LiveProbe: Exploring Continuous Voice Liveness Detection via Phonemic Energy Response Patterns

Hangcheng Cao, *Student Member, IEEE*, Hongbo Jiang, *Senior Member, IEEE*, Daibo Liu, *Member, IEEE*, Ruize Wang, Geyong Min, *Member, IEEE*, Jiangchuan Liu, *Fellow, IEEE*, Schahram Dustdar, *Fellow, IEEE*, and John C. S. Lui, *Fellow, IEEE*

***Abstract*—Voice assistants support contactless smart device control and thus act as a holy grail of human–computer interaction. However, recent studies reveal that an adversary can manipulate devices by vicious voice commands. This security risk is caused by only executing one-time liveness detection and lacking safeguard modules after service activation. Therefore, identifying speaker type (i.e., human articulators or loudspeakers) is critical in protecting voice-driven services during an entire interaction session. In this article, we propose a continuous voice liveness detection approach *LiveProbe*, leveraging unique energy response patterns in frequency bands induced by distinct voice generation mechanisms. The rationality behind LiveProbe is presented in two aspects: human articulator reshapes initial voices by exquisitely coordinated movements of vocal organs, which act as band-pass filters generating unique energy responses; nevertheless, the internal modules of loudspeakers are position fixed and cannot reproduce this response characteristic. To that end, we first work on voice generation mechanisms behind two-type speakers that cause spectrum differences. Then, we elaborately construct signal processing and deep-learning modules to extract liveness features. Especially, our approach does not interfere with normal voice interaction and need not to carry customized sensors. The experiment presents its effectiveness against potential attacks with a false acceptance rate of 0.51%.**

***Index Terms*—Continuous liveness detection, energy response pattern (ERP), voice assistant (VA).**

Manuscript received 18 February 2022; revised 28 October 2022; accepted 9 December 2022. Date of publication 13 December 2022; date of current version 7 April 2023. This work was supported in part by the National Natural Science Foundation of China under Grant U20A20181, Grant 61732017, Grant 61902060, and Grant 61902122; in part by the China Scholarship Council; in part by the Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) under Grant GML-KF-22-23; and in part by the National Social Science Foundation of China under Grant 19ZDA103. *(Corresponding author: Hongbo Jiang.)*

Hangcheng Cao, Hongbo Jiang, Daibo Liu, and Ruize Wang are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, Hunan, China (e-mail: hangchengcao@hnu.edu.cn; hongbojiang@hnu.edu.cn; dbliu@hnu.edu.cn; ruizewang@hnu.edu.cn).

Geyong Min is with the Department of Computer Science, College of Engineering, Mathematics, and Physical Sciences, University of Exeter, EX4 4QF Exeter, U.K. (e-mail: g.min@exeter.ac.uk).

Jiangchuan Liu is with the School of Computing Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada, and also with Jiangxing Intelligence Inc., Nanjing 210000, China (e-mail: jcliu@sfu.ca).

Schahram Dustdar is with the Research Division of Distributed Systems, TU Wien, 1040 Vienna, Austria (e-mail: dustdar@dsg.tuwien.ac.at).

John C. S. Lui is with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China (e-mail: cslui@cse.cuhk.edu.hk).

Digital Object Identifier 10.1109/JIOT.2022.3228819

I. Introduction

TECHNOLOGY advancements in natural language processing [1] make contactless command implementation by voice assistant (VA) practical. For most smart devices, users leverage voice interfaces (e.g., Amazon Echo [2] and Apple Siri [3]) to acquire services, such as sending private messages [4] and completing online payments [5]. This noncontact control way frees users from tediously manual command inputting operations. Benefiting from intrinsic convenience and efficiency, the installation capacity of VAs currently is more than 4 billion [6] around the world, with a market size about 27 billion dollars [7]. Obviously, VA has become a critical module for human–computer interaction and smart home ecosystem. However, existing studies [8], [9], [10], [11] state that numerous attacks have successfully deceived VA systems. Among them, the replaying attack [12], [13], [14], [15] is recognized as the most implementable approach since anyone can easily launch. In this case, adversaries can utilize smartphones playing prerecorded [16] or synthesized voices [17] to illegally control devices through voice-enabled interfaces. To resist these attacks during entire interaction sessions, researchers explore continuous liveness detection mechanisms [18] for differentiating speaker types. For instance, they employ accelerometer [12], mmWave [13], and ultrasound [19] to sense throat vibration and oral structure movements for representing liveness signs; recent studies [14], [20] extract signal attenuation characteristics/sound field fingerprint by multiple microphones, to distinguish speaker type.

To sum up, existing methods require users to either carry/configure additional sensors or obey cumbersome usage restrictions like only allowing to emit voice commands in a preset position. Thus, they impose burdensome involvements on users and are unfriendly. By retrospecting on existing works, a desired liveness detection approach should meet two basic but critical demands: 1) security and 2) user-friendliness. First, considering the drawbacks of one-time user authentication, the continuous way is indeed needed for more secure protection. It continuously verifies speaker type during entire interaction sessions instead of only verifying wake-up words when the service is activated. Second, for ensuring the service quality of VAs, liveness detection processes need to be transparent to users, without disturbing normal voice command control. Specifically, it should not impose overmuch user involvement like carrying customized liveness feature collection sensors [12].

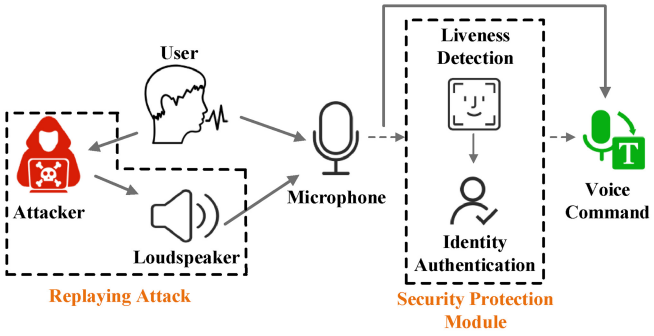


Fig. 1. Classical workflow of the VA system.

In this proposal, we adequately leverage passively collected voices to distinguish speaker types driven by the unforgeability of human articulators (i.e., physiological biometrics) and the uniqueness of multiple vocal organs' coordinated movements (i.e., behavioral biometrics), for meeting the aforementioned two demands. On the one hand, the vocal cord/tract and chest cavity keep intrinsic differences in tissue structure and shape among users. Thus, an adversary is hard to counterfeit such a complicated articulator to imitate human-live voices. On the other hand, to utter specific phonemes, relative positions and states of vocal organs are continuously adjusted, which is a dynamic process. Nevertheless, electronic submodules of loudspeakers are position fixed and hence keeps static in working periods. Relying on the above analysis, there is a question: can we leverage the unforgeable human articulator structure and the unique human-live voice generation mechanism for continuous liveness detection?

In this article, *LiveProbe* extracts liveness traits, i.e., energy response (variation) patterns, during entire human–device interaction sessions. It only utilizes built-in microphones, without imposing extra user involvement. As depicted in Fig. 1, before transforming inputting audios into commands and providing corresponding services, VA verifies speaker type for the security protection goal. To be specific, our method leverages the static energy response pattern (ERP) of each voice frame unit to capture differences in voice spectrum between two-type speakers. The reason for ERP working is that it reflects the special filter properties of human vocal structures (described in Section II-A), which electronic speakers cannot imitate. Furthermore, each phoneme can be disassembled into multiple frames, and response pattern variations across them effectively represent the dynamic vocal organ movement process of human voice generation. By combining the static and dynamic traits, LiveProbe sufficiently represents physiological and behavioral biometrics of human speakers to complete liveness detection.

Although this idea sounds straightforward, the implementation process faces three main challenges. The first one is that voice spectrum derivant extracted by distinct ways are used for distinct usage goals, such as Mel-frequency cepstral coefficient (MFCC) [21] for voice recognition and energy statistic features for activity detection [22]. Thus, elaborately defining and mining ERP for effectively characterizing liveness traits is critical and nontrivial. Second, the built-in microphones of most IoT terminals possess a sampling rate of no less than 44.1 kHz. In this case, processing units require to handle large-size samples per second, which undoubtedly costs significant computing resources and is unacceptable for a continuous detection mechanism. Last but not least, existing phoneme/voice segmentation and recognition technologies just reach a low accuracy in practical application scenes [23] that do not satisfy the demand for continuous liveness detection mechanisms.

To solve the aforementioned problems, we design the following pointed solutions. First, we analyze the structure differences of two-class speakers and thereby revealing distinct voice generation/reconstruct mechanisms; ERP is elaborately defined to represent specific frequency filtering function of human articulator. Second, each phoneme contains multiple time frames, and much spectrum information is repeated among adjacent ones, thus we extract critical frames utilizing a similarity-based judgment approach to reduce computation cost; moreover, we design matching algorithms employing locality sensitive hashing (LSH) [24] for a quick pairing among inputting and registered feature vectors. Third, through confirmatory experiments, we learn that liveness detection failure always happens when using features of only one critical frame; thus, LiveProbe classifies speaker type by multiple data frames of one phoneme and then leverages the decision voting mechanism [25] to jointly determine the final detection result, for improving classifier robustness. In summary, our major contributions of *LiveProbe* are concluded as follows.

1) We propose an easily deployable continuous liveness detection approach LiveProbe for voice-enabled IoT COTS, utilizing only built-in microphones; it does not require additional user involvement and configuring environments.
2) We analyze the differences in structure and voice generation mechanism between human articulators and electronic speakers, thereby extracting ERP-based liveness traits.
3) We innovatively design a quick feature matching method utilizing the LSH function, maximal overlap discrete wavelet transform (MODWT)-based method to reconstruct ERP even if environment factor changes, multiband spectral subtraction way to remove nonlinear background noise, etc.
4) We conduct extensive experiments to evaluate the security and effectiveness of LiveProbe in defending against replaying attacks. The results reveal that our approach can achieve the averaging false acceptance rate (FAR) of 0.51% and false rejection rate (FRR) of 1.12% in house-in and public data sets.

The remaining parts of this article are organized as follows. Section II introduces voice generation mechanisms of two-class speakers and the process of extracting liveness features. Potential attack models and system overview are described in Sections III and IV, respectively. We present the technical modules of LiveProbe and their functions in Section V. The experiment setting and performance evaluation are presented in Section VI. Following that, we discuss limitations and related works in Sections VII and VIII, we finally conclude our work in Section IX.

