Listen to the Music: Evaluating the Use of Music in Audio Based Authentication

Michael Tsai and Vimal Kumar(✉)[0000−0002−4955−3058]

School of Computing and Mathematical Sciences,
University of Waikato, Hamilton 3200, New Zealand
mt296@students.waikato.ac.nz
vkumar@waikato.ac.nz

Abstract. Audio based authentication has been proposed to be used as a second or third factor of authentication in Multi-Factor Authentication (MFA). Previous audio fingerprinting work has mostly used tonal frequencies which are not ideal in authentication as humans do not like sharp tonal frequencies as audio. This work investigates the usage of music as the audio for authentication instead of tonal frequencies. We also compare music with Dual Tone Multi Frequency (DTMF) audio. We present the results of our experimentation over source audio, feature extraction and performance under noise in this paper. The results of our experiments show that music in fact offers advantages such as better accuracy and better performance under noise in addition to sounding pleasant.

Keywords: Authentication; Fingerprinting; Audio; Music; Mobile Devices

1 Introduction

Mobile device fingerprinting, which is the idea of gathering characteristics that could reliably identify each individual mobile phone is useful in a variety of scenarios. For instance, a vendor may wish to determine their target customers and advertise accordingly, in which case mobile device fingerprinting can be used to perform user tracking. Furthermore, this technique can also be used in criminal investigations [4] in which investigators may need to identify if a particular mobile phone was used in criminal activity. Finally, another major application is authentication, which is also the focus of this paper. Due to the ubiquity of mobile devices, it is convenient as well as cost-efficient to adopt them in multi-factor authentication schemes. Traditional mobile device fingerprinting techniques include the usage of IP addresses, cookies and other identifiers such as IMEI (International Mobile Equipment Identity) or UDID (Unique Device Identifier) for iOS devices. However, these methods are often subjected to user modification, which would be unsuitable in a high-security context such as an access control system. Another idea which has been proposed in previous work is sensor fingerprinting. Mobile devices often contain a large number of sensors.

For example, the audio sensors, which consist of the microphone and the speaker provide the basic functionality of recording and playing sounds. Other sensors such as gyroscope, accelerometer, magnetometer, ambient light sensors and others have also contributed to the abundant features in mobile devices. During the manufacturing process of the sensors however, there are often variations which result in inevitable hardware variances. These variances cause the sensors to produce different output when presented with the same inputs, even within the same make and model of the phones. These different outputs can provide enough entropy to construct reliable fingerprints and be used to perform hardware-level device identification [5].

Previous work in the area of audio based fingerprinting [5] [8] has focused on the usage of single or stepped frequencies with some success. We will use the term tonal frequency for these as the sound produced by them is a pure tone. If such fingerprinting techniques are to be used for authenticating humans that carry these mobile phones, some consideration also has to be given to how user friendly the techniques are. Tonal frequencies as it turns out, especially at the higher end of the audio spectrum are not user friendly at all. This in turn means that while audio based fingerprinting may have high accuracy, the technique may not be usable with tonal frequencies and there is a need to look at other types of source audio for it. In this paper we present our results of experimentation with music as the source audio for audio based authentication. In addition to music, we also experimented with Dual Tone Multi Frequencies (DTMF) commonly used in telephony.

2 Background and Literature Review

The technique of physical device fingerprinting has been proposed in previous research. For example, device-specific *clock skew* can be used to fingerprint a physical device [10]. However, it was not until *AccelPrint* [9] that fingerprinting through hardware sensors for the purpose of mobile device identification was utilised. The core concept of sensor fingerprinting is that natural variations in the sensor manufacturing process produce variances that can cause distorted output. This distortion is consistent and can be measured to form a fingerprint of a device.

Further work in sensor fingerprinting was carried out in [2] [3] [5], while [7] [8] [19], have specifically examined many different aspects of audio based fingerprinting. For audio sensors, such as the speaker and the microphone, the variances are exhibited in the frequency response. Bojinov et. al. proposed a technique in [5] that fingerprints both the speaker and the microphone simultaneously by playing a sound and recording it on the mobile phones. Dekker and Kumar in [8] use the same approach. In [8], which our work is based on, the authors experimented with several ideas to improve the existing audio-based sensor fingerprinting technique. These include an increased range of frequencies tested, different length of playback time and other features. The frequencies tested cov-

ered the entire audible range (200Hz to 20000Hz) with varying frequency steps and length of audio.

The types of source audio that can be used in audio based authentication can be divided into two categories - tonal frequencies and non-tonal frequencies. Tonal frequencies are the most common ones used in previous research [2] [5] [8] [19]. The other type includes audio such as music, DTMF, ringtones and human speech as described in [7]. This type of source audio has received less attention in literature. In this paper we use music and DTMF as source audio and evaluate their usefulness in audio based authentication systems.

