \geq 1 \tag{9}$$

In equation (9), x_i is the ith training sample, x_j represents the jth target samples of its k nearest neighbors, and x_l denotes the impostor (i.e., a sample that does not belong to the same class with x_i).

Based on these two intuitive principles, we can further define the loss functions. The first term pulls target neighbors closer by penalizing the large distance between homogeneous samples. This term can be formulated as shown in equation (10):

$$\ell_{pull}\left(M\right) = \sum_{i=1}^{m} \| x_i - x_j \|_M^2 \tag{10}$$

In equation (10), m is the number of training samples. The second term pushes heterogeneous samples away by penalizing the small distance between the impostors and the perimeter. This term can be expressed as shown in equation (11):

$$\ell_{push}\left(M\right) = \sum_{i=1}^{m}\sum_{l}\left(1 - y_{il}\right)\ell_{hinge}\left(z\right) \tag{11}$$

In equation (11), $\ell_{hinge}\left(z\right) = \max\left(0, 1 - z\right)$ is the standard hinge loss function; $z = \| x_i - x_l \|_M^2 - \| x_i - x_j \|_M^2$; and y_{il} = the formula shown in equation (12):

$$y_{il} = \begin{cases} 1, & if \left| y_i = y_l \right. \\ 0, & otherwise \end{cases} \tag{12}$$

The loss functions shown in equations (10) and (11) can be expressed together as shown in equation (13):

$$\ell\left(M\right) = \left(1 - \mu\right)\ell_{pull}\left(M\right) + \mu\ell_{push}\left(M\right) \tag{13}$$

In this equation, $\mu \subseteq \left[0, 1\right]$ is used to balance the two loss functions. Combining the formulas shown in equations (9) and (13), we can formulate the RFF identification problem as shown in equations (14a) and (14b):

$$\min_{M>0} \ell\left(M\right) \tag{14a}$$

$$s.t.\left(9\right) is\ \ satisfied \tag{14b}$$

Note that the above optimization is non-convex and is hard to solve. Nevertheless, because $\ell\left(M\right)$ is a piecewise linear convex function, we can introduce non-negative slack variables ξ_{ijl} to relax equation (9) as a semi-definite program (SDP) as shown in equations (15a) and (15b):

$$\min_{\left\{\begin{matrix} M\succ 0 \\ \xi_{ijl}\geq 0 \end{matrix}\right\}} \left(1-\mu\right)\sum_{i=1}^{m} \| x_i - x_j \|_M^2 + \mu\sum_{i=1}^{m}\sum_{l}\left(1-y_{il}\right)\xi_{ijl} \tag{15a}$$

$$s.t. \| x_i - x_l \|_M^2 - \| x_i - x_j \|_M^2 \geq 1-\xi_{ijl} \tag{15b}$$

In these equations, ξ_{ijl} is a substitution for the hinge loss function $\ell_{hinge}\left(z\right)$. Note that the above optimization is convex and can be efficently solved using the CVX tools (Boyd & Vandenberghe, 2004).

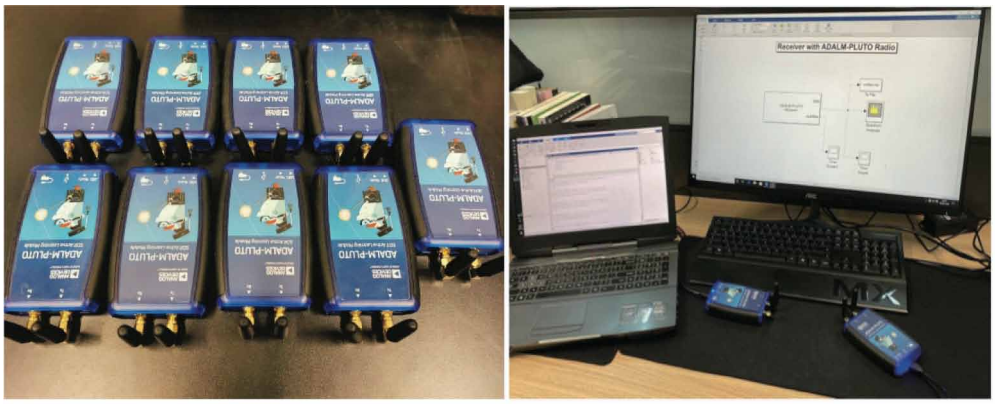
Figure 3(a) shows a photo of eight ADALM-PLUTO SDR devices. Figure 3(b) shows a photo of the experiment platform.

