VoicePop: A Pop Noise based Anti-spoofing System for Voice Authentication on Smartphones

Qian Wang[†], Xiu Lin[†], Man Zhou[†], Yanjiao Chen[*], Cong Wang[‡], Qi Li[§], Xiangyang Luo[¶]
[†]School of Cyber Science and Engineering, Wuhan University, P. R. China.
[*]School of Computer Science, Wuhan University, P. R. China.
[‡]Department of Computer Science, City University of Hong Kong, Hong Kong, P. R. China.
[§]Institute for Network Sciences and Cyberspace, Tsinghua University, P. R. China.
[¶]The State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, P. R. China.

***Abstract*—Voice biometrics is widely adopted for identity authentication in mobile devices. However, voice authentication is vulnerable to spoofing attacks, where an adversary may deceive the voice authentication system with pre-recorded or synthesized samples from the legitimate user or by impersonating the speaking style of the targeted user. In this paper, we design and implement VoicePop, a robust software-only anti-spoofing system on smartphones. VoicePop leverages the *pop noise*, which is produced by the user breathing while speaking close to the microphone. The pop noise is delicate and subject to user diversity, making it hard to record by replay attacks beyond a certain distance and to imitate precisely by impersonators. We design a novel pop noise detection scheme to pinpoint pop noises at the phonemic level, based on which we establish individually unique relationship between phonemes and pop noises to identify legitimate users and defend against spoofing attacks. Our experimental results with 18 participants and three types of smartphones show that VoicePop achieves over 93.5% detection accuracy at around 5.4% equal error rate. VoicePop requires no additional hardware but only the built-in microphones in virtually all smartphones, which can be readily integrated in existing voice authentication systems for mobile devices.**

I. Introduction

Compared with password-based authentication, voice authentication is more convenient since it is hands-free and users do not need to memorize passwords. In recent years, the rapid growth of mobile communications has boosted the use of voice authentication in mobile devices, including smartphone login, mobile banking and e-commerce. For example, Google allows users to unlock their phones of Android operating systems by voice biometrics [1]. Say Tec uses voice biometric solution to support mobile financial services such as online payment and banking [2].

However, since the sound transmits through an open and public channel, the voice authentication system is highly vulnerable to spoofing attacks [3]–[5]. There are two major types of spoofing attacks, namely replay attacks and impersonation attacks [6]. In replay attacks, the adversary pre-records and playbacks the voice sample of the passphrase of a legal user to deceive the authentication system [7]. An adversary can also mimic the voice characteristics and style of a legal user to conduct impersonation attacks [8]. Spoofing attacks may greatly harm the users as the adversary may gain access to the victim's smartphone to steal private information and perform malicious operations.

Traditional methods to defend against replay attacks and impersonation attacks are liveness detection and automatic speaker verification (ASV) system. Liveness detection examines whether the voice is produced by a live user or a speaker, and ASV leverages unique spectral and prosodic features of the user's voice for identity authentication. For example, Zhang *et al.* [9] proposed to capture time-difference-of-arrival (TDoA) changes to the two microphones of the phone in a sequence of phoneme sounds to differentiate the voice from a live user and a replay device, but the user has to hold the phone at a specific position. In [10], the smartphone served as a Doppler radar to transmit a high-frequency acoustic sound from the built-in speaker and monitor the reflections of articulators at the microphone for liveness detection. Unfortunately, the extent of articulatory movements affects the effectiveness of this countermeasure. Chen *et al.* [11] explored the magnetic field emitted from loudspeakers to detect voice replay attacks. However, users need to move the smartphone with a predefined trajectory around the mouth while speaking the passphrase. M Sahidullah *et al.* [12] developed an ASV system against impersonation attacks using the throat microphone which is not available in most smartphones.

In this paper, we propose and implement VoicePop, a novel and practical anti-spoofing system based on *pop noise* that is induced by the user breathing while speaking the passphrase close to the microphone. The recorded voice samples hardly contain the pop noise since the sound of breath is gentle compared to the speech and will die out beyond a certain distance. The pop noise is also subject to user diversity and it is very difficult for attackers to imitate the way of breathing of the legal user. These ideal properties of the pop noise enable our proposed VoicePop system to resist spoofing attacks in voice authentication. To begin with, we conduct phoneme segmentation on the collected voice sample according to spectrogram characteristics. We design a novel pop noise detection algorithm to locate pop noises at the phonemic level. A lack of pop noise is deemed as a replay attack, however, environmental noise and hardware noise may also be wrongly detected as pop noises in the replayed voice samples. To address this problem,

we extract the Gammatone Frequency Cepstral Coefficients (GFCC) features of the detected pop noises for classification to distinguish a genuine voice sample and a replayed one. To defend against impersonation attacks, we leverage the individually unique relationship between phonemes and pop noises to construct a phoneme-pop sequence. A legal user is accepted if the phoneme-pop sequence of the voice sample is similar to that stored in the user profile upon registration, and an impersonation attack is declared otherwise.

VoicePop requires no additional hardware but only the built-in microphones that are available on almost all mobile devices. VoicePop also demands no extra efforts from users except to speak the passphrase as required by current voice authentication systems. As far as we are concerned, we are the first to use the features of pop noise to defend both replay attacks and impersonation attacks. We implement VoicePop on 3 types of smartphones and evaluate its performance with 18 volunteers under different experimental settings. The results verify the effectiveness of VoicePop that achieves over 93.5% detection accuracy at around 5.4% equal error rate. The main contributions of this work are summarized as follows:

- We propose VoicePop, a practical and effective software-only anti-spoofing system for voice authentication based on pop noise, which can be easily integrated in commercial off-the-shelf smartphones.
- We design a novel pop noise detection scheme to defend against replay spoofing attacks, and leverage the individually unique relationship between phonemes and pop noises to generate a phoneme-pop sequence to resist impersonation spoofing attacks.
- We build a fully-functional VoicePop prototype using off-the-shelf smartphones. Extensive evaluation results demonstrate that VoicePop can detect both replay and impersonation spoofing attacks with a high accuracy and a low equal error rate.

