SRVoice: A Robust Sparse Representation-based Liveness Detection System

Jiacheng Shang[†], Si Chen[‡], and Jie Wu[†]
[†]Center for Network Computing, Temple University, Philadelphia, PA 19121
[‡]Computer Science Department, West Chester University of Pennsylvania, West Chester, PA 19383
Email: {jiacheng.shang, jiewu}@temple.edu, schen@wcupa.edu

***Abstract*—voiceprint-based authentication is fast becoming the everyday norm since it is much easier to use and provides better security. However, current voiceprint-based authentication systems are vulnerable to various replay attacks. To tackle the spoofing attacks, we propose a new system that leverages the structural differences between human vocal system and loudspeakers and use the unique vibration pattern of both human vocal cord and throat as a key differentiating factor for liveness detection. Specially, we model the relationship between voices collected by two microphones of a smartphone of each live speaker using sparse representation. Compared with existing systems, our solution does not assume any prior knowledge of the attack method and is easy to operate. Moreover, our solution leverages the audio signals within the vocal frequency range and is robust to jamming attacks using high-frequency audio. Experimental results show that our system can achieve accurate liveness detection for a 6-digit passphrase with a mean true acceptance rate of 99.04% and true rejection rate of 100%.**

***Index Terms*—Voice authentication; Liveness detection; mobile computing.**

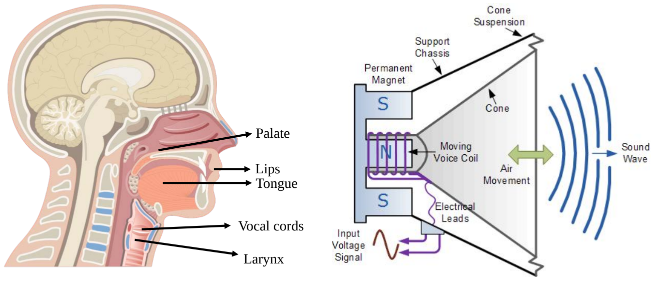
I. Introduction

Voiceprint-based authentication through technology is fast becoming the everyday norm. Compared with password or pattern-based authentication systems, voiceprint-based authentication systems are much easier to use and provides better security. Thanks to the powerful hardware provided by current smartphones and more accurate speaker identification techniques, Many voiceprint-based authentication systems have been deployed on smartphones. For example, by using "Voiceprint" [18] designed by WeChat, users can log into their accounts by speaking a fixed passphrase. Besides, solutions provide by SayPay [12] can fuse mobile payments by leveraging users' voice. VoiceIt and Microsoft have also published different APIs that enable developers to design voiceprint-based authentication solutions. Basically, those systems and APIs ask users to speak a passphrase from the given list of phrases and record several audio samples to register user's voice. If the speaker claims to be of a certain identity, the recorded voices will be used to verify this claim.

However, voiceprint-based authentication systems are vulnerable to various replay attacks. Since voices can be recorded, simulated or even imitated, an attacker can easily steal a person's voice with the availability of high quality and low-cost handy recorders and other recording devices (e.g., smartphones). The leakage of victims' voices cause lots of security issues, which pose a severe threat to voiceprint-based authentication systems [9, 13, 19]. For instance, a strong attacker could impersonate the victim by performing state-of-the-art speech synthesis techniques as long as the attacker acquires enough victim's voice. The synthetic voices are then used to spoof the voiceprint-based authentication systems. Our experiments also show that the attacker can easily spoof WeChat Voiceprint by recording and replaying victim's voice using the speaker of a smartphone. Since voice is considered as unique for each person and a basis for personal authentication [4], victim's safety, reputation, and property are under severe threat if we cannot resist these attacks.

Traditionally, voiceprint-based authentication systems defend against voice-spoofing attack by implementing an automatic speaker verification (ASV) system. This idea has been adopted by many popular application, such as WeChat. The ASV systems compare the extracted features of incoming voices with those that are from the claimed speaker and already registered in the database. However, spoofing attacks against ASV systems are also be improved greatly [6, 9, 19]. An attacker can perfectly impersonate the victim voice by replaying victims' voices to the voiceprint-based authentication systems. Moreover, current ASV systems need to have prior knowledge of specific voice spoofing techniques used by the attacker [5], which greatly limited their abilities against various spoofing attacks. To address this issue, many liveness detection systems are proposed by studying the differences between human vocal system and loudspeakers on how they produce voices. VoiceLive [22] can fight replay attacks by capturing time-difference-of-arrival (TDoA) changes in a sequence of phoneme sounds to the two microphones of the phone. However, it needs the same relative location of user's mouth during authentication, which is hard to satisfy in practice. A liveness detection system is proposed in [21] and can detect a live user by leveraging the unique articulatory gesture of the user when speaking a passphrase. However, it cannot work if the attacker performs a jamming attack using high-frequency audio.