EXPERIMENTAL EVALUATION

Experimental Setup

In our experiment, we used 9 ADALM-PLUTO software-defined radio devices (one receiver and eight transmitters, as shown in Figure 3). These devices were produced in the same batch by Analog

Figure 3. SDR devices and experiment platform



(a) (b)

Devices (ADI) and worked at 2.4 GHz (Analog, 2020). The modulation scheme was designed according to the IEEE 802.15.4 protocol (IEEE, 2020), including spread spectrum, OQPSK modulation, and a half-sine shaping filter (the oversampling factor was 4). After spread spectrum and modulation, each I/Q channel signal had 128 symbols. The symbols were then transmitted to the designated frequency band through carrier modulation through the shaping filter and DAC. The baseband transmitting rate was 1 MHz. The receiver was set at a distance of 0.1 m from the transmitter, where the sampling rate was 4 MHz. All devices were connected to the computer via a universal serial bus (USB) and used Matlab 2019b for data processing.

A total of 7,600 samples (950 samples per device) were obtained in the experiment. To get a reliable and stable model, 80% of the data was used as the training set, and the remaining 20% was used as the test set. Different levels of white Gaussian noise were added to simulate various SNR environments (from 0 dB to 30 dB), where the additive white Gaussian noise (AWGN) channel module in Matlab was used.

Experiment Results

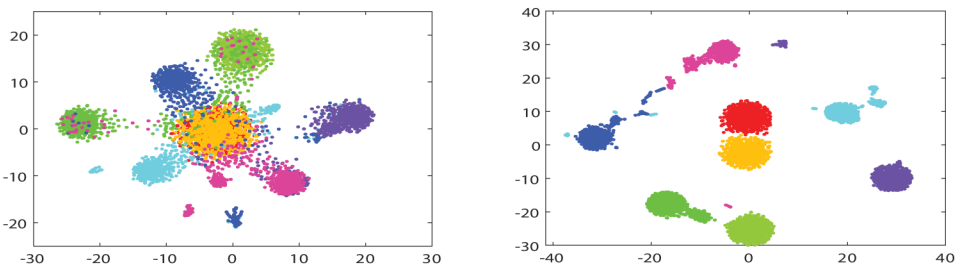
Classic Training and Test Strategy

The visualizations of RFFs with/without metric learning are shown in Figure 4, where the t-Distributed Stochastic Neighbor Embedding (tSNE) technology was used (van der Maaten & Hinton, 2008). It is obvious that the RFF samples in two dimensions are close to each other as shown in Figure 4(a). In contrast, the RFF samples are far away from each other with metric learning as shown in Figure 4(b). The results show that the metric learning will greatly improve the separability for RFFs.

The performances of LMNN, KNN, LDA, and support vector machine (SVM) are compared in Figure 5. It is clear that the identification accuracy of SVM is better than other existing algorithms under high SNR (≥ 13 dB). However, the accuracy will decrease with low SNR, where the conventional LDA method is better than other existing algorithms. Nevertheless, the identification accuracy of LMNN is higher than most existing algorithms with SNR ranging from 0 dB to 30 dB. More detailed results are shown in Table 1. At 30dB SNR, the identification accuracy of KNN, LMNN, and SVM is nearly equivalent, which is about 99%. At 0dB SNR, the identification accuracy of the KNN algorithm quickly drops to 80.42%. Note that the LMNN algorithm maintains 87.29% at 0 dB.

In Figure 5, both KNN and SVM use PCA to reduce dimensionality (95% variance). LDA reduces the sample to seven dimensions. LMNN reduces the sample to 20 dimensions.

Figure 4. Visualization with t-SNE for different sample space: (a) Original sample space, (b) the new sample space after metric learning



(a) (b)