II. Preliminaries

A. Attack Model

Voice authentication system can be text-dependent (requires the same password for enrolment and verification) or text-independent (accept arbitrary utterances from speakers). We primarily focus on the text-dependent authentication system, which is currently the most widely adopted and commercially viable method with a high authentication accuracy [13]. Fig. 1 displays a typical voice authentication system. For the attack model, we consider replay spoofing attacks and impersonation spoofing attacks.

Replay attacks. Replay attacks leverage computers and other peripheral devices (e.g., loudspeaker) to perform voice playback to the microphone of the smartphone. The replay samples that involve the information of the victim's passphrase can be produced by stealthily recording, voice synthesis, and voice conversion. In this paper, we mainly focus on replay attacks by pre-recording since they retain more user characteristics than those generated by synthesis or conversion.

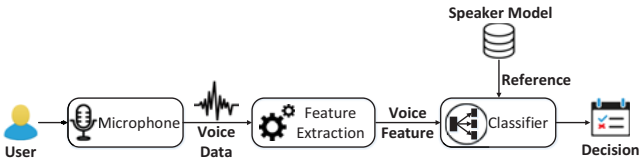

Fig. 1. A typical voice authentication system.

Impersonation attacks. Impersonation attacks can be conducted in two ways. The first is simply to imitate the legitimate user's voice and speaking habit without the help of other devices. The second is more advanced where we consider that the attacker knows the key rationale of our anti-spoofing system and observes how the target user pronounces the passphrase. To perform this type of attacks, we assume that the adversary uses a loudspeaker to replay the pre-recorded voice sample near the microphone while simultaneously impersonating the victim's breathing pattern closely to the microphone.

B. Pop Noise

The human voice is produced through several stages. Air is first expelled from the lung to form an airflow, which then enters the throat, passes through the vocal cords into the vocal tract, and finally bursts out of the mouth to form the sound wave. When the resulting airflow reaches the microphone, if the user's mouth is close enough to the microphone, the captured sound signals will not only contain the speech information but also the plosive burst as the friction between the lips and the airflow, known as the pop noise. In contrast, an attacker who tries to launch a replay attack usually cannot put the microphone of the recording device very close to the user's mouth, thus the recorded voice contains no pop noise. Therefore, by detecting the pop noise, we are able to distinguish the real speech from a live user and the recorded speech from a loudspeaker.

To detect pop noise, we compare the spectrograms of speech signals with and without a pop noise filter using three different smartphones, as shown in Fig. 2. It can be found that pop noise has a high energy in the low frequency (typically 0∼100 Hz), which has been discussed in the prior study [14]. Moreover, the duration of pop noise varies in the range 20∼100 msec based on the way people speak and breathe. Our detection algorithm is based on these observations.

C. Phoneme and Pop Noise

A phoneme is the smallest distinctive unit sound of a language in the human speech production system.There are two categories of phonemes, the vowel and the consonant. A vowel is a sound produced by the airflow through the mouth without hindrance, while a consonant is produced by obstructing the airflow out of the mouth with the teeth, tongue, lips or palate. Since each phoneme features unique physical origin in the human vocal tract system and has its own manner

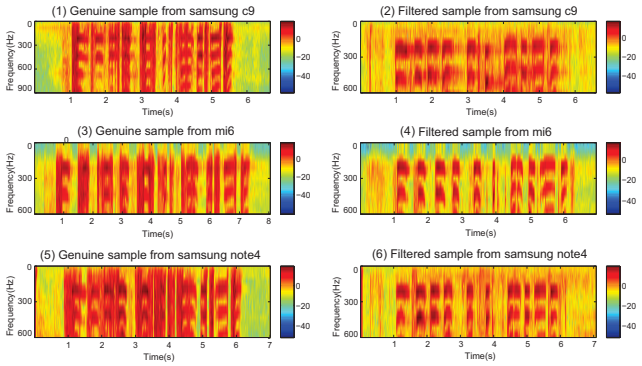

Fig. 2. Spectrogram comparison of samples without (left) and with (right) a pop noise filter using three different smartphones.

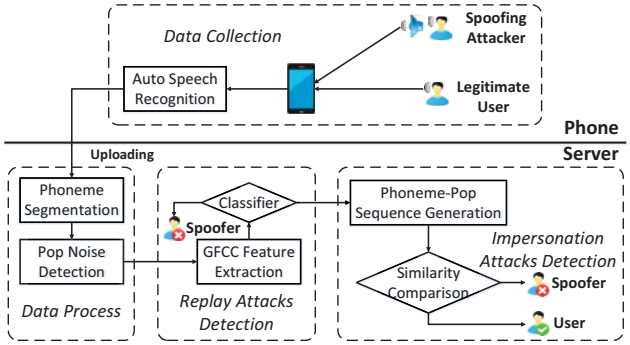

Fig. 3. The architecture of VoicePop.

TABLE I
PHONEMES RANK CORRESPONDING TO POP NOISE.

consonant	articulator	manner	ratio	consonant	articulator	manner	ratio
p	bilabial	stop	0.79	h	glottal	fricative	0.38
t	alveolar	stop	0.69	v	labiodental	fricative	0.35
ʧ	palatal	stop	0.68	w	velar	approximate	0.29
tr	alveolar	affricate	0.68	k,g	velar	stop	0.26
b	bilabial	stop	0.67	dz	alveolar	affricate	0.22
ts	alveolar	affricate	0.67	d	alveolar	stop	0.17
ʃ	palatal	affricate	0.65	ʒ	palatal	stop	0.11
ð	dental	fricative	0.57	n	alveolar	nasal	0.10
ʤ	palatal	affricate	0.50	ŋ	velar	nasal	0.08
dr	alveolar	affricate	0.50	j	palatal	approximate	0.05
θ	dental	fricative	0.43	m	bilabial	nasal	0.04
s,z	alveolar	fricative	0.40	r	alveolar	thrill	0.02
f	labiodental	fricative	0.39	l	alveolar	lateral	0.02
vowel	**articulator**	**manner**	**ratio**	**vowel**	**articulator**	**manner**	**ratio**
ʊ	back	near-close	0.67	ʊə	tongue	centering	0.16
aʊ	tongue	closing	0.39	i:	front	close	0.16
ɔ:	back	open	0.28	ɔɪ,əʊ	tongue	closing	0.15
eə	tongue	centering	0.23	u:	back	near-close	0.14
aɪ	tongue	closing	0.23	ɜ:	central	open-mid	0.13
ʌ	central	open-mid	0.21	a:,ɒ	back	open	0.11
ɪ	front	near-close	0.20	e	front	close-mid	0.08
æ	front	near-open	0.19	eɪ	front	closing	0.07
ə	central	mid	0.17	ɪə	tongue	centering	0.06

