Securing Liveness Detection for Voice Authentication via Pop Noises

Peipei Jiang, Qian Wang, *Senior Member, IEEE*, Xiu Lin, Man Zhou, Wenbing Ding, Cong Wang, *Fellow, IEEE*, Chao Shen, *Senior Member, IEEE*, and Qi Li, *Senior Member, IEEE*

Abstract—Voice authentication has been increasingly adopted for sensitive operations on mobile devices. While voice biometrics can distinguish individuals by their spectral features (such as voiceprints), they are known to be prone to spoofing attacks, where malicious attackers can use pre-recorded or synthesized samples from legitimate users or impersonate the speaking style of the targeted user to deceive the voice authentication system. In this paper, we design and implement a novel software-only anti-spoofing system on smartphones. Our system leverages the *pop noise*, which is generated by the user's oral airflow when speaking the passphrase close to the microphone. The pop noise is delicate and subject to user diversity, making it hard to be recorded by replay attacks beyond a certain distance or to be imitated precisely by impersonators. Specifically, we design a new pop noise detection scheme to pinpoint pop noises at the phonemic level, based on which we establish a theoretical model to calculate the sound pressure level from the speech signal in order to get the estimated pressure signal, and then analyze the consistency with the actual pressure signal extracted from the pop noise. Furthermore, we calculate the similarity score of the unique sequences which describe the individually unique relationship between pop noises and phonemes to resist spoofing attacks. Our evaluation on a dataset of 30 participants and three smartphones shows that our system achieves over 94.79% accuracy. Our system requires no additional hardware and is robust to various factors including authentication angle, authentication distance, the length of passphrase, ambient noise, etc.

Index Terms—Liveness detection, voice authentication, pop noises, anti-spoofing

- *Peipei Jiang is with the Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China, and also with the Department of Computer Science, City University of Hong Kong, Hong Kong SAR 999077, China. E-mail: ppjiang@whu.edu.cn.*
- *Qian Wang, Xiu Lin, and Wenbing Ding are with the School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China. E-mail: {qianwang, lxlyn, shybee}@whu.edu.cn.*
- *Man Zhou is with the School of Cyber Science and Engineering, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China. E-mail: zhouman@hust.edu.cn.*
- *Cong Wang is with the Department of Computer Science, City University of Hong Kong, Hong Kong SAR 999077, China. E-mail: congwang@cityu.edu.hk.*
- *Chao Shen is with the Ministry of Education (MOE) Key Laboratory for Intelligent Networks and Network Security, School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an 710049, China. E-mail: chaoshen@mail.xjtu.edu.cn.*
- *Qi Li is with the Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing 100190, China, and also with Beijing National Research Center for Information Science and Technology (BNRist), Beijing 100084, China. E-mail: qli01@tsinghua.edu.cn.*

Manuscript received 22 July 2020; revised 1 Mar. 2022; accepted 23 Mar. 2022. Date of publication 29 Mar. 2022; date of current version 14 Mar. 2023.
The work of Qian Wang was supported in part by NSFC under Grants U20B2049 and 61822207, and in part by the Fundamental Research Funds for the Central Universities under Grant 2042021gf0006. The work of Qi Li was supported in part by NSFC under Grant 62132011, and in part by BNRist under Grant BNR2020RC01013. The work of Chao Shen was supported in part by NSFC under Grants U21B2018 and 62161160337, and in part by Shaanxi Province Key Industry Innovation Program under Grant 2021ZDLGY01-02. The work of Cong Wang was supported in part by Research Grants Council of Hong Kong under Grants N_CityU139/21 and R6021-20F.
(Corresponding author: Qian Wang.)
Digital Object Identifier no. 10.1109/TDSC.2022.3163024

1 Introduction

Compared with password-based authentication, biometric authentication [2], [3] is more convenient since it is hands-free, and users do not need to memorize passwords. Compared with other biometric authentication, voice authentication is more low-cost, natural and convenient. 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, WeChat adds voice as a new interface, and users can log in through "Voiceprint" [4] generated from their voice passwords. Google allows users to unlock their phones of Android operating systems by voice biometrics [5]. Say Tec uses the voice biometric solution to support mobile financial services such as online payment and banking [6]. With the booming mobile technology, it is foreseeable that more voice authentication based mobile applications will spring up in the future.

However, since the sound transmits through an open and public channel, the voice authentication system is highly vulnerable to spoofing attacks [19], [20], [21]. There are two major types of spoofing attacks, namely, replay attacks and impersonation attacks [22]. In replay attacks, the adversary pre-records and playbacks the voice sample of the passphrase of a legal user to deceive the authentication system [23]. An adversary can also mimic the voice characteristics and style of a legal user to conduct impersonation attacks [24]. 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.

TABLE 1
An Overview of the State-of-the-art Anti-Spoofing Systems

System	Source of Distinctiveness	No Extra Device	No Extra Operation	No Position Constraint‡	Resist Replay Attacks	Resist Impersonation Attacks
VoiceLive [7]	Phoneme location	√	×	√	√	×
VoiceGesture [8]	Mouth motion	√	√	×	√	√
Lippass [9]	Mouth motion	√	√	×	√	√
LVID [2]	Mouth motion	√	√	×	√	√
Wivo [10]	Mouth motion	×	×	√	√	×
Chen *et al.* [11]	Magnetic field	√	×	×	√	√
CaField [12]	Sound field	√	√	√	√	×
Sahidullah *et al.* [13]	Throat vibration	×	√	√	√	×
Shang *et al.* [14]	Throat vibration	√	√	√	√	×
VAuth [15]	Body vibration	×	×	√	√	×
Wang *et al.* [16]	Oral airflow	×	√	√	√	√
Shiota *et al.* [17], [18]	Pop noise	√	√	√	√	×
VoicePop [1]	Pop noise	√	√	√	√	√
VoicePop+ (this work)	Pop noise & oral airflow	√	√	√	√	√

Note that, ‡: it refers to the constraints on relatively stable movements and fixed authentication angles between the user's lips and the microphone.

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.* [7] 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 [8], 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. But users need to move the smartphone with a predefined trajectory around the mouth while speaking the passphrase. VoiceGesture [8], Lippass [9], and LVID [2] are based on lip motions and leverage the theory of Doppler effect, where the smartphone is used as a Doppler radar to transmit high-frequency acoustic signals and monitor the reflection signal of articulators [8] and lip motions [2], [9]. However, the reflection signals are sensitive to the relative position between the smartphone and the user, especially the angles between the mouth, microphone, and speaker. M Sahidullah *et al.* [13] developed an ASV system against impersonation attacks using the throat microphone which is not available in most smartphones. Table 1 summarizes the characteristics of the state-of-the-art anti-spoofing systems. As shown, the attack detection mechanism proposed in this paper has no specific location restrictions except that the user is required to be close to the microphone as possible [1]. Besides, our system does not need any additional authentication equipment, which greatly improves the usability. As an extension to our conference paper, VoicePop [1], VoicePop+ further utilizes the feature of oral airflow pressure and adopts a new speaker verification method, which achieves higher detection rates in defending the replay attacks and impersonation attacks and better robustness against many potentially disrupting factors such as authentication positions, ambient noises, body movement, etc.

In this paper, we propose and implement VoicePop+[2], 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. Our observation is two fold: 1) The pop noise is subject to user diversity; 2) The recorded voice samples hardly contain the pop noise since the sound of the breath is gentle compared to the speech and will die out beyond a certain distance. Thus it is very difficult for attackers to imitate the way the legal user breathes. 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 the spectrogram characteristics. We design a novel pop noise detection algorithm to locate pop noises at the phonemic level. Since pop noise exhibits air pressure from the oral airflow, we can calculate the estimated pressure signal from the sound pressure level and then analyze the consistency between it and the pop noise. If they are inconsistent, the input sample is considered as a replay attack. 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 speaking the passphrase as required by current voice authentication systems. To our best knowledge, we are the first to analyze the characteristics of pop noise to

1. Our evaluation shows that a distance range of 4-12cm is feasible, and the distance of 4cm is recommended.

2. To avoid confusion with the conference version [1], we name the new system presented in this paper VoicePop+.

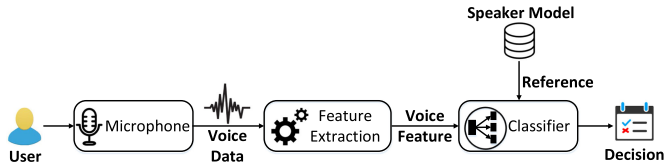


Fig. 1. A typical voice authentication system.

defend both replay attacks and impersonation attacks. We implement VoicePop+ on 3 types of smartphones and evaluate its performance with 30 volunteers under different experimental settings. The results verify the effectiveness of VoicePop+, with over 94.79% detection accuracy. The main contributions of this work are summarized as follows:

- We propose a novel pressure signal estimation scheme that relies only on built-in microphones. Based on the pop noise and its estimated airflow pressures, we extract effective and lightweight features that can well represent the individually unique phoneme-pop sequences with only three dimensions.
- We design a new speaker verification method, including a specific and concise feature extraction scheme by consistency analysis and similarity comparison. By leveraging the lightweight features extracted from pop noises and the airflow pressure, VoicePop+ can effectively and efficiently defend against replay attacks and impersonation attacks.
- We build a fully-functional VoicePop+ prototype using off-the-shelf smartphones. Extensive evaluation results on a dataset of 30 participants demonstrate that VoicePop+ can detect both replay and impersonation spoofing attacks with high accuracies and is robust to interference factors including authentication angles and distances, passphrase length, ambient noises, etc. VoicePop+ achieves an overall accuracy of 94.79%, and still performs well at the distance of 12cm and the authentication angle of 90°, with accuracies of 94.76% and 94.68%, respectively.