Figure 5. Identification accuracy with different training strategies

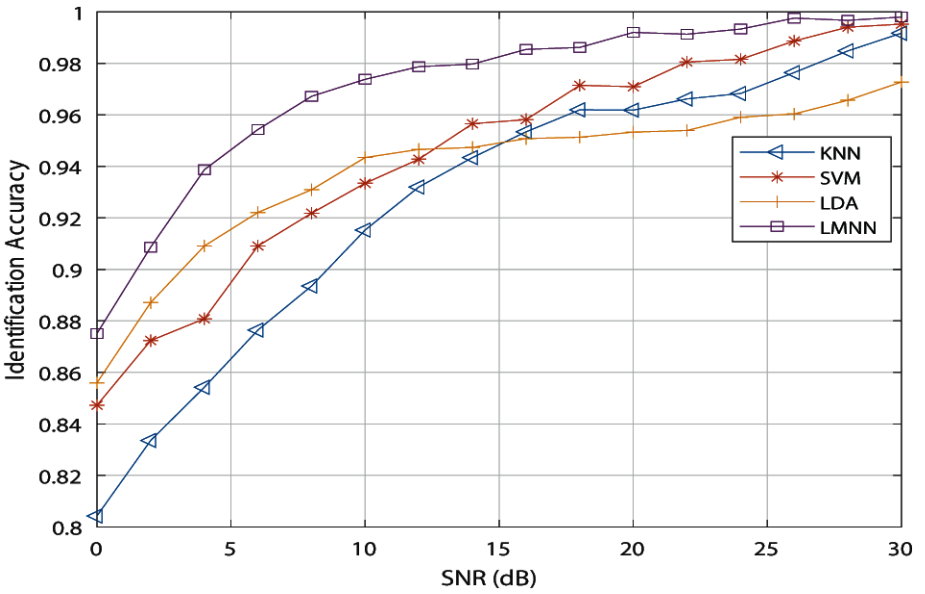


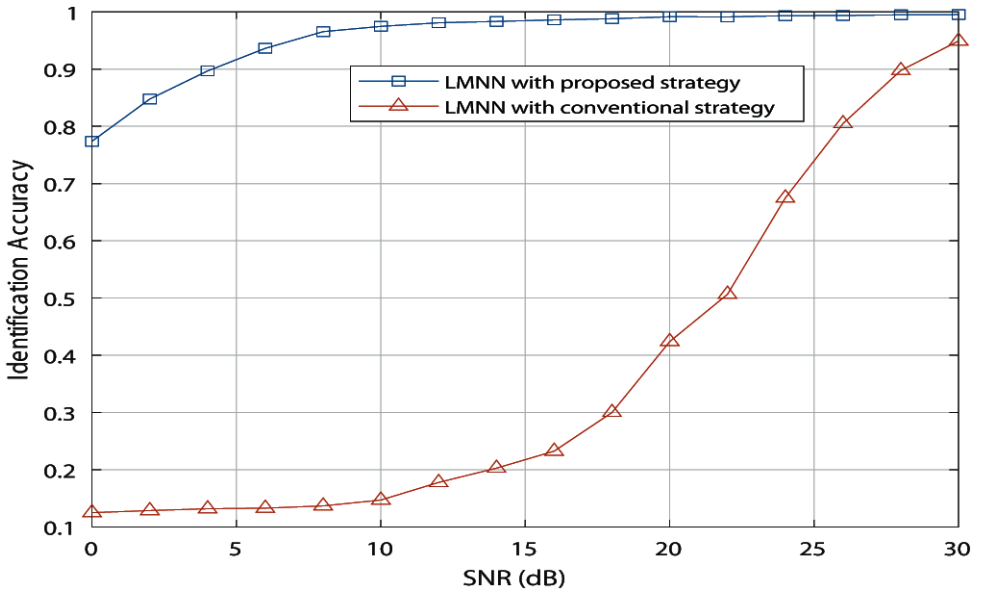
Table 1. Identification accuracy under single SNR scenario

SNR	KNN	LMNN	SVM	LDA
0	0.8042	0.8729	0.8472	0.8589
10	0.9153	0.9683	0.9334	0.9434
20	0.9618	0.9853	0.9709	0.9539
30	0.9917	0.9980	0.9953	0.9728

Novel Training and Test Strategy Based on Mixed SNR

In the conventional training and test strategy, the training set and model parameters must be updated according to different SNRs, which are adopted by most existing works. However, the conventional training and test strategy will result in high memory and time consumption for different SNRs, and this method is difficult to implement in low-cost devices. Moreover, the conventional training and test strategy requires precise estimation of the SNR of test samples and is hard to work in practice. To simplify the conventional strategy, existing works proposed to simplify the training process (i.e., training the algorithm under fixed SNR while testing the algorithm under different SNRs). However, the simplified strategy will result in decreased identification accuracy. Aiming to overcome this problem, we propose a new training and test strategy, where the training is processed on a mixed dataset (consisting of the original received signals with different noise from 0 to 30 dB). It is clear in Figure 6 that the identification accuracy of the LMNN algorithm trained by the conventional strategy drops rapidly with the decrease of SNR (less than 80% below 25 dB). Note that the proposed training and test strategy is much better than the conventional method primarily because the training set and the test set are not independent and identically distributed (i.i.d) (Bishop, 2006). The proposed

Figure 6. Identification accuracy with different training and test strategies



training and test strategy enables the model to directly learn RFF characteristics from mixed SNRs with low complexity.