of pronunciation, the probability of the existence of pop noise when pronouncing different phonemes is different. We conduct an experiment on all 48 phonemes to explore the relationship between the phoneme and the pop noise. We collect speech data from 18 volunteers and rank the phonemes according to the existence probability of pop noise. The existence ratio of pop noise of phoneme X is calculated as $\frac{POP_X}{N_X}$, where N_X is the occurrences of phoneme X and POP_X is the occurrences of the pop noise of phoneme X in all sentences for all people. As shown in Table I, some phonemes require more breathing while some phonemes hardly require any breathing. The existence probability of pop noise in consonants is higher than that in vowels. We find that the phoneme ranking of the existence probability of pop noise is different among users due to their unique vocal systems and utterance styles. Therefore, we can extract and store such information upon registration for user identification.

III. VoicePop: Design Details

A. Overview

The key idea of our anti-spoofing system is to identify a legal user based on the extracted pop noise features and phoneme-pop sequence from the voice sample when the user says the passphrase near the microphone. Fig. 3 depicts the system architecture of VoicePop, which consists of four phases: *data collection*, *data process*, *replay attacks detection*, and *impersonation attacks detection*.

In the first phase, when a user performs an authentication, the built-in microphone captures the user's speech, which is then fed into an automatic speech recognition (ASR) system to obtain the words of the passphrase. If the passphrase is not correct, the user will be rejected directly; otherwise, the recorded sample and text are transmitted to the server in real-time for spoofing attacks detection.

In data process phase, the original sample is first segmented into phoneme units and non-speech periods. In particular, VoicePop partitions and labels the voice sample into phonemes leveraging the forced alignment method, which recognizes the spoken words according to a given text of phoneme sequence using Hidden Markov Models (HMM). Meanwhile, a pop noise detection algorithm is proposed to locate explosive sound periods caused by strong breathing during speaking, which are refined and screened according to the phoneme segmentation result and the predefined user-dependent ranking.

If the pop noise is detected, it conducts replay attacks detection. In this phase, we extract its GFCC feature vectors as input to the Support Vector Machine (SVM) for classification. If the classification result is positive, VoicePop will enter impersonation attacks detection phase, otherwise, a replay spoofing attack is declared.

In the impersonation attacks detection phase, a binary phoneme-pop sequence is generated according to the segmented phonetic units and the located pop noise, which is then compared with that stored in the user profile to compute a similarity score. The user will be accepted only if the score is greater than a threshold. The detection result can be integrated into general voice authentication systems for user identification.

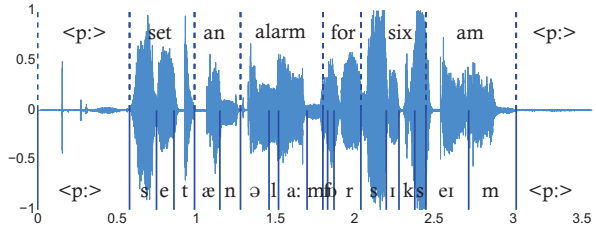

Fig. 4. An example of phoneme segmentation.

B. Phoneme Segmentation

A phoneme is made up of a number of distinctive overtone pitches, which are known as formants. Formants refer to the areas of the sound spectrum where the energy is concentrated. Formants not only determine the sound quality but also reflect the physical characteristics of the vocal tract. Phonemes can be uniquely identified by formants.

To attain phoneme segmentation, we first generate the spectrogram of the voice sample using a spectrum analyzer, then adopt HMM to perform a forced alignment for the obtained voice spectrogram and the pre-defined spectrogram. Given the text of the input speech acquired by an ASR system, the phoneme segmentation tool MAUS [15] first transforms the words into canonical pronunciations according to a standard pronunciation model. Then, a probabilistic graph including all possible results and the corresponding probabilities is produced based on the expected pronunciation of the input words and millions of potential accents. By searching the space of phonemic units, the path of the unit with the highest probability is selected. Finally, the input speech is segmented and labeled at the phonemic level. Fig. 4 illustrates an example of phoneme segmentation for the voice sample of a user saying the passphrase. It is shown that each word and phoneme can be accurately separated.

C. Pop Noise Detection

As we have discussed in Section II-B and Section II-C, pop noise has high energy in low frequencies of the voice sample (comparing the spectrograms in Fig. 2 before and after a pop noise filter), and different phonemes feature different existence probability of the pop noise, while subjecting to user diversity. Based on these observations and prior work [14], we design a novel detection scheme, and the details (illustrated in Fig. 5) are described as follows. The suggested parameters below are mostly empirically determined according to our dataset.

1) ***Non-speech components removal****:* The phoneme segmentation not only partitions phonemes but also separates the speech (phases containing phonemes) and the non-speech components (the silent phases), as shown in Fig. 4. We first remove the non-speech components to improve the accuracy of locating the pop noise since the non-speech components are usually noises or predefined events in the speech that may be wrongly detected as pop noise.