To address limitations of existing solutions, we propose a new liveness detection system for voiceprint-based authentication systems to fight against replay attacks. Our solutions are designed based on the following fact: The human vocal systems and loudspeakers differ a lot for their structures and how they produce voices. Compared with existing systems,



(a) Human vocal structure (b) Speaker's structure

Fig. 1. The differences between human vocal system and loudspeaker.

our solution is ready to use and can be seamlessly deployed on off-the-shelf smartphones and does not assume any prior knowledge of the attacking method and is easy to operate. Moreover, our solution leverages the audio signals within the vocal frequency range and is robust to jamming attacks using high-frequency audio. In order to fight replay attacks, we leverage the unique vibration of vocal cords while a person is speaking a passphrase to a voiceprint-based authentication system. The human voice can be divided into voiced and unvoiced parts. The voiced part is produced by the vibration of vocal cords, while the unvoiced part is produced by the articulators. By attaching a mircrophone to the throat, we can capture the voiced part produced mainly by the vocal cords. Moreover, there is a relationship between the voice captured at the mouth and the throat, and this relationship is unique for different people and different words. Different from the human vocal system, the vocal structure of a loudspeaker has its sound coming from the same place, which means that we cannot observe the relationship if the voice is from a loudspeaker.

We summarize our contributions as follows:

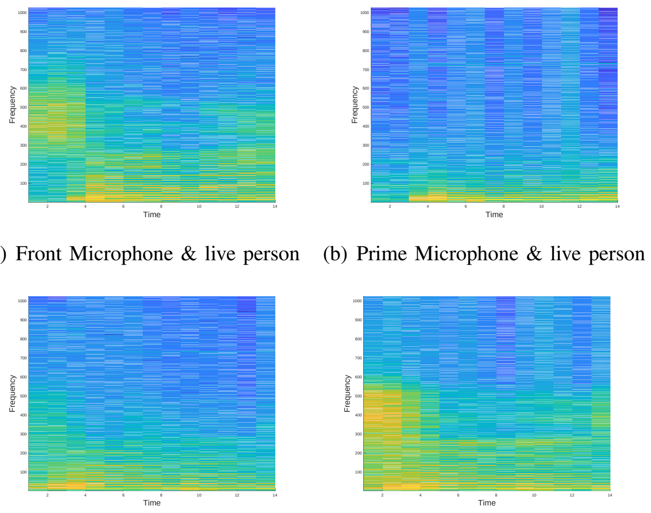
- Our solution leverages existing sensors on most smartphones and can be easily implemented on existing smartphones as a software or plug-in.
- We model the relationship between voices collected by two microphones and prove that the relationship is unique for each word and each person.
- We develop a prototype and conduct comprehensive evaluations. Experimental results show that our system can achieve accurate liveness detection for a 6-digit passphrase with a mean true acceptance rate of 99.04% and true rejection rate of 100%.

II. PRELIMINARIES

In this section, we will first introduce structures of the human vocal systems and loudspeakers. Based on the analysis, we will discuss the key insights that inspire us proposing new solutions and the use case of our system.

A. Background knowledge

In the human vocal system, the vocal cords are the primary sound source to produce voiced phoneme. Besides voiced phoneme, there exist other sound production mechanisms produced by the same general area of the body, involving the production of unvoiced consonants, clicks, whistling and whispering, as shown in Fig.1(a). The mechanism for producing the human voice can generally be subdivided into three parts; the lungs, the vocal folds, and the articulators. The lungs first produce adequate airflows and air pressure to vibrate vocal folds. The vocal cords vibrate and chop up the airflow from the lungs into audible pulses that form the laryngeal sound source. Then, the length and tension of the vocal cords are adjusted to produce 'fine-tune' pitch and tone. The articulators consisting of tongue, palate, cheek, lips further filter the sound generated from the larynx to strengthen it or weaken it. This suggests to us that the audio signals collected near the throat and the mouth may be different, and this difference can only be produced by the human speaker.



(a) Front Microphone & live person (b) Prime Microphone & live person

(c) Front Microphone & loudspeaker (d) Prime Microphone & loudspeaker

Fig. 2. The spectra of audio samples collected from two microphones when the speaker is a live person and a loudspeaker, respectively.

Replay attackers use one or more high-quality loudspeakers to replay victim's voices to the authentication system. As shown in Fig. 1(b), the loudspeakers use a voice coil to translate an electrical signal into an audible sound. When there is an electric current flows through the voice coil, an magnetic field is created around it. When electrical pulses pass through the coil, the direction of the magnetic field is also frequently changed. With a rapidly changing magnetic filed, the coil is attracted to and repelled under the influence the permanent magnet. As a result, the cone attached to the coil will vibrate back and forth, pumping sound waves into the surrounding air. If we put the smartphone near to the loudspeaker, the two microphones of a smartphone around the loudspeaker will capture very similar audio signals.