3 Source Audio Selection

The source audio is an essential part of the audio sensor authentication process. The choice of which may affect the classification accuracy, noise resistance, user experience, authentication time and more. Dekker and Kumar [8], tested different variations of stepped-frequency. In this work, we focus our attention primarily on music as our source audio and investigate, classification accuracy, noise resistance, and other aspects of audio based authentication. For the purpose of comparison, we also investigate sequential and superimposed DTMFs. Below we describe our source audio in detail.

- Music — The use of music in the authentication process provides a pleasing user experience and therefore is preferable to tonal frequency based approaches. As opposed to the stepped frequencies used in [8], which consist of a single frequency at a time, music contains multiple mixed frequencies. The fingerprints extracted will therefore consist of features from the combination of many frequencies simultaneously, which may result in varying classification performance. In this work, we have selected several music samples from the GTZAN dataset [14] [17], a dataset commonly used for music genre classification. The dataset consists of ten music genres, which includes 100 music samples per genre. For our experiments, we have randomly selected one sample per genre, forming a total of ten samples for the music source audio. The selected music samples are listed in Table 1. Furthermore, we have reduced the length of the original music audio from 30 seconds to 3 seconds in order to reduce the total authentication time.
- Sequential and Superimposed DTMF Frequencies – For comparison we have also selected source audio that consists of the frequencies adopted in DTMF (Dual Tone Multi-frequency) tones. The DTMF tones are commonly used in telecommunication systems. More specifically, certain pairs of frequencies are used to represent each key press on a DTMF keypad. For example, when the key 1 is pressed, both 697 Hz and 1209 Hz are played. These frequencies are mainly selected to reduce harmonic interference as well as the possibility of the tones simulated by the human voice [13]. The main reason for adopting these frequencies are that they are not harmonic with each other, therefore they could be played together simultaneously to achieve shorter playback

time. In this type of source audio, we play them both sequentially and simultaneously. We have generated eight tones with the respective frequency of 697, 770, 852, 941, 1209, 1336, 1477 and 1633 Hz. For sequential DTMF audio, each frequency was played for 0.1 and 0.3 seconds. Meanwhile, for superimposed DTMF audio, each frequency was played for 0.1, 0.3 and 0.5 seconds. The 0.5-second sequential DTMF audio was not considered due to the resulting lengthy register and authentication time. Moreover, the selection of these durations was based off the frequency play time determined in [8], in which desirable results were obtained for the stepped frequency source audio. The frequency play time longer than one second was not considered in order to shorten the authentication time required.

4 Feature Extraction

The most common feature extraction methods used in the literature are based on extracting the frequency response at each of the played frequency to form the feature vector. However, this method cannot be directly applied to our source audio since music contains a mixture of different frequencies. Furthermore, there are additional features that can be extracted from a piece of audio [6] that may improve the overall classification performance. Below we discuss the Raw FFT feature used in the literature primarily and other spectral features from [6].

- Raw FFT and its variation – The most basic feature extraction method is to generate the feature vectors directly from the raw FFT output values. This method is easy to implement and can be applied to many source audios, however, this method would result in too many irrelevant features for music, which would drastically increase the resource consumption of the system while also negatively impacting the classification performance. Previous research has adopted some filtering techniques to only extract the frequency responses at the played frequency of the source audio. This method reduces the amount of features, while also having the ability to reject unwanted frequency responses from other sources such as noises.
- Spectral Features – In addition to the FFT values, there are other features that can be used to describe a piece of audio. These include features such as spectral centroid which represents the centre of mass of the spectrum, spectral bandwidth which calculates the average distance to each spectral centroid at each frequency bin and other various spectral features. To observe the effect of each spectral feature, we experimented with them individually. In addition, we also compared all the spectral features combined and evaluated it against directly using the FFT output values. The implementation of these features made heavy use of the Librosa Python package [11]. Furthermore, the number of features were reduced by taking the statistical summary of the features, such that the mean, standard deviation, maximum, minimum and median values were calculated to form the final feature vector [18]. The following describes the list of features implemented.