2) ***Short-Time Fourier Transform****:* We use the Short-Time Fourier Transform (STFT) to acquire the time domain

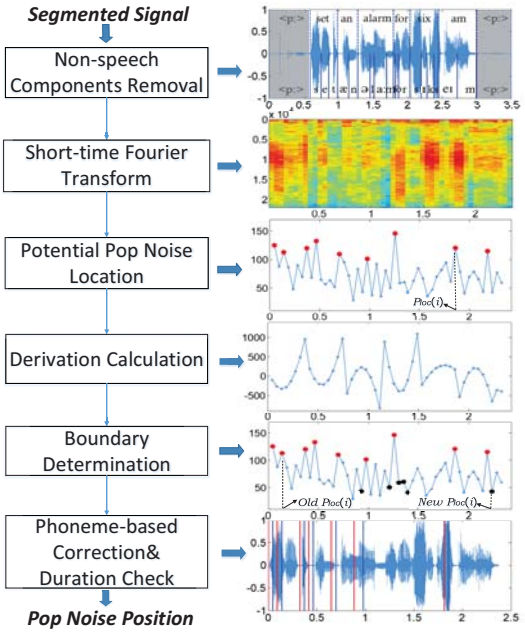

Fig. 5. Pop noise detection.

information such as the frequency distribution changes over time.The STFT divides a long time-domain signal into frames using a fixed window size and then computes the Fourier transformation separately on each frame. The results of each frame along the time dimension are stacked up to reveal the Fourier spectrum for each segment over time. The two-dimensional signal obtained by the STFT expansion is called a sound spectrum diagram. For STFT analysis, we use a Hamming window with a size of 4096 points and an overlapping of 2048 points.

3) ***Potential pop noise location****:* After STFT, we get the frequency distribution of each frame. We first compute the energy within the frequency range 0∼93 Hz (the pop noise energy concentrates on low frequencies) for each frame, denoted as $E(i)$, where i is the index of each frame. This range is selected according to extensive analysis of spectrograms of genuine speech data samples. Then we calculate the standard deviation of the energy for all frames (donated as E_{std}). We determine that potential pop noise exists in the ith frame (donated as $Loc(j)$, where j is the index of selected frames) if $E(i) > 3 \cdot E_{std}$.

4) ***Derivation calculation****:* The previous step pinpoints the peaks of potential pop noises and we need to locate the boundaries to obtain the whole pop noises. To achieve this goal, we take the derivative of the window energy function obtained by polynomial fitting. Specifically, we perform polynomial fitting on discrete energy values $E(i)$ for every eight-point chunk, then we take the derivative of the fitting function to obtain the absolute value of the differential coefficient of every point i, denoted as $D(i)$.

5) ***Boundary determination****:* We find the boundaries of pop noises by searching the vicinity of $Loc(j)$ up to 3 points. If

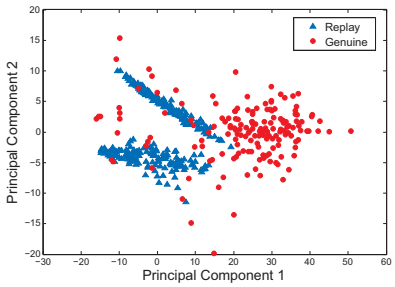

Fig. 6. The feature vectors of the genuine samples and the replay samples.

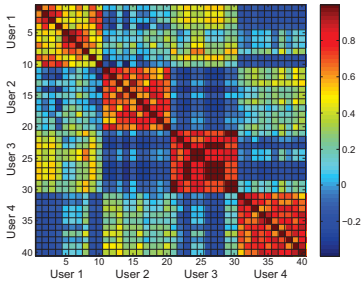

Fig. 7. Phoneme-pop sequence similarity between different pairs of users.

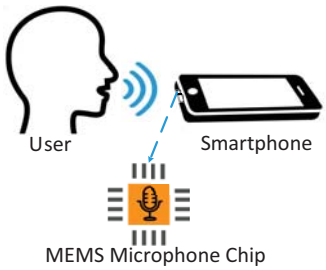

Fig. 8. A typical application of VoicePop.

the nearby point k ($Loc(j) - 3 \leq k \leq Loc(j) + 3$) satisfies the conditions that $E(k) \leq 0.45 \cdot E(Loc(j))$ and $D(k) \geq 0.45 \cdot D(Loc(j))$, we deem that there is a drop near the peak and include point k as part of the pop noise.

*6) **Phoneme-based correction & duration check**:* We conduct phoneme-based correction and duration check for all potential pop noises. Pop noise happens for certain phonemes with a high probability, and everyone's particular phonemes are not the same. Therefore, we only select potential pop noises in the presence of these high-probability phonemes as real pop noises according to personal phoneme probability rank. We also observe that the pop noise typically has a duration within the range 20∼100 msec. Hence, we check the duration of potential pop noises and abandon those out of this range.

D. Replay Attacks Detection

With the above pop noise detection scheme, we can reject most samples of replay attacks that contain no pop noise. Nevertheless, some samples of replay attacks also exhibit "pop noise", which may be caused by speaker-microphone channel noise or environmental noise. Therefore, we further analyze features of pop noises to distinguish samples from genuine users and replay devices. We adopt GFCC features which yield a high accuracy and robustness [16].

We first generate a 64-gammatone filterbank, each filter is defined by a bandwidth B and a center frequency f_c. The center frequencies are distributed in proportion to their bandwidths through the Equivalent Rectangular Bandwidth (ERB) scale. The bandwidth B is calculated as $1.019 \cdot \text{ERB}$, where ERB is given as $24.7 \cdot (4.37 \cdot 0.001 \cdot f_c + 1)$ [17]. The frequency spectrum of each pop noise frame is divided into several overlapping bands (filters) by the filterbank, and we calculate the weighted sum of the FFT magnitudes (log energy) for each filter. Finally, the Discrete Cosine Transformation (DCT) is applied to each log energy to calculate the cepstral coefficients. The number of cepstral coefficients is equivalent to the number of filters. Particularly, We only use the filters of which the center frequencies range from 0 Hz to 104 Hz, because pop noise usually appears in this frequency band.

To address the problem that pop noise may be wrongly detected in some attack samples, we use GFCC features of pop noises to distinguish genuine and attack samples by a two-class SVM classifier. Fig. 6 shows the feature vectors of the genuine samples and the replay samples after applying the Principal Component Analysis (PCA), demonstrating that their feature points are easy to be separated, and thus we can effectively defend replay attacks.

