Effect of the Short Time Fourier Transform on the Classification of Complex-Valued Mobile Signals

Logan Smith[a], Nicholas Smith[a], Surya Kodipaka[b], Ajaya Dahal[a], Bo Tang[a], John E. Ball[a*], and Maxwell Young[b]

[a]Mississippi State University, Department of Electrical and Computer Engineering, 406 Hardy Rd., Mississippi State, MS, USA, 39762
[b]Mississippi State University, Department of Computer Science and Engineering, 665 George Perry St., Mississippi State, MS, USA, 39762

ABSTRACT

Wireless devices identify themselves using media access control (MAC) addresses which can be easily intercepted and mimicked by an adversary. Mobile devices also have a unique physical fingerprint represented by perturbations in the frequency of broadcasted signals caused by differences in the manufacturing process of their hardware components. This unique fingerprint is much more difficult to mimic. The short time Fourier transform (STFT) is used to analyze how the frequency content of a signal changes over time, and may provide a better representation of mobile signals in order to detect their unique fingerprint. In this paper, we have collected wireless signals using the 802.11 a/g protocol, showing the effect on classification performance of applying the STFT when varying the choice of window lengths, augmenting the data with complex Gaussian noise, and concatenating STFTs of different frequency resolutions, achieving state-of-the-art performance of 99.94% accuracy in the process.

Keywords: Short Time Fourier Transform, Wireless Physical Fingerprints

1. INTRODUCTION

Wireless devices have a unique digital identification called the media access control (MAC) address which allow wireless networks to better communicate with them. Wireless systems that rely on the digital MAC address for access to the network are vulnerable to adversarial attacks and more recent MAC address randomization protocols. An unauthorized device can intercept a legitimate device's media access control (MAC) address, copy it, and gain access to the network. More recently, wireless devices such as iPhones^{TM} allow MAC address randomization for privacy; having one MAC address on all wireless networks can link all user activity to that device.[1] It is not viable to allow access based on a set of allowed MAC addresses because legitimate devices can change MAC addresses for legitimate, privacy concerns.

These wireless devices do have a "physical fingerprint" counterpart which does not rely on the digital MAC address, meaning it is unaffected by MAC addresses being mimicked or randomized. The physical fingerprint of each device is created by variations in the manufacturing process of the individual hardware components. This leads to variations in each device's transmitted signals that can uniquely identify it.[2]

The short time Fourier transform (STFT) performs the Fast Fourier Transform (FFT) on different segments of a signal in order to capture changes in the frequency domain overtime. There may be more identifiable differences in the physical fingerprint of each device in the frequency domain; changes in the transmitted signal due to hardware differences may show perturbations in the frequency representation over time.

Herein, our main contributions to the physical fingerprinting research area are:

1. A parameter sweep of different window lengths on the effect of classification performance.

Further author information: (Send correspondence to J.E.B.)
J.E.B.: E-mail: jeball@ece.msstate.edu, Telephone: 1 662 325 4169

2. A comparison of performance on classification accuracy by applying the STFT, augmenting the data with complex Gaussian noise, and combining STFTs at different frequency resolutions.

3. State-of-the-art performance in classification in both (1) and (2).

4. A new wireless physical fingerprinting dataset including 12 devices, two antennas, and channel state information (CSI).

2. BACKGROUND AND RELATED WORK

2.1 Wireless Physical Fingerprinting

The wireless physical fingerprint has been analyzed before by using the transient,[3] which is the very beginning of the wireless signal where it ramps up from the noise floor to a higher magnitude. This signal is inherently complex-valued which has been converted into the magnitude and phase of each signal.[4] To save on memory and computation, some works have used dimensionality reduction techniques such as principle component analysis[5] and utilizing the latent layer of an autoencoder.[6] Others have converted these time domain signals into the frequency domain using the STFT[7] and the wavelet transform.[8]

