Hidebehind: Enjoy Voice Input with Voiceprint Unclonability and Anonymity

Jianwei Qian
Illinois Institute of Technology
Chicago, IL
jqian15@hawk.iit.edu

Haohua Du
Illinois Institute of Technology
Chicago, IL
hdu4@hawk.iit.edu

Jiahui Hou
Illinois Institute of Technology
Chicago, IL
jhou11@hawk.iit.edu

Linlin Chen
Illinois Institute of Technology
Chicago, IL
lchen96@hawk.iit.edu

Taeho Jung
University of Notre Dame
Notre Dame, IN
tjung@nd.edu

Xiang-Yang Li
University of Science and Technology of China
Hefei, Anhui

ABSTRACT

We are speeding toward a not-too-distant future when we can perform human-computer interaction using solely our voice. Speech recognition is the key technology that powers voice input, and it is usually outsourced to the cloud for the best performance. However, user privacy is at risk because voiceprints are directly exposed to the cloud, which gives rise to security issues such as spoof attacks on speaker authentication systems. Additionally, it may cause privacy issues as well, for instance, the speech content could be abused for user profiling. To address this unexplored problem, we propose to add an intermediary between users and the cloud, named *VoiceMask*, to anonymize speech data before sending it to the cloud for speech recognition. It aims to mitigate the security and privacy risks by concealing voiceprints from the cloud. VoiceMask is built upon voice conversion but is much more than that; it is resistant to two de-anonymization attacks and satisfies differential privacy. It performs anonymization in resource-limited mobile devices while still maintaining the usability of the cloud-based voice input service. We implement VoiceMask on Android and present extensive experimental results. The evaluation substantiates the efficacy of VoiceMask, *e.g.*, it is able to reduce the chance of a user's voice being identified from 50 people by a mean of 84%, while reducing voice input accuracy no more than 14.2%.

CCS CONCEPTS

• **Security and privacy** **Pseudonymity, anonymity and untraceability**; **Data anonymization and sanitization**; • **Human-centered computing** *Ubiquitous and mobile computing*;

KEYWORDS

Voiceprint concealment; voice anonymization; voice anonymity; voice privacy; voice security; voice conversion.

ACM Reference Format:
Jianwei Qian, Haohua Du, Jiahui Hou, Linlin Chen, Taeho Jung, and Xiang-Yang Li. 2018. Hidebehind: Enjoy Voice Input with Voiceprint Unclonability and Anonymity. In *The 16th ACM Conference on Embedded Networked Sensor Systems (SenSys '18), November 4–7, 2018, Shenzhen, China.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3274783.3274855

SenSys '18, November 4–7, 2018, Shenzhen, China

ACM ISBN 978-1-4503-5952-8/18/11…$15.00
https://doi.org/10.1145/3274783.3274855

1 INTRODUCTION

Featuring hands-free communication, voice input has been widely applied in keyboard apps (*e.g.*, Google, Microsoft, Sougou, and iFlytek keyboards), voice search (*e.g.*, Microsoft Bing, Google Search), and artificial intelligence virtual assistants (*e.g.*, Apple's Siri, Amazon Echo) on a range of mobile devices. Voice input can greatly ease our lives by freeing us from the time-consuming work of typing on the small screens of mobile devices. It is also one of the major means of human-computer communication for people who are visually impaired. The key technology that powers voice input is speech recognition, also known as speech-to-text conversion, where the vocal input of spoken language is recognized and translated into text. Due to the resource limitation on mobile devices, the speech recognition is usually performed to the cloud server for higher accuracy and efficiency [11, 24], which are two key elements leading to a good user experience. As shown in Fig. 1, the cloud server receives users' speech data from their mobile devices, converts speeches to texts and sends the texts back to the mobile devices. Usually, the cloud server does not discard these speech signals afterwards but stores them in a database, and it may publish or trade them with third-parties for research or business purposes. Most of the existing voice input service providers collect their users' utterances (*i.e.*, speech records). This was validated by recent news and through private communication with several researchers working in the voice input industry. For example, Apple stores the data created when people use Siri and Dictation, two voice-driven services found on its mobile devices, for up to two years, and it also admits that such voice data is being sent to third parties [1]. Google saves our voice inputs to our Google accounts by default (it stores them anonymously if we turn off the Voice & Audio Activity feature) and it is not clear for how long they are stored [5].

