6-2023

# LiVoAuth: Liveness Detection in Voiceprint Authentication with random challenges and detection modes

Rui ZHANG

Zheng YAN

Xueru WANG

Robert H. DENG
*Singapore Management University*, robertdeng@smu.edu.sg



## Citation

ZHANG, Rui; YAN, Zheng; WANG, Xueru; and DENG, Robert H.. LiVoAuth: Liveness Detection in Voiceprint Authentication with random challenges and detection modes. (2023). *IEEE Transactions on Industrial Informatics*. 19, (6), 7676-7688.
**Available at:** https://ink.library.smu.edu.sg/sis_research/8180

# LiVoAuth: Liveness Detection in Voiceprint Authentication with Random Challenges and Detection Modes

Rui Zhang, Zheng Yan *Senior Member, IEEE*, Xuerui Wang, and Robert H. Deng., *Fellow, IEEE*

***Abstract*— Voiceprint authentication provides great convenience to users in many application scenarios. However, it easily suffers from spoofing attacks including speech synthesis, speech conversion and speech replay. Liveness detection is an effective way to resist these attacks. But existing methods suffer from many disadvantages, such as extra deployment costs due to precise data collection, environmental disturbance, high computational overhead and operational complexity. A uniform platform that can offer Voiceprint Authentication as a Service (VAaS) over the cloud is also lacked. Hence, it is imperative to design an economic and effective method for liveness detection in voiceprint authentication. In this paper, we propose a novel liveness detection method named LiVoAuth for voiceprint authentication. It applies a randomly generated vector sequence as Liveness Detection Mode (LDM), corresponding to a random challenge code used for authentication. We implement LiVoAuth and conduct a series of user studies to evaluate its performance in terms of accuracy, stability, efficiency, security, and user acceptance. Experimental results demonstrate its advantages compared with cutting-edge methods.**

***Index Terms*— Identity Authentication, Voiceprint Recognition, Spoofing Attack, Liveness Detection**

## I. Introduction

VOICEPRINT authentication is a technology of user authentication through voiceprint recognition. Voiceprint recognition is the process of identifying a speaker according to personality characteristics contained in his/her speech. It plays an important role for securing many industrial Internet of Things applications, e.g., smart home, Internet of Vehicles, remote voice control, eHealth, etc. Compared with traditional username password authentication methods, voiceprint authentication not only releases users from managing and memorizing complex account information [1], but also avoids the security risk caused by simple password usage [2]. Compared with other types of biometrics [3]–[7], the collection of voiceprint feature data is much simpler, by using a very common microphone without a strict requirement on user distance to his/her equipment. Among many biometric authentication technologies, voiceprint authentication has gradually become one of the mainstream technologies. It shows great practicability and convenience in some special scenes when a user's hands are occupied, e.g., during driving and cooking, and when face features cannot be captured due to face mask wearing. Nowadays, voiceprint authentication has been widely used in daily life, such as voice banking, voice navigation, voice unlocking, and voice shopping.

However, because voice is easy to be copied, forged or imitated, voiceprint authentication is often threatened by spoofing attacks. Speech synthesis, speech conversion and speech replay are three common attacking methods of the spoofing attacks [8]. How to distinguish a natural person's speech signal from synthesized, imitated or replayed speech signals (i.e., liveness detection) is the key to resist the spoofing attacks. Liveness detection determines whether a biometric signal comes from a living human-being.

There are many liveness detection methods for voiceprint authentication, which can be divided into two categories: the methods based on voiceprint features and the methods based on the impact of a user on his/her surrounding environment. The first type of methods has low deployment cost. However, the robustness of this kind of schemes is slightly weak, vulnerable to interference. For example, VoicePop [4] uses pop noise to distinguish real human voice from audio records. However, the analysis shows that this speech feature is easily disturbed by external factors, such as the distance between a user and a microphone, the difference of various microphones, the existence of silencing equipment, and the severity of environmental noise. VoiceLive [9] uses time-difference-of-arrival to model phoneme locations for liveness detection. However, this type of methods usually introduces additional computational overhead and suffers from such a limitation that the number of microphones greatly impact its performance. The second type of liveness detection methods normally depends on additional data acquisition equipment and has a strict requirement on user position. For example, CaField [10] detects the change of fieldprint caused by sound energy, and the method proposed in [11] detects the influence of a user on a wireless signal field to check liveness. These methods require multiple additional high-precision sensors, and the relative positions between a user and the sensors greatly influence the accuracy of liveness detection. Obviously, the above liveness detection methods suffer from low robustness, high computational cost, easy disturbance, high deployment

This work was supported in part by the National Natural Science Foundation of China under Grant 62072351; in part by the Academy of Finland under Grant 308087, Grant 335262, Grant 345072, and Grant 350464; in part by the Open Project of Zhejiang Lab under Grant 2021PD0AB01; and in part by the 111 Project under Grant B16037. (Corresponding author: Zheng Yan.)

Rui Zhang, Zheng Yan and Xuerui Wang are with the State Key Lab of ISN, School of Cyber Engineering, Xidian University, Xi'an 710071, China, e-mail: (zyan@xidian.edu.cn).

Zheng Yan is also with the Department of Communications and Networking, Aalto University, Espoo 02150, Finland.

Robert H. Deng is with the School of Information Systems, Singapore Management University, e-mail: (robertdeng@smu.edu.sg).

cost, or poor usability.

Through investigation, we realize that a series of problems still remain in liveness detection of voiceprint authentication. First, some voiceprint authentication systems sacrifice usability for the sake of gaining high security. Second, some liveness detection methods require accurate biological signal acquisition with highly-precise signal acquisition equipment, so the cost of detection is high. Third, current methods often ignore the effects of various external uncertain factors on the performance of liveness detection and authentication. In addition, due to wide usage of voiceprint authentication in various fields, a general authentication framework is highly expected to provide user Voiceprint Authentication as a Service (VAaS) over the cloud, which could greatly benefit usability due to familiarity and using habit. Therefore, an economic and effective liveness detection method is highly expected, which is preferred to offer VAaS.

To tackle the above problems, in this paper, we propose a novel liveness detection method named LiVoAuth for voiceprint authentication without extra data collection, expensive equipment dependence and complicated operation. It applies a randomly generated vector sequence as Liveness Detection Mode (LDM), corresponding to a random challenge code used for authentication. Based on the user response content spoken in the mode as specified in LDM (e.g., word speaking volume and pause time after speaking a word), user authentication and liveness detection can be performed simultaneously. LiVoAuth can confirm the liveness of a user, which is difficult to imitate. At the same time, LiVoAuth does not make the original authentication process complex, which has no much negatively impact on usability. The computational overhead of LiVoAuth is also low, which ensures its operation efficiency.

We implement a prototype system of LiVoAuth. It consists of three entities: User Agent (UA), User Authentication Provider (UAP) and Relying Party (RP). During user voiceprint authentication, the UAP randomly sends his/her a voice challenge code together with a LDM randomly generated corresponding to the voice challenge code. The user needs to repeat the challenge code according to the requirements specified by the LDM. The UAP judges whether the current speaker is living by verifying if he/she speaks in a way as specified in the LDM and authenticates him/her by checking if he/she correctly repeats the content of the challenge code with a valid voiceprint. We carry out a series of user studies and experiments based on the prototype system. The results show that LiVoAuth has high authentication accuracy, sound liveness detection accuracy, high stability and good efficiency. It is robust to resist spoofing attacks and gains positive feedback on user acceptance. Compared with several cutting-edge related methods, its performance is advanced, especially under additional attacks on liveness detection. Specifically, the contributions of this paper can be summarized as below:

- We propose LiVoAuth, a novel liveness detection method for voiceprint authentication by using a randomly generated voice challenge code with LDM that is also randomly generated accordingly to the voice challenge code.
- We propose LDM as an additional requirement to limit the voice volume, pause time and other factors of user speech when the user responds to a challenge, so as to achieve the goal of liveness detection.
- We implement LiVoAuth to offer a common cloud-based voiceprint authentication service with effective liveness detection.
- We conduct a series of user studies and experiments to evaluate LiVoAuth's performance in terms of accuracy, efficiency, stability, security and user acceptance. The results show LiVoAuth's excellence.
- We compare LiVoAuth with several cutting-edge liveness detection methods to exhibit its advantages.