B. Key insights

In order to resist two types of attacks we considered, we need to leverage the structural differences between human vocal systems and loudspeakers discussed in Section II-A. We observe that human voice can be divided into the voiced and unvoiced parts. In the voiced part, the vocal cord keeps vibrating and generates low-frequency audio signals around the

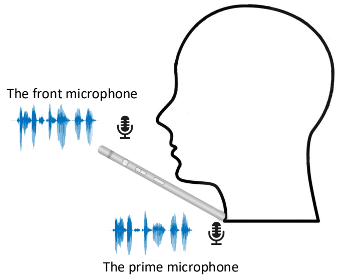


Fig. 3. Use case.

human throat. The vocal cord stops vibrating during unvoiced part, while the mouth vibrates to generate different sounds. We collect voice signals when the user says "Six" at two locations (the throat and the mouth) using two microphones, and the results are illustrated in Fig. 2. It is clear that the audio signal collected near the mouth reserves the information of the unvoiced parts, but the majority of this information is lost in the audio signal collected near the human throat. Also, both audio signals reserve the information of voiced part, while the audio signal collected near throat only contains the information at low frequency. Different from human vocal systems, the cone keeps vibrating for both voiced and unvoiced parts in order to generate sounds. We use a loudspeaker to imitate human vocal system and collect the audio signals in the same way. Fig. 2 also shows the spectrum of the same audio signal played by a loudspeaker and captured by the prime microphone. We can observe that the spectrum shows much more information of unvoiced parts than that collected from the voice of a throat.

C. *Use case*

To defend users from spoofing attacks using our system, the user needs to put the bottom side of the smartphone on the throat while using the normal voice authentication systems, as shown in Fig. 3. To capture the voices from at the throat and the mouth, we leverage two microphones equipped with current smartphones. Specially, the prime microphone is used to capture the low-frequency voice caused only by the human throat, and the front microphone is used to record human voice on the whole frequency band. Two audio signals are well synchronized by smartphones' operating systems. Moreover, he distance between the human throat and the prime microphone must be zero, and the distance between human lips and the front microphone is about $10cm$. Since the distance is pretty short, the time delay between two audio signals is less than 14 samples when the sampling rate is 44,100 samples per second. During authentication, the user speaks a passphrase to the smartphone as usual while keeping the bottom side of the smartphone on the throat.

III. Attack model

The attackers in our system aim to attack the biometric identification of the normal user. In our system, we consider three replay attacks. In each attack, an attacker tries to steal victim's voices and replay them to a voiceprint-based authentication systems. In all attack models, the attackers cannot steal the voice at victim's throat. Also, the attackers can only alter the input of our liveness detection system and cannot get/replace the voice in any middle stage of the liveness detection.

Mimic attack. The mimic attacker tries to attack the voiceprint-based authentication system by imitating victim's voice. In this attack model, the attacker can only physically access victim's smartphone but cannot record victim's voice.

Replay attack. The replay attacker has all the abilities of the mimic attacker and can get victim's voice at the mouth by all means. The replay attacker tries to fool the voiceprint-based authentication system by replaying victim's voice using a loudspeaker.

Reconstruction attack. The replay attacker has all the abilities of the replay attacker. Besides, the reconstruction attacker knows all the details of our solutions and tries to reconstruct the voice at victim's throat based on the voice at victim's mouth. The reconstruction attacker can observe the relationship of his/her own two audio samples and design a low-pass filter to reconstruct the voice at the throat by selecting various cut-off frequencies. Then, the attacker uses two loudspeakers to fool the voiceprint-based authentication system. One loudspeaker replays the voice at victim's mouth to the front microphone, and the other one replays the reconstructed voice to the prime microphone.

IV. System design

A. *Challenges*

In order to design a robust liveness detection system, several challenges need to be addressed.

Representation of the relationship and differences between two voices. The first challenge is how to represent the relationship between two voices collected from two microphones for a live speaker in a proper way, so that we can use the relationship as a pattern to distinguish between a live speaker and a loudspeaker. One possible solution is to compute a mapping function between the spectra of two audio signals based on the following equation:

$$y = M * x \tag{1}$$

where x is the voice sample captured at the mouth, y is the voice sample captured at the throat, and M is the mapping function. However, the solution is not unique, and the optimal solution is hard to be found in polynomial time. A new scheme that requires fewer computation resources needs to be proposed to describe the relationship and differences between two voices. In our system, instead of computing x, we use the differences between two voices in both time domain and frequency domain to represent the relationship and difference. Our experiments show that the differences between two voices are very significant for a live speaker and a loudspeaker.

Uniqueness of the relationship between two voices. Although we can leverage the difference between two voices to distinguish a live speaker or an attacker with a loudspeaker, the system is still under threat of strong attackers who try to reconstruct the voice at victim's throat. It is important to prove that the uniqueness of the relationship for different people.