We face serious **security risks** when our raw voice is exposed to the cloud. Once the cloud collects enough voice samples from a user, it can create an accurate voiceprint and use it to synthesize sound that mimics the user. Because of recent advancements in *speech synthesis* and *voice cloning*, it is becoming increasingly easy to accurately clone a target person's voice given only a few sample speeches for training [10, 15, 50]. It is very likely that this technology may be exploited for various malicious attacks. Voice as a type of biometric information has been widely used in emerging authentication systems to unlock smart devices, gain access to some apps like WeChat, authorize payment like Alipay, and activate virtual assistants like Apple's "Hey Siri" and Google's "OK Google." Synthetic speech can be used to spoof the voice authentication systems and gain access to the user's private resources. They can also be used for fraud, *e.g.*, to authorize bogus charges on the user's credit card [7]. Worse still, the adversary may produce illegal or indecent recordings to frame or blackmail the victim, *e.g.*, [2]. Therefore, voiceprints leaked from speech data can be very dangerous.

There are also two **privacy risks** in the cloud-based voice input services. Firstly, the cloud (or third parties who obtained the voice data from the cloud) may link the speech records to individuals in real life. Simply removing the IDs associated with the speech data is not enough to prevent these users from being de-anonymized. The cloud is still able to identify the speakers of unlabeled speech data via speaker recognition. If the cloud is able to collect some speech samples of the target person from other sources like YouTube and train a voice model of this person, then it can identify the records belonging to this person from the speech database, which leads to an identity privacy breach. Secondly, the cloud can analyze the speech content and learn more detailed information about the person, so identity privacy breach further leads to speech content privacy breach. The cloud may use natural language processing techniques to extract information from a user's voice search history, voice command history, and even SMS messages and emails if they were typed via voice input. Then, the cloud can paint a very accurate picture of the user's demographic categories, personal preferences, interpersonal communications, habits, schedules, travels, and so on. After a user is pinpointed in reality and her personal information is inferred, the follow-up attacks could be more specific and vicious, such as stalking or robbery.

Therefore, we believe it is very necessary to come up with countermeasures and stop the leak of privacy from its source, that is, our smart mobile devices. In this paper, our **goal** is to protect the *voiceprints* (voice biometrics) of voice input users from being disclosed while maintaining the user experience. Notice our main goal is to protect users' voiceprints and mitigate the security risks. We do not seek to guarantee complete identity privacy for users, but our work may be able to strengthen their identity privacy in certain circumstances, *i.e.*, when the speech data is not already associated with users' Personally Identifiable Information (PII). Today's voice input service providers mostly tag users' speech data with PII, so users' identities are inevitably disclosed to them for now. Yet, they usually prohibit their data analysts from seeing users' PII and remove PII when sharing data with third-parties. Our work can strengthen users' identity privacy by preventing them from de-anonymizing them through their voiceprints. Also, anonymous networks such as Tor [6, 8] may be more commonly used in the future to prohibit

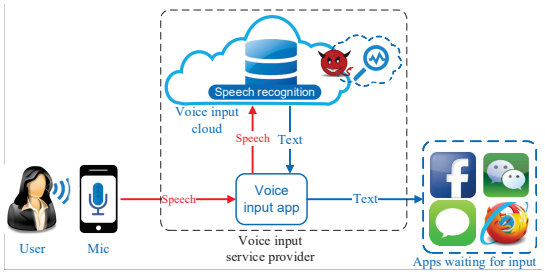


Figure 1: **The workflow of existing insecure cloud-based voice input system. (1) User's voice is sensed by the mic (2) The voice input app accesses the mic for the audio signal. (3) The app uploads the raw audio to the cloud. (4) The cloud carries out speech recognition and sends the text back to the voice input app. Usually the cloud and the app belongs to the same entity. (5) The input app forwards the text to other apps where user input is required.**

the cloud from obtaining PII. We will present more explanations in the discussion (§5). Our research not only directly protects users' voiceprints, but also benefits existing voice input service providers by helping them achieve better data security. Because no individuals will be identified if the data is abused by malicious employees or leaked to hackers, the chance will be lower that the service providers are held responsible for privacy leaks.