1. MFCC
   The MFCC feature describes the spectrum of an audio by taking into account how the human perceives sound by using a Mel scale. Based on [7], we have extracted 13 MFCC coefficients. We then take the statistical summary as stated previously to form the final feature vectors.
2. Poly Features
   Another useful spectral feature to extract is the poly features. This feature fits an n-th degree polynomial function to each frame of the spectrogram. To construct this feature, we have fitted a linear polynomial to each frame and have extracted the first and second array of coefficients.
3. Spectral Centroid
   The spectral centroid feature calculates the mean value per frame. The resulting feature vector consists of the centroids extracted for every frame. It was observed that the difference between the features were more significant than the previous implemented features and may be beneficial in producing a more accurate model.
4. Spectral Contrast
   The spectral contrast feature calculates the difference between the spectral peak and spectral valley at each sub-band. For the implementation of Librosa, the spectral peak is calculated by taking the mean energy in the top quantile and spectral valley the bottom quantile. It was observed that the spectral contrast values did not vary as much as the spectral centroid value, however, it is still superior to the poly features.
5. Spectral Bandwidth
   The spectral bandwidth feature computes the weighted average distance between each frequency and the spectral centroid in each frequency bin. This feature shows the distribution of frequencies relative to the spectral centroid.
6. Spectral Flatness
   The spectral flatness describes how tone-like or noise-like a sound is. It is computed by measuring the frequency distribution at each frequency sub-band. The resulting feature vector contains the spectral flatness for each frame.
7. Spectral Roll-off
   Finally, spectral roll-off computes a roll-off frequency for each frame, such that under which 85% of the spectral energy is contained. The spectral roll-off feature was also found to be quite distinct comparatively, which may be a suitable candidate for feature extraction as well.

Additionally, the spectral features can be concatenate to form a combined feature vector that describes all the above characteristics of a piece of audio.

5 Implementation

Firstly, the audio sensor fingerprints were collected off a number of devices by playing a pre-loaded source audio through their speakers and recording with

their microphones respectively. The collected fingerprints were pre-processed for feature extraction, after which they were utilised for training a machine learning classifier. Finally, the performance of the classifier was evaluated.

5.1 Source Audio Generation

The list of the selected source audio considered in the paper is shown in Table 1, along with the corresponding category, duration and the sampling rate.

	Category	Duration (s)	Sample Rate
dtmf-01	dtmf-seq	01.80	44100
dtmf-03	dtmf-seq	03.40	44100
music-blues-00079	music	03.07	22050
music-classical-00012	music	03.07	22050
music-country-00040	music	03.07	22050
music-disco-00022	music	03.07	22050
music-hiphop-00035	music	03.07	22050
music-jazz-00075	music	03.07	22050
music-metal-00009	music	03.07	22050
music-pop-00044	music	03.07	22050
music-reggae-00016	music	03.07	22050
music-rock-00006	music	03.07	22050
superimposed-dtmf-01	dtmf-sup	01.10	44100
superimposed-dtmf-03	dtmf-sup	01.30	44100
superimposed-dtmf-05	dtmf-sup	01.50	44100

Table 1: Complete list of source audio selected

5.2 Devices Under Test

The devices used in the experimentation are listed in Table 2. A total of 16 devices were used in this research.

Name	Type	Quantity
ASUS Nexus 7	Tablet	2
ASUS ZenFone 3	Phone	2
LG Nexus 4	Phone	3
LG Nexus 5	Phone	3
Motorola Nexus 6	Phone	2
Samsung Galaxy Nexus	Phone	4

Table 2: Device List

5.3 Experiment Architecture

The experiment flow consisted of loading the selected source audio in Table 1 onto each device. Each source audio was played through its speaker and recorded with its microphone. For every device, 20 audio samples were collected and sent to a remote server for further processing.

5.4 Environment

The environment for the experiment comprised of a student lab in a basement of a building. The room was situated at the end of a hallway, thus limiting the amount of outside noise caused by people passing and talking. However, there existed small amount of ambient noise caused by air conditioning and desktop PCs. The audio samples were collected under similar conditions, such that nobody was present in the room during the time of recording. The devices were also placed on the same desk to reduce the number of variables. The process of data collection lasted several weeks, during which no significant changes towards the environment were observed.

6 Classification and Evaluation

With the obtained feature vector derived from using the proposed feature extraction techniques, a machine learning model was trained. Given a set of audio sensor fingerprint features, the model needs to accurately identify the associated device for that fingerprint. To train such a model the audio dataset collected off the devices was split into training and testing sets. The dataset was split into training and testing set based on, the number of phones registered and the size of the training set. The training set was constructed such that *num_phones* amount of phones were first randomly selected from the pool of 16 devices, where *num_phones* indicates the number of registered phones. Afterwards, another *num_train* amount of phones were randomly selected from the 20 audio samples collected off the device, where *num_train* specifies the size of the training set. Similarly, the testing set was constructed first by including the devices used in the training set, while only including those audio samples that were not used in the training set. The resulting training and testing sets were then used in the classification process.