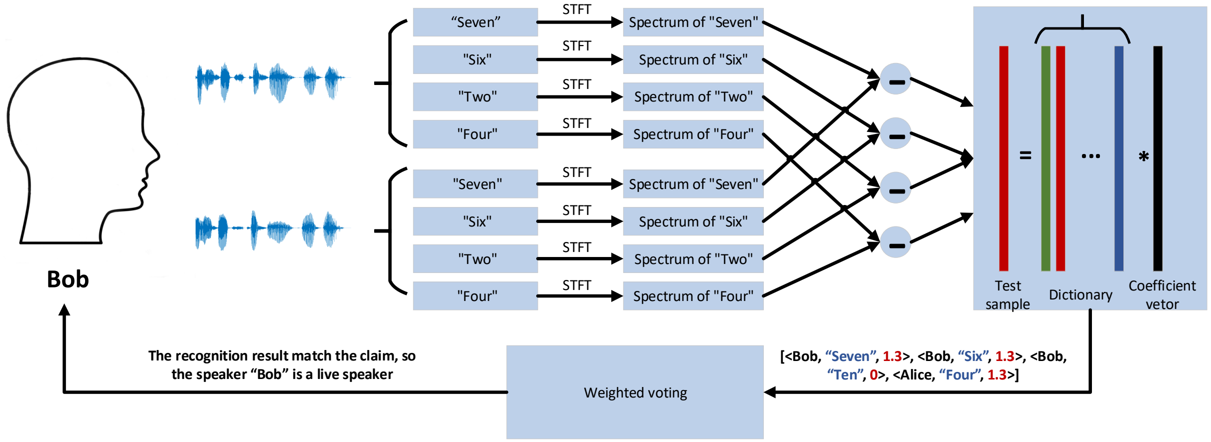


Fig. 4. System overview.

Otherwise, the attacker can easily reconstruct victim's voice by leveraging the relationship existing in his/her own voices. In this paper, we prove that the relationship is unique for each person and each word based on our dataset.

Robust liveness detection under various noises. The acoustic environment is dynamic in practice, which poses a challenge to our liveness detection system. The dynamic noise will alter the feature in an unpredictable way. To solve this problem, we leverage a sparse representation-based classification model that can achieve a great performance even when the input includes noise.

B. System overview

The key idea underlying our liveness detection system is to fully leverage the unique vibration of vocal cords while the user is speaking. When a person utters a passphrase in the way described in Section II-C, the primary microphone at the bottom records the sound mainly produced by the vibration of vocal cords, while the front microphone records the voice coming from the mouth. These two voice collected from a real person have hidden relationships, which can be used to detect if the speaker is a human or a loudspeaker. We explore the difference between two voice in both frequency and time domain and prove that a unique pattern exists for each person and each phoneme. The unique patterns of each person are used to distinguish if the voice is from a loudspeaker or another person.

Fig. 4 shows the pipeline of our system when a speaker "Bob" speaks a 4-digit passphrase to our system. Each voice is first segmented into non-overlapping words through Hidden Markov Model (HMM). Then, we perform Short-Time Fourier Transform (STFT) on the audio sample of each separated word to get the energy distribution at both frequency domain and time domain. A sparse representation-based classification model is designed to determine if the two voices of each word satisfy the relationship of the argued speaker already stored in the dictionary. The final decision is made by involving the liveness detection results of all four words in a weighted voting game. If the voting result is the same as the argued speaker, the speaker is regarded as passing the liveness detection.

C. Word segmentation and feature extraction

The two voices recorded by two microphones include two parts: the passphrase and background noise. The passphrase part contains abundant features of the speaker's voice, while the noise part only records the acoustic noise in the background. In our system, we only focus on the passphrase part in order to reduce the influence of the acoustic noise in the background. Since the audio sample recorded by the front microphone reflects the real voice, we split each audio sample into different words by performing HMM-based word segment techniques [11] on the audio sample of the front microphone.

Also, we need to find features to establish the relationship and differences between two voices collected from two microphones to distinguish whether the voice is from a live speaker or a loudspeaker. In order to capture features on both frequency domain and time domain, we perform STFT on each word and each audio sample with a window size of 46ms based on:

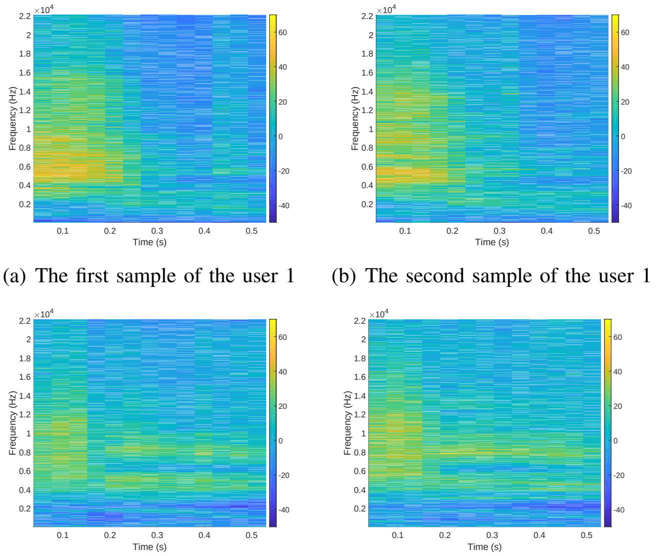
$$X(\tau,\omega) = \sum_{n=-\infty}^{n=+\infty} x[n]w[n-\tau]e^{-j\omega n} \qquad (2)$$

where τ is the time axis, ω is the frequency axis, $x[n]$ is the an audio sample, $w[n]$ is the window, and $X(\tau,\omega)$ is a complex function representing the phase and magnitude of the signal over time and frequency. Then, the spectrogram of the complex function $X(\tau,\omega)$ is computed based on:

$$spectrogram\{x[n]\}(\tau,\omega) \equiv |X(\tau,\omega)|^2 \qquad (3)$$

Different people have various speaking habits, which leads to different speaking time. Also, even the same person can speak the same passphrase in different amounts of time. To eliminate the influence of different speaking time and involve voices of different people in the same classification model, we scale the spectra difference (the image) to the same size.