To prevent voiceprint leak, we are faced with several **challenges**. *Firstly*, speech as a type of unstructured data is hard to sanitize or anonymize. Unlike relational data and graph data [26], privacy policies like k-anonymity cannot be employed directly to protect voiceprints. Interactive protocols involving third parties [22, 23] does not apply either. *Secondly*, we have to sanitize users' speech data without degrading the accuracy of speech recognition to an unacceptable extent. Users should still be able to use high fidelity voice input. *Lastly*, it is hard to efficiently perform speech sanitization in real-time with the restricted resources on the mobile device. The computation overhead should be small enough to induce an acceptable level of latency for voice input apps. The sanitization process should also have a minimal power footprint.

The **naïve method** is to simply perform speech recognition locally. No privacy is leaked in this way, but the usability of offline speech recognition is limited. On one hand, the local speech models are updated less frequently. On the other hand, the cloud is unwilling to provide free offline speech recognition service and it tends to collect speech data as much as possible. For instance, Google Nexus allows us to use offline voice typing but only when we are disconnected from the Internet, which prohibits us from using voice input for many Internet-based services like social media, chat apps, email apps, and browsers. Thus, we cannot expect the cloud to provide unconditional offline speech recognition service for users. As a result, we need to come up with an alternative speech data anonymization mechanism that allows users to stay online.

Solution. To ensure the cloud has access to only the sanitized speech data, our basic solution is to introduce an intermediary, *i.e.*, "VoiceMask", to perturb the speeches. It acts as a module in the operating system (or a third-party voice input app). VoiceMask processes the audio signal received from the microphone and then sends the sanitized speech audio to the voice input apps (or the cloud). It disguises the speaker's voiceprint by randomly modifying the speaker's voice via *robust* voice conversion, which prevents the original voice from being recovered and satisfies differential privacy.

We carefully select tuning parameters to reach the optimal balance of identity privacy and user experience.

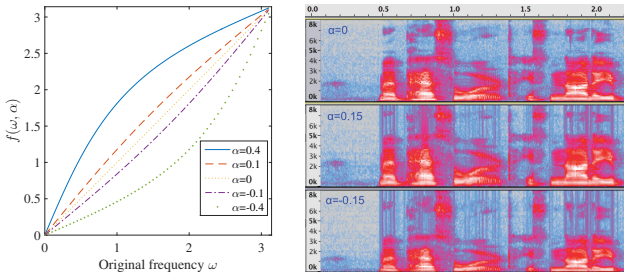
Contributions. To our knowledge, VoiceMask is the first privacy-preserving architecture for voice input on mobile devices. We first reveal the privacy risks in existing voice input apps, and propose two possible secure voice input architectures to prevent the cloud from learning users' voiceprints (§2). *Then*, we propose a technique to protect users' voiceprints from being leaked (§3). Improved over existing voice conversion techniques, VoiceMask is resistant to two potential de-anonymization attacks and guarantees differential privacy. *Finally*, we implement VoiceMask on Android and present an extensive evaluation (§4). The results demonstrate that it decreases the risk of the speaker being identified from 50 people by 84% while inducing only a 14.2% drop in the speech recognition accuracy. Meanwhile, VoiceMask induces little energy consumption and an acceptable delay. We also study the influence of external factors including ambient noise, device brand, and speaker's gender, accent, and motion.

2 PRELIMINARIES

2.1 Security & Privacy Risks

The voice input service provider usually consists of an app on the user end and the cloud that performs speech recognition. It honestly executes the pre-designed protocol and provides a good user experience of voice input. However, most voice input service providers collect their users' speech data. For instance, Apple stores our Siri voice commands for 24 months [1] and Google stores everything we say to Google Now by default [5]. Apple also admitted that it is sharing users' voice data with third parties [1]. Though they claim they will not sell our data in the privacy policy, they do analyze and share our data. There is no guarantee that their data analysts and hackers (*attackers*) do not stealthily abuse our voiceprints and compromise our privacy.