The classification algorithm adopted was selected based on the previous work [8]. In particular, the Random Forest algorithm was selected due to its superiority in performance. We used scikit-learn [12] for implementing Random Forest. Similar to [8], the default parameters were used and no hyperparameter tuning was performed.

7 Evaluating the Effect of Noise

A major challenge of audio fingerprinting is that noises are an inevitable source of disruption to the fingerprint collection process. One way of evaluating the

impact of noise is to collect the fingerprints in various environments and record the associated SNR. Although this should produce the most accurate result, it would require a significant amount of time and resources. Another way to evaluate the impact of noise is to add previously collected noise samples to audio samples. Although simulated, this approach allowed us to compare the effects of different types of noises at different SNR levels. In our approach, it was ensured that the length of noise sample was greater than that of the recorded audio. Therefore, for the processing of noise audio, we first selected a random consecutive portion from the noise audio with the same length as the recorded audio. Afterwards, the noise audio signal was scaled down to the desired SNR by the following formula.

$$SNR_{dB} = 20log_{10}(\frac{A_{signal}}{A_{noise}}) \tag{1}$$

where A_{signal} denotes the RMS value of the original signal and A_{noise} denotes the RMS value of the noise signal. Finally, the noise signal was added onto the recorded audio signal. The resulting signal was then clipped between 0 and 2^{16} (int max) to avoid the issue of cracked sounds.

All the recorded samples were mixed with three types of noises recorded under different scenarios - busy university restaurant, small office noise and data centre noise. The first two noise samples were obtained from the DEMAND dataset [15] while the last noise sample was downloaded from the Freesound website [1].

Name	Type	Sampling Rate
PRESTO_ch01	University Restaurant	16K
OOFFICE_ch07	Small Office	16K
data_centre	Data Centre	44.1K

Table 3: Selected Noise Samples

8 Results

In this section we present the classification accuracy when different source audio are used along with the usage of various feature extraction techniques. There are two main variables in all of the experiments - number of phones registered and number of audio samples available. In the context of the model training stage, this represents the number of classes and the training samples, respectively. We aim to show the performance of the classification model under various conditions when adopted in an authentication system. For example, the number of phones registered may provide an overview of the effectiveness of the system from the beginning of user enrollment to a state where a number of phones have already been enrolled. Meanwhile, the number of training samples indicates the amount of audio samples to collect for the registration process. This amount directly

impacts the time required for users to be enrolled into the system and thus data aggregated by such variable is beneficial to determine the balance between system efficiency and effectiveness. Overall, the results showed a trend in which the accuracy of the classifier slightly decreases as more phones are enrolled into the system. On the other hand, the classifier performance improved as more training samples were available.

8.1 Source Audio Comparison

We first take a look at the accuracy obtained when using various source audio. We used various genres of music as well as sequential and superimposed DTMF tones. We used the combined spectral features for this experiment. We can see in Figure 1 that over 99% of accuracy was achieved across all source audio, suggesting that the use of source audio does not impact classification performance greatly. Nevertheless, slight difference in the performance was observed from these results. For example, over half of the music category source audio achieved an accuracy of over 99.9% while blues, classical, country and disco music are on the slightly lower side. This suggests that it is possible to adopt certain types of music in audio sensor fingerprinting in order to improve the user experience while achieving robust fingerprint identification capability. Furthermore, all results from superimposed DTMF also achieved over 99.9% accuracy, which suggests the possibility of using shorter source audio (1 second) for extracting the audio sensor fingerprints. Since more than one sample is required for registering a particular device, this result suggests that the time required for registering and authenticating a particular device can be significantly reduced.