As we stated in the first challenge, it is hard to compute the optimal M in polynomial time. Instead, we use the difference between two spectra to represent the relationship. If a unique M exists, the unique difference between two spectra should



(a) The first sample of the user 1 (b) The second sample of the user 1

(c) The first sample of the user 2 (d) The second sample of the user 2

Fig. 5. The spectra differences collected from two users when they speak "Six" to our system.

also exist. For each word, we compute the difference between two spectra as follows:

$$S = spectrogram\{x_{front}[n]\} - spectrogram\{x_{prime}[n]\} \tag{4}$$

where x_{front} is the audio sample recorded by the front microphone and x_{prime} is the audio sample recorded by the prime microphone.

D. Liveness detection for a single word

After obtaining the difference of spectra for each word, a robust classification model needs to be designed to detect the liveness of the speaker. Fig. 5 shows four spectra differences collected from two users when they speak "Six" to our system. We can see that the spectra differences produced by the same user are pretty similar. Also, for Figs.5(a) and 5(c), the spectra differences of different users are quite different from each other. Ideally, the spectra difference should be unique for each speaker and each word. A naive way is to treat every entry in the spectra difference as a feature and use machine learning techniques such as Support Vector Machine to recognize which speaker the voice comes from. However, this will involve many irrelevant features and overhead. Even if we try to select good features using algorithms such as Principle Component Analysis (PCA), we find that there is a lack of guidelines to decide which feature to use. Recently, with the theory of compressed sensing, the choice of features is no longer critical as long as the dimension of the feature space is sufficiently large, which is true in our system. Moreover, background noise will introduce occlusions to the spectra. The errors only disrupt a fraction of the spectrum but may break the pattern and reduce system performance. Fortunately, the compressed sensing-based classification model can effectively work even if there exist sparse errors in the feature space.

We build our system based on Sparse Representation-based classification model. There are k distinct object classes in the training data, and each object class refers to a word spoken by a distinct speaker. The n_i given training samples, taken from the i^{th} class, are arranged as columns of a matrix A_i

$$A_i = [v_{i,1}, v_{i,2}, \ldots, v_{i,n_i}] \in R^{m \times n_i} \tag{5}$$

where $v \in R^m$ is a vector that is produced by stacking the columns of computed spectra difference, and m is the number of entries in spectra differences. Each column of A_i is the training spectra difference.

Based on our experiment, we assume that the spectra differences of the same word spoken by the same speaker under different acoustic environments lie on a low-dimensional subspace. As a result, a new sample y of the i^th class can be approximately expressed as:

$$y = \alpha_{i,1}v_{i,1} + \alpha_{i,2}v_{i,2} + \cdots + \alpha_{i,n_i}v_{i,n_i} \quad \alpha \in R \tag{6}$$

For any new spectra difference y, since we do not know which class it belongs to, we define a new matrix $A = [A_1, A_2, \ldots, A_k]$ for all training samples of all available classes, and k is the number of classes we have in the training dataset. The linear representation of y can be expressed as:

$$y = Ax_0 \in R^m \tag{7}$$

By computing x_0, we can obtain the sparse representation of y in terms of dictionary A. Ideally, $x_0 = [0, \ldots, 0, \alpha_{i,1}v_{i,1}, \ldots, \alpha_{i,n_i}v_{i,n_i}, 0, \ldots, 0]$, so we can approximately recover X_0 by solving the following stable l1-minimization problem:

$$\widehat{x_1} = \arg\min ||x||_1 \quad \text{subject to} Ax = y \tag{8}$$

If the dictionary A is large enough, the number of non-zero coefficients in x_0 should be very small, so this convex optimization problem can be efficiently solved via second-order cone programming. The sparsity of x_0 can be satisfied in our system since we only collect a few training samples from the user and most training samples in A are from other speakers. To recognize which speaker produces the voice, we compute the estimation error $E(y)$ for each class.

$$E(y) = mean(||y - A\Delta_i\widehat{x_1}||_1) \tag{9}$$

where $\Delta_i(x_1)$ is the vector that only contains coefficients associated with the i^{th} class. The new sample y is labeled as the word spoken by a speaker whose estimation error is minimal. As long as the label is different from the argued speaker and argued word of the input voice, the spectra difference y is considered coming from the attacker. Fig. 6 shows the estimation errors for 48 object classes, we can see that the error of the ninth class is the lowest, so this testing sample should be labeled as the ninth class, which is "Six" of the first user.

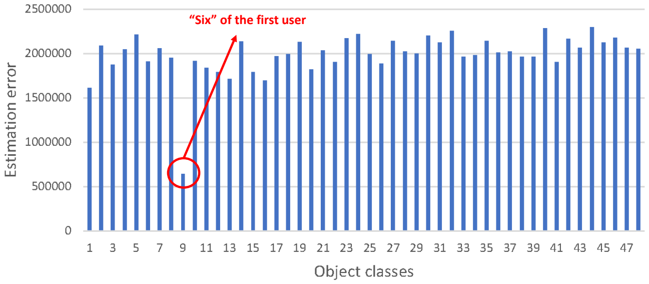


Fig. 6. The estimation errors for 48 classes.