The major security concern is that attackers may extract our voiceprints and generate fake speeches that sound like us with the help of speech synthesis and voice cloning. A generative voice model can be easily trained from a large number of speech samples. This is also feasible when there are only a few samples (as short as 3.7 seconds) thanks to the recent development of few-shot learning [10]. On the other hand, when the attacker only possesses anonymous speech data, she can de-anonymize it via speaker recognition, causing the identity privacy leak. The attacker first gathers the person's voice recording from sources, like her YouTube channel or her posts on online social media, and then trains a voice model that represents her voiceprint to identify her utterances in the stored speech database. For example, in 2014, GoVivace Inc. started to provide speaker identification SDK for telecom companies to infer the callers' identity by matching their voice with a database containing thousands of voice recordings. In our experiment, we can identify one person out of 250 candidates with a 100% success rate provided that we have collected a voice recording of this person with a length as short as 30 seconds. Once a person's utterances are identified, the service provider will be able to mine more private information from the speech content and create a clear profile of the person. Therefore, we believe it is urgent to develop mechanisms to protect voice input users' voiceprints.



(a) Bilinear functions (b) Before & after spectrograms

Figure 2: Bilinear functions: A bilinear function is monotone increasing in the domain $[0, \pi]$ and the range $[0, \pi]$. When $\alpha < 0$, the low frequency part of the frequency axis is compressed and the high frequency part is stretched. When $\alpha > 0$, the low frequency part is stretched while the high frequency part is compressed. Fig. (b) gives an example of the spectrogram of a speech before and after its frequencies are warped.

2.2 Voice Conversion

To protect identity privacy for users of voice input service, we will design a mechanism on top of the voice conversion technology. A voice conversion algorithm modifies a source speaker's voice so that it sounds like another target speaker without changing the language contents. One of the most popular voice conversion paradigms is frequency warping [45], which is based on the well-studied technique of vocal tract length normalization (VTLN) [13, 17]. It is believed that the variety in vocal tract length among speakers causes the variability of their speech waveforms for the same language content. The original purpose of VTLN is to normalize the speaker individuality from the utterance and improve the accuracy of speech recognition [13, 17]. It can be accomplished by rescaling the frequency axis of the voice spectrogram with a warping function to compensate for individual differences in vocal tract length. We can also use VTLN for voice conversion. Given a source utterance, VTLN-based voice conversion processes it in 6 steps: pitch marking, frame segmentation, FFT (fast Fourier transform) to the frequency domain, VTLN, IFFT (inverse fast Fourier transform) to the time domain, PSOLA (pitch-synchronous overlap and add). Pitch marking and frame segmentation aim to split the speech signal into frames that match the pseudo-periodicity of voiced sounds as determined by the fundamental frequency of the voice, so as to make the output synthetic voice have the best audio quality. The key step of voice conversion is VTLN, which modifies the spectrum of each frame using frequency warping, that is, stretching or compressing the spectrum with respect to the frequency axis according to a warping function. One of the most commonly used warping function is the bilinear function [9, 45]. The formula of this function is:

$$f(\omega, \alpha) = \left| -i \ln \frac{z - \alpha}{1 - \alpha z} \right|, \tag{1}$$

where $\omega \in [0, \pi]$ is the normalized frequency, $\alpha \in (-1, 1)$ is a warping factor used to tune the strength of voice conversion, i is the imaginary unit, and $z = e^{i\omega}$. Several examples of the bilinear function are plotted in Fig. 2. Given a frequency-domain data frame, every frequency ω is changed to a new frequency $f(\omega, \alpha)$ by this

Figure 3: Application scenario (1). We add VoiceMask to the operating system and provide two interfaces for apps that need to access the microphone. Untrusted apps like voice input can only access the sanitized voice through Interface 2. Trusted apps that must access the original voice, like call apps, are granted the permission of Interface 1.

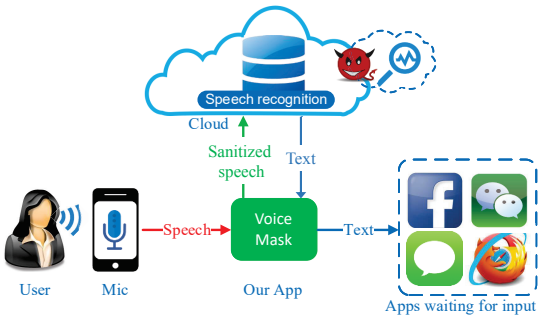


Figure 4: Application scenario (2). The cloud is a specialized speech recognition service provider. VoiceMask is a keyboard app on mobile devices that sanitizes the recorded voice input before sending it to the cloud.

formula. Then all the frequency-warped frames are reverted to the time domain by IFFT. Finally, all the frames are concatenated to generate the output speech, and a technique PSOLA is utilized to improve the speech quality [46]. Therefore, an utterance is produced with the same language content but a different voice. This is the basic idea of protecting the identity privacy, yet it is not that easy. There are several issues to consider, including how to minimize the influence on the audio quality of the output speech, how to select α to achieve a strong privacy guarantee, and how to prevent the voice from being recovered. We will discuss the details in §3.