The structure of the remainder of the paper is as follows. Section II introduces technical background and related work. Section III describes the system model, security model and threat model of LiVoAuth. Section IV describe the design of LiVoAuth, which is an economic and effective liveness detection method for VAaS. User study and experimental results are reported in Section V. Our conclusion is drawn in the last section.

## II. Background and Related Work

This section introduces existing voiceprint recognition methods and reviews related work of liveness detection.

### A. Voiceprint Recognition

I-Vector and X-Vector are widely used voiceprint recognition techniques. I-Vector is an excellent framework in text-independent voiceprint recognition [12], where the contents of registered and authenticated voices are quite different. Scholars have done a lot of optimizations based on I-Vector, including Linear Discriminant Analysis (LDA), Linear Predictive Discriminant analysis (PLDA), Metric Learning, and so on [3]. X-Vector is based on Deep Neural Network (DNN) [9], [13], which can abstract voice features from a large number of samples. DNN can accurately obtain a user's voiceprint information by using a speech with about 10 seconds and can strongly resist noise interference with high robustness. Therefore, X-Vector is suitable for voiceprint feature extraction of short-term speech, while I-Vector has better performance in case that only a small number of speech training samples and long-term speeches are available [14]–[16]. In our prototype, we applied X-Vector for voiceprint recognition since user provided speeches for authentication are short-term.

### B. Liveness Detection Methods

Liveness detection methods can be divided into two categories: the ones based on voiceprint features in user original voices and the ones based on user impact on surrounding environment.

*1) Detection Based on User Voiceprint Features:* Wang et al. proposed VoicePop to identify legitimate users and defend against spoofing attacks through pop noise detection [4]. Pop noise is a kind of distortion of speech that is generated when a human is close to a microphone during speaking. VoicePop detects pop noise for liveness detection without using any additional equipment, nor does it need users to carry out additional operations. This method is low-cost and simple to use. Unfortunately, its detection success rate is slightly lower than that of other methods. It only achieves over 93.5% detection accuracy with around 5.4% Equal Error Rate (EER). on the other hand, researches showed that the pop noise can be reproduced when a loudspeaker plays the same speech

[17]. Thus, its robustness is suspicious under some deliberate interference.

Zhang et al. [9] presented a liveness detection system named VoiceLive. They noticed that each phoneme is uniquely located in the human vocal channel system. Microphones can capture the Time Difference of Arrival (TDoA) of each phoneme, which does not exist under replay attacks. VoiceLive restricts the number and locations of microphones, which, fortunately, does not cause too much trouble to users. VoiceLive is compatible to different mobile phone models, thus it does not require additional equipment to deploy, smartphones with two-channel stereo recording work for VoiceLive. Experimental results show that the detection accuracy of VoiceLive is over 99% and its EER is 1%. Its detection accuracy is high with good usability. But its computational complexity is pretty high.

VOLERE [18] is a privacy-preserving voiceprint authentication system that uses synthesized voiceprint for user authentication in order to prevent voice privacy leakage. The synthesized voiceprint is generated based on the user's voices spoken in different speaking modes, so it is difficult to be forged. In particular, dynamically challenging users to speak some words makes VOLERE have some ability of liveness detection. Unfortunately, the accuracy of VOLERE is not so satisfied. Its ability to resist replay attacks is not fully tested.

Rahman et al. [19] proposed Movee that combines video and accelerometer data to verify liveness. Movee estimates whether motion features in video streams are consistent with the features extracted from an accelerometer sensor stream. It may be a feasible method with the development of 5G network and streaming media technology due to sound network transmission capacity. Movee's accuracy falls into the range of 68-93%, which is not high. Its authentication time takes 6 seconds, which obviously affects its usability. Since its authentication needs user cooperation, learnability and user preference are impacted somehow. In addition, Movee has difficulty to extract information from blur video with low illumination. And its computation complexity is high.

In [20], a novel system called Voice Gesture was proposed for detecting replay attacks. A user's pronunciation gesture may cause Doppler effect. The Doppler shift produced by human speech is larger than that produced by a loudspeaker. The detection accuracy of Voice Gesture is over 99% and its EER is 1%. It does not need any complex operation and extra equipment, which implies high usability for ensuring security. In addition, the time spent for authentication is about 0.5 second, which implies high efficiency. But it has such a disadvantage that users need to hold their phones with their hands when they are using Voice Gesture, which impacts usage convenience and somehow destroys the advantages of voiceprint authentication.

*2) Detection Based on User Impact on Surrounding Environment:* Yan et al. [10] proposed CaField that makes use of a fieldprint to perform liveness detection. The fieldprint is a fingerprint of acoustic energy propagation in the air. The fieldprint of a human speaker is normally different from that of a loudspeaker, which can be employed to detect spoofing attacks. Meng et al. [11] found that different mouth shapes have different shielding effects on a signal [11]. Therefore, when a user speaks between two wireless signal transceivers, the fluctuation of wireless signal can be used as evidence of a living user for liveness detection. However, these two methods need to deploy additional sensors to collect field data. It has strict restriction on the relative position between the user and the sensors. Obviously, their usability is normally not good and their deployment cost is high.

## III. Problem Statement

### A. System Model

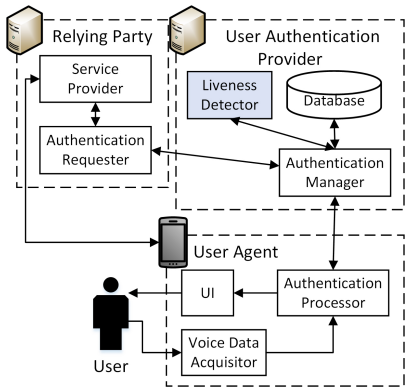



Fig. 1. LiVoAuth system model.

LiVoAuth consists of three types of entities: User Agent (UA), Relying Party (RP), and User Authentication Provider (UAP), as shown in Figure 1. UA is a user-owned personal device, which includes a UI that interacts with the user, a voice data acquisitor that collects user voices and an authentication processor for communication with UAP. RP contains an authentication requester and a service provider. The authentication requester is responsible for requesting UAP for user registration and authentication. When user authentication is passed, the service provider of RP provides services to the user. Multiple RPs could exist to offer various online services based on user voiceprint authentication. UAP provides a cloud-based authentication service to different RPs and their corresponding users, and stores users' voiceprint information. UAP consists of three components: authentication manager that performs voiceprint authentication; database that stores user identities and voiceprint information; liveness detector that is deployed to detect user liveness by generating and verifying LDMs to assist the authentication manager to provide an authentication result.

### B. Security Model

In our study, we put forward the following assumptions to establish the security model of LiVoAuth:

- We assume that UA is trusted by its user. It works as designed. The data stored and processed in UA cannot be stolen or distorted.
- The data entered into UA cannot be ensured as trusted. An attacker may inject interference when the user inputs verification voice. The attacker may also raise replay attacks and use other methods to disguise a legal user via his/her UA.
- We assume that before uploading user voiceprint data, UA adopts a certain manner like encoding and encryption to ensure that the data cannot be leaked, e.g., synthesized voiceprint generated based on different speaking modes can be adopted to avoid voiceprint information leakage [18].
- We assume that UAP works as designed. It cannot be fully trusted. It may be maliciously deceived by an

attacker and may be attacked to provide wrong authentication results. Herein, we assume that the integrity of UAP database is ensured by applying some advanced technology for data integrity assurance [21].

- We assume that RP is trusted. It provides its service to a user once he/she can pass the authentication offered by UAP. In addition, due to profit conflict and business difference, UAP and RP do not collude.
- We assume that the communication channels among UA, RP and UAP are secure with mutual authentication due to the employment of security protocols [22].

### C. Threat Model

We focus on preventing spoofing attacks in voiceprint authentication, including replay attacks and artificial synthesis attacks (e.g., conversion and impersonation attacks). We assume that an attacker knows the detailed algorithms of LiVoAuth. The attacker can obtain voiceprint information through voice recording, synthesis and re-editing. In addition, the attacker may obtain the voiceprint information stored in the UAP database by means of SQL injection, for example.