The remainder of this paper expands as follows. Preliminaries are presented in Section 2. We describe the detailed design of VoicePop+ in Section 3. Sections 4 and 5 presents the evaluations of the VoicePop+. We review related works in Section 6 and finally conclude our work in Section 7.

2 Preliminaries

2.1 Attack Model

VoicePop+ is designed for smartphone voice authentication, where the unique acoustic features of a user's voice are used to verify his/her identity in a convenient and reliable way. Voice authentication system can be text-dependent (requires the same password for enrollment and verification) or text-independent (accept arbitrary utterances from speakers). We primarily focus on the text-dependent authentication system, which is more widely adopted and commercially viable with a high authentication accuracy [25]. Fig. 1 displays a typical voice authentication system.

We make the following assumptions about the attacker.

Acoustic Attacks Only. An attacker can only attack the voice authentication system by sound. We do not consider scenarios in which an adversary attacks the hardware or operating system of a voice authentication system and interferes with the communication process of the authentication. Before the attack, the attacker can obtain the target user's voice sample by eavesdropping or from public resources but cannot obtain the stereo recording directly from the authentication system. We also assume that during the attack, the adversary can attack from any location without being noticed by the user.

State-of-the-art Software and Hardware. Attackers can use state-of-the-art equipment, including the microphone and speaker hardware of any type and quality, and can use the most advanced speech synthesis or conversion technology.

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. The replay attack includes two steps, i.e., the record phase and the replay phase. In the record phase, the adversary can stealthily record the victim's passphrase at a distance larger than 30cm. This assumption is reasonable because the recording distance cannot be very short (i.e., 2-6cm to the user's mouth). Otherwise, users can easily notice the illegal recording of the attacker. In the replaying phase, the adversary replays the recorded voice sample in front of the authentication device to perform the replay attack. In this step, the adversary can control the replay distance, i.e., the adversary can replay the recorded voice samples within 6cm from the authentication device. The record and replay devices can be high-quality professional voice recorders and loudspeakers.

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 attack, 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 close to the microphone.

2.2 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 contain not only 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 recording device's microphone very

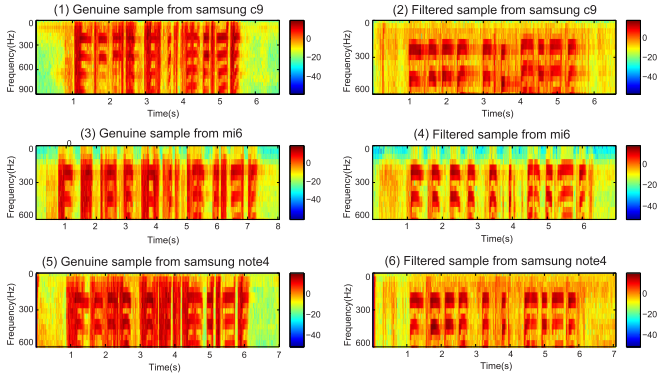


Fig. 2. Spectrogram comparison of samples without (left) and with (right) a pop noise filter using three different smartphones.

close to the user's mouth, and 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 in Fig. 2. We can find that pop noise has high energy in the low frequency (typically 0∼100 Hz), which has been discussed in the prior study [17]. Moreover, the duration of pop noise varies in the range 20∼100ms based on the way people speak and breathe. Our detection algorithm is based on these observations.

2.3 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. Each phoneme sound originates from different physical places in the human vocal tract system and is articulated in a certain manner with a specified degree of stricture in the oral tract and the escape channel. Since the tongue position is the most important physical feature that distinguishes one vowel from another [26], the articulation manners of vowels differentiate mainly according to the shape of the mouth. The range of positions of consonants is wider, and the typical 6 types of articulation manners of consonant are nasal, stop, fricative, affricate, approximate, and lateral [7]. Since each phoneme features unique physical origin in the human vocal tract system and has its own manner of pronunciation, the existence probability 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. As shown in our earlier conference paper [1], we have collected speech data from 18 volunteers and ranked the phonemes according to the existence probability of pop noise. The existence ratio of the pop noise of phoneme X is calculated as $\frac{POP_X}{N_X}$, where N_X is the occurrence number of phoneme X, and POP_X is the occurrence number of the pop noise of phoneme X for all people. As shown in Table 2, 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.

TABLE 2
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

2.4 Airflow Pressure and Pop Noise

According to [16], different phonemes will cause different airflow pressure levels due to different vocalizations. With regards to the pop noise we discuss in this paper, we observe that the plosive burst of pop noise will cause a higher airflow pressure. In other words, the phoneme that contains pop noise usually has higher airflow pressure than other phonemes. Based on this observation, we design a pressure estimation algorithm and use the consistency of the pop noise and its airflow pressure to distinguish the replay of recorded voices from real voices.

3 VoicePop+: Design Details

3.1 Overview

The key idea of our anti-spoofing system is to identify a legal user based on the located pop noise, the extracted oral airflow pressure signal, 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 three phases: *data collection*, *data process*, and *speaker verification*.

In the first phase, when a user performs authentication, the built-in microphone captures the user's speech, which will then be 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 attack detection. Note that when the voice is recorded, it will be denoised first, which is a built-in function of the smartphone and is not the focus of this paper, and thus we will omit the description of this step in the following sections.

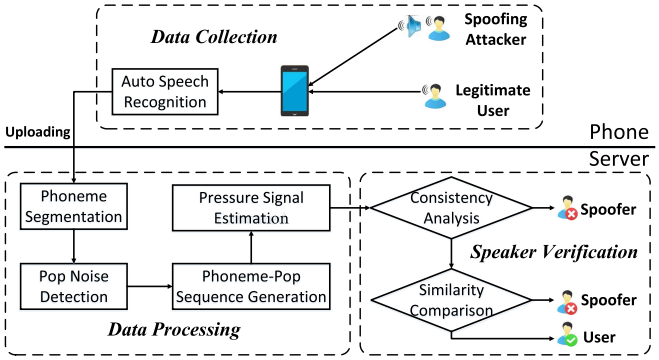


Fig. 3. The architecture of VoicePop+.

In the 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. According to the segmented phonetic units and the located pop noise obtained in the above steps, a binary phoneme-pop sequence is generated, which describes the presence of pop noises at each phoneme. Then we calculate the sound pressure level corresponding to each phoneme position from the speech signal and convert it into the pressure signal through a theoretical model.

In the speaker verification phase, we first conduct a consistency analysis between the estimated pressure signal and the phoneme-pop sequence that reflects the real pressure signal, and calculate two consistency scores. Then, we compare the authentication phoneme-pop sequence with that stored in the user profile and compute a similarity scores. We further combine the above three scores and construct a 3-dimensional feature. Finally, we train a binary logistic regression model to obtain an optimal decision boundary (i.e., the weights and thresholds for each score) that can distinguish the spoofers and the legitimated users. The detection result can be integrated into general voice authentication systems for user identification.

Note that compared to the conference version [1], we add a new step in the data processing phase and design a new speaker verification scheme. Specifically, besides the pop noise, we propose a new method to extract the oral airflow pressure signal for the consistency analysis in the verification phase. Instead of using the generic GFCC features and SVM classifier, we propose a more specific and concise verification method, which reduces the workload of training and improves the efficiency of the system. Due to the adoption of new features and speaker verification method, VoicePop+ is more robust to authentication positions, ambient noises, body movement, etc.

3.2 Phoneme Segmentation

A phoneme is made up of a number of distinctive overtone pitches, as known as formants. Formants refer to the area 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.

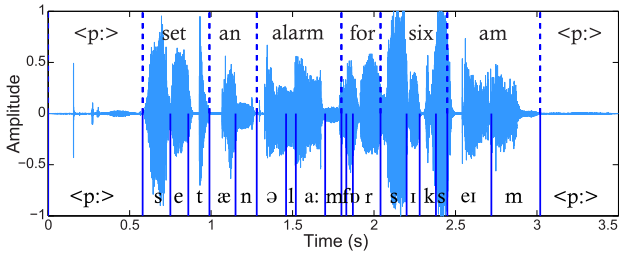


Fig. 4. An example of phoneme segmentation.

To attain phoneme segmentation, we first generate the spectrogram of the voice sample using a spectrum analyzer and 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 [27] 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.

3.3 Pop Noise Detection

As we have discussed in Sections 2.2 and 2.3, 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 [17], 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.

3.3.1 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, for the reason that they often have similar characteristic of high energy at low frequencies.

3.3.2 Short-Time Fourier Transform

We use the Short-Time Fourier Transform (STFT) to acquire the time domain information such as the frequency

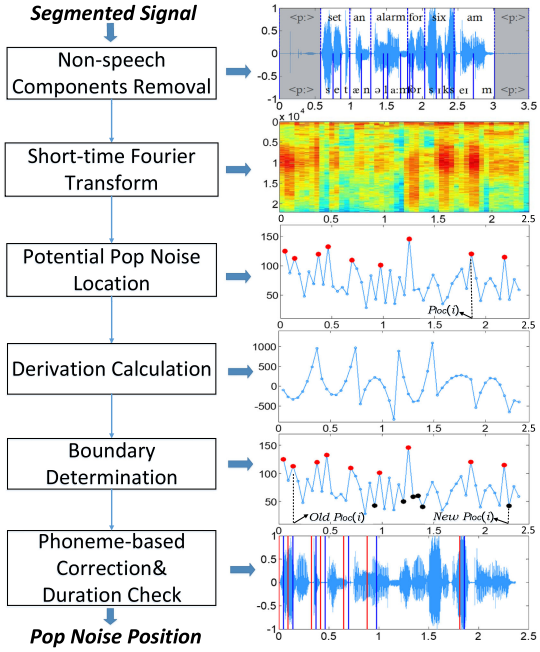


Fig. 5. Pop noise detection.

distribution changes over time. STFT is a time-frequency analysis technique especially for non-stationary signals that can determine the sinusoidal frequency and phase content of local sections of a signal. 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.