2.3 Application Scenarios

We present two privacy-preserving voice input scenarios where VoiceMask can be used and compare their pros and cons. In Fig. 3, VoiceMask is incorporated into the operating system. It provides two types of app permissions to read the microphone. Trusted apps that must access the original voice, like call apps, are granted the permission of Interface 1. Untrusted apps that do not have to access the original voice, like voice input, can only access the masked voice through Interface 2. In Fig. 4, the cloud is a server dedicated to providing speech-to-text service. It does not install apps on the mobile end. Instead, VoiceMask (like a keyboard app) bridges the communication between the user input, the cloud, and third-party apps. It will be an open source app to win users' trust. VoiceMask accesses the raw audio, perturbs it, and produces sanitized audio. After the sanitized audio is sent to the cloud and the corresponding transcript is sent back, VoiceMask revokes some of previous perturbations on it and restores it to the desired transcript. Finally, the transcript is copied to the input box of any other app on the device that has requested user input before. VoiceMask is locally deployed in mobile devices, and it is independent of the cloud service provider. Such separation prevents the cloud from collecting extra private information from the user's device. Please note this paper aims to defend against voice input service providers. There are certainly many other apps undermining our privacy, like browsers and call apps, but how to defend against them is not the focus of this paper.

Comparison: First of all, both scenarios can protect users' voiceprints from being leaked to voice input service providers. As for protecting users' identity privacy, they both have pros and cons. The *first* scenario is more practical today and does not require installing an extra app. Nonetheless, it cannot stop voice input service providers from collecting our PII and associating it to the speeches. On the contrary, the *second* scenario is more ideal and secure in concept as it separates voice input from apps relying on voice input. Information is more secure in a decentralized system. Though many existing apps have their own voice input feature such as voice-based virtual assistants, privacy-aware users can choose to tap the input box and use VoiceMask instead. However, the specialized speech recognition service provider can access sanitized data only, so it might lack the incentive to provide the service. This may be overcome by charging users as privacy-aware users may be willing to pay. Another shortcoming is that users need to install the VoiceMask app so there could be a trust issue, which can be addressed by open source.

3 VOICEPRINT CONCEALMENT VIA ROBUST VOICE CONVERSION

3.1 Basic Voice Conversion

When we disguise the speaker's voiceprint by randomly modifying her/his voice to another voice, it is required that the speech content can still be accurately recognized when it is uploaded to the cloud so that user experience is not degraded unacceptably. User experience has two aspects: voice input accuracy and delay. We will discuss accuracy first and leave the discussion on the latency in the experiment. Since voiceprint does not have a clear definition, we quantify privacy with the accuracy of speaker recognition. The more utterances can be correctly identified, the more dangerous are users' voiceprints, which implies worse privacy. Researchers have not found a clear line between the voice features used for speech recognition and those used for speaker recognition. They both use short-term spectral features like MFCC and/or LPCC, extracted in frame-level (usually 20-30ms). Conversely, prosodic features like pitch, intonation, duration, and intensity are speaker-dependent and less useful for speech recognition. In this work, we utilize voice conversion to change the speaker's pitch to hide the voiceprint. As aforementioned, the parameter α in the bilinear function tunes the extent of distortion of the output voice. Setting $\alpha < 0$ would produce a deeper (more like low-pitched) output voice; setting $\alpha > 0$

would produce a sharper (more like high-pitched) output voice. The output voice is not distorted at all when α is 0. (See Fig. 2 for the reason.) We want to select the best α that can bring a considerable drop in the speaker recognition accuracy whereas the decrease of the speech recognition accuracy is minimized. We refer to the best parameter values that balance privacy and user experience as the *proper range*, and we experimentally found α's proper range is $A = [-0.10, -0.08] \cup [0, 08, 0.10]$ (see details in §4.2).