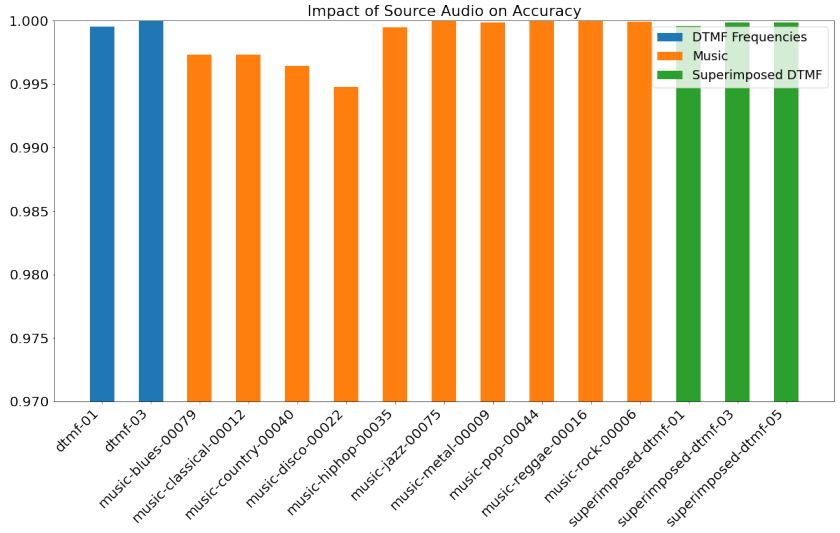
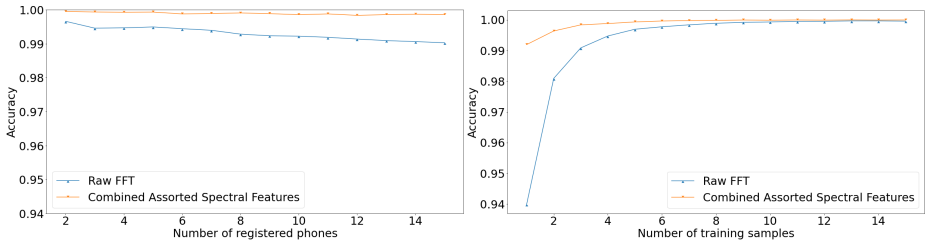


Fig. 1: Source Audio Accuracy Result

8.2 Feature Comparison

We then compared the various features that could be extracted and used for audio based authentication. Dekker and Kumar [8] only targeted stepped frequency source audio but to generalise the audio based authentication idea to a wider range of source audio such as music we need more nuanced features. Additionally, different features produce different length of feature vectors. This could impact the training and testing time of the ML model, thus in turn affect the registration and authentication time. Last but not least, different source audio may require different feature extraction methods in order to reach the peak performance, thus a comparison of such techniques could provide insight into the best combination of source audio and the corresponding features to extract.

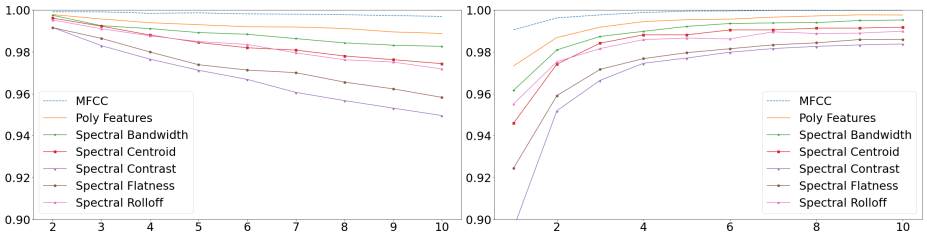


(a) Varying number of registered phones (b) Varying the number of training samples

Fig. 2: Raw FFT versus Combined Assorted Spectral Features

Raw FFT versus Combined Assorted Spectral Features Figure 2a shows the accuracy comparison of the two different features used at various number of phones registered. The first conclusion that could be drawn from this result is that the accuracy decreases as more phones were registered. In fact, we also experimented with filtered FFT as a feature but it degrades severely with the number of phones and therefore we have excluded it from the discussion in this paper. It can be easily observed that the use of combined spectral features produced the highest accuracy of nearly 100% while raw FFT achieved slightly worse result albeit maintaining its accuracy around 99%. Similar pattern could also be observed from the aspect of training samples. Figure 2b shows that the combined spectral features achieved over 98% accuracy with only one training sample, whereas the raw FFT approach started at approximately 93%. Despite the fact that the accuracy of raw FFT features converged with that of the combined spectral features at around 9 training samples, it should be noted that this number of training samples indicates a significantly longer registration time. For example, the *dtmf-01* source audio would take around 16 seconds of play time (exclusive of pause time) while the music source audio would take around 27

seconds. On the other hand, the *dtmf-01* source audio would only take around 5 seconds of play time and the music source audio would take around 9 seconds with combined spectral features. Furthermore, the number of raw FFT features are significantly more than that of the combined spectral features resulting in more intensive computation. Therefore the combined spectral features outperform the raw FFT features.



(a) Impact of Number of Registered Phones (b) Impact of Number of Training Samples

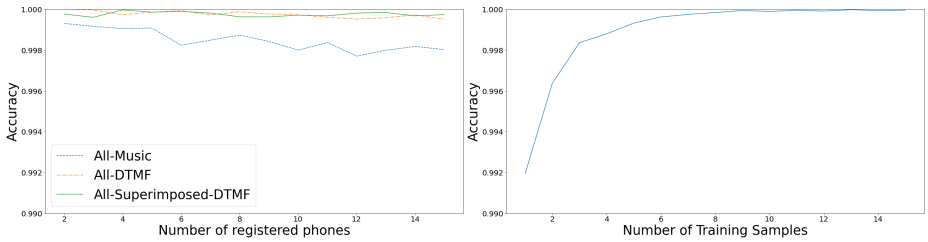
Fig. 3: Comparison Between Individual Spectral Features