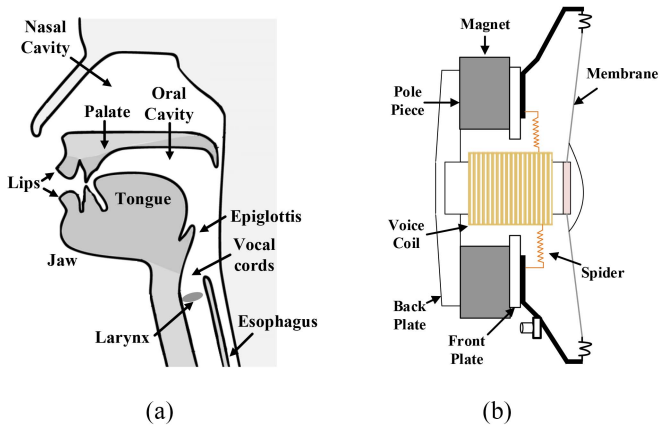


Fig. 2. Structure illustration of two speakers, presenting obvious differences. (a) Human articulator. (b) Electronic loudspeaker.

II. Background and Preliminaries

In this section, we introduce voice generation mechanisms of human articulator and electronic loudspeakers. Then, we verify the feasibility of distinguishing speaker type by analyzing phoneme-level ERP.

A. Voice Generation Mechanism

The process of human-live voice generation consists of two main stages: 1) initial voice formation and 2) reshaping. When a user speaks, airflow is expelled from the lungs and then passes through vocal cords. The contracted larynx blocks airflow, which drives vocal cords to vibrate and thereby generate initial voices; the initial outcome only covers fundamental signal frequencies decided by the tension and length of vocal cords. If the larynx is in a relaxed state, air can smoothly flow through along with causing slightly vocal cord vibration and a noise-like whispering. Subsequently, multiple vocal organs, such as throat and palate in Fig. 2(a), jointly adjust postures to construct a phoneme-specific airflow transmission pipe for reshaping the initial voice. After reshaping, its amplitude and frequency are adapted to carry user-specific voice biometrics information and human articulator actually plays the role of a filter [8], [26]. Finally, voice is emitted from human mouth/nose cavities and spreads out in surrounding environments. The aforementioned description reveals that human-live voice is a product of physiological and behavioral traits, for uniquely representing user identity. Therefore, even if individuals utter identical words, there are inherent differences in their voices.

In Fig. 2(b), we display a typical structure of electronic loudspeakers deriving most existing audio equipment designs. It as a common transducer converts electrical energy into acoustic signals. As turning on a loudspeaker, internal energized voice coils are subjected to forces from surrounding magnetic fields; the coil's vibration direction and magnitude are consistent with incoming currents. Meanwhile, voice coils drive membranes to vibrate and collide with air, hence forming voice waves radiating to surrounding spaces. Compared with human articulators, loudspeakers emit sounds only relying on membrane vibration determined by currents, lacking an initial voice reshaping stage. To spoof VAs, an adversary replays electrical inputting sources like synthesized/recorded audio samples by loudspeakers.

Summary: Unlike loudspeakers with fixed electronic components, human-live voice generation requires coordinated movements of multiple vocal organs. Even if outputting individual phonemes, there exist dynamic adjustment processes of vocal tract posture. For instance, to pronounce vowel /e/, people have to open mouths and control their tongues close to lower teeth; while they naturally stretch the tongue and drive airflow to open lips for emit consonant /b/. Particularly, resonant cavity posture changes affect initial voice formation and propagation paths, thereby bringing unique phonemic spectrum variation patterns. However, loudspeakers only rely on membrane vibration to output voices and hence cannot reproduce this complex variation characteristic.

B. Energy Response Pattern Derived From Spectrum

Spectrum represents the amplitudes of signal harmonic bands in frequency domain and thereby helping researchers conduct component analysis. For instance, most existing user authentication methods leverage features, such as linear prediction Cepstral coefficient (LPCC) [21], derived from spectrum information. However, these features are ineffective in directly applying to liveness detection tasks. In this section, we define *energy response (variation) pattern* that contains unique liveness characteristics of human-live voices.

Voice is regarded as a carrier conveying rapid variation information and hence capturing its dynamic characteristic requires high resolution methods. In this case, *LiveProbe* applies continuous wavelet transform (CWT) [27] performing frequency decomposition refinement of signals, to output spectrum information in millisecond time frames. Processing original voice samples by CWT, we obtain spectrum vectors denoted as $\{X_1, X_2, \ldots, X_N\}$ from N frames. The energy (amplitude) of frequency bands in the nth vector is expressed as $X_n = \{x_n^1, x_n^2, \ldots, x_n^M\}$, where M is the number of frequency bands. For extracting the ERP of the nth frame, named ERP_n, we first obtain the normalized energy ratio (NER) by the following equation:

$$\text{NER}_n = \left\{ \frac{x_n^1}{\text{sum}(X_n)}, \frac{x_n^2}{\text{sum}(X_n)}, \ldots, \frac{x_n^M}{\text{sum}(X_n)} \right\} \tag{1}$$

where $\text{sum}(\cdot)$ is a accumulative function and x_n^m is the mth scalar of X_n. If a speaker owns equal energy response distribution, it can be denoted as $\overline{\text{ERP}_n}$, that is, $\{(1^1/M), (1^2/M), \ldots, (1^M/M)\}$. We implement a subtraction operation between NER and $\overline{\text{ERP}}$ for depicting each speaker's phoneme-level ERP, that is, $\{(x_n^1/\text{sum}(X_n)) - (1/M), [x_n^2/\text{sum}(X_n)] - [1/M], \ldots, [x_n^M/\text{sum}(X_n)] - [1/M]\}$. Moreover, ERPs are variable induced by the posture adjustment of vocal organs as speaking, which is a critical trait employed to capture dynamic liveness characteristics. Therefore, we propose the energy response variation pattern (ERVP) among multiple *critical time frames*[1] of phonemes to

[1]The ERPs corresponding to adjacent time frames exhibit high similarity, thus we screen out only representative ones named *critical time frame* to further extract liveness characteristics of human-live voices.

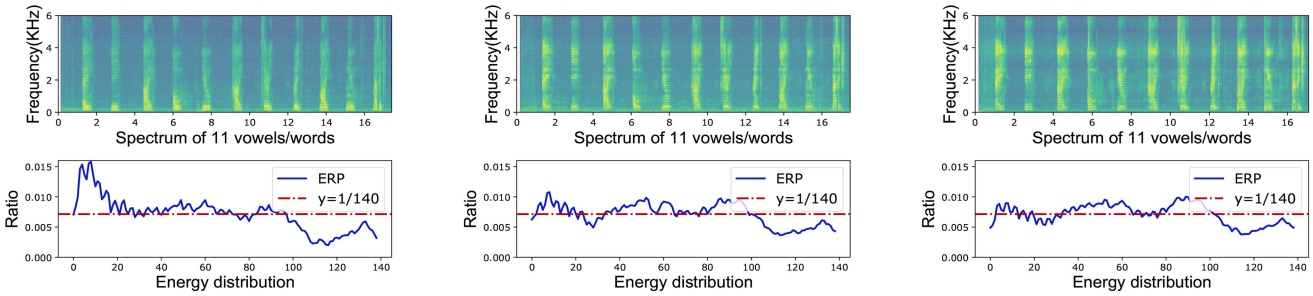


Fig. 3. Voice spectrum of five vowels and a phase "*Hi, Siri; read my new message*" consisting six words is uttered by (a) user and (b) and (c) replayed by two loudspeakers; moreover, the corresponding *ERP*s of vowel /a/ are displayed in the second-row subfigures.

describe this dynamic process, which is defined as follows:

$$\text{ERVP} = \{\text{ERP}_2 - \text{ERP}_1, \text{ERP}_3 - \text{ERP}_1, \ldots, \text{ERP}_N - \text{ERP}_1\}. \quad (2)$$

As depicted in (2), we execute a subtraction operation between the nth and 1st frames. The rationality for employing ERP/ERVP to achieve liveness detection is summarized in two respects: first, human vocal cavities act as band-pass filter banks remaining user-specific frequency parts [8], [28] and leading spectrum energy to concentrate in low-frequency bands; due to intrinsic structure differences as described in Section II-A, loudspeakers lack such frequency filter modules of the human articulator, and electromagnetic noise [29] generated by internal electronic components inevitably interferes with original voices and spread to higher frequencies. Second, even if uttering individual phonemes, people make evident human cavity structure adjustment, which does not occur in electronic speakers. This process leads ERPs to vary in multiple voice frames and thereby rendering ERVP to represent unique liveness characteristics.

C. Feasibility of Designing LiveProbe

In this section, we explore the feasibility of employing ERP/ERVP to tell apart two-class speaker types. A user first utters five vowels (i.e., /a/, /e/, /i/, /o/, /u/) and an example sentence "*Hi, Siri. Read my new message*." Meanwhile, the microphone SONY ECM-W2BT with high-fidelity frequency response performance records and saves this audio at a sampling rate of 44.1 kHz. Then, with controlling other environment variation interference factors like speaker position, we leverage two common portable devices (i.e., HUAWEI Meta30 Pro and iPad 8th-Gen) to play the recorded audios. Following that, we apply the CWT method to convert human-live and replaying voices to the frequency domain, for presenting signal components in continuous time frames. In this process, original voices are divided into multiple data segments with a window length of 25 ms, and the overlapping length between adjacent segments is set as 10 ms. In addition, a Hanning window multiplies segmented data to smooth the boundary samples and thereby alleviating the impact of spectrum leakage. For ease of analysis, we reserve time intervals between words/letters. The first row figures in Fig. 3 depict that spectrum energy of human-live voices are mainly concentrated in a fraction of frequency bands (i.e., less than 2 kHz); while the

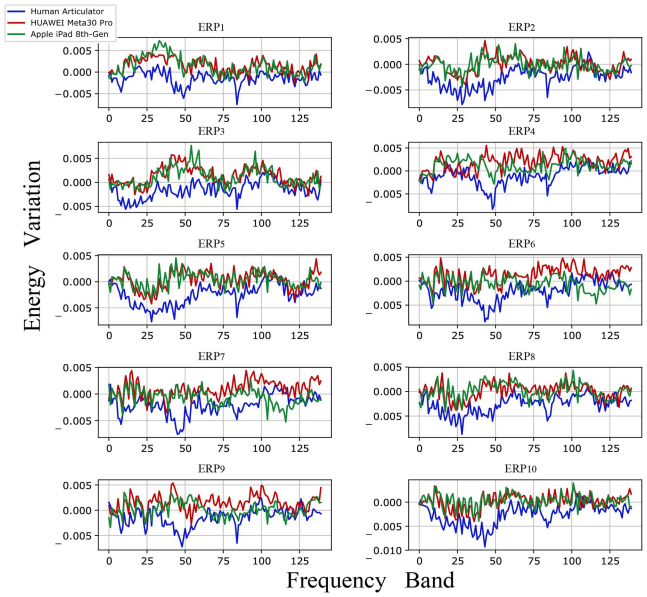


Fig. 4. There are extracted *ERVP*s of human articulator (blue line), HUAWEI Meta30 Pro (red), and Apple iPad 8th-Gen (green); each ERVP consists of ten ERPs, named from ERP_1–ERP_{10}.

energies of replaying audios are scattered over a much wider range (i.e., up to 6 kHz).