E. Impersonation Attacks Detection

To resist impersonation spoofing attacks, we propose a phoneme-pop sequence generation algorithm (shown in Algorithm 1) to identify the unique relationship between phonemes and pop noises for each individual user. Through extensive experiments, we find that the positions of pop noises are different for different people. This is because each person has his/her own speaking style and special vocal system. Therefore, we build a binary phoneme-pop sequence for the passphrase of each user upon registration and store them (one authentication trail produces one sequence) in the user profile. This sequence describes which phonemes of the passphrase pop noises appear along with. When a user performs identity authentication using VoicePop, we compute the sequence and leverage a combined-scheme using two similarity scores to make a comparison.

We use two methods for similarity calculation. The first is the Pearson correlation coefficient [18], which is used to measure the degree of linear correlation between two sequences. The coefficient value is within the range [-1, 1], with an absolute value near 1 indicating a strong linear correlation, while a value near 0 indicating a lack of linear correlation. Fig.7 shows the similarity (Pearson correlation coefficients) of the computed phoneme-pop sequences for the same passphrase spoken by four different users. Each user speaks the passphrase for 10 times. We observe that the correlation coefficients for the same user under different trials are very high (around 0.8), while the correlation coefficients are below 0.5 between different users. This confirms the individual diversity in phoneme-pop sequences. Particularly, we calculate the average value of the correlation coefficients between the sequence of the input sample and each sequence in the user profile.

The second method is the contact ratio (the ratio of the number of same elements to that of all elements). We build a sequence of existence probability of pop noise for all m

Algorithm 1 Phoneme-Pop Sequence Generation Algorithm.

Require: The number of located pop noises n, number of segmented phonemes m, the set of start and end boundaries of pop noises $\{ST\_pop_i\}_{i=1}^{n}$ and $\{ET\_pop_i\}_{i=1}^{n}$, the set of start and end boundaries of phonemes $\{ST\_pho_j\}_{j=1}^{m}$ and $\{ET\_pho_j\}_{j=1}^{m}$.

Ensure: Binary phoneme-pop sequence $\{S_j\}_{j=1}^{m}$.

```
Initial S_j = 0, j = 1, 2, ..., m;
j = 1;
for i = 1 → n do
    /*Find corresponding phoneme index of ST_pop_i*/
    while (j < m) ∧ (ST_pop_i < ST_pho_j) do
        j++;
    end while
    /*If the pop noise only exists in current phoneme*/
    if (j < m) ∧ (ET_pop_i < ST_pho_{j+1}) then
        S_j = 1;
    else
        /*Find all remaining phonemes*/
        while (j < m) ∧ (ET_pop_i > ST_pho_{j+1}) do
            S_j = S_{j+1} = 1;
            j++;
        end while
        j--;
    end if
end for
return S
```

TABLE II
EXPERIMENTAL DEVICES.

maker	model	authenticate	record	replay	maker	model	record	replay
Mi	Mi6	√	√	√	Huawei	Mate10		√
Samsung	C9 pro	√	√	√	Earise	Al-101		√
Samsung	S7 edge	√	√		Sony	Icd-ux565f	√	√
Huawei	Mate8		√		Hivi	M200mkIII		√

phonemes in the passphrase of the user upon registration, denoted as $\{P_j\}_{j=1}^{m}$, and store it in the user profile. To calculate the similarity score between P_j of the user profile and the phoneme-pop sequence of the input sample, we first process the sequence P_j by resetting $P_j = 0$ if $P_j \leqslant 0.2$ and resetting $P_j = 1$ if $P_j \geqslant 0.8$. We neglect phonemes whose P_j is between 0.2 and 0.8 as they are less reliable. We then check the contact ratio for elements of P_j being 0 or 1.

Finally, we develop an impersonation attack detection scheme by combining the similarity scores of the Pearson correlation and the contact ratio. We set thresholds for the two similarity scores respectively, and accept the user only if both scores are higher than their corresponding thresholds, otherwise, the user will be denied access to the system and an impersonation spoofing attack will be declared.

IV. EVALUATION

We implement a prototype of VoicePop with the typical client-server architecture: a mobile application running on Android and a processing server running on a ThinkPad server with Intel(R) Core(TM) i7-7500U 2.70 GHz CPU and 8 GB RAM. The mobile application is designed to record user voice samples, recognize the words of speech, and upload the raw voice data and the corresponding text to the server in real time. The sampling rate of the speech signals on smartphones is 44.1 kHz. We use three different smartphone models running Android 6.0 KitKat for authentication, as shown in Table II. At the server side, the received data is fed into a processing pipeline as described in Section II. Since VoicePop detects pop noises caused by user breathing while speaking, we require the users to speak close to the microphone, but do not request them to hold the phone at a specific place or distance. Fig. 8 shows a typical use case, and the effective distance will be discussed in Section IV-D.

Data collection. We recruit 18 volunteers (13 males and 5 females) with the age ranging from 20 to 27 to participate in the experiments. The participants are undergraduate and graduate students who are instructed to perform voice authentication with VoicePop. To build the user profile, including GFCC features of genuine pop noises, phoneme-pop sequences, and pop noise existence probability sequences, we ask each participant to speak a passphrase five times upon registration. Then, each participant performs legitimate authentications for 10 times for testing. The passphrase of each participant is randomly selected from a pre-defined set of commands and the lengths of the passphrases range from 3 to 10 words.

Attacks. We evaluate our system under replay attacks and impersonation attacks. For replay attacks, we first employ six microphone models, including the professional recorder and the built-in microphones of mobile devices, to pre-record the passphrase when a legitimate user is performing voice authentication. Then, we use five speaker models, including the standalone speakers and built-in speakers of mobile devices, to playback the recorded voice samples in front of the smartphone for voice authentication. The models of microphones and speakers are listed in Table II. Each speaker conducts 10 trials for each participant and each passphrase.

For impersonation attacks, we allow adversaries to listen to the voice samples of legitimate users in order to imitate the speed of talking, the breath style and so on. We recruit 5 volunteers as adversaries and each impersonates 3 participants for 10 trails for each passphrase. In addition, an attacker may replay a participant's voice sample and impersonate the breathing in front of the speaker.