E. *Liveness detection for a passphrase*

After getting the liveness detection results of each separate word, we need to make a rule to combine these results and produce the final decision. A good decision rule should make sure to provide a high passing rate for a live speaker and a low pass rate for a loudspeaker. A straightforward way is to include classification results of all words of the passphrase in a voting game where all players have the same weight. However, we find that the liveness detection may fail on detecting some live speakers for those words that do not have unvoiced phoneme. If a passphrase contains many words without unvoiced phoneme, the voting game may give a wrong liveness detection result. To solve this problem, we assign different weights to different words. In the voting game, each player is a tuple $< speaker, word, weight >$. The $speaker$ and $word$ are the recognition results for each spectra difference. If the recognized word does not match the word that the speaker said, the weight is 0. Otherwise, the weight of each word is defined as follows:

$$Weight(w) = 1 + log^{(1+N_{unvoiced}(w))} \tag{10}$$

where w is a word in the passphrase and $N_{unvoiced}(w)$ is the number of the unvoiced phonemes in word w. Here, we use the logarithmic equation to measure the gain of unvoiced phonemes, so that the word with multiple phonemes will have more weights but will not dominate the voting process. A speaker is recognized as a live speaker if and only if the voting result is the same as the claimed identity.

V. EVALUATION

A. *Experiment methodology*

Experiment setup In order to evaluate the effectiveness of our system, we build a prototype on two types of smartphones with different sizes (LG Nexus 5 and MOTO Nexus 6). Both of the two types of smartphones run on Android. The smartphones are used to capture audio signals in two channels. We design a simple graphical user interface (GUI) to help users collect audio signals. The application starts capturing user's voice in two channels as soon as the user presses the button and stops data collection immediately when the user releases the button. After data collection on smartphones, audio signals are sent to a local server for further validation. The server runs on a MacBook Pro with 2.9GHz Intel Core i5 processor and 8GB 1867 MHz DDR3 memory. The data is collected from 7 volunteers (3 males and 4 females) with different accents.

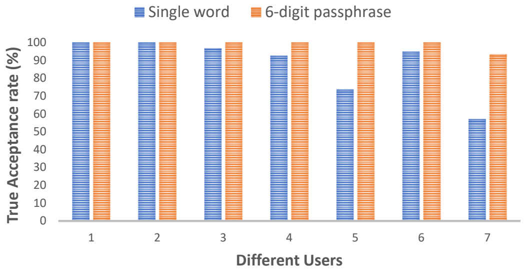


Fig. 7. True acceptance rates for a single word and a passphrase.

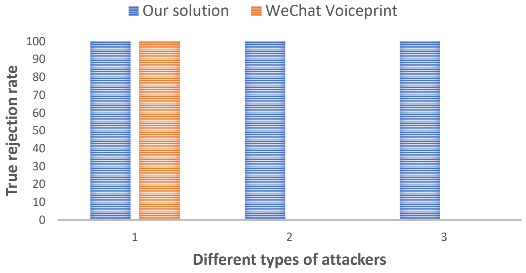


Fig. 8. True rejection rates against three attacks.

Performance Metrics In our experiments, we use the following performance metrics to evaluate the validation performance of our system. True acceptance rate is defined as the rate at which a user is correctly accepted by the system and considered as a real person. True rejection rate is defined as the rate at which an attacker is correctly rejected by the system.

B. *True acceptance rate for live speaker*

A good liveness detection system should provide high acceptance rate for a live speaker. To evaluate the system performance on detecting live speakers, we ask each volunteer to repeat speaking a 6-digit passphrase to our system. Five samples among them are used to build the dictionary for classification, and 45 samples are used as the testing data. Fig. 7 shows the acceptance rate of a single word and a whole passphrase for 7 volunteers. We can observe that our system can provide a high acceptance rate of at least 93.3% for a 6-digit passphrase. Also, our weighted voting scheme ensures the high true acceptance of a passphrase even the true acceptance rate of a single word is relatively low. For example, in some rare cases,our system can only provide an acceptance rate of 57% for each single word, while the acceptance rate is improved to 93.3% for the whole passphrase. On average, our system can provide a mean true acceptance rate of 99.04%. With a longer and more diverse passphrase, our system is expected to provide better robustness and performance.

C. *True rejection rate for three types of attackers*

The true rejection is another important performance metric to evaluate the security of liveness detection systems. In practice, we always would like to reject as many attackers as

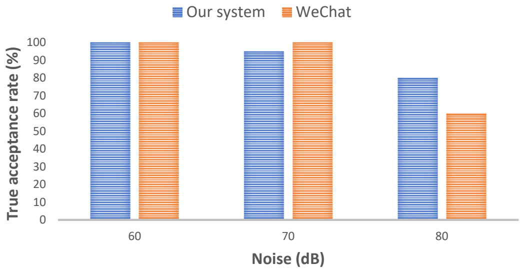


Fig. 9. True acceptance rates under different levels of noise.

we can to secure the system. We collect 30 samples under three types of attacks and evaluate the true rejection rate under the same setting in Section V-B. Also, we compare the performance of our system with that of WeChat Voiceprint for the first two types of attacks. To evaluate the performance under the replay attack, we use the speaker of a Nexus 5 smartphone to record and replay victim's voice to the target smartphone. From Fig. 8, it is clear that our system can defend all three types of attacks with a high true rejection rate of 100%, while WeChat Voiceprint cannot defend the replay attack. The evaluation results show that our liveness detection system can largely improve the security of voice-based authentication system against replay attacks and does not influence the user-experience of live speakers. Moreover, our system cannot be fooled bya reconstruction attacker even when the attackers know all implementation.