2.2 STFT

The STFT is a two-dimensional transform that allows some control over time and frequency resolution simultaneously. A one-dimensional signal is scanned using a time-based filter with limited time extent. Common choices for a filter are the Hamming window or a Gaussian window. The STFT slides the window across the data, and for each window position, performs a FFT to convert the data to the frequency domain. There are several parameters in STFT analysis: the window length (in samples), the window shape, the overlap between windows, and the FFT order. A longer time window will have higher frequency resolution and lower time resolution, while a short window has high time resolution and lower frequency resolution. The FFT order controls the FFT computational resolution. The time window length controls the FFT physical frequency resolution.[9] For each time window position, the FFT is stacked into a column vector. These FFT vectors are then concatenated to form a two-dimensional complex matrix (or image). Typically, engineers work with the magnitude squared image, called the spectrogram, since the STFT is a complex-valued two-dimensional image.

The STFT can tease out time-varying frequency components in a non-stationary signal. Using a FFT on the original image blurs these together, so the STFT provides richer information on the time/frequency content at the expense of being more computationally complex.

The STFT has been used with a support vector machine (SVM) to classify four wireless devices of the same make and model up to 80.5% accuracy.[7] Additionally, it has been used on the radio frequency (RF) signal of drones for classifying the specific number of drones in an area, up to four drones,[10] and classifying five generated RF signal transmitters using a CNN, achieving 99.70% accuracy.[11]

2.3 MLP

An MLP is a feedforward NN. The MLP, given sufficient neurons, non-linear activations, and adequate training, can approximate any function to arbitrary accuracy and precision.[12] A typical MLP is made up of several layers of neurons. Each neuron has a set number of inputs and an output. The output is generally a non-linear function of the dot product of the input vector plus a bias, that is, $f(\mathbf{x}) = f(\mathbf{x} \cdot \mathbf{w} + b)$, where the neuron's input is $\mathbf{x}$, the neuron's learned weight and bias vectors are $\mathbf{w}$ and b, respectfully, and the nonlinear activation is $f()$. Each layer constists of a stack of the neurons, and a typical MLP has dense connections, meaning every neuron in layer $N+1$ receives inputs from each output in layer N.

3. METHODS

3.1 Dataset

Using a USRP B210 software defined radio (SDR) with an LP0965 antenna, we collected wireless signals from the 12 cellphones listed in Table 1. Each device is placed 3.35 meters up and 1.98 meters to the right of the SDR. From each device, we captured 3000 complex-valued signals from two antennas (1500 from each antenna) where each signal is trimmed to get the transient, short training field (STF), and long training field (LTF) as shown in Figure 1. The STF and LTF are used for synchronization, so they, along with the transient, are viable information for learning the physical fingerprint since they do not contain the MAC address. We did not include the signal (SIG) symbol which is after the LTF due to it containing information on the length of the message. Although this still does contain the MAC address, it is variable message-by-message and would likely detract from learning the physical fingerprint of each device.

All devices used IEEE802.11g modulation on Channel 11 (2.462GHz) and are sampled at 20MHz. The receiver gain is set to 45.6 dB. The signal-to-noise (SNR) for the signals varies between 25-40 dB.

We also collected the CSI, using the first 128 samples which correlated with the LTF. The CSI is a complex-valued vector describing the channel distortion the signal experiences. For this work, CSI is calculated per OFDM symbol using the least-squares method.[13] IEEE802.11g has 52 subcarriers in the frequency domain, and each subcarrier corresponds to one complex sample in the CSI. The other 12 samples per symbol in the CSI are zeroed around the origin.

Identifier	Device Type	MAC Address
0	IPhone SE	A0:D7:95:1F:75:F7
1	Samsung Galaxy J2 Prime	F4:71:90:A1:D1:2C
2	IPhone SE	E0:5F:45:73:3D:F1
3	Moto G4 Plus	68:C4:4D:97:89:9E
4	IPhone SE	F0:79:60:7D:A2:13
5	Oppo R10	00:08:22:36:D8:FB*
6	Samsung Galaxy S5	10:A5:D0:D2:F0:2B
7	Samsung Galaxy S4	C0:BD:D1:0A:9F:EA
8	Samsung Galaxy S4	C0:BD:D1:0B:B7:9A
9	Samsung Galaxy S4	F4:09:D8:7D:55:98
10	Google Nexus	D0:13:FD:63:63:87
11	Google Pixel 2XL	B4:F1:DA:E8:A7:61*

Table 1: List of phones used in the dataset, along with their identifier in the provided code and MAC Address. *These devices have been witnessed randomizing WiFi MAC addresses.