To further quantify this energy response difference, the second row of Fig. 3 displays the averaging NERs of one critical frame of vowel /a/ emitted by a user and two loudspeakers. The horizontal red line $y = (1/140)x$ is on behalf of equal energy response of 140 frequency bands as drawn. For LiveProbe, a large difference between one response distribution and $y = (1/140)x$ means that spectrum energy is only concentrated in a few bands. We count difference values by comparing the three NERs with $y = (1/140)x$ to obtain ERPs, respectively, and the value 0.352 referring to the human voice is significantly larger than replaying ones (i.e., 0.196 and 0.181). This result is consistent with our aforementioned analysis, that is, vocal organs act as a band-pass filter, and voice energy mainly concentrates in only a few bands. Subsequently, we extract the ERVP of one critical frame /a/ and other 10 ones as shown in Fig. 4. There are two evident distinctions in terms of envelope and amplitude among the three speakers' ERVPs. To measure the ERP variation of ten frames, we count

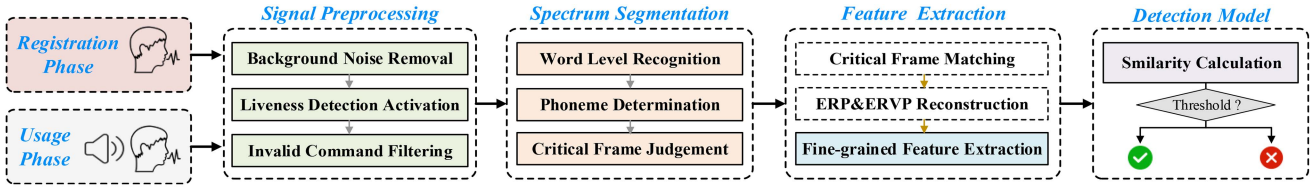


Fig. 5. Technical module overview of *LiveProbe*, consisting of *Signal Preprocessing*, *Spectrum Segmentation*, *Feature Extraction*, and *Detection Model*.

the amplitude of three speakers' ERVPs. Human-live speakers own the largest value 3.13 that larger than 1.26 and 0.91 belonging to loudspeakers. The result illustrates that human articulator generates greater intraclass ERP variation than electronic ones during speaking, which verifies our inference about the effects on energy response induced by articulator structure adjustment in Section II-B.

III. Threat Model

Our approach focuses on resisting spoofing attacks, which originate from replaying recorded and synthesized human-live voices. Referring to the source of malicious voices, attacks can be classified into two main types as follows.

Recording Attack: To deceive VAs, an adversary collects victim voices and replays them through electronic speakers. VAs may be unable to distinguish human-live and spoofing samples, hence being manipulated by malicious commands. The victim voice can be readily obtained in multiple ways [11], [30], [31], such as conversations in daily life, online video conferences, and phone calls. To be specific, benefiting from easy acquisition of voice samples and without requiring customized devices, this recording way is currently one of the most practical.

Synthesizing Attack: In addition to directly recording authentic users' voices, the adversary leverages synthesizing technology to modify captured audios for deceiving VAs. Existing methods can be divided into three categories, namely, ultrasonic modulation [17], voice spectrum modification [10], and deepfake-driven spoofing [32]. For instance, DolphinAttack [17] modulates voice baseband signals onto a high-frequency carrier wave (i.e., larger than 20 kHz) to operate target devices; Wang et al. [10] carefully analyzed the nonuniform frequency response caused by loudspeaker hardware defects before launching attacks, and then modified spectrums to "reproduce" human-live voices. Nevertheless, the synthesizing-based approaches require an adversary to devote enormous efforts to exploring hardware characteristics and configuring cumbersome environments.

Consistent with previous works, we assume that an adversary has obtained victim voices and launches attacks without spatiotemporal restriction. In LiveProbe, we verify security protection performance under potential threats in Section VI-D, including modulated spectrum [10], DolphinAttack [17], and deepfake-driven spoofing [32] attacks.

IV. LiveProbe Overview

The basic idea of LiveProbe is to detect spoofing samples by comparing newly inputting audios with previously registered profiles of users with claimed identities. As depicted in Fig. 5, it consists of four modules and ten data processing units. In the *Registration Phase*, users utter phonemes to register personal identity under operation guidance. Then, LiveProbe removes the background noise and detects effective voice inputting events. In *Spectrum Segmentation*, denoising audio signals are segmented into phonemes and we abstract critical frames of each phoneme by removing redundant parts. Subsequently, fine-grained features, including statistics, local, and wave characteristics, are extracted to present liveness traits in *Feature Extraction*. Finally, these registered features on behalf of user identity are stored in our database.

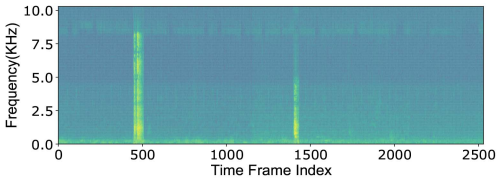
The usage period shares similar process modules with registration phase except for adding *Critical Frame Matching* and ERP&ERVP *Reconstruction* units. As obtaining newly inputting feature vectors, LiveProbe requires to match it with the most similar one in registered profiles. Moreover, due to variations in environment factors, we leverage an MODWT-based approach to reconstruct the polluted ERP/ERVP and hence ensuring the consistent of matched feature pairs. Finally, we feed these pairs into the trained support vector machine (SVM) to judge whether they are belonging to the same user in *Detection Model*.

V. LiveProbe Design

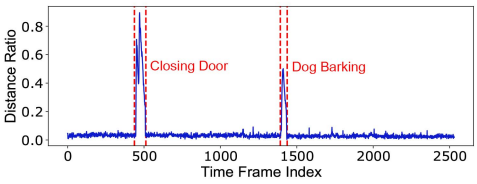
In this section, we introduce the technical details of LiveProbe from inputting voices to liveness feature extraction, and finally completing speaker-type detection.

A. Signal Preprocessing

Background Noise Removal: Environment noise carried by electromagnetic interference and daily activities is ubiquitous in human daily lives. Thus, voices perceived by microphones are always polluted to some extent. For mitigating its impacts on the following voice analysis, denoising operation on original inputting voices is a critical step. In terms of spectrum energy variation patterns, noise can generally be divided into two types, namely, steady and unsteady modes. Nevertheless, the second one imposes negligible impacts on liveness detection performance due to sustaining for only a short time compared with human voice and being filtered in the *Invalid Command Filtering* module. Subsequently, we only focus on eliminating the steady noise. Considering that noise affects frequency bands covered by human voices is nonuniform and we have to subtract spectrum increment brought by it in each frequency band. To solve this issue, we leverage multi-band spectral subtraction [33] to enhance target command voices disturbed by additive noise, and simultaneously without introducing new distortion. Note that the premise of successful



(a)



(b)

Fig. 6. Background noise patterns from a living room. (a) Spectrum of background noise. (b) Euclidean distance ratio between 1st frame and subsequent ones.

denoising is background noise being in steady state. We collect voice for 30 s as noncommand inputting in a 65-m^2 restroom. Fig. 6(a) depicts background noises always keeping steady spectrum patterns except for unsteady ones emerging such as sounds of the closing door and a dog barking. Moreover, we further measure the steady state by calculating Euclidean distances among frames as shown in Fig. 6(b), which displays that noises in steady state have consistent distance ratios and hence are stable. Relying on this observation, we acquire an enhanced spectrum of inputting voice in the mth frequency band of data segmentation X_n

$$\left|\hat{x_n^m}\right|^2 = \left|x_n^m\right|^2 - \alpha_m \beta_m \left|\psi_n^m\right|^2 \tag{3}$$

where $x_n^{m\wedge}$ is the energy pattern of denoising voice, ψ_{n}^m is the estimated noise, and m is the frequency band index. α_m is an over-subtraction weight and β_m is empirically chosen leveraging 1-s noninputting audio. α_m is updated as $\lambda_1 \cdot \log_{10}([|x_{\mathrm{n}}^m|^2/|\psi_{\mathrm{n}}^m|^2]) + \lambda_2$. The details of λ_1 and λ_2 value selection can refer to [33]. After the nonuniform spectrum energy subtraction, the enhanced voice version is derived from original inputting. Note that our noise removal approach acts as a complementation module of built-in noise suppression technologies of COTS devices, to jointly remove noise interference.

Liveness Detection Activation: For IoT devices equipping with built-in VAs, their microphones silently monitor inputting behind the scenes. Once voice inputting audios rather than background noise are detected, signal processing modules turn into running states and then complete speaker authentication/command recognition. As new voices present, captured signal energy bursts in the time domain and thus we employ a threshold-based approach to detect them. LiveProbe calculates the mean energy of samples in a 25-ms window. If its energy is higher than the preset threshold $u + 3 \cdot \sigma$ referring to normal distribution [34], LiveProbe immediately activates subsequent voice processing modules. u and σ are the mean and standard deviation of sample energies in noncommand inputting windows, respectively.