3.3.3 Potential Pop Noise Location

After STFT, we get the frequency distribution of each frame. We first compute the energy within the frequency range 0∼170 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 $E(i) > 3 \cdot E_{std}$.

3.3.4 Derivation Calculation

The previous step pinpoints the peaks of potential pop noises, and then 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 $D(i)$.

3.3.5 Boundary Determination

We find the boundaries of pop noises by searching the vicinity of $Loc(j)$ up to 3 points. If the nearby point k, where $Loc(j) - 3 \leq k \leq Loc(j) + 3$, 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.

Algorithm 1. Phoneme-Pop Sequence Generation Algorithm

Input: 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 $\{ET\_pho_j\}_{j=1}^{m}$.
Output: 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
  j--;
  /*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 = 1;
      S_{j+1} = 1;
      j++;
    end while
    j--;
  end if
end for
return S
```

3.3.6 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∼100ms. Hence, we check the duration of potential pop noises and abandon those out of this range.

3.4 Phoneme-Pop Sequence Generation

Through the extensive experiments shown in Section 2.3, we find that the positions of pop noises are different for different people. This is because every person has his/her own speaking style and special vocal system. Therefore, we build a binary phoneme-pop sequence for each user's passphrase 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.

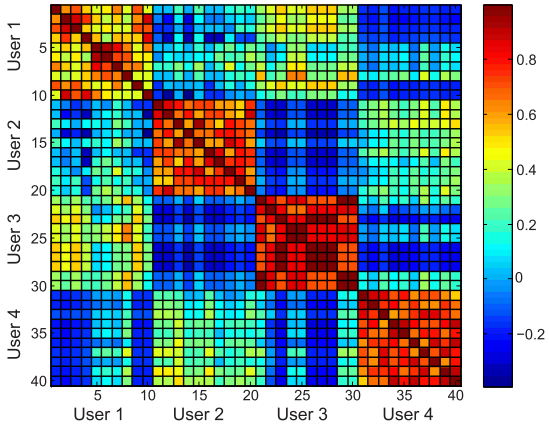


Fig. 6. Phoneme-pop sequence similarity between different pairs of users.

We design a phoneme-pop sequence generation algorithm (shown in Algorithm 1) to identify the unique relationship between phonemes and pop noises for each user. Here we briefly generalize the algorithm. For the first pop noise period, the algorithm begins to scan the phonemes to find the first phoneme the pop noise appeared in and then determines whether the appearance of the pop noise involves more than one phoneme. If there is pop noise in one phoneme, we set the element of the sequence to 1 corresponding to this phoneme's position. For the second and later pop noise periods, we repeat the above steps until all the sequences are generated.

To prove the user uniqueness of the phoneme-pop sequence, we use the Pearson correlation coefficient, which is used to measure the degree of linear correlation between two sequences, to calculate the similarity between different users. The coefficient value is within the range [-1, 1], where an absolute value near 1 indicates a strong linear correlation, while a value near 0 indicates a lack of linear correlation. Fig. 6 shows the similarity (i.e., 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.

3.5 Pressure Signal Estimation

Besides generating the phoneme-pop sequence, we intend to extract the estimated pressure signal of the oral airflow from the speech signal and then conduct the consistency analysis with it. The pressure signal is calculated from the sound pressure level according to the theoretical model, and the calculation of the sound pressure level depends on the energy of the signal. The whole pressure signal generation process is divided into four parts: acoustic power acquisition, sound power level obtaining, sonority scale calculation, and pressure scale conversion.

3.5.1 Energy Power Acquisition

We calculate the energy power of each phoneme from the frequency domain. Recall that we use STFT to analyze the voice signals in the time domain in Section 3.3.2. Here in this step, we use STFT to calculate the spectrum energy within the time range of each segemented phoneme. After STFT, we will obtain the starting time of each window, denoted as $\{T_k\}_{k=1}^{l}$, where l means the number of STFT time windows, and the energy distribution with frequency in each window, denoted as $\{P_{bc}\}_{b=1,c=1}^{b=B,c=l}$, where B means the number of STFT frequency windows. Therefore, the obtained energy is a two-dimensional vector. The x-coordinate represents the frequency distribution, and the y-coordinate represents the time distribution. According to each phoneme's time period, we first determine which windows are corresponding to it, and calculate the average window energy value in these windows as the final energy of the phoneme. For the i-th phoneme, its energy power $EnergyP_j$ can be calculated as

$$EnergyP_i = \frac{\sum_{c=j}^{j+M-1}\sum_{b=1}^{B}|P_{bc}|}{M}, \tag{1}$$

where j and M are the first time window and the number of the time windows in the time period of the i-th phoneme, respectively. It is worth noting that this step calculates the energy power directly from the voice signal recorded by the built-in microphones and does not require any specialized device to record the airflow and obtain the energy power, which is more applicable than [16].

3.5.2 Sound Power Level Obtaining

For the $i-$th phoneme, its sound power level SPL_i can be obtained from the waveform by using the corresponding timestamp. For an input speech signal, the sound power level (SPL_i) in decibels (dB) is derived as follows:

$$SPL_i = 10 * log_{10}\left(\frac{EnergyP_i}{EnergyP_{ref}}\right), \tag{2}$$

Then we can use SPL_i to standardize the audio energy sequence obtained above by reference energy value, which we denote as $AcousticP_i$. The formula is as follows:

$$AcousticP_i = EnergyP_{ref} * 10^{\frac{SPL_i}{10}}, \tag{3}$$

where $EnergyP_{ref}$ is the reference acoustic power, which is typically 10^{-12} watts (i.e., 0 dB).

3.5.3 Sonority Scale Calculation

After calculating the acoustic power sequence of the segmented phonemes, we then construct the sonority hierarchy of the phoneme sequence. Sonority is the scalar of phonemes, which refers to the loudness of other phonemes. In general, vowels are louder than consonants because of the different pronunciation manner and physical origin. The sonority of each phoneme can be expressed as the ratio of the power of the current phoneme in the speech signal to the weakest power in all phonemes. Sonority is calculated by comparing each phoneme's energy with the weakest energy in speech. The expression is as follows:

$$Sonority_i = \frac{AcousticP_i}{AcousticP_{min}}, i = 1, 2, ..., m, \tag{4}$$

where $Sonority_i$ represents the sonority of the i-th phoneme, and $AcousticP_{min}$ is the minimum power among all phonemes.

3.5.4 Pressure Scale Conversion

The last step is to convert the sonority scale into the airflow pressure sequence. There have been several studies about the relationship between sonority and oral airflow pressure. Wang *et al.* [16] build a relationship model to estimate the oral airflow pressure according to the phoneme sonority scale, depending on the conclusion in [28] that the sonority is inversely correlated with oral airflow pressure and the observation in [29] that the correlation coefficient between sonority and the oral airflow pressure is approximate -0.84. Here we use this theoretical model to estimate the pressure signal. The formula is given as

$$Pre_i = \mu * Sonority_i + \nu, i = 1, 2, ..., m, \tag{5}$$

where Pre_i represents the estimated pressure of the i-th phoneme, μ is the correlation coefficient between sonority and pressure, and ν is a constant term used to adjust the pressure coordinates to positive values, which is set to 169.85 in our experiments[3]. We construct the sonority hierarchy and the estimated pressure for the speech signal "set an alarm for six am", as illustrated in Fig. 7.

3.6 Speaker Verification

In this part, we will make a double check according to the obtained characteristics above to ensure that VoicePop+ can resist both replay attacks and impersonation attacks. Specifically, we conduct the consistency analysis based on the phoneme-pop sequence and the estimated pressure signal and the similarity comparison based on the phoneme-pop sequence. We then construct a 3-dimensional feature from the consistency and similarity scores, and finally use logistic regression to verify the speakers.

3.6.1 Consistency Analysis

In this step, we use the phoneme-pop sequence and the estimated pressure signal for consistency analysis. We find that the phoneme that contains pop noise usually has higher airflow pressure than other phonemes, because pop noises are formed by plosive bursts. Based on this observation, we design two consistency scores, i.e., $pro1$ and $pro2$, for the phonemes with low and high airflow pressure, respectively.

Recall that the phoneme-pop sequence describes the phoneme positions where pop noises occur, so we think that the pressure values should be very different for the element with value 0 and the element with value 1 in the sequence. Therefore, we use the following formula to calculate the consistency score to represent the consistency of those phonemes whose pressure is low:

$$pro1 = \frac{\Sigma_{i=1}^{m} \mathbb{I}(S_i = 0 \wedge Pre_i <= mean(Pre))}{m_0}, \tag{6}$$

3. Since we only compares the relative pressures of the phonemes, the choices of μ and ν will not affect the experimental results.

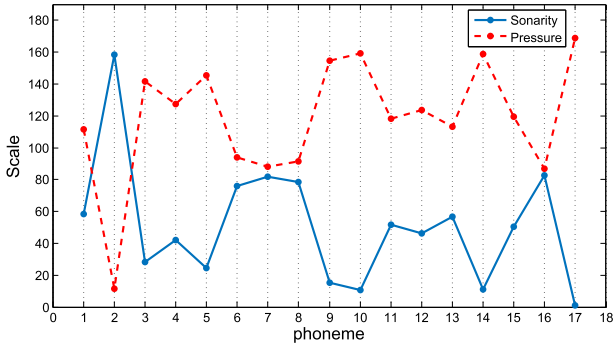


Fig. 7. Scale constructions for sonority and pressure.