D. Performance under different acoustic environments

In practice, a user can use the voice biometrics-based authentication system in various acoustic environments. A good liveness detection system should provide high true acceptance in any acoustic environment as long as the voice biometrics-based authentication system can work. To evaluate the performance of our system performance under different levels of noise, we use an Amazon Echo speaker to play random kinds of music. In order to simulate different levels of noise, we adjust the volume of Amazon Echo speaker. Under each noise level, we ask a user to speak a passphrase for 20 times to each system. Fig. 9 shows the true acceptance rates of our system and WeChat Voiceprint under three different noise levels. We can observe that both our system and WeChat Voiceprint can achieve near 100% true acceptance rate when the noise level is about 60 and 70 dB, which means our system will not influence the user experience even if the user is in noisy scenarios such as on the streets. Even when the noise level is raised to about 80 dB, our system can still provide a higher true acceptance rate of 80% than WeChat Voiceprint. These experimental results show that our liveness detection system will not influence the original voice biometrics-based authentication in various experiments.

E. Performance under hardware influences

In most cases, users will enroll the voice and use the voice biometrics-based authentication system on the same smartphone, but they may also update their smartphones or switch to another brand at any time, which will introduce hardware changes and influence the system performance. To study the influence introduced by different hardware on different smartphones, we ask a user to enroll his voice on two different smartphones to build the dictionary. Then, we classify the test samples collected on one smartphone using the training samples on another smartphone, and the results are shown in Fig. 10. We can observe that our system can provide almost the same true acceptance rate on two smartphones for all 7 volunteers. Also, our system can provide 100% true rejection rate for all users. Experimental results show that our system can deal with hardware changes.

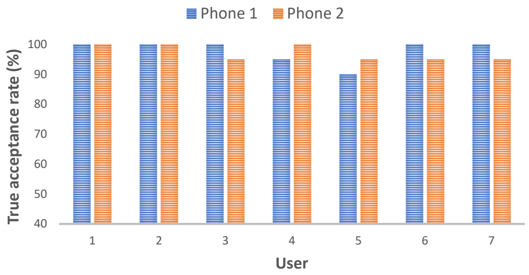


Fig. 10. True acceptance rates under hardware changes.

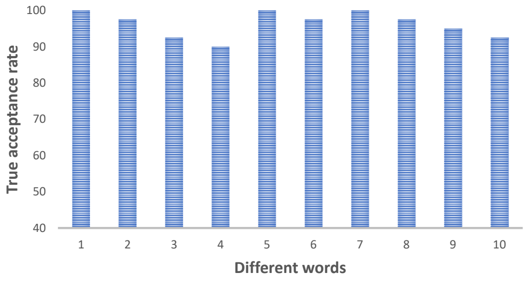


Fig. 11. True acceptance rates for different words.

F. Performance for different words

Different words contain different numbers of voiced and unvoiced phonemes. Also, the speaking time of different words is also different. In our system, we only use numbers as words. To evaluate our system performance for different numbers, we collect 40 testing samples from each of 7 users, and the results are shown in Fig .11. It is clear that our system can provide high acceptance rate of at least 90% for all words. We notice that the true acceptance rate of word "Four" is lower than that of other words. This is because the speaking habits of "Four" may change for some volunteers. Since our dictionary only contains 5 samples of each word spoken by each volunteer, the limited number of training samples cannot reflect the habit change. This can be further improved by involving more training samples in the dictionary.

VI. Related work

A. Voiceprint-based authentication applications and APIs

Since voiceprint-based authentication is efficient and comfortable for users, many applications have integrated voiceprint-based authentication into their system. For example, by using "Voiceprint" [18] designed by WeChat, users can log into their accounts by speaking a fixed passphrase. Besides, SayPay [12] offers a solution that fuses mobile payments by leveraging users' voice.

B. Automatic Speaker Verification (ASV) System.

An automatic speaker verification system is identify speaker's identity based on speech sample submitted by a user [15]. Recently, many ASV systems have been deployed on smartphones and online commerces [3, 7, 10]. The problem that ASV systems need to answer can be abstracted to a binary classification problem where the user must be justified as either a genuine speaker or as an imposter [8]. Existing ASV systems can be divided into Text-independent ASV systems and Text-dependent ASV systems. Text-independent ASV systems are able to accept arbitrary utterances, i.e., different speaking habit and languages, from speakers [2]. The text-dependent ASV systems are widely adopted by most authentication applications due to its higher recognition accuracy with fewer required utterances.