Invalid Command Filtering: Existing VAs on IoT devices activate voice-enabled services after verifying wake-up words like "Hi, Siri." Subsequently, LiveProbe implements liveness detection during interaction sessions no matter what users speak. However, valid user commands and unsteady noises like pet barking always co-exist in one space. Therefore, LiveProbe conducts a filtering operation on original voices to remain only effective inputting, to avoid costing extra computing resources. We first obtain daily human commands and steady noise audios in public data sets [35], [36]; their labels are set as 1 and 0, respectively. Moreover, we collected two-class audios in daily life as supplements. The number of positive and negative samples are both 1300. Then, common LPCC and statistical features (i.e., standard deviation, mean absolute deviation maximum, standard deviation, variance, mean, and entropy) are leveraged to represent two-class audio characteristics. Finally, considering the excellent performance of the SVM in the small-size sample classification tasks [37], we select it as a classifier to complete the validness judgment of original inputting audios. Model parameters called radial basis function (RBF) and class weight are set to "nonlinear kernel" and "balanced." SVM is trained offline and hence time efficient.

B. Spectrum Segmentation

Word and Phoneme Determination: Phonemes determination is the basis for further extracting ERP/ERVP from original inputting commands. In LiveProbe, this process is divided into two steps. We first utilize an offline automatic speech recognition model CMUSphinx [38] to identify words in inputting commands. Due to each word consisting of fixed phonemes, our task is to match critical frames and their features with corresponding phonemes. Following that, LiveProbe obtains a feature vector, including 12 MFCC coefficients and its new delta version [39] from each frame. To recognize each phoneme from its framewise feature vector, we trained a DNN model with two hidden layers and each with 300 units; since TIMIT[2] consists of 61 phonemes, we construct 3-states HMM with for each phoneme and hence 183 classes altogether. All the labels of frames are obtained by an alignment from a joint HMM-DNN system. For the DNN module, parameters are set as follows: stochastic gradient descent is utilized with a mini-batch size of 128; the learning rate is 0.1 for each mini-batch; cross-entropy is selected as a training criterion; the sigmoid is set as activation function. In Fig. 7, we present the likelihood of each phoneme state of another popular wake-up word "Alexa." In LiveProbe, we take the frame with the highest probability as the first critical frame. All remaining ones having likelihoods larger than experimentally acquired threshold 0.08 are regarded as alternative keyframes.

Critical Time Frame: LiveProbe leverages a loop to determine other critical frames in turn in this unit. Each phoneme consists of multiple time frames, and adjacent parts share similar spectrum information. Therefore, it is needed to prune

[2]English speech contains a limited set of 61 basic phonemes, as described in the TIMIT Acoustic Speech Corpus [35].

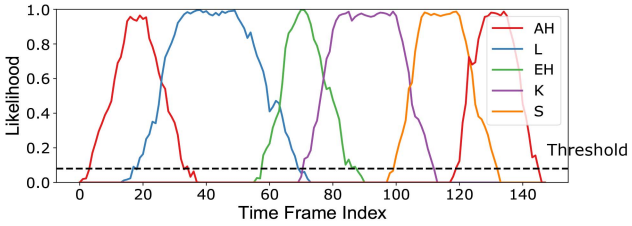


Fig. 7. Phoneme recognition employing an HMM-DNN model.

redundancy frames and retain only critical ones, which reduces the computational resource consumption and meanwhile represents liveness information. The most direct and effective way to find critical frames is by calculating the similarity of any two frames. High similarity means that the information they contain is duplicated and hence just retaining one of them is reasonable. We first put the 1st frame f_1 of a phoneme into the set S_{frame} and choose the latest frame in S_{frame} as *current frame*. Then, we calculate the similarity between retaining frames and the current one. Finally, adding the frame to S_{frame} if its cosine similarity is small. LiveProbe loops this process until all critical frames are added in S_{frame}.

C. Feature Extraction

In this section, LiveProbe rapidly and accurately matches inputting voice and registered profiles in *Critical Frame Matching*. Subsequently, we introduce the process of choosing the signal part with high robustness when environment factor changes, for completing ERP/ERVP reconstruction. Finally, LiveProbe extracts effective fine-grained liveness features and then feeds them into our classifier model for distinguishing speakers.

Critical Frame Matching: On the one hand, voice-enabled devices always support multiple users to register for expanding usage base, hence increasing data storage cost. On the other hand, obtaining more voice samples about authentic users means being more comprehensive to characterize identity information; thus we add legitimate samples after each successful liveness detection to the database (as described in Section V-D). The above two steps undoubtedly take increments in data matching time and hence a compromise for user experience. To get out of this dilemma, we utilize a quick and accurate data matching approach relying on LSH and its LSH Function Family [24] is defined as follows: $\Psi = \{\ell : \Re \to \mu\}$ is expressed by $(D, cD, \theta_1, \theta_2)$, which is sensitive for any $\rho_1, \rho_2 \in \Re$:

1) $\|\rho_1, \rho_2\|_s \le D$ then $\Pr_\Psi[\ell(\rho_1) == \ell(\rho_2)] \ge \theta_1$;
2) $\|\rho_1, \rho_2\|_s \ge cD$ then $\Pr_\Psi[\ell(\rho_1) == \ell(\rho_2)] \le \theta_2$.

where $\|\rho_1, \rho_2\|_s$ is the distance of feature vectors ρ_1 and ρ_2 and $\Re$ is the domain of all feature vectors. In LSH, c is larger than 1 and $\rho_1 > \rho_2$. Given a feature vector pair μ, we project two vectors in a hash table by the Euclidean distance-based LSH function $h_{a,b} : \mathbb{R}^n \to \mu$

$$h_{a,b}(v) = \left\lfloor \frac{a^T v + b}{W} \right\rfloor \tag{4}$$

where a is a random vector with all elements chosen independently from a Gaussian distribution and W is the width of each hash bucket. b is a real number randomly selected from the bucket interval $[0, W)$. As two vectors belong to the same speaker, they have identical hashed addresses and hence leading to a bucket collision in hashing. Referring to [24], the probability of ρ_1 and ρ_2 collision for the defined LSH in (4) is

$$\Pr_{a,b}\left[h_{a,b}(\rho_1) = h_{a,b}(\rho_2)\right] = \int_0^W \frac{1}{d}\chi_s\left(\frac{t}{d}\right)\left(1 - \frac{t}{W}\right)dt \tag{5}$$

where $\|\cdot\|_s$ measures the distance of two vectors, and $\chi_s(\cdot)$ is the probability density function of s-stable distribution. We can learn from (5) that with a distance W, the probability becomes larger when d is smaller. Jointly considering (4) and (5), the LSH project similar feature vectors into the same hash bucket. In LiveProbe, we input the matched vectors into ERP/ERIP reconstruction module for restoring polluting parts induced by environment factor changes.

ERP/ERVP Reconstruction: The main environment factor is relative position changes between users and voice-enabled devices, which makes ERP/ERVP differences across inputting voices and registered ones becoming larger, hence rejecting successful detection of authentic users. The forming process of these differences is complex and unpredictable, determined by the multipath effect and energy attenuation. To overcome its interference, we first employ MODWT [40] to perform multiscale and high-resolution difference analysis of sensed signals at multiple positions. MODWT essentially decomposes original signals into multiple scales containing detail υ_k and approximation ς_k coefficients in frequency bands as follows:

$$\upsilon_k^{(L)} = \sum_{z \in Z} x_z \bar{g}_{n-2^L L_k}^{(L)} \tag{6}$$

$$\varsigma_k^{(l)} = \sum_{z \in Z} x_z \bar{h}_{n-2^l L_k}^{(l)} \tag{7}$$

where L is the number of decomposition levels (set as 8 in LiveProbe) and $l \in \{1, 2, \ldots, L\}$. By experiment analysis, distinct positions present υ_k and ς_k variations only in a few bands. We attribute this case to a reason: the energy distribution of human voice is concentrated in a few frequency bands, and they can still ensure the dominant status on energy distribution even if the position changes. Based on this observation, we calculate relative coefficient changes of inputting and registered ones at each level (making $\upsilon_k^{(1)}$ as an instance)

$$\text{diff} = \frac{\text{sum}\left(\upsilon_k^{(1)} - \overline{\upsilon_k^{(1)}}\right)}{\text{sum}\left(\upsilon_k^{(1)}\right)} \tag{8}$$

where $\overline{\upsilon_k^{(1)}}$ is the detail coefficient of an inputting frame. Then, we replace coefficients of the inputting data using the registered one when their relative changes *diff* larger than 0.86. This empirical choosing is decided by our experiments as described in Section VI-E. Finally, the coefficients of all levels are obtained, and the inverse transformation is used to complete time signal reconstruction using current coefficients. Fig. 8 displays ERPs of one inputting voice frame captured from four positions (i.e., 0°&1 m, 90°&1 m, 0°&2 m, and 90°&2 m) relative the device HUAWEI Meta30 Pro and their reconstructed

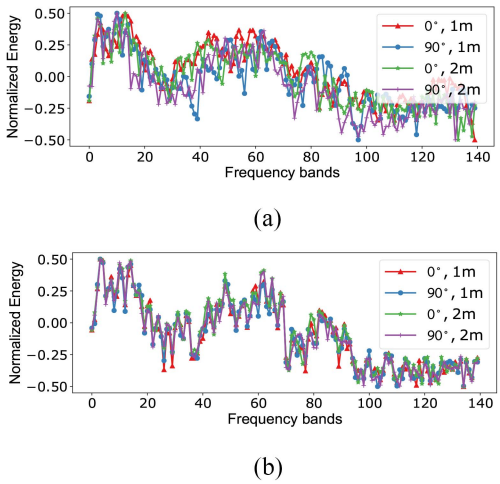


Fig. 8. Original ERPs of one critical frame in /a/ depicted in (a) and their reconstructed versions presenting in (b).

versions. We can observe that they share consistent envelopes even if relative positions change.