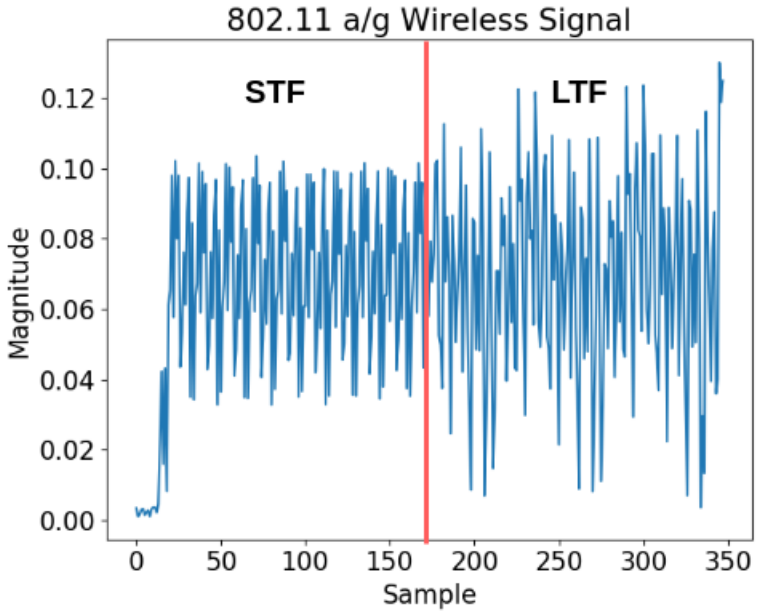

Figure 1: A captured wireless signal. The transient is the initial ramp-up shown in the firsts 30 samples. The LTF and STF each successively account for the next 160 samples each.

The data was partitioned using 67% of the data for training with the rest for testing. We did not use a validation dataset since parameters were not tuned throughout experimentation. Sklearn's train_test_split function[14] was utilized with the stratify parameter, ensuring an equal representation of each class is available in both the training and test set. This ensures a run does not have a lucky draw of easier classes to differentiate in the test case, such as the three iPhone^{TM} SEs, which would effect interpretation of results. Additionally, the random_state argument was set to ensure the same split occurred for all experiment settings.

We also created three different datasets for the various experiment settings.

1. Ant 1 - The signals collected from antenna 1.
2. Ant 1&2 - Signals from antenna 2 are concatenated to those from antenna 1.
3. Ant 1&2&CSI - The CSI is concatenated to Ant 1&2.

For all datasests, we took the magnitude of the complex-valued signals. Code for all experiments and datasets are available on GitHub at https://github.com/loganriggs/STFT_wifi_physical_fingerprint

3.2 MLP

We implemented classification using a MLP model with four layers of 64 neurons each using the sklearn implementation.[14] The alpha value represents the regularization parameter and was set to 1×10^{-4} to prevent overfitting. Max_iter is the maximum amount of iterations the MLP will run if it has not converged yet. This value was set to 1,000 to allow enough iterations to converge without costing too much time on each run. We used the Adam optimization algorithm due to being appropriate for sparse and noisy gradients.[15]

3.3 STFT

We convert the data into the STFT using the scipy signal library.[16] The nPerSeg parameter represents the size of the window length. Larger values equate to larger resolution in the frequency domain and a smaller resolution in the time domain. The noverlap argument is the overlap between segments, where larger overlaps will make the output larger with more interpolations between each FFT segment. This value was set to 3 based on empirical results. For all STFTs, we flatten them into a one-dimensional signal to fit into the MLP.

3.4 Window Length Parameter Sweep

Using the Ant 1&2&CSI dataset, we sweep the window length from 5 to 200 spaced apart exponentially to have an even spacing of the frequency resolutions. The best-of-10 results were calculated.

3.5 Settings

The MLP had a specified random state set which ensured the same weight initialization occurred, making the result deterministic. This was not used for the parameter sweep because that experiment calculated best-of-10, while the following settings were only run once.