Metrics. We adopt six metrics to evaluate the performance of our system. False Accept Rate (FAR) is the likelihood that the system wrongly accepts a spoofing attack as a legitimate user. True Accept Rate (TAR) is the probability that the system correctly identifies a legitimate user. Receiver Operating Characteristic (ROC) curve explains the relationship between TAR and FAR under various detection threshold. False Reject Rate (FRR) is the probability that the system mistakenly declares a legitimate user as a spoofing attack. Equal Error Rate (EER) defines the rate when FAR equals FRR. Accuracy measures the likelihood that the system accepts legitimate users and rejects attacks.

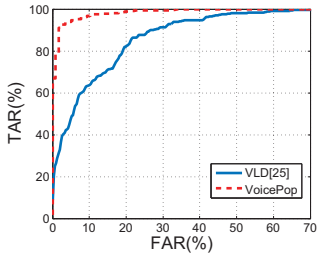

Fig. 9. Overall ROC curves.

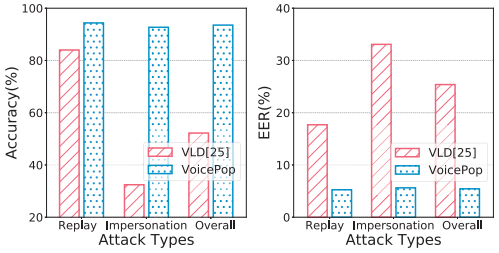

Fig. 10. Overvall accuracy and EER.

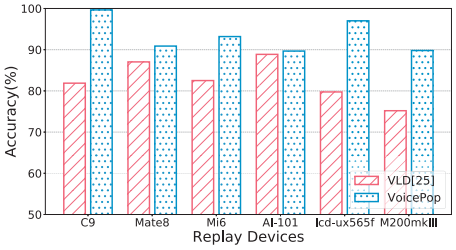

Fig. 11. Accuracy under different replay devices.

A. Overall Performance

We confirm the effectiveness of our system against replay attacks and impersonation attacks by comparing with the baseline according to [19]. In [19], Sayaka Shiota *et al.* proposed the pop noise detector combined with the phoneme information to detect the existence of pop noises but did not use the features of pop noises for further classification. We construct the baseline by using VLD in [19] to detect pop noises, then extract features to detect spoofing attacks. Fig. 9 shows the ROC curves under both attacks. We observe that our system achieves more than 93% TAR with less than 2% FAR, which confirms the effectiveness of our system in defending against spoofing attacks. Moreover, Fig. 10 demonstrates the overall accuracy and EER under both attacks. It is shown that our system attains an overall accuracy of 93.5% and an EER of 5.4% under replay and impersonation attacks, far outperforming VLD that has an accuracy of merely 52.2% and an EER as high as 25.4%. These results verify that VoicePop is highly effective against both replay and impersonation attacks.

Replay attacks. We take a closer look at the performance of VoicePop under different replay devices as listed in Table II. As shown in Fig. 11, the accuracy of VoicePop is relatively stable under different replay devices, and it is always more effective in replay detection than VLD. These results demonstrate the robustness of VoicePop against replay spoofing attacks.

Impersonation attacks. We also dig deeper into impersonation attacks. We consider three ways of attacks: pure impersonation attack, playback with random breath, and playback with breath impersonation. Fig. 12 shows that VoicePop has a superior performance over the baseline under all three ways of attacks while the baseline is quite vulnerable to impersonation spoofing attacks. This is because VLD only detects the existence of pop noise without extracting individually unique features. We leverage the unique relationship between phonemes and pop noises of each individual to extract location sequence features. This feature is user-dependent, and the attacker can hardly impersonate the breathing in precise synchronization at the phonemic level.

B. Impact of Authentication Distance

Since the pop noise caused by breathing while speaking is mild and directional compared with speech, we study the impact of the distance between the microphone and the user's mouth to find the effective distance range. We ask 3 volunteers to perform 10 trails at different distances. Fig. 13 presents the accuracy of three different phones. In particular, the accuracy is satisfactory when the distance ranges from 2 cm to 6 cm for all phones but decreases sharply beyond 6 cm. When the distance is larger than 12 cm, the accuracy drops to below 20%, which means the microphone cannot capture the pop noise information. This is because the breath while speaking is gentle and its power decreases as the distance increases, thus the microphone can hardly capture it. In addition, the accuracy is also be degraded when the distance is too short since a strong breath will affect the stability of phoneme-pop sequence.

C. Impact of Authentication Phone

As we know, the microphones of different smartphones have diverse frequency selectivity [20]. A user may register in VoicePop using one phone but perform authentication with another one. Thus we study the performance of our system on different smartphones. Specifically, we use one phone to record the information for registration and use the other two phones for authentication. As shown in Fig. 14, we observe that VoicePop can resist spoofing attacks with an accuracy of 94.7%, 87.7%, and 88.1% when using Mi6, S7, and C9 as the phone for registration while the other two for authentication. Although the performance of VoicePop will inevitably be worse when using different phones for registration and authentication, it still produces relatively accurate detection results. This result demonstrates that VoicePop is robust and compatible with different phone models.

D. Impact of Passphrase Length

Generally, a longer passphrase provides a stronger security but increases the authentication time. We categorize all passphrases into three types according to the length of words (2∼4, 5∼7 and 8∼10). Fig. 15 illustrates the accuracy for different lengths of passphrases for three phones. We observe that for each smartphone, the accuracy is improved with the increase in passphrase length. The overall accuracy of three phones of the longest passphrase is 94.3% , while the shortest passphrase gets an average accuracy of 91.4%. This is because that a longer passphrase contains more pop noises and more distinctions in the phoneme-pop sequence. It is also shown that VoicePop is able to achieve a very high anti-spoofing

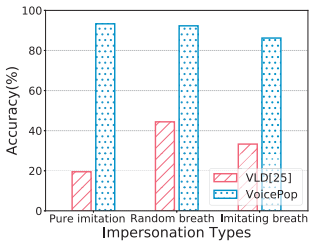

Fig. 12. Accuracy under three ways of impersonation attacks.

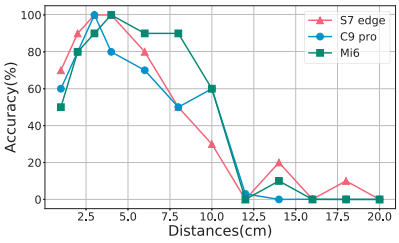

Fig. 13. Impact of authentication distance.