Fine-Grained Feature Extraction: After obtaining reconstructed ERP/ERVP, fine-grained features are extracted to represent uniquely static articulator structure and dynamic human vocal organ adjustment. LiveProbe completes each ERP's feature extraction from envelopes in both time (signal morphology) and frequency (signal composition) domains. We first focus on statistic features due to their stability on local anomalies, which capture value/distribution properties of ERP envelope. To be specific, LiveProbe applies skewness to measuring the symmetry of value distribution' left and right areas, kurtosis to estimating the tailedness differences compared to a normal distribution, and crest factor to counting the extreme peak significance for one distribution. Following that, we divide 140 frequency bands into 14 subblocks and calculate their local features. This operation is inspired by that the envelope and amplitude across frequency bands are significantly different as shown in Figs. 3 and 4, thus their contributions to the uniqueness of ERP/ERVP are distinct. The local feature vector consists of mean absolute value, median absolute deviation, variance, entropy, power, interquartile range, and root mean square. Furthermore, we search for the local minimums to locate and segment each wave cycle in ERP envelopes and then extract features as the aforementioned local part. To sum up, the feature vector of each ERP covers three components, that are statistics, local, and wave characteristics. Considering that LiveProbe is mounted on smart terminals with limited computation resources and traditional machine learning models cannot handle high-dimensional data, we choose cost-effective feature parts to feed into our detection model. To achieve this goal, we leverage principal component analysis [41] that is a common way to remove data redundant information for complete dimensionality reduction and retain top-71 feature for each ERP. We also visualize final feature vectors from five speakers in 2-D space using t-SNE [42] in Fig. 9. We can observe that the distributions of human and electronic speakers are significantly diverse, indicating the effectiveness of our feature extraction and dimensionality reduction.

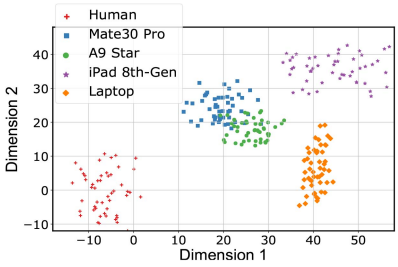


Fig. 9. Feature vector distributions of five speakers in 2-D space.

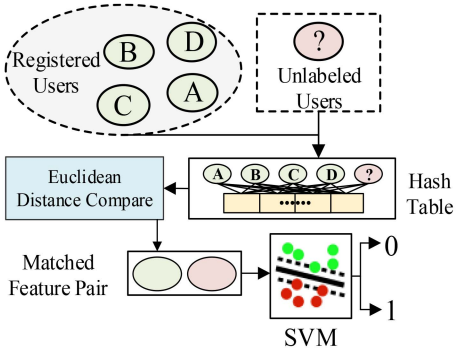


Fig. 10. Structure of the liveness detection model in LiveProbe (features of one phoneme's critical frames as inputting).

D. Detection Model

Compared to existing common one-time mechanisms, LiveProbe's detection model needs to meet extra two goals: 1) ensuring user experience when detection failures occur (not caused by spoofing samples) in a few frames and 2) securely utilizing newly adding authentic samples that may dilute the similarities of registered profiles. Fig. 10 displays the structure of our liveness detection model, making four registered users an example. Subsequently, we introduce its workflow after receiving inputting feature vectors corresponding to critical frames of one phoneme. Due to multiple user data stored in our database, LiveProbe leverages the LSH function to project inputting vectors into hash buckets. Then, the Euclidean distances between unlabeled and registered users' features are obtained. A registered user has the smallest distance (i.e., the largest similarity) is selected as the inputting vector's candidate identity and his/her features combined with inputting ones are formed feature pairs. Finally, they are fed into trained SVM, and the detection result is output as 0/1. The role of SVM is to judge if two feature vectors of one pair belong to the same user.

Goal One: Voice signals are sensitive to surrounding environments and thus detection failure inevitably occurs even inputting authentic user voices. For LiveProbe, ensuring the successful passing of authentic samples is critical to user experience. Inspired by the decision voting mechanism [25], our model adequately considers the detection results of multiple frames in each phoneme to improve model robustness. If the number of outputting detection results 1 is more than 0, current users are authentic, and vice versa.

Goal Two: In common cases, the number of registered samples determines the comprehensiveness of representing user

identity information. Nevertheless, LiveProbe only requires users to utter a few voice commands to shorten registration times and the total expenditure takes about 2 min. This case makes our method obtaining not enough registration data and thus LiveProbe chooses to add successfully verified inputting feature vectors to the database after each liveness detection. Note that newly added samples should not dilute the similarities of existing registration ones, for ensuring system security. Thus, we first obtain the averaging values of all dimensions in registration features belonging to each phoneme and thereby obtain the centroid feature vector. Then, LiveProbe calculates the Pearson correlation coefficient (PCC) [43] of newly inputting and centroid vectors and then adds inputting samples to the database if their PCC is larger than the experimentally empirical similarity threshold 0.9.

SVM Training: The role of SVM in LiveProbe is to judge two matched feature vectors whether belonging to the same user. We traverse every register feature vector and select five nearest ones in the hash bucket while their PCC larger than 0.9, labeled as 1. Then, the same number of spoofing samples are random selected and combine with registered ones to form feature vectors, labeled as 0. After feeding these samples into initial SVM, it leverages the standard grid search method to adjust parameters, i.e., loss: {epsilon_insensitive, squared_epsilon_insensitive}, tol: {0.2, 0.15, 0.1, 0.05, 0.01}, max_iter: {200, 500, 1000}.

VI. Implementation and Evaluation

In this section, we introduce experiment setups, such as utilized IoT device types and performance evaluation environments. Subsequently, the process of constructing human voice and spoofing sample data sets is described. Finally, we display overall liveness detection performance, the effectiveness of LiveProbe to defend potential attacks, and the impacts of experiment factor adjustment.

A. Experiment Setup

LiveProbe collects registered voices of authentic users in a quiet office (with background noise about 20 dB) and then evaluates detection performance at a living room of personal residences (about 30 dB) and a standard lab (about 45 dB) at our school. HUAWEI Mate30 Pro is denoted as the legal recording device and authentic users hold it with habitual positions to complete registered voice inputting. Other devices, including smartphones (Samsung A9 star, Apple IPhone12, and HUAWEI P30), loudspeakers (Newmine BT51, Baidu DuSmart, SONY SRS-XB13, and JBL GO2), and tablet PC (HUAWEI MatePad Pro and Apple iPad 8th-Gen), are leveraged to replay spoofing audios. Relying on real usage scenarios, users adjust the relative distances between speakers and devices from 1 to 4 m with an interval of 1 m. For the built-in microphones of these devices, we set the default sampling rate as 44.1 kHz. For obtaining every detection result, LiveProbe inputs all matched frames of one phoneme. Last but not least, we do not apply any time and space limitation to attackers to fully measure LiveProbe security.

B. Data Collection

In this section, we introduce the data set construction of LiveProbe, which consists of two parts, i.e., self-collected house-in and public.

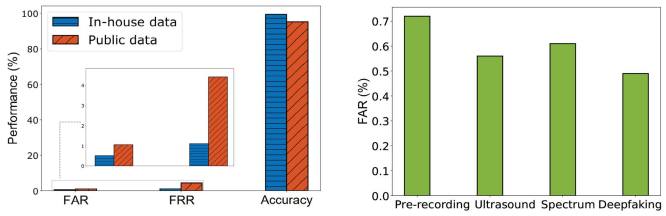
House-in Data Collection: There are 24 users (12 males and 12 females) participating in the experiments whose ages ranged from 22 to 34. They are undergraduate/graduate students recruited by our institute. Before data collection, we inform them about the purpose of LiveProbe and ask them to emit voices as usual for operating VAs. Participants can freely select commands from full list[3] of the Apple VA. Each of them chooses 45 commands from 15 categories in the personal office, and 15 commands are used as registration profiles while others as verification ones. Each user also offers 30 commands in the residence and lab, respectively, for evaluating detection performance across environments. Finally, the human voice data set contains commands with a total number amounting to 2520 (i.e., $15 \times 3 \times 24$ and $15 \times 2 \times 24 \times 2$) commands. Moreover, we obtain spoofing commands by replaying collected human voices employing nine electronic speakers and recording them with the same smartphone Meta30 Pro. By adjusting the relative distance of 1, 2, 3, and 4 m, every device replays randomly selected 15 verification commands of all participants. We totally obtain 12 960 (i.e., $4 \times 9 \times 15 \times 24$) spoofing voice samples by the self-collected way. Note that as verifying the effects of experiment setting parameters on detection performance, we will collect additional data as supplement parts that are described in the corresponding sections.

Public ASVspoof 2017 Data Set: To comprehensively evaluate LiveProbe performance, we also utilize human voices and spoofing samples of a public replaying attack database ASVspoof 2017. The spoofing voice samples collected span about 170 sessions with distinct environment settings. It employs 26 malicious devices consisting of professional audio equipment, smartphones, tablets, etc. We select a subdata set named "Evaluation" to test LiveProbe, due to it owning the largest number of samples (i.e., 1298 registered profiles and 12 008 spoofing ones from 24 speakers). After filtering unrecognized/low-similarity parts, we finally obtain 952 registered and 10 249 spoofing samples. 50% of human voices is leveraged to register identity information and the remaining part to verify liveness detection performance.

C. Metrics

We apply three metrics to evaluate the experiment performance of LiveProbe. FAR measures the rate of a spoofing voice wrongly accepted by LiveProbe and being classified as a legal identity. FRR presents the rate of authentic samples falsely rejected by the liveness detection and regarded as a negative label. *Accuracy* is on behalf of the overall probability that our approach accurately accepts authentic and rejects spoofing samples. One satisfactory liveness detection mechanism should keep a high accuracy while low FAR and FRR.

[3]https://www.insightcruises.com/pdf/mm17_pdfs/LeVitus/Full_list_Siri_Commands.pdf



(a) (b)

Fig. 11. Liveness detection performance of LiveProbe on house-in and public data sets in (a), while the FARs under three potential attacks in (b).

D. Overall Performance Evaluation

Verifying Overall Liveness Detection Performance: We verify the performance of LiveProbe in two data sets, which are house-in and public. The voice samples of all users and electronic devices are feeding into our models and we compare their output labels with the true ones. Fig. 11(a) depicts the averaging values of three metrics. In self-collected data, LiveProbe owns the FAR of 0.51%, FRR of 1.12%, and Accuracy up to 99.17%. Nevertheless, FRR is increased to 4.42% as inputting public data, and FAR is increased by 0.56%. By analyzing the original voice signals, we reveal the reason as follows: compared with in-house data, users utter registered voice phrases much more quickly in the public one and LiveProbe cannot obtain enough time frames and liveness features to represent speaker-type information. Fortunately, although the performance evaluated by ASVspoof 2017 is a little worse, it still owns satisfactory security and user-friendliness. Moreover, LiveProbe can further improve the accuracy by combining labels of multiple phonemes as described in *Impact of Utilized Phoneme Amount*.