According to the above assumptions, the system may encounter the following attack scenarios:

1) When the attacker is familiar with a target user's timbre and voice habits through observation and learning, he can directly imitate the user's voice for user authentication, so as to obtain the access rights of RP services.
2) When the attacker obtains the voice of a target user that was successfully authenticated before, if the challenge code of authentication is consistent with the acquired audio content, he can directly execute a replay attack, so as to obtain the access rights of RP services.
3) When the attacker obtains the voice of a target user who has been successfully authenticated before, if the challenge code of authentication is inconsistent with the acquired audio content, he can re-edit and split the audio, synthesize the challenge code by following LDM, and then perform a replay attack to imitate the user, so as to obtain the access rights of RP services.
4) When the attacker obtains the voiceprint template information stored in the UAP database, he may use such information to perform replay attacks in order to gain access to the RP services.

## IV. LiVoAuth Design

This section describes the details of LiVoAuth design. Its security analysis is provided in Appendix I.

### A. Liveness Detection Method

LiVoAuth refers to a text-independent voiceprint user authentication system with liveness detection. Its user does not have to use a same password for authenticate every time. In each authentication process, a textual Challenge Code (CC) is randomly generated. Refer to Figure 2, corresponding to the CC used for voiceprint authentication, we randomly generate a vector sequence, called LDM for the purpose of liveness detection. Both CC and LDM are generated to challenge the user for authentication and liveness detection at the same time in order to double prevent spoofing attacks, such as replay attacks and impersonation attacks. Note that although a voiceprint recognition algorithm does not rely on audio content, UA still needs to upload voice files containing textual information. Because UAP needs to confirm whether the user's voice response meets CC regarding speaking content and LDM regarding speaking mode. For detecting LDM, the short-term energy and the short-term zero crossing rate of speech are calculated to determine voice volume and pause time and see if they can match with LDM. For matching with CC, a speech recognition toolkit [23] is applied to check if the user speaks the same content as CC. For matching a voiceprint sample $VP_{te}$ with a target one $VP_{ta}$, features of $VP_{te}$ are extracted and input into the neural network of X-vector to gain a feature vector and compare it with that of $VP_{ta}$.

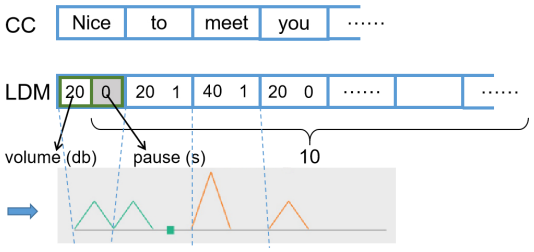



Fig. 2. An example of challenge code and LDM structure.

Each vector in the LDM corresponds to each word in the CC in order. Each vector contains several pieces of information about speaking mode for liveness detection. Taking the LDM in Figure 2 as an example, the vector in LDM contains two elements, i.e., we set two modes of word speaking: volume and pause time, for liveness detection. For example, the volume of the first word "Nice" is 20dB and its pause time is 0 second. Note that the above ranges can be randomly set. Considering that exaggerated volume, improper pause setting, and too long CC may have a negative impact on user experience, we set the range of volume to 10 40 dB with a step of 5 dB; set the range of pause time to 0 2 seconds with a step of 0.5 seconds. We also limit the length of CC to 10 words for gaining sound user experience based on several rounds of user tests. We collect a batch of daily expressions with the satisfied length to form a challenge code warehouse. LiVoAuth randomly selects one from the warehouse each time as the CC and randomly generates an LDM with the same length of the CC. In order to facilitate the user to obtain LDM information, a graphic prompt is shown to the user, as shown in Figure 2. During authentication, the user is expected to repeat the challenge code by following the speaking mode indicated by the LDM, e.g., speak each word of the CC with the indicated volume and pause time in the LDM.

Next, UA sends the recorded user voice to UAP for authentication. It should be noted that the pause in the speech is used for liveness detection, which should be preserved by UA. But UA needs to detect and delete muted voice at the beginning and the end of a voice file during preprocessing. After UAP receives the voice file and finishes LDM detection, it deletes the pause part in the voice for later processing for the purpose of improving subsequent processing efficiency.

To detect the volume of voice $VP$, the following method is applied. We first get a frame signal $S$ after preprocessing $VP$. If the length of the frame is $k$, the volume of the frame is:

$$volume = 10 \lg \left( \sum_{i=1}^{k} S_i^2 \right) \tag{1}$$

The pause time of a speech can also be judged based on voice volume. But a common method is to measure short-term energy or short-term zero crossing rate. We use a right-angle window $h(n)$ for pretreatment:

$$h(n) = \begin{cases} 1, & 0 \le n \le N-1 \\ 0, & \text{other} \end{cases} \quad (2)$$

Then, the short-time average energy of the $n$th frame signal $S$, denoted $E_n$, is:

$$E_n = \sum_{m=n-N+1}^{n} [S(m)\omega(n-m)]^2 \quad (3)$$

The short-time zero crossing rate of the $n$th frame signal $S$, denoted $Z_n$, is:

$$Z_n = \sum_{m=n-N+1}^{n} |sym[S(m)] - sym[S(m-1)]|\omega(n-m), \quad (4)$$

where $sym[]$ is a symbolic function,

$$sym[S(n)] = \begin{cases} 1, & S(n) \ge 0 \\ -1, & S(n) < 0 \end{cases} \quad (5)$$

$$\omega(n) = \begin{cases} 1/(2N), & 0 \le n \le N-1 \\ 0, & \text{other} \end{cases} \quad (6)$$

In a recording, the short-term energy and the short-term zero crossing rate after the beginning of speech are significantly greater than those in silence. Due to the interference of environmental noise and unvoiced phonemes, the short-term energy and the short-term zero crossing rate in silence are not completely equal to zero. Therefore, it is necessary to select an appropriate threshold value for each of them. When either of their values is greater than the corresponding threshold, it can be recognized that a speech has begun. The pause duration can be calculated by determining the beginning point and the end point of a voice.

We notice that a computing machine can easily imitate a specified volume and pause time, attackers may also launch an attacking voice by applying a speech synthesis algorithm [24], [25]. In fact, LDM can be changed according to practical requirements to adapt to multiple application scenarios. For example, we can modify the setting of LDM and add other elements, such as speaking speed, into it. Subsequent tests show that changing speaking speed can well resist imitation attacks since it is difficult for attackers to pass liveness detection by adjusting audio playback speed, because the voiceprint characteristics of the played audio with speaking speed adjustment are greatly changed. On the other hand, existing speech synthesis algorithms usually have high computation overhead. The quality of generated speeches is not good enough. Thus, it is difficult to launch an attack based on this method. A series of tests are conducted in Section V to verify this issue.

## B. User Authentication Based on LiVoAuth

*1) User Registration:* Figure 3 shows the procedure of user registration in LiVoAuth.

1) Registration request. If a user wants to access RP, the UA of the user raises a Personal Registration Command (PRC) and sends the PRC and UA address (UA_add) to RP;
2) Registration forward. When RP receives the PRC, it packages it with its own ID (RP_id) and send them to UAP;

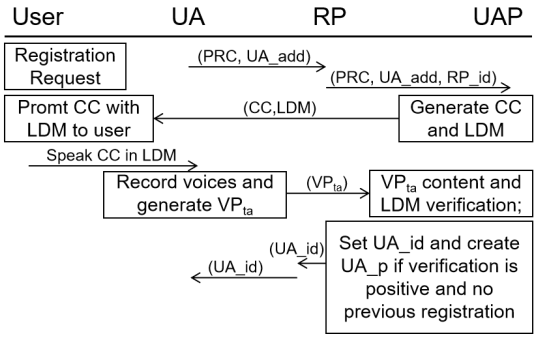



Fig. 3. The procedure of user registration.