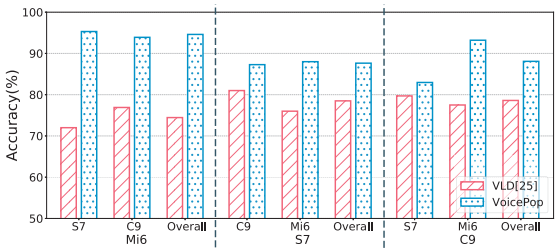

Fig. 14. Impact of authentication phone.

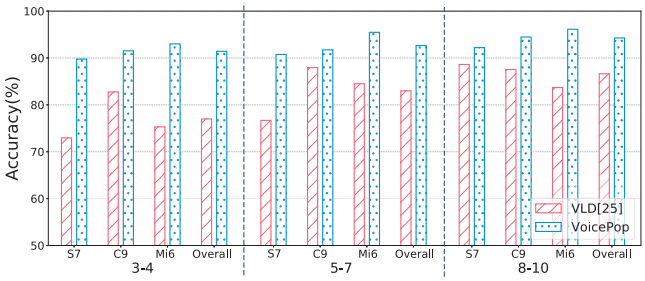

Fig. 15. Impact of passphrase length.

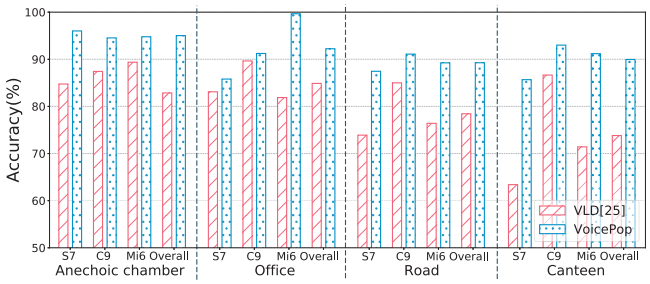

Fig. 16. Impact of ambient noise.

effectiveness even when the length of the passphrase is less than 5.

E. Impact of Ambient Noise

The ambient noise usually has high energy at low frequencies [21], [22], which may interfere with the pop noise detection, thus we evaluate the impact of ambient noise on the performance of VoicePop. We use three types of smartphones in four different environments (anechoic chamber, office, road, and canteen) with various degrees of ambient noise. As shown in Fig. 16, we can see that the overall accuracy of VoicePop of three phones in four different environments is all above 89%, and is as high as 95.1% in the anechoic chamber. The main reason is that the features of pop noise are different from those of ambient noise, and can be leveraged for differentiation by the SVM classifier. This demonstrates that VoicePop is robust to ambient noise.

V. Related Work

Voice spoofing attacks. The voice biometrics systems have been adopted by a large number of mobile devices for user authentication. However, numerous studies have shown that voice authentication is vulnerable to spoofing attacks [3]–[5]. There are mainly two types of attacks: replay attacks and impersonation attacks. Replay samples can be produced by stealthily recording, voice synthesis, and voice conversion. T. Kinnunen *et al.* [7] discovered that the EER of voice authentication systems increased from 1.76% to 30.71% under replay attacks. Voice synthesis techniques concatenate voice segments from multiple samples to reconstruct the passphrase of the legitimate user [23]. Recently, Adobe developed a system VoCo [24] to enable users to edit texts and synthesize corresponding speeches of a given speaker with only 20 minutes of voice samples, which may pose severe potential threats to voice authentication systems Voice conversion attacks convert the attacker's voice sample into the victim's based on the known acoustic model of the victim using voice morphing techniques [25], [26]. Impersonation attacks are launched by attackers who mimic the voice characteristics and speaking behavior of the victim [5]. Wu *et al.* [6] showed that pure impersonation may produce similar speaking pattern and rate of the victim, but it's nearly impossible for the impersonators to fake the spectral characteristics like formants.

Voice anti-spoofing. The traditional method of defending against replay attacks are liveness detection [3], [9]–[11], [27], [28], which examines whether the voice is produced by a live user or a speaker. For example, VoiceLive [9] measured the time-difference-of-arrival (TDoA) changes to the two microphones of the smartphone to pinpoint the sound origins within a live user's vocal tract for liveness detection, but the user has to hold the phone at a specific position. In [10], the smartphone was used as a Doppler radar to transmit a high-frequency acoustic sound and monitor the reflections of articulators at the microphone, but the extent of articulatory movements affects the effectiveness of this countermeasure. Chen *et al.* [11] checked the magnetic field emitted from loudspeakers to detect machine-based spoofing attacks, whereas users need to move the smartphone with a predefined trajectory around the mouth while speaking the passphrase. As far as we are concerned, we are the first to use the features of pop noise to defend both replay attacks and impersonation attacks. Sayaka Shiota *et al.* [14] proposed the pop noise detector, which combines the single- and the double-channel to detect pop noise. They

further incorporated the phoneme information for pop noise detection in [19]. However, their studies rely on the specific microphone model and can not be applied to mobile devices. In contrast, our pop noise detection scheme is designed for voice authentication in mobile devices. We specifically address the problem that pop noise may be wrongly detected in the replay audio. The experiment results also confirm that our pop noise based authentication system is effective against various ways of attacks and is robust to different phone models and ambient noises.

VI. Conclusion

We presented VoicePop, a practical and effective software-only anti-spoofing system for voice authentication on smartphones. VoicePop identifies a live user by detecting pop noise naturally incurred by user breathing while speaking close to the microphone. We leveraged the individually unique relationship between phonemes and pop noises to detect both replay and impersonation spoofing attacks. Extensive experiments confirmed that VoicePop is robust in resisting various types of voice spoofing attacks with different smartphones under diversified environments. VoicePop can be readily integrated into existing voice authentication systems on smartphones with no additional hardware modification, and we believe it has a promising future application.