Detecting Potential Attacks: Replaying attacks is regarded as the most practical way to spoof existing user authentication/liveness detection systems. These malicious voice samples are always from four sources (i.e., prerecording authentic user voices in public scenarios/private areas, modulating baseband signals to frequency ranges greater than 20 kHz for ultrasound versions [17], modifying collected voice spectrum to relieve the distortion induced by electronic speakers [10], and generating spoofing fake samples by deep learning [32]) and all launched by replaying way. We regard 15 verification samples of each registered user as original voices, then modulate and modify them referring to [10] and [17], while generating 1000 faking samples by [32]. Finally, we emit these spoofing voices and detect them by LiveProbe. The FARs under four attacks are shown in Fig. 11(b) and the simplest replaying prerecording voices is 0.71 in surprise, while others are 0.56, 0.62, and 0.49. By analyzing the working principles behind the last three ways, although trying to keep the initial characteristics of human voices, they inevitably introduce new distortion caused by nonlinear frequency responses of electronic devices and imperfect spectrum trim. In fact, the value of FAR should be lower because we currently have no time and space limitations for attackers, which is impossible in real scenes. To sum up, LiveProbe can ensure liveness detection security due to the averaging FAR being 0.63% as defending against potential attacks.

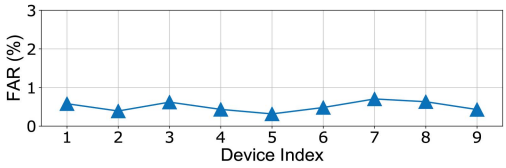
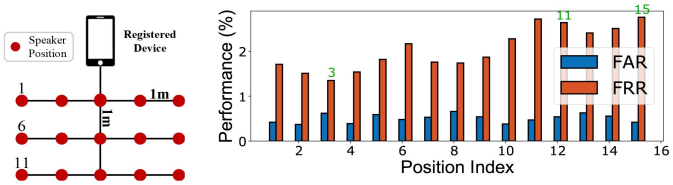


Fig. 12. FARs using nine different replaying devices.



(a) (b)

Fig. 13. Detection performance as speakers at different positions. (a) Illustration of speaker position. (b) FARs and FRRs on distinct positions.

E. Effects of Experiment Factor On Liveness Detection

Impact of Electronic Speaker Type: Electronic speakers replaying audio cause distinct degrees of distortion because of hardware defects. Thus, we investigate the detection performance by employing nine portable devices of smartphones (i.e., 1) HUAWEI Mate30 Pro; 2) Xiaomi Mi 10s; 3) OPPO Reno6; 4) Samsung Galaxy S7; 5) Samsung Galaxy S6), laptops (i.e., 6) Dell Inspiron 3511; 7) Lenovo New Air 14), and loudspeakers (i.e., 8) Newmine BT51; and 9) SONY SRS-XB12) to lunch attacks. The built-in microphones of these devices vary in size and quality, thus we leverage them to fully evaluate LiveProbe's performance. In Fig. 12, FAR values across devices own small fluctuation (i.e., 0.38%). The result indicates that employing common replaying loudspeakers presents consistent attack abilities and their types have little impact on detection accuracy.

Impact of Device Position: As we know, when relative position changes, emitted voices experience attenuation and multiple path effects, thereby making received signals vary. Thus, the same speaker launches command at different positions and may lead to inconsistent detection results. To overcome this issue, we have proposed ERP/ERVP *Reconstruction* in LiveProbe. In this section, we evaluate its effectiveness when launching voices at 15 positions as displayed in Fig. 13(a). There is a 1-m interval between any two adjacent positions. We let users or place electronic speakers at red spots in order and account corresponding FAR/FRR. By analyzing experimental results, we reveal that larger relative distances bring a slight FRR increase, because of energy attenuation affecting ERP/ERVP. For instance, the FRR values of 3rd and 15th positions are 1.32% and 2.76%. Nevertheless, such a small fluctuation takes negligible impacts on user experiences when joint liveness detection by multiple phonemes.

Impact of Utilized Phoneme Amount: For obtaining enough features to represent user identity and improve system robustness as failure judgment occurring in a few time frames, LiveProbe leverages feature vectors from multiple phonemes to complete liveness detection each time. We adjust the number of phonemes from 1 to 7 and then calculate the performance

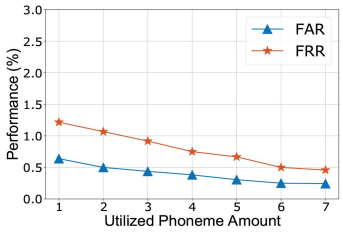


Fig. 14. Detection performance of FAR and FRR as using distinct amounts of phonemes.

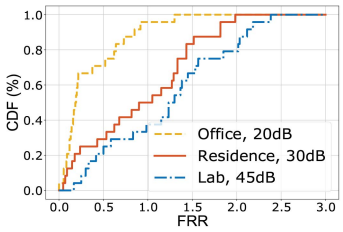


Fig. 15. FRRs under three environment noise levels.

of LiveProbe. As shown in Fig. 14, both FAR and FRR are decreased with increasing phoneme amount. When setting it as 7, FAR is 0.24% and FRR is 0.45%, presenting a satisfactory accuracy. Nevertheless, processing more phonemes each time brings large resources and time costs. Thus, choosing phoneme amount is a balance issue according to practical scenarios. In LiveProbe, the amount is selected as 4 relying on experiment results and computation resources.

Impact of Environment Noise Level: The liveness detection performance of LiveProbe in distinct noise environments is one critical indicator of measuring system robustness. Therefore, we let users register personal voices in a quiet office (20 dB) and then evaluate FRR in other two rooms, i.e., a living room (30 dB) and a lab (45 dB). Each user offers extra 30 voice commands in each environment in addition to the basic database. We can observe in Fig. 15 that FRR becomes large with noise decibel increasing, but the averaging difference value is only 0.28%. Thus, after the process of *Background Noise Removal* module, the effects of the background noise of common usage scenarios are significantly relieving.

Impact of ERP/ERVP Reconstruction Threshold: In ERP/ERVP *Reconstruction*, LiveProbe requires to set a threshold whether should be replaced to judge coefficients in the current frequency range. In fact, the threshold setting is the result of jointly considering security and user-friendliness of LiveProbe. To be specific, if it is small, lots of original information of inputting voice is discarded and hence the ratio of successful spoofing events becomes large; on the contrary, some authentic samples cannot pass the liveness detection because of a few differences between it and registered ones. We adjust the threshold from 0.70 to 1.00 with a step of 0.02 and the values of FAR/FRR are presented in Fig. 16. As it is 0.86, there are similar values of two metrics, which is the best choice to balance detection accuracy referring to the definition of Equal Error Ratio. Therefore, the default value of threshold is 0.86 in LiveProbe.

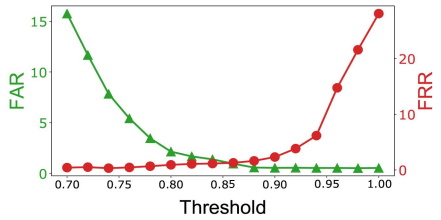


Fig. 16. Detection performance variation as adjusting the ERP/ERVP reconstruction threshold.

VII. Discussion and Limitation

Current audios are collected when devices and users co-locate in one room. Nevertheless, some users hope to manipulate devices in a long range such as crossing multiple rooms. In this case, voice signals will encounter severe multipath interference and energy attenuation, which induces ERP distortion. However, not only for liveness detection, the performance of all voice-related services are compromised in non-line-of-sight (NLOS) scenarios. Thus, we should track the latest progress of voice recognition in NLOS scenes to ensure the performance of liveness detection mechanism.

LiveProbe extracts liveness features relying on voice signals sensed by a built-in microphone. Nevertheless, current voice-enabled devices own different numbers of built-in microphones. For example, Bluetooth loudspeakers always have more than six ones. A large number of microphones means that the devices can obtain more voice signals propagating from multiple directions and positions to represent human liveness traits. Therefore, in future work, we should make LiveProbe compatible with distinct device types to further improve system implementability.

Except for verifying the effectiveness of typical attacks (i.e., recording, spectrum modulation, ultrasound injection, and deepfake types), a recent study [44] proposed laser injection for attacking VA systems, which can physically convert light to sound and make VAs conduct malicious commands. The implementation of this attack needs complicated device settings, including a telephoto lens, laser mount, audio amplifier, laser current driver, etc. Therefore, we fail to replicate such a complex experiment configuration. In future work, we will continue to replicate it and verify the effectiveness of LiveProbe in resisting it.

VIII. Related Work

The cases of spoofing VAs through replaying [45], synthesizing [10], and modulating [17] ways increasing the risk of privacy leakage, receive human continuous and extensive attention. Among them, the implementation strategy of replaying is simple and surprisingly effective, resulting in 31% equal error rate (EER) [46]. Therefore, researchers put significant efforts into building liveness detection modules to defend against this attack type. Existing detection works are usually divided into two categories classified by distinct security levels, namely one-time and continuous mechanisms.

The earliest studies verifying user identity always lack continuous protection measures during entire interaction sessions.

VoiceLive [9] proposes that human speaking motion brings dynamic time-difference-of-arrival changes reflected by audio signals, which is unique and only exists in human voices. Another study VoiceGesture [47] employs a smartphone to transmit high-frequency sounds from a built-in speaker and sense the reflections by a microphone, to represent Doppler shifts induced by human articulatory motion as liveness traits. Moreover, Shiota et al. [48] and Wang et al. [49] declared that distinguishing speaker types by detecting exhalation noises of human utterance is effective, while they need to keep mouth very close to microphones. Recent works also leverage extra sensing approaches, such WiFi [45], [50] and mmWave [13], [51] to capture user-specific oral organ movements induced by speaking. Expect for analyzing structural differences between electronic speakers and humans, some existing systems require users to reproduce/utter random actions [16] or phases [52] to ensure that they are authentic users. Compared with a continuous detection mechanism verifying user identity during interaction sessions, the above-mentioned one-pass methods possess lower security. Moreover, the collection of liveness traits used in them requires either exquisitely configuring environments or equipping with customized sensing sensors.