3) Registration challenge. After receiving the request information, UAP randomly selects a CC from dataset **A** and creates its LDM randomly and sends (CC, LDM) to UA if it registers for the first time;
4) Target voiceprint generation. UA guides the user to repeat the CC by following the speaking mode specified in the LDM and collects the user's voices to complete feature extraction and generate target voiceprint features $VP_{ta}$. Then UA transfers $VP_{ta}$ to UAP;
5) User profile creation. When UAP receives $VP_{ta}$ sent by UA. UAP checks if the user correctly repeats the contents of CC as challenged by following the mode specified in the LDM. If the verification is positive and the user has not previously registered into UAP (i.e., no record in its database matched with $VP_{ta}$), UAP creates a user profile (UA_p) that contains a newly created unique user ID (UA_id) linked to RP_id, and makes the user's $VP_{ta}$ as a target template. Then, UAP sends UA_id to RP and UA. Note that, if UAP finds that the user has registered already or the registration suffers from some problems, it sets UA_id as null.
6) Registration notification. If UA_id is not null, RP treats registration successful and keeps UA_id. Otherwise, the registration fails.

*2) User Authentication:* Figure 4 illustrates the procedure of user authentication based on LiVoAuth.

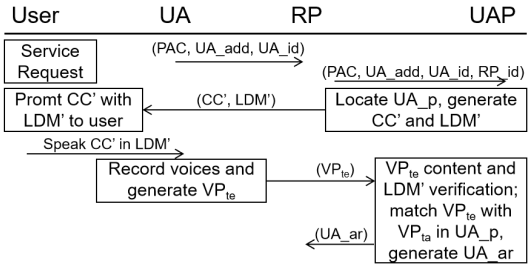



Fig. 4. The procedure of user authentication.

1) Authentication request. If the user wants to access a RP service, he/she initiates UA to send a Personal Authentication Command (PAC), UA_add and UA_id to RP.
2) Authentication forward. RP forwards PAC to UAP together with UA_add, UA_id and RP_id.
3) Authentication challenge. When UAP receives the above authentication request, it first randomly selects a challenge code ($\mathrm{CC}'$) from dataset **B** and randomly generates its $\mathrm{LDM}'$, then sends them to UA;

4) Test voiceprint generation. The user speaks the CC accordingly by following the mode specified in the LDM$'$. UA collects user voices to generate $VP_{te}$ and send it to UAP;
5) User verification. UAP checks if the user correctly repeats the contents of CC$'$ as challenged with the same speaking mode as specified in the LDM$'$, and verifies whether $VP_{te}$ can match with $VP_{ta}$. Then an authentication result (UA_ar) is generated and sent to RP.
6) Service provision. If UA_ar is positive, RP allows the user to access its service. Otherwise, RP rejects service access.

*3) Registration Update and Account Deletion:* Registration update can be performed by combining the authentication procedure and the registration procedure. After the user passes authentication, UAP allows him/her to register again and replaces the old $VP_{ta}$ with a new one. Account deletion can be performed by firstly verifying user legitimacy. After the user passes authentication, UAP can delete the user profile linked to a requesting RP.

## V. User Studies and Experimental Results

We implemented a prototype system of LiVoAuth and tested its performance in terms of authentication accuracy, stability, efficiency, security, and user acceptance. We adopted the following evaluation metrics:

1) False Negative Rate (FNR) indicates the probability that the system wrongly rejects a legitimate user.
2) False Positive Rate (FPR) is the probability that the system mistakenly accepts an attacker or an illegal user as a legitimate user.
3) Equal Error Rate (EER) is defined as the rate when FNR equals FPR.
4) Accuracy (AA) is the probability that the system correctly accepts an eligible user or rejects an illegal user. The sum of accuracy and EER equals to 1.
5) Stability (St) indicates the accuracy over time. With time flying, the system should still work stablely and effectively.
6) Efficiency (Ef) can be indicated by the time consumed in an authentication process. It includes user operation time and system response time. The system response time includes system processing time and data transmission time.
7) Security indicates the ability of the system to work properly under attacks. Liveness Detection Rate (LDR) of negative cases is used to measure this metric.
8) User Acceptance (UsAc) concerns perceived ease of use (Eu), usefulness (Us), playfulness (Pl), UI (In) and attitude of acceptance of a system (At). It represents a user's willingness to accept the system.

### A. Implementation and User Study

We conducted a series of user studies based on the implemented LiVoAuth prototype system. Due to the coronavirus pandemic, we conducted our user studies online. In order to facilitate experiment management and avoid troubles caused by installation and configuration, we deployed UA as an online web service, shown in Figure 5. The server was setup on a desktop computer running Windows 10 Operating System, with 16G RAM and broadband network connection. RP and UAP developed with the Python language were also located in the same desktop.

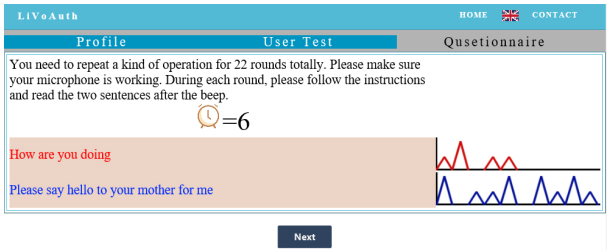



Fig. 5. UI of the LiVoAuth prototype system.

In our user test, there is no need for participants to "try out" before their formal test. After reading the project introduction and other information on the home page of UA, the participants can directly do their test. When a participant enters the UA homepage, he/she first sees a brief introduction to LiVoAuth and a description of user test. After understanding and agreeing with privacy statements, the participant can enter the test. As shown in Figure 5, it contains three parts: basic information collection, system test and questionnaire survey. Firstly, we collect the basic information of the participant for later analysis. Each participant needs to provide personal information, such as gender, nationality, contact information, as well as relevant technical background of using intelligent voice systems. Then, we start a system test. We ask the participant to register voiceprint information, repeat the authentication operation for 10 times, update the registration information once, and repeat the authentication operation for another 10 times. That is each participant records a total of 22 pieces of voices. During the above test, the CC and the LDM that the participant should follow are displayed, meanwhile the system provides voice prompts. At last, we ask the participant to fill in a questionnaire for collecting his/her feedback on the usage experience of the LiVoAuth prototype system in order to evaluate its usability and user acceptance.

We obtained the first batch of participant volunteers from our relatives and friends through direct invitation. Then, they further invited additional people to participate in our user study. Overall, 50 participants from five countries joined. There were 21 males and 29 females, aged between 14 and 54. Among them, 17 had used intelligent voice interaction systems, including smart speaker, smart home assistant, and vehicle voice assistant. But only 7 of them had used voiceprint authentication services. Refer to Appendix II for detailed background information of participants. The data including authentication time, authentication results, and voiceprints of each participant were recorded for the purpose of evaluating the performance of LiVoAuth.

### B. Experimental Results and Analysis

*1) Accuracy, Stability and Efficiency:* For evaluating accuracy and efficiency, we used all registered 100 voiceprints by 50 participants for 2 times as matching targets. During the user test, each of the 50 participants performed 20 authentication operations, thus in total 1000 voiceprints were collected. Each of these voiceprints was compared to the registered ones during user authentication in order to test authentication accuracy. The same user test was conducted for the second

time after 6 months for testing the stability of LiVoAuth. There were 20 participants in the second round of user test, all of them participated in the first round of test. We conducted the second round of test in order to verify that the possible changes of user voiceprint, environmental factors and other factors do not have an obvious negative impact on the normal operation of LiVoAuth over time.

Figure 6 shows the Detection Error Tradeoff (DET) curves of LiVoAuth in the two rounds of tests. We can see that the authentication accuracy of LiVoAuth reached 99.3%, which is at a leading level compared with the existing works (refer to Table I). Even after half a year, its accuracy still remained 99.1%. After screening and verification, the possible influencing factors on authentication accuracy could be that the adolescents inside the participants are in their sound change period and their voiceprint characteristics have changed during the past half year. However, through dynamic learning and registration update, this negative phenomenon can be eliminated to a great extent.