Original: We perform classification while keeping the data in its original form consisting of the magnitude of the complex wireless signal. This is to give a comparison against the STFT version.

STFT: For the window length parameter, we use a value of 200 in all STFT settings based on the results from the sampling frequency parameter sweep.

STFT-Augmented: Similar to the STFT, except we augmented the data with Gaussian noise. We took the original training dataset and concatenated two copies of itself on the end. These copies had a random Gaussian vector of mean 0 and standard deviation of 0.5×10^{-3} added to it in the complex domain before the magnitude is calculated. This prevents the network from memorizing a wireless signal, noise and all, and instead incentivizes the network to identify the specific signal itself.

STFT-Concatenated: Similar to the STFT, except we calculated using the window length parameters 10, 30, and 200. We flatten the STFT images into one-dimension before concatenating the three. The motivation is to include components with both a higher time resolution and a higher frequency resolution, including the value 200 which had the greatest performance in the window length parameter sweep. This is to create a fair comparison with the STFT-Augmented which has three times the amount of data by adding noise, whereas this setting has 3x due to concatenating STFTs at varying frequency and time resolutions.

4. RESULTS AND DISCUSSIONS

For the window parameter sweep, the best-of-10 results are shown in Table 2. Values near 200 had the greatest performance. Since this parameter reflects frequency and timing resolution, the best performance was found at a very high frequency resolution.

Window Length (samples)	Time Resolution (microseconds)	Frequency Resolution (MHz)	Accuracy (%)
5	0.25	8.00	99.60
10	0.50	4.00	99.70
20	1.00	2.00	99.80
30	1.50	1.33	99.72
50	2.50	0.80	99.80
100	5.00	0.40	99.80
200	10.00	0.20	**99.94**

Table 2: Results from sweeping the window length parameter. As shown, values near 200 had the greatest accuracy overall. This corresponds to a more fine-grained frequency resolution.

The results for the four settings are shown in Table 3. The STFT marginally improves classification performance for all datasets. Augmenting the STFT produced a meaningful improvement in the smallest Ant 1 dataset, while concatenating produced worse performance on net compared to the STFT.

Setting	Ant 1	Ant 1&2	Ant 1&2&CSI
Normal	96.56	98.34	**99.15**
STFT	98.74	99.74	**99.94**
STFT Augmented	99.19	99.74	**99.94**
STFT Concatenated	98.46	99.68	**99.68**

Table 3: Classification accuracy of various settings. Bold font represents the maximum value of that row.

The STFT-Concatenated setting performed worse than simply augmenting, even though both have similar amounts of data. It may be the case that some machine learning problems benefit from both high timing and frequency resolutions, these results imply that classification of wireless physical fingerprints may only benefit with high frequency resolutions.

Another trend is greater performance when adding more data, whether that is the signals from antenna 2 or the CSI. Our previous work used complex neural networks to achieve 98.81% accuracy on nine devices with approximately 100 signals each, showing better performance after augmenting the data.[17] This work scored 99.94% as shown in Table 3 with three more devices, making it the more difficult classification problem. Our work's improved classification accuracy is likely due to having 15-45 times more data, meaning our previous work was data constrained.

Chen et al. achieved 80.5% accuracy classifying four wireless devices of the same make and model.[7] We do not have four of the same devices, but three iPhone^{TM} SEs and three Samsung^{TM} Galaxy S4s as shown in Table 1, achieving a much higher classification accuracy.

Zong et al. reached a high classification accuracy of 99.7% with five RF transmitters.[11] Although our work achieved a higher accuracy of 99.94% accuracy with 12 classes, it is more difficult to compare since their transmitter signals were generated while this work's datasets were collected from physical phones.

5. CONCLUSIONS AND FUTURE WORK

In conclusion, we applied a parameter sweep of the window length parameter which controls the frequency and timing resolution. Values around 200 were shown to have the greatest performance, implying a very high frequency resolution provided a better representation of the physical fingerprint than those with more timing resolutions.