As is well known, after wake-up word activation, VAs backstage monitor voice inputting and execute corresponding instructions. Therefore, continuous authentication/liveness detection is critical for voice-enabled systems' user privacy protection. However, such research is just emerging and existing works still face challenges in moving toward practical scenarios. For instance, Lin et al. [53] declared that using continuous-wave radar can construct a user-specific high-resolution cardiac motion-sensing platform to continuously verify user identity. VAuth [12] installs an accelerometer on wearable devices like eyeglasses attaching to human skin for sensing voice-dependent body surface vibrations for ensuring the commands belonging to the owner. However, the above two works need customized sensors and hence cannot be directly applied to voice-enabled COTS devices. Furthermore, CaField [8] and EarArray [14] propose that a physical acoustic field is created as the sound propagates over the air, and energy attenuation level can readily distinguish speaker types. Recently, VibLive [54] transmits ultrasounds and receives reflected versions to obtain bone-conducted vibrations and air-conducted voices when users speak. Although these methods perform well on detection accuracy, they require speakers at preset/registered positions toward microphones across distinct liveness detection sessions and hence inevitably compromising user experiences.

Problem Definition: After reviewing the existing works, there are two critical problems that should be solved, which impede the implementation of desired voice liveness detection mechanism. On the one hand, wearing extra/customized sensors for feature collection can extremely compromise user experience. On the other hand, one-pass mechanisms cannot protect user privacy security during entire service sessions. To overcome the two challenges, LiveProbe improves the performance (i.e., security and user-friendliness) of continuous detection mechanisms in the following two aspects: first, the proposed ERP effectively representing structure differences between human articulator and electronic loudspeaker, can continuously track speaker type and hence ensuring high security; second, it only employs the built-in microphone for liveness trait collection without modifying devices; users need not carry any additional sensors hence user-friendliness.

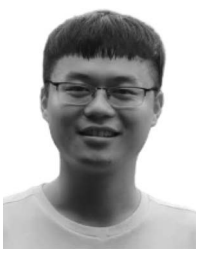
IX. Conclusion and Feature Work

In this article, we have proposed a continuous liveness detection mechanism named LiveProbe relying on unique ERPs for voice-enabled IoT COTS. To overcome the challenges induced by environment factor variation and limited computation resources, we have proposed a series of ingenious approaches. For instance, we have leveraged the MODWT-based method to reconstruct polluted voices in multiscale frequency bands and utilize the LSH-based to quickly match similar feature vectors. Finally, we have conducted experiments using in-house and public data sets to evaluate the detection performance of LiveProbe, with a satisfactory 0.51% FAR and 1.12% FRR. With recent advances in beamforming technology in voice recognition, IoT devices can capture high-quality and targeted human audio, thereby further improving existing liveness detection mechanism performance. Especially, inspired by the study focusing on AI for next generation computing [55], the edge computing technology can help devices with limited computation resources endow voice-driven services. Moreover, along with the progress of AI, the neural network structure and performance can be further optimized, which will promote the development of VAs.

References

[1] J. Hirschberg and C. D. Manning, "Advances in natural language processing," *Science*, vol. 349, no. 6245, pp. 261–266, 2015.
[2] "Amazon echo." Wikipedia. 2021. [Online]. Available: https://en.wikipedia.org/wiki/Amazon_Echo
[3] "Apple siri." WiKipedia. 2021. [Online]. Available: https://en.wikipedia.org/wiki/Siri
[4] Z. Guo, Z. Lin, P. Li, and K. Chen, "SkillExplorer: Understanding the behavior of skills in large scale," in *Proc. USENIX Security Symp.*, 2020, pp. 2649–2666.
[5] I.-Y. Kwak, J. H. Huh, S. T. Han, I. Kim, and J. Yoon, "Voice presentation attack detection through text-converted voice command analysis," in *Proc. ACM CHI*, 2019, pp. 1–12.
[6] "Number of digital voice assistants in use worldwide from 2019 to 2024." Statista. 2021. [Online]. Available: https://www.statista.com/statistics/973815/worldwide-digital-voice-assistant-in-use/
[7] "Global voice recognition market size 2020 and 2026." Statista. 2021. [Online]. Available: https://www.statista.com/statistics/1133875/global-voice-recognition-market-size/
[8] 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 CCS*, 2019, pp. 1215–1229.
[9] L. Zhang, S. Tan, J. Yang, and Y. Chen, "VoiceLive: A phoneme localization based liveness detection for voice authentication on Smartphones," in *Proc. ACM CCS*, 2016, pp. 1080–1091.
[10] S. Wang, J. Cao, X. He, K. Sun, and Q. Li, "When the differences in frequency domain are compensated: Understanding and defeating modulated replay attacks on automatic speech recognition," in *Proc. ACM CCS*, 2020, pp. 1103–1119.
[11] M. E. Ahmed, I. Kwak, J. H. Huh, I. Kim, T. Oh, and H. Kim, "Void: A fast and light voice liveness detection system," in *Proc. USENIX Security Symp.*, 2020, pp. 2685–2702.
[12] H. Feng, K. Fawaz, and K. G. Shin, "Continuous authentication for voice assistants," in *Proc. ACM MobiCom*, 2017, pp. 343–355.

[13] H. Li et al., "VocalPrint: Exploring a resilient and secure voice authentication via mmWave biometric interrogation," in *Proc. ACM Sensys*, 2020, pp. 312–325.
[14] G. Zhang, X. Ji, X. Li, G. Qu, and W. Xu, "EarArray: Defending against DolphinAttack via acoustic attenuation," in *Proc. ISOC NDSS*, 2021, pp. 1–14.
[15] P. Partila, J. Tovarek, G. H. Ilk, J. Rozhon, and M. Voznak, "Deep learning serves voice cloning: How vulnerable are automatic speaker verification systems to spoofing trials?" *IEEE Commun. Mag.*, vol. 58, no. 2, pp. 100–105, Feb. 2020.
[16] S. Chen et al., "You can hear but you cannot steal: Defending against voice impersonation attacks on smartphones," in *Proc. IEEE ICDCS*, 2017, pp. 183–195.
[17] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, "Dolphinattack: Inaudible voice commands," in *Proc. ACM CCS*, 2017, pp. 103–117.
[18] Z. Akhtar, C. Micheloni, and G. L. Foresti, "Biometric liveness detection: Challenges and research opportunities," *IEEE Security Privacy*, vol. 13, no. 5, pp. 63–72, Sep./Oct. 2015.
[19] L. Lu et al., "LipPass: Lip reading-based user authentication on smartphones leveraging acoustic signals," in *Proc. IEEE INFOCOM*, 2018, pp. 1466–1474.
[20] Y. Meng et al., "Your microphone array retains your identity: A robust voice liveness detection system for smart speakers," in *Proc. USENIX Security*, 2022, pp. 1077–1094.
[21] Y. Abdulaziz and S. M. S. Ahmad, "Infant cry recognition system: A comparison of system performance based on mel frequency and linear prediction cepstral coefficients," in *Proc. IEEE CAMP*, 2010, pp. 260–263.
[22] A. Bertrand and M. Moonen, "Energy-based multi-speaker voice activity detection with an ad hoc microphone array," in *Proc. IEEE ICASSP*, 2016, pp. 85–88.
[23] "Evaluate and improve custom speech accuracy." Microsoft. 2021. [Online]. Available: https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/how-to-custom-speech-evaluate-data
[24] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proc. 20th Annu. Symp. Comput. Geometry*, 2004, pp. 253–262.
[25] A. A. A. Zamil, S. Hasan, S. M. J. Baki, J. M. Adam, and I. Zaman, "Emotion detection from speech signals using voting mechanism on classified frames," in *Proc. IEEE ICREST*, 2019, pp. 281–285.
[26] T. Ito, E. Z. Murano, and H. Gomi, "Fast force-generation dynamics of human articulatory muscles," *J. Appl. Physiol.*, vol. 96, no. 6, pp. 2318–2324, 2004.
[27] O. Rioul and P. Duhamel, "Fast algorithms for discrete and continuous wavelet transforms," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 569–586, Mar. 1992.
[28] M. Aljasem et al., "Secure automatic speaker verification (SASV) system through sm-ALTP features and asymmetric bagging," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3524–3537, 2021.
[29] J. Ming, T. J. Hazen, J. R. Glass, and D. A. Reynolds, "Robust speaker recognition in noisy conditions," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 5, pp. 1711–1723, Jul. 2007.
[30] H. Malik, "Securing speaker verification system against replay attack," in *Proc. 46th Int. Conf. Audio Forensics, Audio Eng. Soc. Conf.*, 2012, p. 1.
[31] A. Javed, K. M. Malik, A. Irtaza, and H. Malik, "Towards protecting cyber-physical and IoT systems from single-and multi-order voice spoofing attacks," *Appl. Acoust.*, vol. 183, Dec. 2021, Art. no. 108283.
[32] Y. Gao, R. Singh, and B. Raj, "Voice impersonation using generative adversarial networks," in *Proc. IEEE ICASSP*, 2018, pp. 2506–2510.
[33] S. Kamath and P. C. Loizou, "A multi-band spectral subtraction method for enhancing speech corrupted by colored noise," in *Proc. IEEE ICASSP*, 2002, p. 4164.
[34] Y. L. Tong, *The Multivariate Normal Distribution*. New York, NY, USA: Springer, 2012.
[35] "TIMIT acoustic-phonetic continuous speech corpus." Linguistic Data Consortium. 2021. [Online]. Available: https://catalog.ldc.upenn.edu/LDC93s1
[36] "2000 HUB5 english evaluation speech." Linguistic Data Consortium. 2021. [Online]. Available: https://catalog.ldc.upenn.edu/LDC2002S09
[37] L. Zhou, T. Lin, X. Zhou, S. Gao, Z. Wu, and C. Zhang, "Detection of winding faults using image features and binary tree support vector machine for auto-transformer," *IEEE Trans. Transp. Electrific.*, vol. 6, no. 2, pp. 625–634, Jun. 2020.
[38] "Open source speech recognition toolkit." CMU. 2015. [Online]. Available: https://cmusphinx.github.io/
[39] X. Zheng, Z. Wu, H. Meng, and L. Cai, "Learning dynamic features with neural networks for phoneme recognition," in *Proc. IEEE ICASSP*, 2014, pp. 2524–2528.
[40] D. B. Percival and A. T. Walden, *Wavelet Methods for Time Series Analysis*, vol. 4. Cambridge, U.K.: Cambridge Univ. Press, 2000.
[41] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 2, no. 4, pp. 433–459, 2010.
[42] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.
[43] H. Y. B. Jacob, C. Jingdong, and C. Israel, "Pearson correlation coefficient," in *Noise Reduction in Speech Processing*. Dordrecht, The Netherlands: Springer, 2009, pp. 1–4.
[44] T. Sugawara, B. Cyr, S. Rampazzi, D. Genkin, and K. Fu, "Light commands: Laser-based audio injection attacks on voice-controllable systems," in *Proc. USENIX Security*, 2020, pp. 2631–2648.
[45] P. Swadhin, S. Wei, B. Ghufran, and Q. Lili, "Combating replay attacks against voice assistants," *ACM Ubicomp/IMWUT*, vol. 3, no. 3, pp. 1–26, 2019.
[46] T. Kinnunen, M. Sahidullah, H. Delgado, and M. Todisco, "The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," in *Proc. Interspeech*, 2017, pp. 1–5.
[47] 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 CCS*, 2017, pp. 57–71.
[48] S. Shiota, F. Villavicencio, J. Yamagishi, N. Ono, I. Echizen, and T. Matsui, "Voice liveness detection algorithms based on pop noise caused by human breath for automatic speaker verification," in *Proc. ISCA Interspeech*, 2015, pp. 239–243.
[49] Q. Wang et al., "VoicePop: A pop noise based anti-spoofing system for voice authentication on smartphones," in *Proc. IEEE INFOCOM*, 2019, pp. 2062–2070.
[50] Y. Meng et al., "WiVo: Enhancing the security of voice control system via wireless signal in IoT environment," in *Proc. ACM MobiHoc*, 2018, pp. 81–90.
[51] H. Li et al., "VocalPrint: A mmWave-based unmediated vocal sensing system for secure authentication," *IEEE Trans. Mobile Comput.*, vol. 22, no. 1, pp. 589–606, Jan. 2023.
[52] "VocalPassword." Nuance. 2015. [Online]. Available: http://www.nuance.com/ucmprod/groups/enterprise/@webenus/documents/collateral/nc_015226.pdf
[53] F. Lin, C. Song, Y. Zhuang, W. Xu, C. Li, and K. Ren, "Cardiac scan: A non-contact and continuous heart-based user authentication system," in *Proc. ACM MobiCom*, 2017, pp. 315–328.
[54] L. Zhang, S. Tan, Z. Wang, Y. Ren, Z. Wang, and J. Yang, "VibLive: A continuous liveness detection for secure voice user interface in IoT environment," in *Proc. ACM ACSAC*, 2020, pp. 884–896.
[55] S. S. Gill et al., "AI for next generation computing: Emerging trends and future directions," *Internet Things*, vol. 19, Aug. 2022, Art. no. 100514.