where m_0 is the total number of those phonemes with low probabilities of pop noises, and $\mathbb{I}(x)$ is an indicator function. If the condition in parentheses is true, $\mathbb{I}(x)$ adds 1 to its counting value. The formula indicates that when the phoneme period does not contain pop noise, the pressure caused by the oral airflow at this phoneme position should be lower than the average of the overall sequence pressure.

For the phonemes whose pressure is relatively high in a phoneme sequence, we use the following formula to calculate another score $pro2$ for a double check:

$$pro2 = \frac{\Sigma_{i=1}^{m} \mathbb{I}(S_i = 1 \wedge Pre_i > mean(Pre))}{m_1}, \tag{7}$$

where m_1 is total number of those phonemes with low probabilities of pop noises. $pro2$ indicates that when pop noise occurs in some phonemes, the pressure of these phonemes should be higher than the average value.

Since the adversaries cannot record valid pop noises in the replay attacks, the replayed samples contain few pop noises. Even if there are pop noises detected in the replayed samples, the pop noises are not caused by natural airflows from human. Therefore, the consistency scores $pro1$ and $pro2$ of the recorded samples will be very different from real samples, and we use $pro1$ and $pro1$ as part of the features to distinguish the fake samples and real samples.

Taking impersonation attacks into consideration, we design a similarity comparison method based on the individually unique phoneme-pop sequence. When the user enrolls in the system, he or she is required to perform the authentication trail for 5 times. Thus, we collect 5 phoneme-pop sequences for each user and store them in his/her user profile. Using these sequences, we can get a probability rank through the following formula:

$$Pro_i = \frac{\Sigma_{i=1}^{5} \mathbb{I}(S_i = 1)}{5}. \tag{8}$$

Then we use the following formula to calculate the similarity score $pro3$:

$$pro3 = \frac{\Sigma_{i=1}^{m} \mathbb{I}(Pro_i >= 0.6 \wedge S_i == 0)}{m} + \frac{\Sigma_{i=1}^{m} \mathbb{I}(Pro_i <= 0.2 \wedge S_i == 1)}{m}. \tag{9}$$

It is worth mentioning that from the perspective of reliability, we choose those phonemes whose probability

of containing pop noise is larger than 0.6 and calculate their amount of inconsistency. Namely, if the phoneme's probability satisfies the above condition while the corresponding element in the phoneme-pop sequence is 0, the statistic value will be increased by 1. Similarly, if the phonemes' probability is less than 0.2, we count the number of elements with a value of 1 at these phoneme positions. If the sequence is similar, the value of $pro3$ will be low, indicating that there will be fewer sequence errors.

3.7 Verification Method

Finally, we construct the feature from $pro1$, $pro2$, and $pro3$ and leverage a binary logistic regression model to find the best classification boundary. The feature can be represented as follows,

$$\mathbf{f} = (pro1, pro2, pro3). \tag{10}$$

Theoretically, according to the definitions of the scores, a single $pro1$, $pro2$, or $pro3$ can roughly divide the true samples and false ones by setting a threshold and comparing the score with it. Therefore, the linear combination of the scores is linearly separable, which is suitable to adopt a binary logistic regression model here.

Specifically, we consider a standard logistic function,

$$h_\theta(\mathbf{f}) = \frac{1}{1 + e^{-\theta^T \mathbf{f}}} = Pr(Y = 1|\mathbf{f}; \theta), \tag{11}$$

where $h_\theta(\mathbf{f})$ denotes the probability of predicting Y as 1 for an input $\mathbf{f}$, and θ is the function parameter. Note that $\theta^T \mathbf{f}$ is the decision boundary we need. To obtain an optimal decision boundary, we use the following loss function,

$$Loss(h_\theta(\mathbf{f}), Y) = \begin{cases} -\log\left(h_\theta(\mathbf{f})\right), & if \quad y = 1, \\ -\log\left(1 - h_\theta(\mathbf{f})\right), & if \quad y = 0. \end{cases} \tag{12}$$

Then we can use the gradient descent algorithm to find the optimal θ that minimizes the loss. Finally, we calculate $h_\theta(\mathbf{f})$ and compare it with 0.5. If $h_\theta(\mathbf{f}) > 0.5$, we regard the input passphrase as a true sample. Otherwise, we consider this authentication as a spoofing one.

4 Implementation

We implement a prototype of the typical client-server architecture, as shown in Fig. 8, and build it on several smartphone testbeds to evaluate and validate the performance and effectiveness of our system. Our prototype is consisting of two parts: 1) a mobile application running on Android and 2) a processing backend running on one ThinkPad server with Intel(R) Core(TM) i7-7500U 2.70 GHz CPU and 8 GB of RAM.

4.1 Mobile Application

The mobile application is designed for users to record acoustic data at a sampling rate of 44.1kHz, which then recognize the words of the speech and upload the raw acoustic data and its corresponding text to the server in real-time.

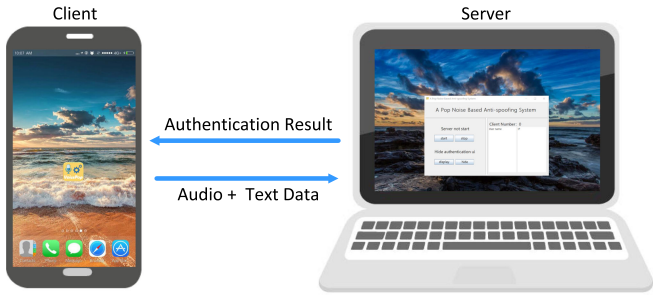


Fig. 8. The implementation overview of VoicePop+.

4.2 Server Backend

The implemented server liveness detection program, which is primarily coded by JAVA, has two main functionalities: data receiving and result feedback and data processing and verification.

Data Receiving and Result Feedback. The server communicates with the mobile phone via secure socket protocol and is capable of receiving the users' acoustic data. After the data processing pipeline, it will return the verification result directly to the user's phone. We implement this communication model by JAVA language.

Data Processing and Verification. At the server side, the received data is fed into a processing pipeline just as we described in Section 3, including phoneme segmentation, pop noise location, phoneme-pop sequence generation, consistency analysis, and similarity comparison. We implement this part by matlab and package the processing function as a JRE package, which is available for our server written in JAVA to invoke.

5 Evaluation

In this section, we evaluate the proposed anti-spoofing system under both replay attacks and impersonation attacks. We also evaluate the robustness of our system under different angles and distances between the microphone and the user's mouth, different lengths of passphrases, different speaking speed, different types of smartphones, different body movements and different environments.

5.1 Experimental Setting

5.1.1 Data Collection

We recruit 30 volunteers (18 males and 12 females) to participate in the experiments. We use three different smartphone models running Android 6.0 KitKat for authentication, as shown in Table 3. The participants are undergraduate and graduate students who are instructed to perform voice authentication with VoicePop+. Since VoicePop+ detects pop noises caused by user breathing while speaking, we require the users to speak close (i.e., within 12cm) to the microphone, but do not request them to hold the phone at a specific place or distance. Fig. 9 shows a typical use case, and we will discuss the effective distance and the impact of authentication angles later.

To build the user profile, including phoneme-pop sequences and pop noise existence probability sequences, we ask each participant to speak a passphrase five times upon registration. The passphrase of each participant is randomly selected from a pre-defined command set, and the

TABLE 3
Devices Used in the Experiments

Maker	Model	Authentication	Record	Replay
XiaoMi	Mi6	√		
Samsung	C9 pro	√	√	
Samsung	S7 edge	√		
XiaoMi	Mi10S		√	√
Huawei	Mate10		√	√
Samsung	Note4		√	
Phillips	VTR8060		√	√
Amazon	Echo dot			√
Hivi	M200mkIII			√
Earise	Al-101			√

lengths of the passphrases range from 3 to 10 words. Then, each participant chooses three passphrases and performs 10 trails for each passphrase.

5.1.2 Attack Methods

We evaluate our system under replay attacks and impersonation attacks.

Replay Attack. In the replay attack, the adversary holds a microphone (e.g., a smartphone or a professional voice recorder) in front of the legitimate user at a distance of 30cm and records the passphrase when the legitimate user is performing voice authentication. Then, the adversary replays the pre-recorded passphrase in front of the authentication smartphone's microphone at a distance of 4cm by a speaker (e.g., a smartphone or a loudspeaker). For each passphrase of each participant, we conduct 10 replay attack trials.

Impersonation Attack. We consider three types of impersonation attacks: playback with random breath and playback with breath impersonation. Specifically, the adversary replays the target user's voice sample by using a loudspeaker and imitates the user's breath (or just randomly breathe) in front of the smartphone at the same time for voice authentication. We also conduct a training process for each adversary [30]. Before the training, the adversaries can only listen to the recorded voice samples once, and the imitated breathing can be regarded as "random breathing". In the training phase, the volunteers are asked to observe the target user's breathing (especially the speed of talking and the breath style) in the authentication phase ten times and then imitate the breathing of the target user. To guarantee the quality of training, we then ask the target user to listen to the imitated voice samples produced by the volunteer and give specific feedback to him/her for better imitation (e.g., breathe more lightly, breathe more quickly, etc.). The volunteer will adaptively imitate the target user's breathing with the help of the feedbacks. For each volunteer, the data collection of the training process is considered done if and only if all the imitations are sufficiently satisfied by the corresponding target user. Then, we allow the adversary to imitate the legitimate users within a limited number of times, e.g., five attempts. This setting is practical and reasonable because the voice authentication system may be locked if the user fails multiple authentications. In our experiments, we recruit 10 volunteers as adversaries, and each impersonates 3 participants for 10 trails.