Regarding LiVoAuth efficiency, our test counted the average authentication processing time (i.e., computational cost) and data transmission time (i.e., transmission cost) required for each authentication, as shown in Figure 7. We can see that the average computational cost increases almost linearly with the increase of the number of words/Chinese characters in the challenge code. Some fluctuation may be caused by the difference of the number of syllables. Notably, this cost increase is more stable for Chinese than English and Finnish because all Chinese characters are monosyllabic. In general, the computational cost required for a single authentication is less than 35 milliseconds, which is quite efficient. As for the average transmission time, when the server was located in Helsinki, Finland, the transmission time for the participants in China is longer than the participants in Finland due to network delay. A similar result was gained when we located the server in China in a supplemented experiment, where other configurations were completely consistent with the previous experiment except that the server was located in China. As shown in Figure 7 (c), we can see that the transmission time for the participants in Finland is longer than the participants in China. Generally, the participants need to wait no more than 0.5 seconds to use the system due to data transmission delay, which is acceptable in practical system usage. The length of the challenge code (i.e., the number of words/Chinese characters in the challenge code) has less impact on the transmission time compared with the distance between the server and the participant location because the highest bound of the challenge code is small.

We further tested the effect of a number of factors, such as gender, age, nationality and environmental noise, on the authentication accuracy of LiVoAuth (including liveness detection). Figure 8 shows our testing results. We can see from Figure 8(a) that different genders and ages have little impact on authentication accuracy, which can reach over 98%. The highest authentication accuracy is achieved for Chinese, the accuracy of other speaking languages is lower than Chinese with varying degrees. This may be caused by the accuracy of Baidu speech-to-text toolkit [23] used in our system, in addition to the influence of accent. This effect can be reduced by appropriately relaxing the threshold of challenge code content check and giving a certain fault tolerance rate.

However, environmental noise has a great impact on the accuracy of LiVoAuth. Refer to Figure 8(b), the accuracy of the system in a noisy environment is lower than that in a quiet environment, and this problem is particularly obvious when the length of challenge code is too short. In order to investigate the main reason of negative impact of environmental noise on LiVoAuth, we separately analyzed LiVoAuth performance in case that there is only one element considered in LDM, either volume change or pause time (refer to Figure 8(c)(d)). Subsequently, two tests were performed with newly collected data. We invited 10 participants. In each test, they provided 200 voice samples when either volume change or pause time is considered in LDM generation. The test with the result shown in Figure 8(c) only considers volume changes. Its pause time in LDM is uniformly set to 0. The test with the result shown in Figure 8(d) only considers pause time. The volume in LDM is uniformly set to 30dB. It can be seen that ambient noise has a great impact on speaking volume and a small impact on speaking pause. Notably, when the length of the challenge code becomes short, the accuracy of authentication decreases. For overcoming this problem, we tested with a data set with noise reduction. The test results show that the accuracy of LiVoAuth is improved. In general, its average accuracy is over 99.4%. Regarding the tests under different noise levels shown in Figure 8 (b), the accuracy of LiVoAuth is improved as a whole. The accuracy of the test with the highest upper limit of accuracy is over 99.4% under the noise of 20 40dB. The accuracy of the test whose noise exceeds 60dB has the biggest improvement regarding the lower limit of accuracy, which is increased by 4% compared with that without noise reduction. Based on our tests, we believe that the average accuracy is acceptable when the length of the challenge code is no less than 4.

*2) Performance under Spoofing Attacks:* In order to verify the performance of LiVoAuth under spoofing attacks, we conducted a series of experiments as described below. The experimental dataset came from our user tests. The 100 voiceprints registered by the participants were used as a target set. The voice samples generated in the first 10 times of authentication operations of all participants in the first round of test, in total 500 voice samples were used as positive samples. The voice samples generated in the last 10 times authentication operations of all participants, in total 500 voice samples were marked as negative samples. They were used to simulate different degrees of spoofing attacks.

- Imitation attack. It is assumed that the attacker is very familiar with the voice of the target user and can imitate his/her sounds very similar in hearing. The participants in the user test do not have such imitation ability. So we try our best to find a few cases from the Internet: when a target voice is specified, someone can speak or sing in a voice very similar to it. This method is a real live attack. The voice used for the attack sounds more natural than the voice generated by any algorithms, and can avoid most of the existing detection methods reviewed in Section II. However, in our test, we found that even though there is little auditory difference between the imitated speech and the target speech, the speech recognition algorithm can still distinguish them, and the success rate of LiVoAuth achieves 100%.
- Direct replay. Assuming that an attacker collects some

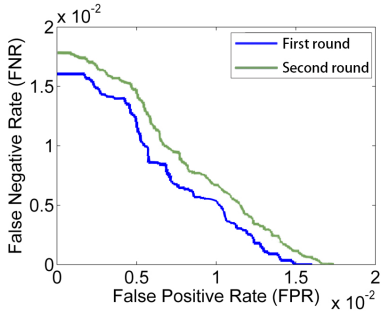
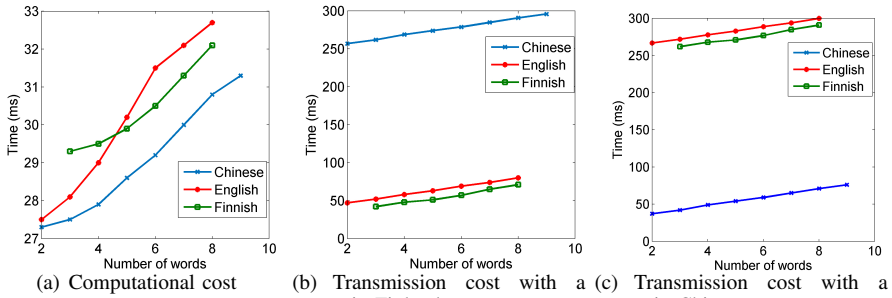
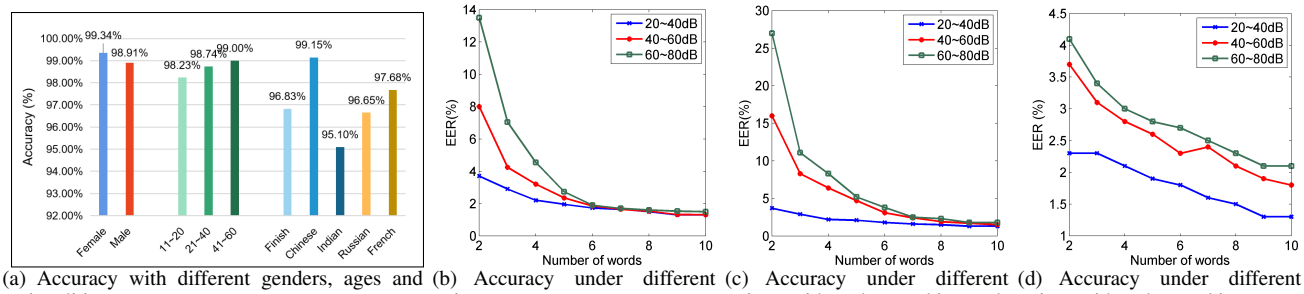



Fig. 6. DET curves of LiVoAuth.




(a) Computational cost (b) Transmission cost with a server in Finland (c) Transmission cost with a server in China

Fig. 7. Efficiency of LiVoAuth




(a) Accuracy with different genders, ages and nationalities (b) Accuracy under different noises (c) Accuracy under different noises with only speaking volume in LDM (d) Accuracy under different noises with only speaking pause in LDM

Fig. 8. Authentication accuracy of LiVoAuth affected by different factors

participant voices, the simplest way is to play these recordings directly. When we input negative samples to raise attacks directly without any update, the detection success rate of LiVoAuth is 100%. However, When we select the audio consistent with the content of the challenge code and raise a replay attack by playing such audio, the detection success rate is 98.7%. It is found that when the length of the challenge code is too short, the possibility of using the same or similar LDM increases. There are such examples in the experimental samples that were not detected.