Acknowledgment

Qian's research is supported in part by the NSFC under Grants 61822207 and U1636219, the Equipment Pre-Research Joint Fund of Ministry of Education of China (Youth Talent) under Grant 6141A02033327, the Outstanding Youth Foundation of Hubei Province under Grant 2017CFA047, and the Key Program of Natural Science Foundation of Hubei Province under Grant 2017CFA007. Yanjiao's research is supported in part by the NSFC under Grant 61702380, the Hubei Provincial Natural Science Foundation of China under Grant 2017CFB134, and the Hubei Provincial Technological Innovation Special Funding Major Projects under Grant 2017AAA125. Cong's research is supported in part by the Research Grants Council of Hong Kong under Grants CityU 11276816, CityU 11212717, CityU C1008-16G and the NSFC under Grant 61572412. Qi's work is supported in part by the NSFC under Grants 61572278 and U1736209. Xiangyang's research is supported by the Plan for Scientific Innovation Talent of Henan Province under Grant 2018JR0018. Dr. Yanjiao Chen is the corresponding author.

References

[1] Google smart lock. https://get.google.com/smartlock/.
[2] Say Tec. https://www.say-tec.com/.
[3] Z. F. Wang, G. Wei, and Q. H. He, "Channel pattern noise based playback attack detection algorithm for speaker recognition," in *Proc. of IEEE ICMLC*, 2011, pp. 1708–1713.
[4] P. L. De Leon, M. Pucher, J. Yamagishi, I. Hernaez, and I. Saratxaga, "Evaluation of speaker verification security and detection of hmm-based synthetic speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 8, pp. 2280–2290, 2012.
[5] R. Hautamäki, T. Kinnunen, V. Hautamäki, T. Leino, and A.-M. Laukkanen, "I-vectors meet imitators: On vulnerability of speaker verification systems against voice mimicry," in *Proc. of INTERSPEECH*, 2013, pp. 930–934.
[6] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *Speech Communication*, vol. 66, pp. 130–153, 2015.
[7] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. W. D. Evans, J. Yamagishi, and K. Lee, "The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," in *Proc. of INTERSPEECH*, 2017, pp. 2–6.
[8] M. Shirvanian and N. Saxena, "Wiretapping via mimicry: Short voice imitation man-in-the-middle attacks on crypto phones," in *Proc. of ACM CCS*, 2014, pp. 868–879.
[9] L. Zhang, S. Tan, J. Yang, and Y. Chen, "Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones," in *Proc. of ACM CCS*, 2016, pp. 1080–1091.
[10] 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 ACM CCS*, 2017, pp. 57–71.
[11] 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 IEEE ICDCS*, 2017, pp. 183–195.
[12] M. Sahidullah, D. A. L. Thomsen, R. G. Hautamäki, T. Kinnunen, Z. H. Tan, R. Parts, and M. Pitkänen, "Robust voice liveness detection and speaker verification using throat microphones," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 44–56, 2017.
[13] S. Koga, S. Makihara, and Y. Yamanouchi, "Score normalization in playback attack detection," in *Proc. of IEEE ICASSP*, 2010, pp. 1678–1681.
[14] S. Shiota, F. Villavicencio, J. Yamagishi, N. Ono, I. Echizen, and T. Matsui, "Voice liveness detection for speaker verification based on a tandem single/double-channel pop noise detector," in *Proc. of Odyssey*, 2016, pp. 259–263.
[15] T. Kisler, F. Schiel, and H. Sloetjes, "Signal processing via web services: The use case webmaus," in *Proc. of ADHO DH*, 2012, pp. 30–34.
[16] X. Zhao and D. Wang, "Analyzing noise robustness of mfcc and gfcc features in speaker identification," in *Proc. of IEEE ICASSP*, 2013, pp. 7204–7208.
[17] B. C. Moore, "Simplified derivation of auditory filter shapes." *Journal of Speech & Hearing Research*, vol. 34, no. 6, pp. 1439–1439, 1991.
[18] H. M. Walker and J. Lev, "Statistical inference," *Journal of the American Statistical Association*, vol. 49, no. 266, pp. 53–76, 1953.
[19] S. Mochizuki, S. Shiota, and H. Kiya, "Voice livness detection based on pop-noise detector with phoneme information for speaker verification," *Journal of the Acoustical Society of America*, vol. 140, no. 4, pp. 3060–3060, 2016.
[20] M. Zhou, Q. Wang, T. Lei, Z. Wang, and K. Ren, "Enabling online robust barcode-based visible light communication with realtime feedback," *IEEE Transactions on Wireless Communications*, vol. 17, no. 12, pp. 8063–8076, 2018.
[21] M. Zhou, Q. Wang, J. Yang, Q. Li, F. Xiao, Z. Wang, and X. Chen, "Patternlistener: Cracking android pattern lock using acoustic signals," in *Proc. of ACM CCS*, 2018, pp. 1775–1787.
[22] M. Zhou, Q. Wang, K. Ren, D. Koutsonikolas, L. Su, and Y. Chen, "Dolphin: Real-time hidden acoustic signal capture with smartphones," *IEEE Transactions on Mobile Computing*, vol. PP, pp. 1–1, DOI: 10.1109/TMC.2018.2 842 771, 2018.
[23] S. K. Ergunay, E. el Khoury, A. Lazaridis, and S. Marcel, "On the vulnerability of speaker verification to realistic voice spoofing," in *Proc. of IEEE BTAS*, 2015, pp. 1–6.
[24] Adobe VoCo. http://www.bbc.com/news/technology-37899902.
[25] 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*, 2015, pp. 599–621.
[26] M. Pal, G. Saha, M. Pal, and G. Saha, "Spectral mapping using prior re-estimation of i-vectors and system fusion for voice conversion," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 11, pp. 2071–2084, 2017.
[27] H. Feng, K. Fawaz, and K. G. Shin, "Continuous authentication for voice assistants," in *Proc. of ACM MobiCom*, 2017, pp. 343–355.
[28] L. Lu, J. Yu, Y. Chen, H. Liu, Y. Zhu, Y. Liu, and M. Li, "Lippass: Lip reading-based user authentication on smartphones leveraging acoustic signals," in *Proc. of IEEE INFOCOM*, 2018, pp. 1466–1474.