We also performed classification in many settings showing a marginal increase in performance due to performing the STFT. This aids in validating that the physical fingerprint is better represented in the frequency domain. A smaller performance was realized by replicating the signal and adding complex Gaussian noise to the replications. This along with increased performance due to adding both the second antenna and CSI points to a small amount of being data constrained. Finally, we failed to increase performance by combining different STFTs of the signal at different frequency resolutions. This is further validated by it performing worse than the augmented data even though both settings have a similar amount of data.

Future work includes applying similar methods to long term evolution (LTE) signals due to them having similar identification randomization related problems. It would also be beneficial to use this dataset and possibly MLP models for outlier detection of wireless devices in an online setting.

ACKNOWLEDGMENTS

This research is supported by the National Institute of Justice (NIJ) grant 2018-75-CX-K002.

REFERENCES

[1] Apple, "Use private wi-fi addresses in ios 14, ipados 14, and watchos 7," (March 2021). [Online; posted 05-November-2020].

[2] Danev, B., Zanetti, D., and Capkun, S., "On physical-layer identification of wireless devices," *ACM Computing Surveys (CSUR)* **45**(1), 1–29 (2012).

[3] Klein, R., Temple, M. A., Mendenhall, M. J., and Reising, D. R., "Sensitivity analysis of burst detection and RF fingerprinting classification performance," in [*2009 IEEE International Conference on Communications*], 1–5, IEEE (2009).

[4] Suski II, W. C., Temple, M. A., Mendenhall, M. J., and Mills, R. F., "Using spectral fingerprints to improve wireless network security," in [*IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference*], 1–5, IEEE (2008).

[5] Padilla, J., Padilla, P., Valenzuela-Valdés, J., Ramírez, J., and Górriz, J., "RF fingerprint measurements for the identification of devices in wireless communication networks based on feature reduction and subspace transformation," *Measurement* **58**, 468–475 (2014).

[6] Torres-Sospedra, J., Montoliu, R., Martínez-Usó, A., Avariento, J. P., Arnau, T. J., Benedito-Bordonau, M., and Huerta, J., "Ujiindoorloc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," in [*2014 international conference on indoor positioning and indoor navigation (IPIN)*], 261–270, IEEE (2014).

[7] Chen, S., Xie, F., Chen, Y., Song, H., and Wen, H., "Identification of wireless transceiver devices using radio frequency (rf) fingerprinting based on stft analysis to enhance authentication security," in [*2017 IEEE 5th International Symposium on Electromagnetic Compatibility (EMC-Beijing)*], 1–5, IEEE (2017).

[8] Klein, R. W., Temple, M. A., and Mendenhall, M. J., "Application of wavelet-based RF fingerprinting to enhance wireless network security," *Journal of Communications and Networks* **11**(6), 544–555 (2009).

[9] Orfanidis, S. J., [*Introduction to signal processing*], Prentice-Hall, Inc. (1995).

[10] Xu, C., Chen, B., Liu, Y., He, F., and Song, H., "Rf fingerprint measurement for detecting multiple amateur drones based on stft and feature reduction," in [*2020 Integrated Communications Navigation and Surveillance Conference (ICNS)*], 4G1–1, IEEE (2020).

[11] Zong, L., Xu, C., and Yuan, H., "A rf fingerprint recognition method based on deeply convolutional neural network," in [*2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*], 1778–1781, IEEE (2020).

[12] Cybenko, G., "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems (MCSS)* **2**(4), 303–314 (1989).

[13] Kay, S. M., [*Fundamentals of statistical signal processing*], Prentice Hall PTR (1993).

[14] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research* **12**, 2825–2830 (2011).

[15] Kingma, D. P. and Ba, J., "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980* (2014).

[16] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al., "Scipy 1.0: fundamental algorithms for scientific computing in python," *Nature methods* **17**(3), 261–272 (2020).

[17] Smith, L., Smith, N., Hopkins, J., Rayborn, D., Ball, J. E., Tang, B., and Young, M., "Classifying wifi" physical fingerprints" using complex deep learning," in [*Automatic Target Recognition XXX*], **11394**, 113940J, International Society for Optics and Photonics (2020).