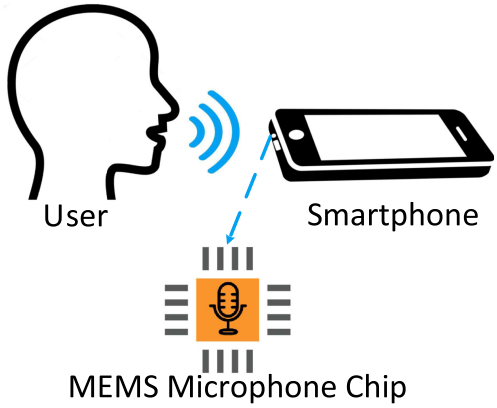


Fig. 9. A typical application of VoicePop+.

5.1.3 Devices

Table 3 lists the devices used in our experiments. For user authentication, we use three off-the-shelf smartphones, including a XiaoMi Mi6, a Samsung C9 pro, and a Samsung S7 edge, to record the user's passphrases upon registration and verify the users in the authentication phase. In the replay attack, we employ 5 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. The professional recorder we use is a Phillips VTR8060 voice digital recorder with high-quality microphones. We also use five speaker models, including the standalone loudspeakers and built-in speakers of mobile devices, to playback the recorded voice samples. Specifically, we use a popular commercial voice assistant, Amazon Echo dot, a small and lightweight loudspeaker, Earise Al-101, and a professional loudspeaker, Hivi Swans M200mkIII. The Hivi speaker can produce high definition and room-filling sound at 120W RMS.

5.1.4 Metrics

We adopt three metrics to evaluate the performance of our system: True Positive Rate (TPR), True Negative Rate (TNR), and accuracy. TPR is the probability that the system correctly identifies a legitimate user. TNR is the probability that the system correctly detects spoofers. TPR and TNR measure the accuracy of the system for user identification and spoofer detection, respectively. Accuracy measures the likelihood that the system accepts legitimate users and rejects attacks.

5.2 Overall Performance

We confirm the effectiveness of our system against replay attacks and impersonation attacks by comparing it with the baseline (VLD) [18] and the conference version of VoicePop [1]. In [18], Sayaka Shiota *et al.* proposed the pop noise detector combined with the phoneme information to detect the existence of pop noises, but the replayed samples were easily recognized as legitimate samples under their proposed algorithm. However, VLD does not consider using the characteristics of the pop noise for further classification, nor does it consider the impersonation attack when the adversary replays the audio and mimics breathing at the

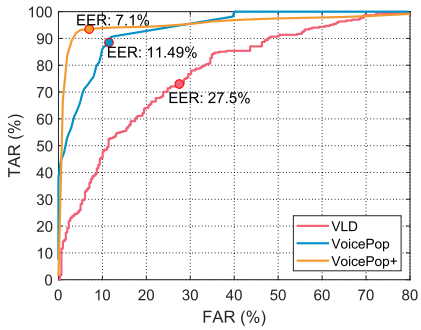


Fig. 10. Overall ROC curves with EERs.

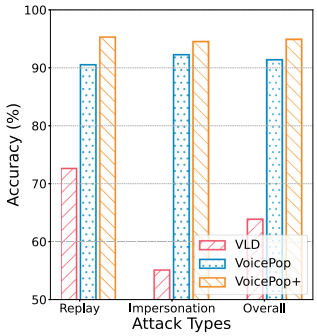


Fig. 11. Overall accuracy.

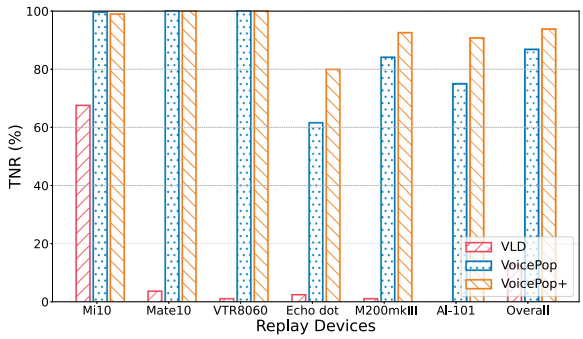


Fig. 12. TNR under different replay devices.

same time. Figs. 10 and 11 demonstrate the overall ROC curves and accuracies of VoicePop+, VoicePop, and VLD under the spoofing attacks. It is shown that our system achieves an overall accuracy of 94.79% and EER of 7.1% under replay and impersonation attacks, far outperforming the baseline VLD that has an accuracy of 63.97%. Besides, the overall accuracy of VoicePop is 91.29%, indicating that the new speaker verification scheme in VoicePop+ is more effective than the direct use of GFCC and SVM in VoicePop. This also shows that the combination of pop noise and its airflow pressure can improve the detection rate of the pop noise-only feature.

Replay Attacks. Here we take a closer look at the performance of VoicePop+ under different record and replay devices listed in Table 3. As shown in Fig. 12, the TNR of VoicePop+ is relatively stable under different replay devices, and it is always more effective than VLD in replay detection. Since VLD only detects the existence of the pop noise, and in the replay samples, it is very easy to detect the "pop noise" wrongly under their algorithm. Therefore, VLD

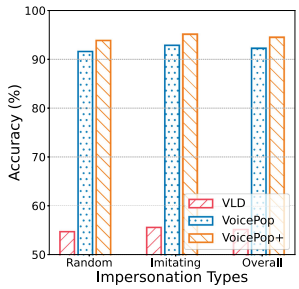


Fig. 13. Accuracy under different impersonation attacks.

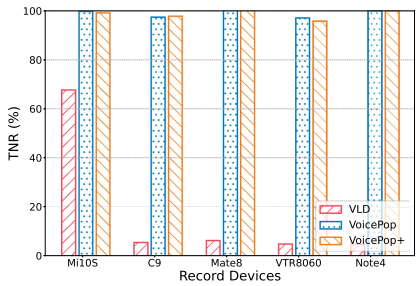


Fig. 14. TNR under different record devices.

tends to classify a false sample as a true one, and the TNR of VLD is very low. These results demonstrate the robustness of VoicePop against replay spoofing attacks. We also find that all the three schemes have a decline in TNR when facing the Hivi speakers. This indicates that a high-quality replay of the recorded voice can improve the attack success rate, but the cost of the device will be high.

Fig. 14 shows the performance of VoicePop+ under different record devices. Both VoicePop+ and VoicePop achieve high TNRs in all devices. Interestingly, the TNR under the professional recorder, i.e., Phillips VTR8060 is slightly lower. This indicates that a professional recorder can help improve the attack success rate, but the probability of being detected is still high. Another interesting finding is that VLD performs much better when facing Mi10S than other devices. This may be because the false samples in the training dataset for the three schemes are constructed from false passphrases recorded by Mi10S. Thus, VLD can perform better under Mi10S, but it fails to distinguish the recorded phassphrases from other devices, indicating that VLD is not robust when facing different record devices. However, in the meantime, both VoicePop+ and VoicePop can perform well even if the training dataset does not include the recorded passphrases from the attacking devices.

Impersonation Attacks. We also dig deeper into impersonation attacks. Since pure impersonation attacks can be directly recognized by the voice authentication system (e.g., the voiceprint verification), we do not consider such attacks here. We consider two ways of attacks: playback with the random breath and playback with breath impersonation. As shown in Fig. 13, VoicePop+ has a superior performance over the baseline under two 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. VoicePop+ leverages the unique relationship between phonemes and pop noises of each individual to extract location sequence

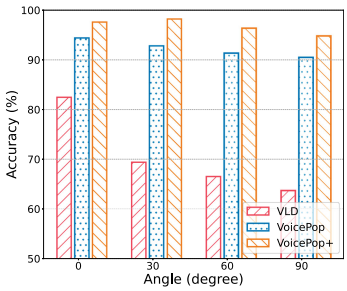


Fig. 15. Impact of authentication angles.

features. This feature is user-dependent, and the attacker can hardly impersonate the breathing in precise synchronization at the phonemic level. Compared with VoicePop, the accuracies of VoicePop+ are 93.76% and 95.04% under the random attack and the imitating attack, respectively, which improves the performance of VoicePop (i.e., 91.51% and 92.78%). The results also show that the training phase for the adversaries can slightly improve the attack success rate, but it is still hard for the adversaries to imitate the victim's breathing style even if they have observed the victim's speech for quite a long time.

5.3 Impact of Authentication Angle

In practice, it is hard to strictly constrain the authentication angles between the microphone and the user's mouth. When performing the authentication, the angle may be different from that in the register phase, which may lead to an inconsistency of the register profile and the recorded feature in the authentication phase. Hence, we evaluate the performance of VoicePop+ under different authentication angles. Fig. 15 shows the accuracy of VLD, VoicePop, and VoicePop + under different authentication angles. Results show that VoicePop+ is robust to the authentication angle, and the accuracy remains 94.55% when the angle is 90°. The accuracy decreases of VoicePop+ and VoicePop are both about 4%, which is much smaller than that of VLD, i.e., a 20% decline, showing that the pop noise detection and extraction schemes in VoicePop+ and VoicePop are more robust to different positions between the microphone and user's mouth.

5.4 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. Fig. 16 presents the accuracy of VoicePop+, VoicePop, and VLD

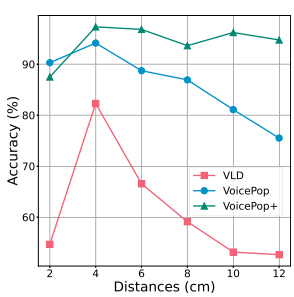


Fig. 16. Impact of authentication distances.

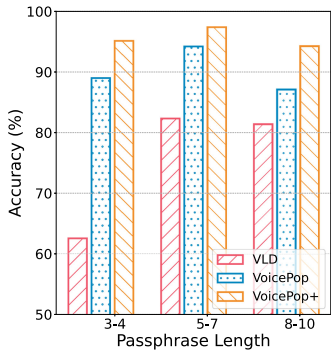


Fig. 17. Impact of passphrase lengths.