- Indirect playback. Suppose that user voices can be segmented and re-edited into an audio that meets the content of challenge code. When we simply spliced negative samples, obtained an audio consistent with a target challenge code and raised an attack, the detection success rate is about 99%. When we further processed the forged challenge code and tried to make it meet the LDM, the detection success rate was reduced. In order to overcome this issue, we further improve our design. First, we make the length of CC more than 4 characters. Second, we introduce additional elements into LDM, e.g., speed, intonation, and strength of a word. Considering that overly complex LDM may cause inconvenience to users, LiVoAuth does not generate LDM by involving all possible elements in a single verification process. Instead, LiVoAuth randomly selects two of them to generate LDM, and provides corresponding graphic and voice prompts on the interface of UA to instruct user response. After optimization, the detection success rate can be maintained above 97.9%.
- Speech synthesis. Suppose that an attacker uses the known user voices to extract voiceprint features, and synthesizes a similar audio with a technical means to forge the user's voice. We attempted this attack based on some existing speech synthesis method [24], [25]. The attack success rate based on [25] is very low, less than 10%. We speculate its reason as the small size of input corpus. Since it is difficult to collect a large amount of corpus for training although the attacker may be able to obtain user voiceprint information from the UAP or social networks. The size of the corpus the attacker can obtain is normally small, so this test is acceptable. The attack success rate based on [24] is a bit higher than that of [25]. Based on our test, this attack has high adaptability to LDM. However, it takes a long time, usually more than 10 minutes, to generate an audio that can be used for attack due to its high computational complexity. If we set a proper response time limitation for authentication challenge response in LiVoAuth, this attack is difficult to succeed due to challenge response timeout.

*3) User Acceptance:* We also interviewed all participants with a questionnaire designed based on Technology Acceptance Model (TAM) [26]. It consists of 15 statements in terms of five items (i.e., Eu, Us, Pl, In and At), shown in Appendix III. Each statement is measured by a score from 1 to 5, which indicates a participant's attitude from totally disagree to totally agree. We calculated the average score and standard deviation of each item to measure user acceptance on LiVoAuth. The result is shown in Figure 9.

As can be seen from Figure 9, the overall feedback of participants on the system is good, with an average score over 4.1. However, we noticed a bit low score on ease of use. This implies that the participants feel that the system is not easy or convenient to use, or its response to human interaction may not meet the participants' expectations. In order to pursue the reason of the problem and improve the system, we interviewed several participants. The interview questions include how they

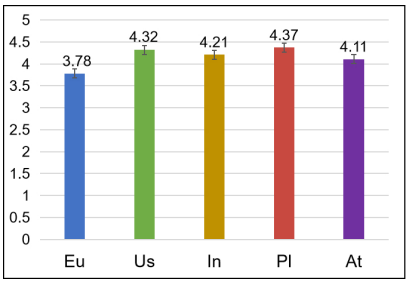



Fig. 9. Feedback of user acceptance.

feel about each step of the experiment and their views on some open questions, as listed in Appendix IV. We found that it is not easy to respond LDM with an expected speaking volume. It is hard for some participants to respond with high quality. Some participants said: "Too long challenge code does cause some problems, such as the difficulty of matching challenge code characters with LDM one by one, which made me confusing." "However, the length of the current setting is generally acceptable, and the CCs I encountered have only 4 to 6 characters." "It may be better if the prompt design of the challenge can be optimized." Moreover, speaking environment, the distance between the participant and his/her equipment, headphone and microphone usage also impact ease of use. Some participants think that speaking by following a volume request in a public place might be troublesome, worrying about disturbing others or revealing passwords and voiceprints, especially when the volume required by LDM is high. But there is no such a trouble in a vehicle or at home. By summarizing and analyzing our interview results, we believe that if we relax determination thresholds in LDM check and pay more attention to the relative changes of the speaking volumes of two concatenated words, users do not need to speak with required volumes strictly and accurately. In this case, LiVoAuth becomes easier to use. After modifying the system design accordingly, we performed the second-round of test and re-interviewed several participants who participated in the first-round of test. They affirmed the improvement of our system.

*4) Comparison:* Based on the evaluation metric proposed above, we compared LiVoAuth with some cutting-edge related works in Table I. Note that Movee is a video-based linevess detection method, while other methods are related to voiceprint. We can see that LiVoAuth achieves superior authentication accuracy and LDR. Its operational efficiency and user acceptance are also at the forefront. In order to compare the performance of liveness detection, we designed several additional attacks and simulated them. The simulated attacks include pop noise simulation, humanoid sound source array, voice gesture imitated by moving devices, etc. Under the same conditions, the LDRs of the methods listed in Table I are negatively affected with varying degrees. Among them, VoiceGesture is the most affected. Its LDR is reduced to less than 50%. On the other hand, when using mobile device to simulate voice gesture movement, even if a recording is played, it still produces a relatively large Doppler frequency shift. Thereby, VoiceLive is greatly affected, its LDR is reduced to about 76%. Notably, due to the specificity of different people's vocal positions, one sound source array model can only forge the vocal model of one user. Since it is difficult to imitate the characteristics of the source array by modifying the voice data, each attack takes time to adjust the position of the sound source array. VoicePop is relatively less affected by those attacks because the pop noise produced by different people also has specificity. Although pop noise is easy to simulate, it is difficult to simulate it with correct specific characteristics. Thus, its LDR is still kept at the level of 90% under the above attacks. In short, LiVoAuth is the least negatively affected by those attacks, its accuracy of liveness detection can still remain above 95%. The results show that LiVoAuth has better robustness against the above spoofing attacks, compared with other existing methods.

TABLE I
COMPARISON OF DIFFERENT METHODS

| Method | AA | St | Ef | LDR | UsAc | LDR under additional attack |
|---|---|---|---|---|---|---|
| VoicePop [4] | 94.6% | - | medium | 93.5% | high | 90% |
| Movee [19] | - | - | low | 68-93% | low | - |
| VoiceLive [9] | 99% | - | high | **99%** | high | 76% |
| VoiceGesture [20] | 99% | - | high | **99%** | medium | 50% |
| LiVoAuth | **99.3%** | **99.1%** | high | 97.9% | high | **95%** |

*5) Discussion:* Based on the above experimental tests and comparisons, we can see that LiVoAuth has excellent performance on accuracy, efficiency, stability, user acceptance and robustness of liveness detection. Its deployment cost is very low. At the same time, LiVoAuth has low restrictions on its user in terms of position, distance, and posture. However, LiVoAuth still has some shortcomings. First, the success rate of liveness detection is not top-notch. Considering that too short challenge code could make the LDM embedded in it useless, we can improve this weakness by limiting the shortest length of the challenge code and optimizing the LDM generation method. Second, the robustness of LiVoAuth seems to be easily questioned. This problem is also caused by the design of LDM. The LDM used in the current system is too simple and looks easy to be imitated. However, in practice, LDM can be designed by introducing additional elements to build various forms to avoid this problem. Finally, user feedback shows that the ease of use is not so satisfied. Some participants mentioned in the interview that the prompt information should in the current interface of UA is difficult to read. They hope to change it into a one-to-one format, where the challenge code is put above the LDM. In addition, the prompts such as timers should be optimized to improve user experience.

## VI. CONCLUSION

In this paper, we proposed LiVoAuth, an economic and effective liveness detection method for voiceprint authentication. It applies a form of random voice challenge with random LDM to resist spoofing attacks, thus enhancing the security of authentication. A series of user studies and experimental tests based on a prototype system show that LiVoAuth has high authentication accuracy, sound liveness detection accuracy, high stability and high efficiency. LiVoAuth is robust to resist various spoofing attacks and gains positive feedback on user acceptance. Compared with existing cutting-edge works, its performance is advanced. We will further improve its usability and ease of use to promote its adoption in practice.

## REFERENCES


[1] M. Golla and M. Dürmuth, "On the accuracy of password strength meters," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1567–1582. [Online]. Available: https://doi.org/10.1145/3243734.3243769

[2] M. Golla, M. Wei, J. Hainline, L. Filipe, M. Dürmuth, E. Redmiles, and B. Ur, ""what was that site doing with my facebook password?": Designing password-reuse notifications," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1549–1566. [Online]. Available: https://doi.org/10.1145/3243734.3243767

[3] Z. Rui and Z. Yan, "A survey on biometric authentication: Toward secure and privacy-preserving identification," *IEEE Access*, vol. 7, pp. 5994–6009, 2019.

[4] Q. Wang, X. Lin, M. Zhou, Y. Chen, C. Wang, Q. Li, and X. Luo, "Voicepop: A pop noise based anti-spoofing system for voice authentication on smartphones," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 2062–2070.