Comparison Between Individual Spectral Features Figure 3a shows the accuracy comparison of the seven different spectral features used. It was observed that the MFCC feature performed the best out of all the spectral features, which is consistent with the findings in [7]. The next high-performing features were the poly features and the spectral bandwidth, followed by the spectral centroid and spectral roll-off. Two of the worst performing features were the spectral contrast and spectral flatness, which dropped from approximately 99% accuracy to around 95%- 96% accuracy with 10 registered phones. The issue may have lied in the fact that these extracted features did not have as much variation as the other features, thus increasing the difficulty for the Random Forest classifier. From the aspect of training samples as shown in Figure 3b, we can also observe the same pattern. The MFCC feature achieved approximately 99% with one training sample, which increased to near 100% around five training samples. On the other hand, the spectral contrast performed the worst by achieving less than 90% using one training sample. Nevertheless, it still achieved over 95% with three training samples.

8.3 Impact of phones registered

It was observed during all the experiments that the performance of the classifiers decreased as more phones were registered. Figure 4a shows the accuracy obtained at different number of registered phones for different types of source audio. The features used were the combined spectral features, and the results were averaged over all the audio within each type. The graph showed several fluctuations but

an overall trend of decrease in accuracy. Despite the accuracy being maintained above 99.7% for the maximum number of registered phones available in this experiment, the scalability of this phenomenon for a larger number of phones was unclear due to lack of phones and time available for testing. A definite answer to this would require further experiments with a significantly larger pool of devices.



(a) Impact of number of phones (b) Impact of number of training samples

Fig. 4: Impact of phones and training samples when considered with various audio types and combined spectral features

8.4 Impact of Training Samples

The number of training samples also has a noticeable impact on the performance of the classifier. Figure 4b shows that the average accuracy across all source audio under the adoption of combined spectral features increased with the number of training samples. It could be observed that with one training sample, an average of 98.8% accuracy was achieved. However, with nine training samples, the accuracy increased to near 100%. As mentioned previously, the number of training samples affect the time required for registering a device.

8.5 Impact of Noise

In order to evaluate the suitability of audio sensor fingerprinting in real-world scenario, we conducted experiments to simulate noise in the classification process. This was performed by adding the noises outlined in Table 3 at SNR levels of 5, 10 and 15 to the training set as well as the testing set. The processed audio then had the combined spectral features extracted for training and classification. Figure 5 shows the impact of noises on accuracy for an SNR of 10.

The result shows that with a fixed SNR of 10, the university restaurant noise had a greater impact on the accuracy compared to the small office and the data centre noise. It also indicated that under certain scenario, the accuracy of the classifier could decrease to around 90% or even lower. This showed that the proposed technique may not maintain its high performance in noisy environments

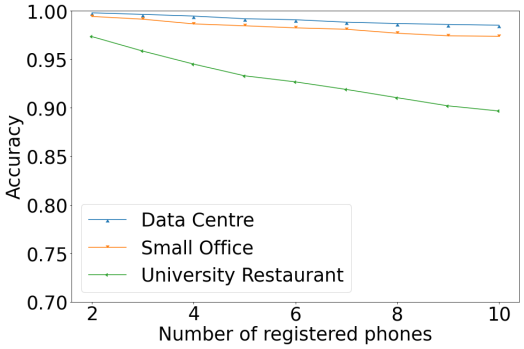


Fig. 5: Impact of Noise Types on Accuracy, SNR=10

such as a university cafeteria, however, it could still be applicable in places with less amount and variation of noise such as a small office or a server room.

Different Source Audio The result from the noise simulation was aggregated based upon different types of source audio in order to discover potential noise-resistant source audio. It was found that stepped frequency as used in the earlier work of Dekker and Kumar [8] perform poorly with noise. As our focus in the paper is on music we will skip the details of stepped frequencies but more information can be found in [16]. However, as Figure 6 shows, the noise simulation hardly impacted the classification performance when music was used. Despite some noticeable difference in performance under the impact of university restaurant noise, the overall accuracy was above 98%. This result indicates that the music source audio may be more suitable for collecting audio sensor fingerprints in a noisy environment.