under different authentication distances. In particular, the accuracy of VoicePop+ is satisfactory when the distance ranges from 2cm to 12cm, with a minimal accuracy of 87.54% under 2cm. The accuracy is degraded when the distance is too short, since a strong breath will affect the accuracy of pressure estimation and the stability of the phoneme-pop sequence. For VoicePop, the accuracy decreases as the distance increase when the distance is larger than 4cm. Since the breath while speaking is gentle, and its power decreases as the distance increases, the microphone may miss a few pop noises. VoicePop+ performs much better than VoicePop when the distance is larger than 10cm, and the accuracy remains over 94% at a distance of 12cm. This shows that the airflow pressure used in the feature can really help distinguish the samples when the pop noise detection is unstable in larger distances. We also notice that the results are a little different from those given in the conference version. In the conference version, when the distance is larger than 10cm, the accuracy suddenly drops to below 20%. This is because this experiment was conducted on a small dataset of three volunteers, where individual deviations may have occurred and caused this inconsistency. In this paper, the dataset includes much more speech samples from 30 volunteers, which statistically eliminates more individual biases. Besides, all the three schemes perform the best at the distance of 4cm, which shows that this is the best distance to capture the pop noise. Hence, we recommend setting 4cm as the default authentication distance.

5.5 Impact of Passphrase Length

Generally, a longer passphrase provides 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. 17 illustrates the accuracy of different lengths of passphrases. It is shown that VoicePop+ can achieve a very high anti-spoofing effectiveness even when the length of the passphrase is less than 5. We also observe that medium-length passphrases perform slightly better. This is an interesting finding because the accuracy should improve with the increase in passphrase length theoretically. Indeed, a longer passphrase contains more pop noises and more distinctions in the phoneme-pop sequence. Nonetheless, a longer passphrase may contain more changes (e.g., speaking speed), thereby increasing the possibility of inconsistency with the profile. But we emphasize that the accuracy of VoicePop+ is still high in long

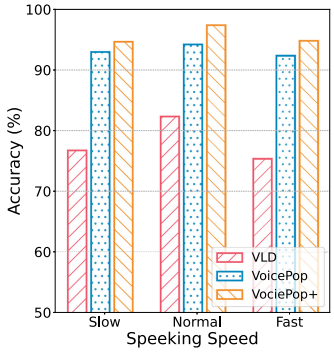


Fig. 18. Impact of the speaking speed.

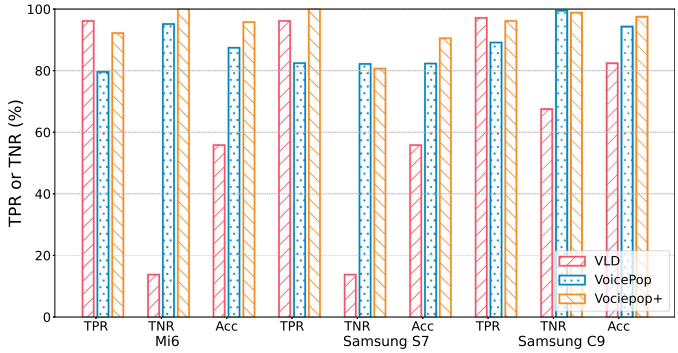


Fig. 19. Impact of authentication phones. "Acc" is short for accuracy.

passphrases, and the users can choose a long passphrase according to their preferences.

5.6 Impact of Speaking Speed

Different people usually speak at different speeds, and in different situations, one person may also speak at different speeds. Therefore, we further verified the performance of the system at different speaking speeds. Fig. 18 shows the accuracy for different lengths of the passphrases at three kinds of speeds. We divide the speaking speed into three types: fast, which corresponds to speaking one word for 0.4 seconds on average, normal, which corresponds to 0.65 seconds for a word, and slow, 1 second for a word. We observed that the accuracy was the highest for each length of passwords when at normal speaking speed. When the users register in, they use normal speaking speed, so the sequences of the two are more matching. We can also observe that the accuracies when speaking at slow and normal speed are not much different. However, when the user speaks quickly, the accuracy drops a lot. On the one hand, fast speaking may lead to poor pronunciation, which will affect the phoneme segmentation result. On the other hand, there will be linking between the words when speaking too fast, which will eat up a phoneme, resulting in the instability of the phoneme-pop sequence. However, the accuracy remained at 94.76% when using the longest passwords.

5.7 Impact of Authentication Phone

As we know, the microphones of different smartphones have diverse frequency selectivity [31]. Thus we study the performance of our system on different smartphones. As shown in Fig. 19, we observe that VoicePop+ can resist spoofing attacks with accuracies of 95.61%, 90.71%, and 97.33% when using Mi6, S7, and C9 as the phone for authentication, respectively. Although the accuracy when using the S7 phone for authentication was low, it remained above 90%. The results demonstrate that VoicePop+ is robust and compatible with different phone models.

5.8 Impact of Body Movement

Body movement will affect the breathing pattern and also the relative position between the user and the smartphone. To evaluate whether VoicePop+ is robust to body movement, we let the volunteers walk slowly or quickly during the authentication phase, and the results are given in Fig. 20. The accuracies of VoicePop+ remains as high as 94% when the user walks quickly, showing the robustness of VoicePop+ to body movements. We also notice that VoicePop has a high TNR and low TPR (i.e., 94% and 90%), while VLD has a high TPR and low TNR (i.e., 99% and 30%). It indicates that when the movement is unstable, VLD tends to accept a sample, while VoicePop tends to reject a sample.

5.9 Impact of Ambient Noise

The ambient noise usually has high energy at low frequencies [32], [33], which may interfere with the pop noise detection, and 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. 21, we can see that the overall accuracies of VoicePop+ of three phones in four different environments are all above 94%, and the accuracy is as high as 97.6% in the anechoic chamber. We can see that the environment does have a little impact on the performance of the system, but not too much. This is because we have enabled two features in the consistency analysis that increases the double threshold to mitigate the random impact of the environment. As for VLD, the accuracy drops a lot in noisy environments. The results demonstrate that VoicePop is robust to ambient noise.

5.10 Efficiency

Compared with VoicePop, a big advantage of VoicePop+ is its high efficiency. In VoicePop+, we extract low-dimensional features from the pop-phoneme sequence and airflow

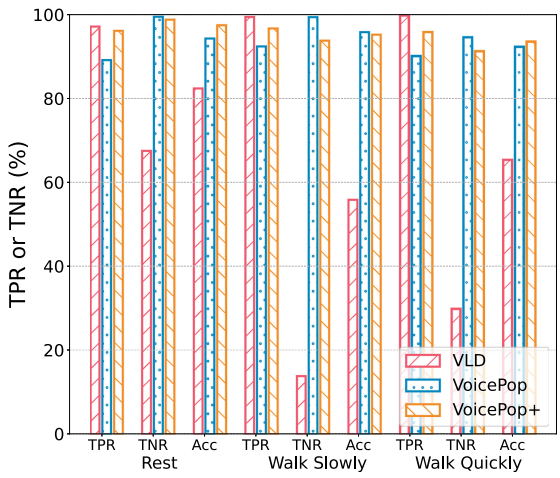


Fig. 20. Impact of body movements.

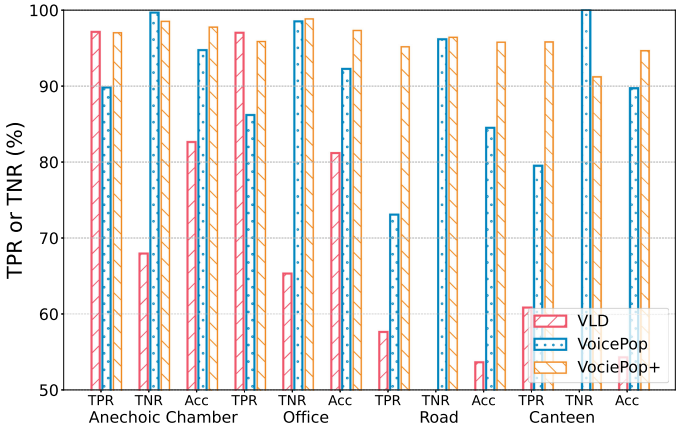


Fig. 21. Impact of ambient noises.

pressures and adopt the lightweight classification model, i.e., a binary logistic regression model. However, VoicePop directly adopts the generic GFCC feature and a SVM classifier, whose training process is complicated and time-consuming. Therefore, VoicePop+ reduces much workload of training and improves the efficiency. We test the running time of the computational cost of VoicePop+ and VoicePop in the training and test phases. The training dataset and the test dataset contain 290 and 595 passphrase samples, respectively. We run VoicePop+ and VoicePop for ten times and record the running time in the data processing, training, and testing phases. Table 4 shows the time cost of the two schemes. Overall, VoicePop+ runs over 10 times faster than VoicePop. We can see that the data processing overhead of VoicePop+ is slightly higher than VoicePop. This is because VoicePop+ additionally involves preesure estimation in the data processing module. As for the speaker verification part, VoicePop+ not only reduces the workload of the server in the training phase, but also performs the liveness detection faster in the authentication phase, and thus can bring a better user experience.

5.11 Suitability Evaluation for Deployment on Device

Our evaluation so far focuses on the authentication scenarios that involve a client and a remote authentication server. For some special cases that do not require a remote authentication server (e.g., smartphone login), we have further investigated the possibility of implementing VoicePop+ on device, i.e., whether the trained machine learning model can perform the authentication directly on the mobile device. The feasibility test of deploying the authentication module on Android yields promising results. Specifically, running the speaker verification module on the trained machine learning model locally takes about 51s for 100 trials, which suggests that the lightweight of the machine learning models in our VoicePop+ design is indeed suitable to run on the mobile device. The remaining bottleneck, as indicated by our test, is running the data processing module, which can be quite heavy for the mobile devices, i.e., about 382s for 100 trials. We plan to develop the full-fledged offline version of VoicePop+ on mobile devices as our future work.

TABLE 4
Evaluation on Efficiency of VoicePop and VoicePop+

	Size	VoicePop+	VoicePop
Data processing	885	70.6s	67.6s
Training	290	27.3s	383.8s
Testing	595	61.7s	551.2s

6 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 [19], [20], [21]. 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. Kinnunen *et al.* [23] discovered that the Equal Error Rate (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 [34]. Recently, Adobe developed a system VoCo [35] 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. Mukhopadhyay *et al.* [36] synthesized the victim's voice by using the user's speech fragments and the existing speech synthesis tools and achieved an attack rate of more than 80%. Similarly, Sizov *et al.* [37] showed that the attackers could threaten the voice authentication system through synthetic attacks, resulting in a high misjudgment rate of the system. Zhou *et al.* [38] synthesized hidden voice commands to stealthily control the voice controllable system (VCS) of autonomous driving cars. 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 [39]. Bonastre *et al.* [40] used the voice conversion technology to assess Gussian Mixture Model-Universal Background Model (GMM-UBM) and Joint Factor Analysis (JFA) based speaker recognition systems, the results show that the EER under both models increased from 8.5% and 4.8% to 32.6%, and 24.8% respectively. Impersonation attacks are launched by attackers who mimic the voice characteristics and speaking behavior of the victim [21]. Wu *et al.* [22] showed that pure impersonation might produce similar speaking pattern and rate of the victim, but it is nearly impossible for the impersonators to fake the spectral characteristics like formants.