[5] N. K. Ratha, J. H. Connell, and R. M. Bolle, "An analysis of minutiae matching strength," in *Audio- and Video-Based Biometric Person Authentication*, J. Bigun and F. Smeraldi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 223–228.

[6] D. Gonzalez-Jimenez and J. L. Alba-Castro, "Toward pose-invariant 2-d face recognition through point distribution models and facial symmetry," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 413–429, 2007.

[7] C. C. Queirolo, L. Silva, O. R. P. Bellon, and M. Pamplona Segundo, "3d face recognition using simulated annealing and the surface interpenetration measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 206–219, 2010.

[8] X. Wang, Z. Yan, R. Zhang, and P. Zhang, "Attacks and defenses in user authentication systems: A survey," *Journal of Network and Computer Applications*, vol. 188, p. 103080, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804521001028

[9] L. Zhang, S. Tan, J. Yang, and Y. Chen, "Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 1080–1091.

[10] 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 *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1215–1229. [Online]. Available: https://doi.org/10.1145/3319535.3354248

[11] Y. Meng, Z. Wang, W. Zhang, P. Wu, H. Zhu, X. Liang, and Y. Liu, "Wivo: Enhancing the security of voice control system via wireless signal in iot environment," in *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. Mobihoc '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 81–90. [Online]. Available: https://doi.org/10.1145/3209582.3209591

[12] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," 08 2017, pp. 999–1003.

[13] S. Kaman, K. Swetha, S. Akram, and G. Varaprasad, "Remote user authentication using a voice authentication system," *Information Security Journal: A Global Perspective*, vol. 22, pp. 117 – 125, 2013.

[14] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *2016 IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 165–170.

[15] G. Bhattacharya, J. Monteiro, J. Alam, and P. Kenny, "Generative adversarial speaker embedding networks for domain robust end-to-end speaker verification," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6226–6230.

[16] Z. Yi, Z. Chen, H. Cai, W. Mao, M. Gong, and H. Zhang, "Bsd-gan: Branched generative adversarial network for scale-disentangled representation learning and image synthesis," *IEEE Transactions on Image Processing*, vol. 29, pp. 9073–9083, 2020.

[17] P. Jiang, Q. Wang, X. Lin, M. Zhou, W. Ding, C. Wang, C. Shen, and Q. Li, "Securing liveness detection for voice authentication via pop noises," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2022.

[18] R. Zhang, Z. Yan, X. Wang, and R. Deng, "Volere: Leakage resilient user authentication based on personal voice challenges," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2022.

[19] M. Rahman, U. Topkara, and B. Carbunar, "Movee: Video liveness verification for mobile devices using built-in motion sensors," *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1197–1210, 2016.

[20] L. Zhang, S. Tan, and J. Yang, "Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 57–71. [Online]. Available: https://doi.org/10.1145/3133956.3133962

[21] D. S. Wei, S. Murugesan, S.-Y. Kuo, K. Naik, and D. Krizanc, "Enhancing data integrity and privacy in the cloud: An agenda," *Computer*, vol. 46, no. 11, pp. 87–90, 2013.

[22] A. Ullah, M. Azeem, H. Ashraf, A. A. Alaboudi, M. Humayun, and N. Jhanjhi, "Secure healthcare data aggregation and transmission in iot—a survey," *IEEE Access*, vol. 9, pp. 16 849–16 865, 2021.

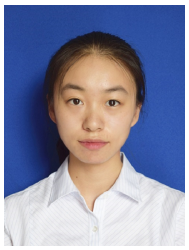
[23] (2021). [Online]. Available: https://ai.baidu.com/tech/speech/asr

[24] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," 2017.

[25] J. Yamagishi, B. Usabaev, S. King, O. Watts, J. Dines, J. Tian, Y. Guan, R. Hu, K. Oura, Y.-J. Wu, K. Tokuda, R. Karhila, and M. Kurimo, "Thousands of voices for hmm-based speech synthesis–analysis and application of tts systems built on various asr corpora," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 984–1004, 2010.

[26] Z. Yan, Y. Chen, and Y. Shen, "A practical reputation system for pervasive social chatting," *Journal of Computer and System Sciences*, vol. 79, no. 5, pp. 556–572, 2013.

[27] (2020). [Online]. Available: https://librosa.org/doc/main/index.html

**Rui Zhang** received the B.Sc. degree in Computer Science and Technology from China University of Mining and Technology, Xuzhou, China, in 2016. Now she is studying for the PhD degree in Xidian University, Xi'an, China, major in cyber security. Her research interests are in information security, user authentication and privacy preservation.

**Zheng Yan** received the D.Sc. degree in technology from the Helsinki University of Technology, Espoo, Finland, in 2007. She is currently a Professor in the School of Cyber Engineering, Xidian University, Xi'an, China. Her research interests are in trust, security, privacy, and security-related data analytics. Dr. Yan is an area editor or an associate Editor of IEEE INTERNET OF THINGS JOURNAL, Information Fusion, Information Sciences, and so on. She served as a General Chair or Program Chair for numerous international conferences, including IEEE TrustCom 2015 and IFIP Networking 2021. She is a Founding Steering Committee co-chair of IEEE Blockchain conference. Her achieved awards include 2021 N²Women: Stars in Computer Networking and Communications, Nokia Distinguished Inventor Award, Aalto ELEC Impact Award, the Best Journal Paper Award issued by IEEE Communication Society Technical Committee on Big Data and the Outstanding Associate Editor of 2017 and 2018 for IEEE Access.

**Xuerui Wang** received the B.E. degree in Computer Science and Technology from Xidian University, Xi'an, China, 2018. She is currently pursuing the master degree in Xidian University, Xi'an, China. Her main research interests are in attack and defense in authentication systems.

**Robert H. Deng** (M'03-SM'10-F'17) is AXA Chair Professor of Cybersecurity, Director of the Secure Mobile Centre, and Deputy Dean for Faculty & Research, School of Computing and Information Systems, Singapore Management University. His research interests are in the areas of data security and privacy, network security, and applied cryptography. He received the Outstanding University Researcher Award from National University of Singapore, Lee Kuan Yew Fellowship for Research Excellence from SMU, and Asia-Pacific Information Security Leadership Achievements Community Service Star from International Information Systems Security Certification Consortium. He serves/served on the editorial boards of ACM Transactions on Privacy and Security, IEEE Security & Privacy, IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Information Forensics and Security, Journal of Computer Science and Technology, and Steering Committee Chair of the ACM Asia Conference on Computer and Communications Security. He is a Fellow of IEEE and a Fellow of Academy of Engineering Singapore.

## APPENDIX I
## SECURITY ANALYSIS

*Proposition 1:* Spoofing attack cannot be carried out by a simple replay.

*Proof:* The challenge codes in the process of registration and authentication come from different datasets, each challenge code and its LDM are randomly selected. The probability that the contents of CCs and their LDMs of two challenges are the same is pretty low.

The registration challenge code is randomly selected from set $\mathbf{A} = CC_i, i = 1, ..., I$. The authentication challenge code is randomly selected from set $\mathbf{B} = CC_j, j = 1, ..., J$. For $\forall CC_i \in \mathbf{A}$ and $\forall CC_j \in \mathbf{B}$, $CC_i \neq CC_j$. The parameters of LDM is specified in Section IV-A. The length of LDM $l$ is assumed to be equally distributed over its scope. Its maximum length is $L$, the amount of selectable first element (e.g., voice volume) is $n$, and the amount of selectable second element (e.g., pause time) is $m$.

It is assumed that the attacker obtains any voice of the user in the registration or authentication, noted $VP$, and directly implements a replay attack. The probability that $VP$ meets the CC requirements for the next authentication is: $P_1 = \frac{I}{I+J} \times 0 + \frac{J}{I+J} \times \frac{1}{J}$; The probability that $VP$ meets the LDM requirements is: $P_2 = \frac{1}{L-1} \sum_{l=2}^{L} \left(\frac{1}{n}\right)^l \left(\frac{1}{m}\right)^{l-1}$. Since the returned voice content needs to be checked to confirm that it is consistent with the CC with the same speaking mode specified in its LDM, the probability of replay attack by recording and replaying is $P = P_1 \times P_2 = \frac{1}{(I+J)(L-1)} \sum_{l=2}^{L} \left(\frac{1}{n}\right)^l \left(\frac{1}{m}\right)^{l-1}$. Given $I = 100$, $J = 1000$, $L = 10$, $n = 7$, and $m = 5$, $P = 0.000043\%$. This is a very low probability, thus the replay attack is difficult to succeed. ■

*Proposition 2:* It is difficult for an attacker to obtain an audio through compilation for raising a spoofing attack.