Different SNR Finally, the three types of noise were simulated under different SNR to provide insight into the effectiveness of audio sensor fingerprinting under various amount of background noise. The results for the university restaurant noise in Figure 7a showed that in an environment that is heavily polluted with noise (under SNR of 1), the accuracy could drop to an average of 65% across all source audio with a single training sample. The situation could be improved to achieve over 80% accuracy with at least three training samples. However, the performance gain from increasing the number of training samples would eventually become marginally small. The results for the small office and data centre are presented in Figure 7b and Figure 7c respectively. Similar patterns could be observed in the graphs, however, the impact of these two source audio is much less than the university restaurant noise. In the case of data centre noise, the accuracy could still be maintained well above 90% under an SNR of one. Meanwhile for the small office environment, the accuracy could be maintained above 90% if SNR is larger than 5. It should also be noted that the accuracy

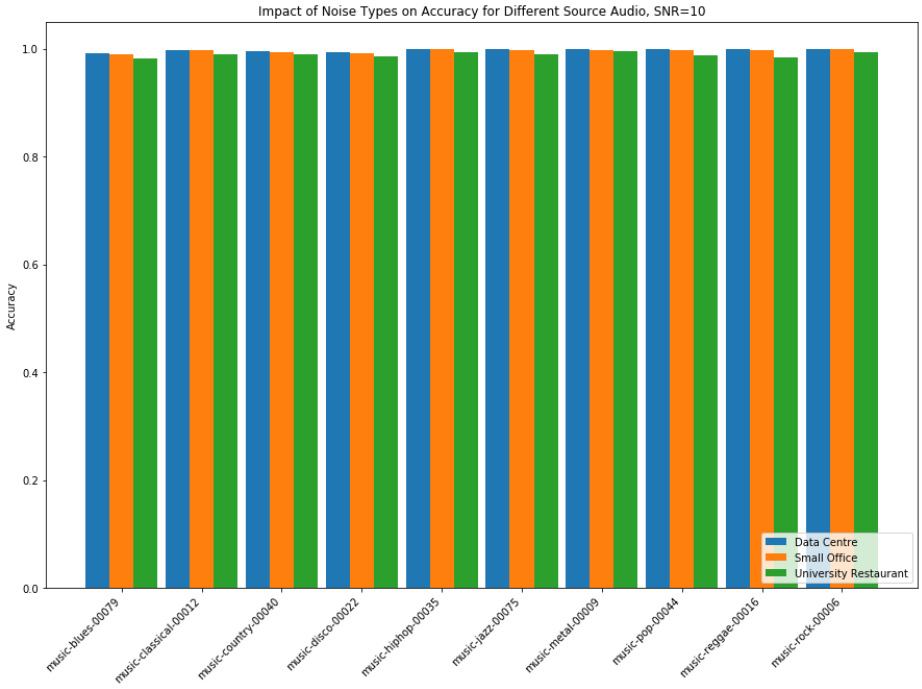
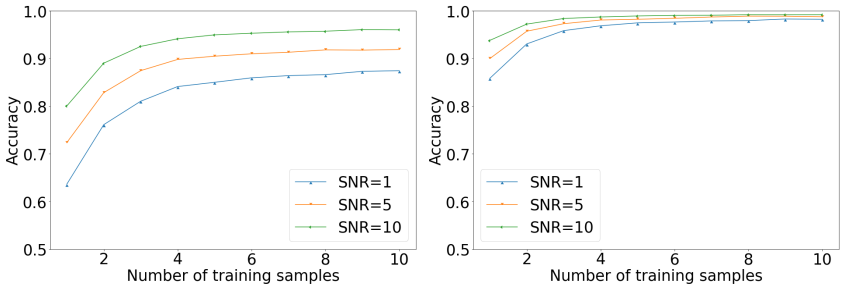


Fig. 6: Impact of Noise Types on Accuracy for Different Source Audio, SNR=10

obtained was trained on the combined spectral features and averaged over all the selected source audio. This could potentially be improved by pairing each source audio sample with their best-performing features, creating an even more robust audio sensor fingerprinting system.

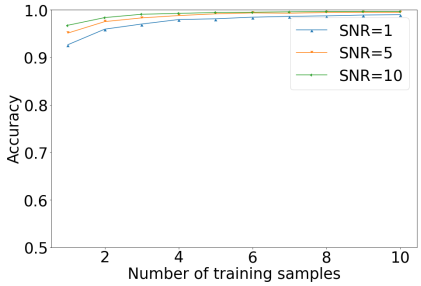
9 Conclusion

The source audio and feature extraction evaluation in this paper show various interesting results. Our evaluation shows that audio with various mixed frequencies such as music and DTMF can provide high classification accuracy with very short samples. We provided evidence that the combined spectral features work best with the music source audio. It was also observed that the MFCC feature was the most important feature from the combined spectral features, and that it might be feasible to extract only the MFCC features for classification in order to achieve faster training and classification time while maintaining certain level of identification accuracy. Our evaluation with noise show that different types of noises have different effect on the classification accuracy. Therefore, this technique may require more refinement in environments such as a noisy university restaurant, however, it may still work in other environments such as a data



(a) Uni. Restaurant Noise

(b) Small Office Noise



(c) Data Centre Noise

Fig. 7: Impact of Noise at Different SNR

centre and a small office. The results have also provided a measurement of accuracy with different levels of noise, which should help to determine the maximum tolerable noise in an environment for this technique to be viable.