Our VoicePop+ can resist the replay attacks based on stealthily recording, voice synthesis, and voice conversion. VoicePop+ is mainly based on pop noise that is induced by the user breathing while speaking the passphrase close to the microphone. In practice, stealthily recording cannot be too close to the legitimated user, otherwise the user will easily notice the recording. At a relatively long distance (i.e., 30cm), stealthily recording cannot capture effective pop noise, and thus can be detected by VoicePop+. Besides, it is also hard for voice synthesis and voice conversion techniques to forge natural pop noises. Since pop noises are

induced by physical breathing, software-only synthesis cannot forge such subtle user physical characteristics. As for the impersonation attacks, we first find the individually uniqueness of pop noises. Different people have different phonation style and breathing style, and such unique characteristics are hard to observe and imitate. It is also unlikely for the adversary to obtain the characteristics of the pop noises of the target users and forge the pop noises. Thus, the adversary cannot use the computer to program the target user's pop noises, either. Our evaluation shows that it is highly unlikely to imitate the victim's breathing patterns.

Voice Anti-Spoofing.The traditional method of defending against replay attacks is liveness detection [7], [8], [9], [11], [15], [19], which examines whether the voice is produced by a live user or a speaker. The current methods can be summed up in two ways. The first builds the security mechanism based on the characteristics of the human voice production system. The second is mainly to obtain support from extra hardware, such as using additional sensors to collect other signals while acquiring voice signals and analyzing the consistency of the two signals for attack detection.

VoiceLive [7] 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 [8], 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. Furthermore, since it relies on the reflection of the acoustic signals, the relative positions between the user's mouth and the microphone and different positions of the microphones and speakers in the phone will also affect the performance. In [12], a text-independent speaker verification method was proposed to detect spoofing attacks based on loudspeakers, and the key point of this method is to use the acoustic biometrics embedded in the sound field to build a "fieldprint", which can be used to distinguish loudspeakers from real people. 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.

In the research of the second kind of liveness detection method, Feng *et al.* [15] proposed to use the accelerometer to record the vibration signal of the body surface and analyzed the consistency of it and the voice signal to determine the identity of users. Lei *et al.* [41] designed a physical presence based access control for home digital voice assistants (HDVAs), which uses wireless fidelity (WiFi) sensing technology to detect human activities in the room. Only when user activities are detected, hdva equipment can execute voice commands. Similarly, Yan *et al.* [10] used the WiFi signal to detect the oral movement of users when they are speaking and extracted features from voice and WiFi signals. By measuring the correlation between the two signals, it determines whether the command is issued by the real user. However, this method is vulnerable to the impact of the environment. Wang *et al.* [16] proposed to detect the pressure of the oral air flow through the external sensor, and then judge the pressure in the voice signal to detect the attack. The common problem of the above sensor-based detection methods is that they need to use additional devices, and the calculation cost is high, and the scheme is not easy to popularize.

Compared with the anti-spoofing systems that need an extra device [10], [13], [15], [16], VoicePop+ relies only on off-the-shelf smartphones that are equipped with a microphone, which can be readily integrated into existing voice authentication systems on smartphones with no additional hardware modification. Compared with the systems based on mouth motion [2], [8], [9], VoicePop+ is more robust to the position between the user's mouth and the microphone. The systems [2], [8], [9] rely on the refection signals, which are sensitive to the positions of the speaker, microphone, and the mouth. VoicePop+ relies on the pop noises and the airflow pressure, which can be detected in different speaking positions within 12cm. Other systems based on sound field [12] and throat vibration [14] do not explicitly resist impersonation attack. Thus, whether the distinctiveness have individual uniqueness remains to explore, which is an interesting and promising future research direction. We conclude that pop noise is a robust voice biometrics that does not rely on extra device, and it can be used for individually classification to resist impersonation attacks.

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.* [17] 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 [18]. However, their studies rely on the specific microphone model and cannot perform well when 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. Compared to our previous work [1], we further leverage the pressure signal of the oral airflow to perform consistency analysis to improve the efficiency and robustness of VoicePop.

7 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 also leveraged the sound pressure level to get the estimated pressure signal and compare it with the actual pressure signal extracted from the pop noise to resist replay attacks. We also used the individually unique relationship between phonemes and pop noises to detect impersonation spoofing attacks. Extensive experiments confirmed that VoicePop+ is robust in resisting various types of voice spoofing attacks with different smartphones under diversified environments with an average detection accuracy of 94.79%. VoicePop+ can be readily

integrated into existing voice authentication systems on smartphones with no additional hardware modification. We believe VoicePop+ has a promising future application.

References

[1] Q. Wang *et al.*, "VoicePop: A pop noise based anti-spoofing system for voice authentication on smartphones," in *Proc. IEEE Conf. Comput. Commun.*, 2019, pp. 2062–2070.

[2] L. Wu, J. Yang, M. Zhou, Y. Chen, and Q. Wang, "LVID: A multimodal biometrics authentication system on smartphones," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1572–1585, 2020.

[3] M. Zhou *et al.*, "Securing face liveness detection using unforgeable lip motion patterns," 2021, *arXiv:2106.08013*.

[4] Voiceprint: The New WeChat Password, 2015. [Online]. Available: https://blog.wechat.com/tag/voiceprint/

[5] How to Unlock Your Phone with Google Assistant, 2019. [Online]. Available: https://www.techadvisor.com/how-to/google-android/unlock-phone-google-assistant-3689107/

[6] Say-Tec Build a Better B2B and B2C Experience, 2020. [Online]. Available: https://www.finnovant.com/say-tec

[7] L. Zhang, S. Tan, J. Yang, and Y. Chen, "VoiceLive: A phoneme localization based liveness detection for voice authentication on smartphones," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 1080–1091.

[8] L. Zhang, S. Tan, and J. Yang, "Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2017, pp. 57–71.

[9] L. Lu *et al.*, "LipPass: Lip reading-based user authentication on smartphones leveraging acoustic signals," in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 1466–1474.

[10] M. Yan, Z. Wang, Z. Wei, P. Wu, and L. Yao, "WiVo: Enhancing the security of voice control system via wireless signal in IoT environment," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2018, pp. 81–90.

[11] S. Chen *et al.*, "You can hear but you cannot steal: Defending against voice impersonation attacks on smartphones," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst.*, 2017, pp. 183–195.

[12] C. Yan, Y. Long, X. Ji, and W. Xu, "The catcher in the field: A fieldprint based spoofing detection for text-independent speaker verification," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2019, pp. 1215–1229.

[13] M. Sahidullah *et al.*, "Robust voice liveness detection and speaker verification using throat microphones," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 1, pp. 44–56, Jan. 2018.

[14] J. Shang, S. Chen, and J. Wu, "Defending against voice spoofing: A robust software-based liveness detection system," in *Proc. IEEE 15th Int. Conf. Mobile Ad Hoc Sensor Syst.*, 2018, pp. 28–36.

[15] H. Feng, K. Fawaz, and K. G. Shin, "Continuous authentication for voice assistants," in *Proc. 23rd Annu. Int. Conf. Mobile Comput. Netw.*, 2017, pp. 343–355.