*Proof:* We introduce LDM to work with CC during authentication in LiVoAuth. In order to pass authentication, the returned challenge code audio should meet the speaking style specified by the LDM. Even if the attacker can obtain enough samples by collecting the user's past voices and can construct an audio $VP$ consistent with the content of CC through compilation, which means the probability that $VP$ meets the requirements of CC is $P_1 = 100\%$. The authenticate is still hard to be successful because the audio mostly does not comply with the LDM due to its randomness and a large number of possible composition results of multiple vector elements with different values. The parameters of LDM is the same as that of Proof 1 The probability of $VP$ meets the requirements of LDM is $P_2 = \frac{1}{L-1} \sum_{l=2}^{L} \left(\frac{1}{n}\right)^l \left(\frac{1}{m}\right)^{l-1}$. The probability of spoofing attack is $P = P_1 \times P_2$. Given the same values of $L$, $n$, $m$ as in Proof 1, $P = 0.047\%$, which is still very low while the compilation cost is very high, so that spoofing in this way is not feasible. ■

*Proposition 3:* It is difficult for an attacker to obtain an audio through speech synthesis for raising a spoofing attack.

*Proof:* As far as we know, there is a technology to synthesize a similar speech by using a voiceprint feature template of a user and establishing the user's voiceprint model through machine learning. But so far, the computational overhead of this technique is still very high. For example, a toolkit named Librosa [27] takes about 15 minutes to generate a 30-second .wav audio file. But the authentication time of LiVoAuth is much shorter than the time spent to generate a synthesized audio using a known voiceprint feature template. Thus, it is hard to raise a spoofing attack through speech synthesis in LiVoAuth if we set a proper authentication timeout threshold. (Refer to Figure 7 about LiVoAuth's efficiency.) ■

## APPENDIX II
## BACKGROUND INFORMATION OF PARTICIPANTS

Table II shows the background information of all participants.

TABLE II
BACKGROUND INFORMATION OF PARTICIPANTS.

| User | Gender | Age | Nationality | Envi[1] | TB[2] |
|---|---|---|---|---|---|
| 1 | Female | 28 | Chinese | office | × |
| 2 | Male | 45 | Finnish | office | × |
| 3 | Female | 31 | Finnish | office | × |
| 4 | Male | 28 | Chinese | office | ✓ |
| 5 | Female | 27 | Indian | office | × |
| 6 | Female | 26 | Indian | office | × |
| 7 | Male | 28 | Indian | office | × |
| 8 | Male | 37 | French | office | × |
| 9 | Male | 45 | Finnish | canteen | × |
| 10 | Female | 39 | Finnish | canteen | × |
| 11 | Female | 51 | Chinese | canteen | × |
| 12 | Female | 24 | French | quiet | × |
| 13 | Male | 51 | Chinese | canteen | × |
| 14 | Female | 49 | Chinese | canteen | × |
| 15 | Male | 14 | French | quiet | × |
| 16 | Male | 53 | Chinese | canteen | × |
| 11 | Male | 43 | Finnish | canteen | × |
| 12 | Female | 45 | French | canteen | × |
| 13 | Female | 17 | Chinese | quiet | × |
| 14 | Female | 18 | Chinese | quiet | ✓ |
| 15 | Female | 18 | Chinese | quiet | ✓ |
| 16 | Female | 24 | Chinese | quiet | × |
| 17 | Male | 26 | Chinese | office | × |
| 18 | Male | 21 | Chinese | office | × |
| 19 | Male | 27 | Chinese | office | × |
| 20 | Female | 46 | Chinese | office | × |
| 21 | Female | 51 | Chinese | canteen | × |
| 22 | Female | 47 | Chinese | canteen | × |
| 23 | Female | 28 | Chinese | canteen | × |
| 24 | Female | 31 | Chinese | canteen | × |
| 25 | Female | 28 | Chinese | office | × |
| 26 | Male | 25 | Chinese | office | × |
| 27 | Male | 27 | Chinese | office | × |
| 28 | Female | 23 | Chinese | office | × |
| 29 | Female | 23 | Chinese | office | × |
| 30 | Male | 17 | Chinese | quiet | × |
| 31 | Male | 17 | Chinese | quiet | ✓ |
| 32 | Female | 18 | Chinese | quiet | × |
| 33 | Male | 15 | Chinese | quiet | × |
| 34 | Male | 39 | Chinese | canteen | × |
| 35 | Male | 41 | Chinese | canteen | × |
| 36 | Male | 53 | Chinese | canteen | ✓ |
| 37 | Male | 52 | Chinese | canteen | × |
| 38 | Female | 24 | Chinese | office | × |
| 39 | Female | 24 | Chinese | office | × |
| 40 | Male | 28 | Chinese | office | × |
| 41 | Female | 25 | Chinese | quiet | × |
| 42 | Female | 22 | Chinese | quiet | × |
| 43 | Male | 24 | Chinese | quiet | × |
| 44 | Female | 24 | Chinese | quiet | × |
| 45 | Male | 25 | Chinese | quiet | × |
| 46 | Female | 54 | Chinese | canteen | ✓ |
| 47 | Female | 53 | Chinese | canteen | × |
| 48 | Male | 53 | Chinese | canteen | ✓ |
| 49 | Male | 52 | Chinese | canteen | × |
| 50 | Male | 18 | Chinese | quiet | × |

Envi: Environments
TB: Technical background. ✓- the user had used voiceprint authentication. × - the user had never used voiceprint authentication.

## APPENDIX III
## QUESTIONNAIRE FOR TESTING USER ACCEPTANCE

The questionnaire is designed based on a 5-point Likert scale, consisting of 15 statements with regard to five aspects, i.e., perceived ease of use, usefulness, user interface, playfulness and usage attitude in future).

1) **Perceived ease of use**

S1: I think it is convenient for me to use the system as an authentication method in my daily life.
S2: I think it is easy for me to use the system.
S3: I think it is fast and accurate to complete authentication.

2) **Usefulness**
S4: The system is better than password-based authentication.
S5: The system can help me manage my accounts in the Internet applications.
S6: The system is a helpful application.

3) **Interface**
S7: The system provides a fashionable way of authentication.
S8: The system has a simple way to interact.
S9: The system provides a good user interface for authentication operation.

4) **Playfulness**
S10: Using the system for authentication makes me happier than entering user ID and password.
S11: The system provides a joyful method of authentication.
S12: The system is an interesting application.

5) **Attitude**
S13: I would like to use the system.
S14: The system is very cool.
S15: I prefer using the system compared with the authentication system based on user ID and password.

## APPENDIX IV
## SEMI-STRUCTURED INTERVIEW OUTLINE

This section provides a semi-structured interview outline. We interviewed some user test participants. The interview was conducted according to the outline. Before the interview, we explained the purpose of the interview to the interviewees, obtained the consent of all interviewees in advance, and recorded the contents of all interviews.

Before the in-depth interview, here are some questions for collecting the background information of participants:

1) What kind of intelligent voice assistant or other voice system have you used?
2) What authentication methods do you usually use in your daily life?
3) For which transactions or applications do you need to perform user authentication?

With regards to usage experience of LiVoAuth, we directly ask the following questions:

1) Compared with other similar systems you have used, what is the difference between them and LiVoAuth?
2) Do you feel bothered during LiVoAuth usgae?

If the answer to Question 5 is “Yes”, further answer Question 7 and 8; If the answer is “No”, please answer Question 6.

1) What kind of application scenario do you think LiVoAuth is suitable for?
2) What do you think LiVoAuth bothers you during usage?
3) Do you have any suggestions on improving LiVoAuth with regard to its interface design and functionalities?