In summary, the overall results showed that mobile device classification could achieve over 99% accuracy with various feature extraction mechanisms on multi frequency audio such as music and DTMF. Such audio are also more resistant to noise than tonal frequencies and provide a more pleasant experience to the user during authentication. Taken together this presents a very good case for music to be used in audio based authentication.

Acknowledgments This work was supported in part by Sir William Gallagher Cyber Security Scholarship at the University of Waikato

References

1. Freesound. `https://freesound.org"`. Accessed: 2020-12-26
2. Amerini, I., Becarelli, R., Caldelli, R., Melani, A., Niccolai, M.: Smartphone fingerprinting combining features of on-board sensors. IEEE Transactions on Information Forensics and Security **12**(10), 2457–2466 (2017)
3. Amerini, I., Bestagini, P., Bondi, L., Caldelli, R., Casini, M., Tubaro, S.: Robust smartphone fingerprint by mixing device sensors features for mobile strong authentication. Electronic Imaging **2016**(8), 1–8 (2016)

4. Baldini, G., Steri, G.: A survey of techniques for the identification of mobile phones using the physical fingerprints of the built-in components. IEEE Communications Surveys & Tutorials **19**(3), 1761–1789 (2017)
5. Bojinov, H., Michalevsky, Y., Nakibly, G., Boneh, D.: Mobile device identification via sensor fingerprinting. arXiv preprint arXiv:1408.1416 (2014)
6. Cano, P., Batlle, E., Kalker, T., Haitsma, J.: A review of audio fingerprinting. Journal of VLSI signal processing systems for signal, image and video technology **41**(3), 271–284 (2005)
7. Das, A., Borisov, N., Caesar, M.: Do you hear what i hear? fingerprinting smart devices through embedded acoustic components. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, pp. 441–452 (2014)
8. Dekker, M., Kumar, V.: Using Audio Characteristics for Mobile Device Authentication, pp. 98–113 (2019). https://doi.org/10.1007/978-3-030-36938-56
9. Dey, S., Roy, N., Xu, W., Choudhury, R.R., Nelakuditi, S.: Accelprint: Imperfections of accelerometers make smartphones trackable. In: NDSS. Citeseer (2014)
10. Kohno, T., Broido, A., Claffy, K.C.: Remote physical device fingerprinting. IEEE Transactions on Dependable and Secure Computing **2**(2), 93–108 (2005)
11. McFee, B., Lostanlen, V., Metsai, A., McVicar, M., Balke, S., Thomé, C., Raffel, C., Zalkow, F., Malek, A., Dana, Lee, K., Nieto, O., Mason, J., Ellis, D., Battenberg, E., Seyfarth, S., Yamamoto, R., Choi, K., viktorandreevichmorozov, Moore, J., Bittner, R., Hidaka, S., Wei, Z., nullmightybofo, Hereñú, D., Stöter, F.R., Friesch, P., Weiss, A., Vollrath, M., Kim, T.: librosa/librosa: 0.8.0 (2020). https://doi.org/10.5281/zenodo.3955228. URL `https://doi.org/10.5281/zenodo.3955228`
12. 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., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
13. Rey, R.: Engineering and operations in the Bell system. AT&T Bell Laboratories (1983)
14. Sturm, B.L.: An analysis of the gtzan music genre dataset. In: Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies, pp. 7–12 (2012)
15. Thiemann, J., Ito, N., Vincent, E.: Demand: a collection of multi-channel recordings of acoustic noise in diverse environments. In: Proc. Meetings Acoust (2013)
16. Tsai, M.: Optimisation of audio-based sensor fingerprinting system for mobile device authentication. Master's thesis, The University of Waikato (2021)
17. Tzanetakis, G., Essl, G., Cook, P.: Automatic musical genre classification of audio signals (2001). URL `http://ismir2001.ismir.net/pdf/tzanetakis.pdf`
18. VanderPlas, J.: Python data science handbook: Essential tools for working with data. " O'Reilly Media, Inc." (2016)
19. Zhou, Z., Diao, W., Liu, X., Zhang, K.: Acoustic fingerprinting revisited: Generate stable device id stealthily with inaudible sound. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, pp. 429–440 (2014)