In Figure 6, one strategy is to learn from the training set with mixed noise. The other is to directly learn from the training set with high SNR. The training set (760×8 samples) is from the original received signals without noise. The text set ($190 \times 8 \times 16$ samples) is from the original received signals with different noise from 0 to 30 dB (the step size is 2, a total of 16 levels).

In Figure 7, the parameter selection is the same as Figure 5.

In the last example, we compared the performance of the proposed algorithm with other existing algorithms in the new training and test strategy. The results in Figure 7 show that the LMNN algorithm maintains stability and high identification accuracy under different SNRs. Moreover, to evaluate the overall performance of the algorithms under different SNRs, we used test samples from 16 different levels of SNRs to calculate the average identification accuracy of the two strategies, as shown in Table 2. Obviously, the proposed training and test strategy is better than existing strategies. The average identification accuracy of LMNN algorithm is about 95.58%, which is the higher than other methods.

CONCLUSION

We proposed a new RFF identification method based on metric learning. The nonlinear characteristics of the RF front end from eight SDR devices were used as RFFs. Different from existing works, our research is based on using the LMNN algorithm for RFF identification, which can be formulated as a convex optimization. Moreover, we proposed a novel training and test strategy based on mixed SNR to improve the performance of existing RFF identification methods. Experimental results showed that the proposed LMNN algorithm under the new training strategy achieved 95.58% identification accuracy, which is better than existing methods. Recognizing RFFs remains a challenging task, particularly when the channel environment is unknown and there is a lack of prior information on signal modulation schemes. Although our proposed algorithms showed promising results, further evaluation is necessary to assess their effectiveness in a more complex environment.

Figure 7. Identification accuracy of different algorithms trained under mixed SNR

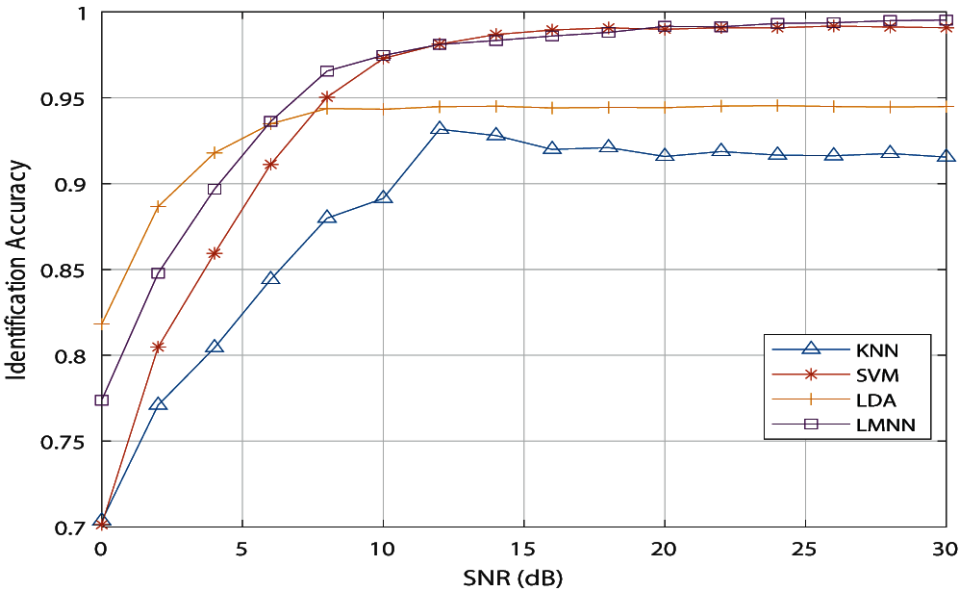


Table 2. Average identification accuracy under different training strategy

Algorithm	KNN	LMNN	SVM	LDA
Strategy I	0.3132	0.3732	0.2987	0.4174
Strategy II	0.8809	0.9558	0.9433	0.9307

DATA AVAILABILITY

The data used to support the findings of this study are included within the article.