C. Voice Spoofing Attack.

In voice spoofing attacks, an attacker aims to attack the biometric identification of the user.These attacks can divide into two categories: voice replay attack and voice synthesis attack. In [16], it shows that an attacker can overcome text-dependent ASV systems by concatenating speech samples from multiple short voice segments of the target speaker. To fight against simple but effective voice replay attacks, a few research papers have been published [16, 17]. However, all these countermeasure systems suffer high false acceptance rate (FAR) compared to respective baselines. A voice synthesis attack is designed in [1] can break ASV system by generating artificial speech from text input. Various voice conversion attacks are proposed in [14, 20] in which the attacker converts the spectral and prosody features of his or her own speech and makes it resembles to the victim's speech.

VII. Conclusion

In this paper, we propose a new system that can detect the liveness and the identity of the speaker. Specially, we leverage the structural differences between human vocal system and loudspeakers and use the unique vibration pattern of both human vocal cord and throat as a key differentiating factor. Compared with existing systems, our solution does not assume any prior knowledge of the attacking method and is easy to operate. Moreover, our solution leverages the audio signals within the vocal frequency range and is robust to jamming attacks using high-frequency audio. We evaluate the performance of our system with 7 volunteers, and experimental results show that our system can achieve accurate liveness detection using a 6-digit passphrase with a mean true acceptance rate of 99.04% and true rejection rate of 100%.

Acknowledgement

This research was supported in part by NSF grants CNS 1757533, CNS 1629746, CNS 1564128, CNS 1449860, CNS 1461932, CNS 1460971, and IIP 1439672.

References

[1] F. Alegre, R. Vipperla, N. Evans, and B. Fauve. On the vulnerability of automatic speaker recognition to spoofing attacks with artificial signals. In *Proc. of EUSIPCO*, 2012.

[2] J. P. Campbell. Speaker recognition: A tutorial. *Proceedings of the IEEE*, 85(9):1437–1462, 1997.

[3] S. Chen, K. Ren, S. Piao, C. Wang, Q. Wang, J. Weng, L. Su, and A. Mohaisen. You can hear but you cannot steal: Defending against voice impersonation attacks on smartphones. In *Proc. of ICDCS*, pages 183–195. IEEE, 2017.

[4] K. Delac and M. Grgic. A survey of biometric recognition methods. In *Proc. of IS&T*, volume 46, pages 16–18, 2004.

[5] N. Evans, J. Yamagishi, and T. Kinnunen. Spoofing and countermeasures for speaker verification: a need for standard corpora, protocols and metrics. *IEEE Signal Processing Society Speech and Language Technical Committee Newsletter*, pages 2013–05, 2013.

[6] A. Janicki, F. Alegre, and N. Evans. An assessment of automatic speaker verification vulnerabilities to replay spoofing attacks. *Security and Communication Networks*, 9(15):3030–3044, 2016.

[7] K. A. Lee, B. Ma, and H. Li. Speaker verification makes its debut in smartphone. *IEEE signal processing society speech and language technical committee newsletter*, 2013.

[8] H. Melin. *Automatic speaker verification on site and by telephone: methods, applications and assessment*. PhD thesis, KTH, 2006.

[9] D. Mukhopadhyay, M. Shirvanian, and N. Saxena. All your voices are belong to us: Stealing voices to fool humans and machines. In *Proc. of Esorics*, pages 599–621. Springer, 2015.

[10] Nuance. Nuance vocal password. http://www.nuance.com/, 2013.

[11] U. Reichel. PermA and Balloon: Tools for string alignment and text processing. In *Proc. of Interspeech*, page 4 pages, Portland, Oregon, 2012.

[12] SayPay. http://saypaytechnologies.com/.

[13] M. Shirvanian and N. Saxena. Wiretapping via mimicry: Short voice imitation man-in-the-middle attacks on crypto phones. In *Proc. of CCS*, pages 868–879. ACM, 2014.

[14] Y. Stylianou. Voice transformation: a survey. In *Proc. of ICASSP 2009*, pages 3585–3588. IEEE, 2009.

[15] R. Togneri and D. Pullella. An overview of speaker identification: Accuracy and robustness issues. *IEEE circuits and systems magazine*, 11(2):23–61, 2011.

[16] J. Villalba and E. Lleida. Detecting replay attacks from far-field recordings on speaker verification systems. In *Proc. of BIOID*, pages 274–285. Springer, 2011.

[17] Z.-F. Wang, G. Wei, and Q.-H. He. Channel pattern noise based playback attack detection algorithm for speaker recognition. In *Proc. of ICMLC*, volume 4, pages 1708–1713. IEEE, 2011.

[18] WeChat. Voiceprint. http://thenextweb.com/apps/2015/03/25/wechat-on-ios-now-lets-you-log-in-using-just-your-voice/.

[19] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li. Spoofing and countermeasures for speaker verification: A survey. *Speech Communication*, 66:130–153, 2015.

[20] Z. Wu and H. Li. Voice conversion and spoofing attack on speaker verification systems. In *Proc. of APSIPA*, pages 1–9. IEEE, 2013.

[21] L. Zhang, S. Tan, and J. Yang. Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication. In *Proc. of CCS*, pages 57–71. ACM, 2017.

[22] L. Zhang, S. Tan, J. Yang, and Y. Chen. Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones. In *Proc. of CCS*, pages 1080–1091. ACM, 2016.