[16] Y. Wang, W. Cai, T. Gu, W. Shao, Y. Li, and Y. Yu, "Secure your voice: An oral airflow-based continuous liveness detection for voice assistants," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, 2019, pp. 1–28.

[17] 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. Odyssey*, 2016, pp. 259–263.

[18] S. Mochizuki, S. Shiota, and H. Kiya, "Voice livness detection based on pop-noise detector with phoneme information for speaker verification," *J. Acoust. Soc. Amer.*, vol. 140, no. 4, pp. 3060–3060, 2016.

[19] Z. F. Wang, G. Wei, and Q. H. He, "Channel pattern noise based playback attack detection algorithm for speaker recognition," in *Proc. IEEE Int. Conf. Mach. Learn. Cybern.*, 2011, pp. 1708–1713.

[20] 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 Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 8, pp. 2280–2290, Oct. 2012.

[21] 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. Conf. Int. Speech Commun. Assoc.*, 2013, pp. 930–934.

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

[23] T. Kinnunen *et al.*, "The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," in *Proc. Conf. Int. Speech Commun. Assoc.*, 2017, pp. 2–6.

[24] M. Shirvanian and N. Saxena, "Wiretapping via mimicry: Short voice imitation man-in-the-middle attacks on crypto phones," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2014, pp. 868–879.

[25] S. Koga, S. Makihara, and Y. Yamanouchi, "Score normalization in playback attack detection," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2010, pp. 1678–1681.

[26] A. Jongman, "Acoustics of american english speech: A dynamic approach," *Lang. Speech*, vol. 38, no. 1, pp. 115–118, 1995.

[27] T. Kisler, F. Schiel, and H. Sloetjes, "Signal processing via web services: The use case WebMAUS," in *Proc. Digit. Humanities Conf.*, 2012, pp. 30–34.

[28] J. A. Gierut, "Phonological complexity and language learnability," *Amer. J. Speech Lang. Pathol.*, vol. 16, pp. 6–17, 2007.

[29] S. Parker, "Quantifying the sonority hierarchy," Ph.D. dissertation, Dept. Linguistics, Univ. Massachusetts Amherst, Amherst, MA, USA, 2002.

[30] C. M. Tey, P. Gupta, and D. Gao, "I can be you: Questioning the use of keystroke dynamics as biometrics," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2013.

[31] M. Zhou, Q. Wang, T. Lei, Z. Wang, and K. Ren, "Enabling online robust barcode-based visible light communication with realtime feedback," *IEEE Trans. Wireless Commun.*, vol. 17, no. 12, pp. 8063–8076, Dec. 2018.

[32] M. Zhou, Q. Wang, K. Ren, D. Koutsonikolas, L. Su, and Y. Chen, "Dolphin: Real-time hidden acoustic signal capture with smartphones," *IEEE Trans. Mobile Comput.*, vol. 18, no. 3, pp. 560–573, Mar. 2019.

[33] M. Zhou *et al.*, "Stealing your android patterns via acoustic signals," *IEEE Trans. Mobile Comput.*, vol. 20, no. 4, pp. 1656–1671, Apr. 2021.

[34] R. Valle, J. Li, R. Prenger, and B. Catanzaro, "Mellotron: Multispeaker expressive voice synthesis by conditioning on rhythm, pitch and global style tokens," in *Proc. IEEE Int. Conf. Acoust.*, 2020, pp. 6189–6193.

[35] Adobe VoCo 'Photoshop-for-voice' causes concern, 2016. [Online]. Available: http://www.bbc.com/news/technology-37899902

[36] D. Mukhopadhyay, M. Shirvanian, and N. Saxena, "All your voices are belong to us: Stealing voices to fool humans and machines," in *Proc. 27th Eur. Symp. Res. Comput. Secur.*, 2015, pp. 599–621.

[37] A. Sizov, E. Khoury, T. Kinnunen, Z. Wu, and S. Marcel, "Joint speaker verification and anti-spoofing in the *i*-vector space," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 4, pp. 821–832, Apr. 2015.

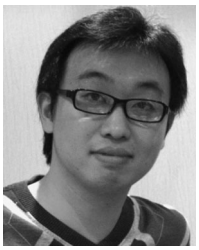
[38] M. Zhou, Z. Qin, X. Lin, S. Hu, Q. Wang, and K. Ren, "Hidden voice commands: Attacks and defenses on the VCS of autonomous driving cars," *IEEE Wireless Commun. Mag.*, vol. 26, no. 5, pp. 128–133, Oct. 2019.

[39] 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 Trans. Audio, Speech, Language Process.*, vol. 25, no. 11, pp. 2071–2084, Nov. 2017.

[40] J. Bonastre, D. Matrouf, and C. Fredouille, "Artificial impostor voice transformation effects on false acceptance rates," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2007, pp. 2053–2056.

[41] X. Lei, G.-H. Tu, A. X. Liu, C.-Y. Li, and T. Xie, "The insecurity of home digital voice assistants-vulnerabilities, attacks and countermeasures," in *Proc. IEEE Conf. Commun. Netw. Secur.*, 2018, pp. 1–9.

Peipei Jiang received the BE degree in information security from Wuhan University, China, in 2019. She is currently working toward the PhD degree with the School of Cyber Science and Engineering, Wuhan University, China. She is also a joint PhD student with the City University of Hong Kong, Hong Kong, China. Her research interests include network security and AI security.



Qian Wang (Senior Member, IEEE) is a professor with the School of Cyber Science and Engineering, Wuhan University, China. He received the National Science Fund for Excellent Young Scholars of China in 2018. He is also an Expert of the National 1000 Young Talents Program of China. His research interests include AI security, data storage, search and computation outsourcing security and privacy, wireless systems security, Big Data security and privacy, and applied cryptography. He was a recipient of the 2016 IEEE Asia-Pacific Outstanding Young Researcher Award and the 2018 IEEE TCSC Award for Excellence in Scalable Computing (early career researcher). He is also a co-recipient of several Best Paper and best student paper awards from ICICS'21, IEEE DSC'19, IEEE ICDCS'17, IEEE TrustCom'16, WAIM'14, and IEEE ICNP'11. He serves as associate editors for *IEEE Transactions on Dependable and Secure Computing (TDSC)*, *IEEE Transactions on Information Forensics and Security (TIFS)*, and *IEEE Internet of Things Journal (IoT-J)*. He is a senior member of the ACM.

Xiu Lin received the BE degree in information security, in 2017, and the MS degree in cyberspace security, in 2020, from Wuhan University, China. Her research interests include mobile computing and mobile security. She was the recipient of the first prize in the "National College Student Information Security Contest, China" in 2016.

Man Zhou received the PhD degree in cyberspace security, in 2021, and the BE degree in information security, in 2016, from Wuhan University, China. He is currently an associate professor with the School of cyber science and engineering, Huazhong University of Science and Technology. His research interests include AI system security, mobile security, and IoT security. He was the recipient of "National scholarship for graduate students, China" in 2016–2018 and 2020.

Wenbing Ding received the BE degree in the information security from Wuhan University, China, in 2021. He is currently working toward the master's degree with the School of Cyber Science and Engineering, Wuhan University, China. His research interests include network security and IoT security.

Cong Wang (Fellow, IEEE) is a professor with the Department of Computer Science, City University of Hong Kong. His research interests include data and network security, blockchain and decentralized applications, and privacy-enhancing technologies. He has been one of the founding members of the Young Academy of Sciences of Hong Kong since 2017, and has been conferred the RGC research fellow in 2021. He received the Outstanding Researcher Award (junior faculty) in 2019, the Outstanding Supervisor Award in 2017 and the President's awards in 2019 and 2016, all from the City University of Hong Kong. He is a co-recipient of the Best Paper Award of IEEE ICDCS'20, ICPADS'18, MSN'15, Best Student Paper Award of IEEE ICDCS'17, and the IEEE INFOCOM Test of Time Paper Award 2020. His research has been supported by multiple government research fund agencies, including National Natural Science Foundation of China, Hong Kong Research Grants Council, and Hong Kong Innovation and Technology Commission. He has served as associate editors for *IEEE Transactions on Dependable and Secure Computing (TDSC)*, *IEEE Transactions on Services Computing (TSC)*, *IEEE Internet of Things Journal (IoT-J)*, *IEEE Networking Letters*, and *Journal of Blockchain Research*, and TPC co-chairs for a number of IEEE conferences and workshops. He is a member of the ACM.

Chao Shen (Senior Member,IEEE) is currently a professor in the school of electronic and information engineering, Xian Jiaotong University of China. He serves as the associate dean with the School of Cyber Security, Xian Jiaotong University. He is also with the Ministriy of Education Key Lab for Intelligent Networks and Network Security. He was a research scholar in Carnegie Mellon University from 2011 to 2013. His research interests include network security, human computer interaction, insider detection, and behavioral biometrics.

Qi Li (Senior Member, IEEE) received the PhD degree from Tsinghua University. He is currently an associate professor with the Institute for Network Sciences and Cyberspace, Tsinghua University. His research interests are in network and system security, particularly in Internet and cloud security, mobile security, and Big Data security. He is currently an editorial board member of *IEEE Transactions on Dependable and Secure Computing* and *ACM Diabetes Research and Clinical Practice*.