CONFLICTS OF INTEREST

We declare that there is no conflict of interest regarding the publication of this paper.

FUNDING STATEMENT

This research received no external funding.

REFERENCES

Analog Devices. (2020). *ADALM-PLUTO: Software-defined radio active learning module* [Software]. https://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html#eb-overview

Baldini, G., Giuliani, R., Steri, G., & Neisse, R. (2017). Physical layer authentication of Internet of Things wireless devices through permutation and dispersion entropy. In *Proceedings of the IEEE Global Internet of Things Summit (GIoTS)*. (pp. 1–6). IEEE. doi:10.1109/GIOTS.2017.8016272

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press. doi:10.1017/CBO9780511804441

Brik, V., Banerjee, S., Gruteser, M., & Oh, S. (2008) Wireless device identification with radiometric signatures. In *MobiCom '08: Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*. (pp. 116–127). Association for Computing Machinery. doi:10.1145/1409944.1409959

Danev, B., Heydt-Benjamin, T. S., & Capkun, S. (2009). Physical-layer identification of RFID devices. In *Proceedings of the 18th Conference on USENIX Security Symposium* (pp. 199–214). Association for Computing Machinery.

Danev, B., Zanetti, D., & Capkun, S. (2012). On physical-layer identification of wireless devices. *ACM Computing Surveys*, *45*(1), 1–29. doi:10.1145/2379776.2379782

IEEE Standard for Low-Rate Wireless Networks. (2020). Retrieved July 10, 2020, from http://standard.ieee.org/standard/802.15.4-2015.html

Nguyen, N. T., Zheng, G., Han, Z., & Zheng, R. (2011). *Device fingerprinting to enhance wireless security using nonparametric Bayesian method. In Proceedings IEEE INFOCOM*. IEEE. doi:10.1109/INFCOM.2011.5934926

Patel, H., Temple, M. A., & Ramsey, B. W. (2014). Comparison of high-end and low-end receivers for RF-DNA fingerprinting. In *Proceedings of the 2014 IEEE Military Communications Conference*. (pp. 24–29). IEEE. doi:10.1109/MILCOM.2014.13

Peng, L., Hu, A., Zhang, J., Jiang, Y., Yu, J., & Yan, Y. (2019). Design of a hybrid RF fingerprint extraction and device classification scheme. *IEEE Internet of Things Journal*, *6*(1), 349–360. doi:10.1109/JIOT.2018.2838071

Peng, L., Zhang, J., Liu, M., & Hu, A. (2019). Deep learning based RF fingerprint identification using differential constellation trace figure. *IEEE Transactions on Vehicular Technology*, *69*(1), 1091–1095. doi:10.1109/TVT.2019.2950670

Pignatiello, J. J. Jr, & Runger, G. C. (1990). Comparisons of multivariate CUSUM charts. *Journal of Quality Technology*, *22*(3), 173–186. doi:10.1080/00224065.1990.11979237

Rasmussen, K. B., & Capkun, S. (2007). Implications of radio fingerprinting on the security of sensor networks. In *Proceedings of the Third International Conference on Security and Privacy in Communications Networks and the Workshops (SecureComm 2007)* (pp. 331–340). doi:10.1109/SECCOM.2007.4550352

Rehman, S. U., Alam, S., & Ardekani, I. T. (2014). An overview of radio frequency fingerprinting for low-end devices. *International Journal of Mobile Computing and Multimedia Communications*, *6*(3), 1–21. doi:10.4018/IJMCMC.2014070101

van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*, 2579–2605. https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

Wang, W., Sun, Z., Piao, S., Zhu, B., & Ren, K. (2016). Wireless physical-layer identification: Modeling and validation. *IEEE Transactions on Information Forensics and Security*, *11*(9), 2091–2106. doi:10.1109/TIFS.2016.2552146

Wang, W., Sun, Z., Ren, K., & Zhu, B. (2017). User capacity of wireless physical-layer identification. *IEEE Access : Practical Innovations, Open Solutions*, *5*, 3353–3368. doi:10.1109/ACCESS.2017.2674967

Weinberger, K. Q., & Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, *10*(2). https://www.jmlr.org/papers/v10/weinberger09a.html

Wheeler, C. G., & Reising, D. R. (2017). Assessment of the impact of CFO on RF-DNA fingerprint classification performance. In *Proceedings of the International Conference on Computing, Networking and Communications (ICNC)*. (pp. 1–5). doi:10.1109/ICCNC.2017.7876111