Hangcheng Cao (Student Member, IEEE) is currently pursuing the Ph.D. degree with the College of Computer Science and Electronic Engineer, Hunan University, Changsha, China.

From 2021 to 2022, he studies as a joint Ph.D. student with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. He has published papers in ACM Ubicomp/IMWUT 2020, IEEE ICDCS 2021, ACM MobiCom Workshop 2022, and IEEE TMC. His research interests lie in the area of IoT security, particularly user authentication on smart devices, side-channel attacks, and data anomaly detection.

Mr. Cao served as the reviewer for IEEE TRANSACTIONS ON MOBILE COMPUTING, *ACM Transactions on Sensor Networks*, and *Wireless Networks*.

Hongbo Jiang (Senior Member, IEEE) received the Ph.D. degree from Case Western Reserve University, Cleveland, OH, USA, in 2008.

He is currently a Full Professor with the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. He was a Professor with Huazhong University of Science and Technology, Wuhan, China. His current research focuses on computer networking, especially, wireless networks, data science in Internet of Things, and mobile computing.

Prof. Jiang has been serving on the editorial board of IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE TRANSACTIONS ON MOBILE COMPUTING, *ACM Transactions on Sensor Networks*, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, and IEEE INTERNET OF THINGS JOURNAL. He was also invited to serve on the TPC of IEEE INFOCOM, ACM WWW, ACM/IEEE MobiHoc, IEEE ICDCS, and IEEE ICNP. He is an Elected Fellow of The Institution of Engineering and Technology, a Fellow of The British Computer Society, a Senior Member of ACM, and a Full Member of IFIP TC6 WG6.2.

Daibo Liu (Member, IEEE) received the Ph.D. degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2018.

He was a Visiting Researcher with the School of Software, Tsinghua University, Beijing, China, from 2014 to 2016, and the Department of Electrical and Computer Engineering, University of Wisconsin–Madison, Madison, WI, USA, from 2016 to 2017. He is currently an Assistant Professor with the College of Computer Science and Electronic Engineering, Hunan University, Changsha, China. His research interests cover the broad areas of low-power wireless networks, mobile and pervasive computing, and system security.

Dr. Liu is a member of ACM.

Ruize Wang received the B.S. degree from Xiangtan University, Xiangtan, China, in 2021. He is currently pursuing the master's degree with the College of Computer Science and Electronic Engineer, Hunan University, Changsha, China.

His current research focuses on wearable-based sensing, including human activity recognition, localization, and authentication.

Geyong Min (Member, IEEE) received the Ph.D. degree in computing science from the University of Glasgow, Glasgow, U.K., in 2003.

He is a Professor of High-Performance Computing and Networking with the Department of Computer Science, College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter, U.K.

Dr. Min was awarded the Outstanding Leadership Awards from IEEE International Conferences HPCC'2012, IUCC'2011, CIT'2010, ScalCom'2009, and HPCC'2008 and one Outstanding Service Award from ISPA'2006. He is an Editorial Board Member of eight international journals, including IEEE TRANSACTIONS ON COMPUTERS and serves as the guest editor for 18 international journals. He has chaired/co-chaired 30 international conferences/workshops and served as the committee member of 120 professional conferences/workshops and 100 papers in the leading international journals, including IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE NETWORK, IEEE INTELLIGENT SYSTEMS, and IEEE INTERNET COMPUTING.

Jiangchuan Liu (Fellow, IEEE) received the Ph.D. degree from The Hong Kong University of Science and Technology, Hong Kong, China, in 2003.

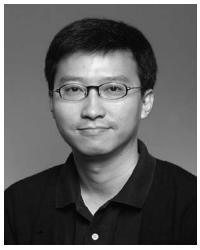
He is a University Professor with the School of Computing Science, Simon Fraser University, Burnaby, BC, Canada. He was an EMC-Endowed Visiting Chair Professor with Tsinghua University, Beijing, China, from 2013 to 2016. In the past, he worked as an Assistant Professor with The Chinese University of Hong Kong, Hong Kong, and as a Research Fellow with Microsoft Research Asia, Beijing. His research interests include multimedia systems and networks, cloud and edge computing, social networking, online gaming, and Internet of Things/RFID/backscatter.

Dr. Liu is a Steering Committee Member of IEEE TRANSACTIONS ON MOBILE COMPUTING and the Steering Committee Chair of IEEE/ACM IWQoS from 2015 to 2017. He is the TPC Co-Chair of IEEE INFOCOM'2021. He has served on the editorial boards of IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE TRANSACTIONS ON BIG DATA, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, and IEEE INTERNET OF THINGS JOURNAL. He is a Fellow of The Canadian Academy of Engineering and an NSERC E.W.R. Steacie Memorial Fellow.

Schahram Dustdar (Fellow, IEEE) received the Ph.D. degree in business informatics from the University of Linz, Linz, Austria, in 1992.

He is currently a Full Professor of Computer Science (Informatics) with a focus on Internet Technologies heading the Distributed Systems Group, TU Wien, Vienna, Austria.

Prof. Dustdar is a recipient of multiple awards, including the TCI Distinguished Service Award in 2021, the IEEE TCSVC Outstanding Leadership Award in 2018, the IEEE TCSC Award for Excellence in Scalable Computing in 2019, the ACM Distinguished Scientist in 2009, the ACM Distinguished Speaker in 2021, and the IBM Faculty Award in 2012. He is an Associate Editor of IEEE TRANSACTIONS ON SERVICES COMPUTING, IEEE TRANSACTIONS ON CLOUD COMPUTING, *ACM Computing Surveys*, *ACM Transactions on the Web*, and *ACM Transactions on Internet Technology*, as well as on the editorial board of IEEE INTERNET COMPUTING and IEEE COMPUTER. He is the Founding Co-Editor-in-Chief of *ACM Transactions on Internet of Things* as well as the Editor-in-Chief of *Computing* (Springer). He is an Elected Member of the Academia Europaea: The Academy of Europe, where he is the Chairman of the Informatics Section, as well as an Asia–Pacific Artificial Intelligence Association President in 2021 and a Fellow in 2021. He is an EAI Fellow in 2021 and an I2CICC Fellow in 2021. He is a member of the 2022 IEEE Computer Society Fellow Evaluating Committee in 2022.



John C. S. Lui (Fellow, IEEE) was born in Hong Kong. He received the Ph.D. degree in computer science from the University of California at Los Angeles, Los Angeles, CA, USA, in 1992.

He is currently the Choh-Ming Li Professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong (CUHK), Hong Kong, where he was the Chairman from 2005 to 2011. His current research interests are in communication networks, network/system security (e.g., cloud security and mobile security), network economics, network sciences (e.g., online social networks and information spreading), cloud computing, large-scale distributed systems, and performance evaluation theory.

Prof. Lui received various departmental teaching awards and the CUHK Vice-Chancellors Exemplary Teaching Award. He is also a co-recipient of the Best Paper Award in the IFIP WG 7.3 Performance 2005, IEEE/IFIP NOMS 2006, and SIMPLEX 2013. He has been serving on the editorial board of the IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, *Performance Evaluation*, and the *International Journal of Network Security*. He is an Elected Member of the IFIP WG 7.3, a Fellow of the Association for Computing Machinery (ACM), and a Senior Research Fellow of the Croucher Foundation, and was the Chair of the ACM SIGMETRICS.