3.2 De-Anonymization Attacks

If we set α to a fixed value, the cloud may discover it by decompiling the apk of VoiceMask, and then reverse the voice conversion to recover the original voice. *Reversing attack* is possible because the warping functions are invertible. For instance, the bilinear function in Eq. 1 is invertible, *i.e.* $BI(\omega, -\alpha)$ is the inverse function of $BI(\omega, \alpha)$. Given α, the attacker can partially recover the original frequency axis from the warped one, and thus reverse VTLN. We found by experiment that the recovered voice sounds very close to the original one. This reveals the insecurity of the basic voice conversion. To resist the reversing attack, VoiceMask needs to randomly choose α from the proper range A every time. Then, the cloud receives speeches in different voices even when they are from the same user, so it is very difficult for it to extract the user's voiceprint.

Now, although unable to reverse the voice conversion, the cloud can still reduce ("partially reverse") it to a weaker level so that it can achieve a higher speaker recognition accuracy. The bilinear function $f(\omega, \alpha)$ has a property:

$$f(f(\omega, \alpha_1), \alpha_2) = f(\omega, \alpha_1 + \alpha_2). \quad (2)$$

In other words, applying voice conversion twice to a speech with α_1 and α_2 successively yields exactly the same output as applying the voice conversion once with $\alpha = \alpha_1 + \alpha_2$ does. Suppose the cloud has received many sanitized speeches from users whose voices have been converted with $\alpha \in [0.08, 0.10]$. It can apply a second voice conversion to these speeches with the expected value $\alpha_2 = -0.09$. Now the produced speeches are actually the output speeches of voice conversion with $\alpha \in [-0.01, 0.01]$, which has much weaker distortion strength than the originally sanitized speeches. Consequently, the cloud can achieve better accuracy than we originally expected when performing speaker recognition on these speeches. We refer to this process as *reducing attack*, and we say a function is *reducible* if it has the property in Eq. (2). We refer to reversing attack and reducing attack together as *de-anonymization attacks*. To our knowledge, de-anonymization attacks on voice conversion have not been studied in prior work. Besides the bilinear function, other common warping functions are also reversible and reducible. Therefore, simply transforming voice cannot really conceal users' voiceprints. Though using bilinear function solely is insecure, it is still worth studying, as it will help us to determine the proper range of warping factors for the compound warping function in §3.3.

3.3 Compound Warping Functions

To design a warping function that is resistant to the reducing attack, our technique is to compound two different warping functions. Here we introduce another commonly used warping function, the quadratic function [34]: $g(\omega, \beta) = \omega + \beta\left(\frac{\omega}{\pi} - \left(\frac{\omega}{\pi}\right)^2\right)$, where

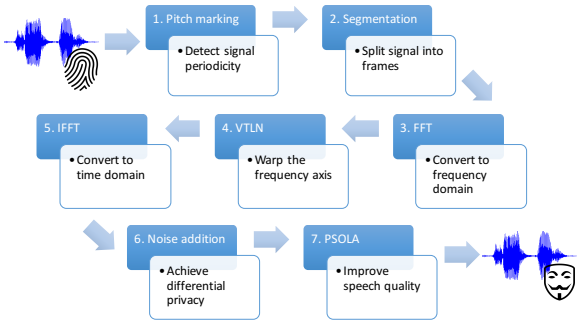


Figure 5: The rationale of our robust voice conversion. In Step 4, we compound two warping functions to warp the frequency so as to prevent de-anonymization attacks. In Step 6, we add Laplace noise to the audio signal to achieve differential privacy. The rest steps are introduced in Section 2.2.

$\omega \in [0, \pi]$ is the normalized frequency and $\beta > 0$ is a warping factor. Similar to α in the bilinear function, β determines the distortion strength of this function. The output voice turns deeper when $\beta < 0$ and sharper when $\beta > 0$. The compound function of f and g is denoted by $h(\omega, \alpha, \beta) = g(f(\omega, \alpha), \beta)$. Since the two independent parameters α, β are used in combination, they have a much bigger proper range, which is ring like (see details in Fig. 9(c)). This prohibits the attacker from conducting the reducing attack. Now every time VoiceMask perturbs a speech from the user, it randomly picks a pair of values for α, β from this range as the warping factors. This mechanism ensures the cloud is unable to reverse or reduce the voice conversion.

However, it is challenging to learn the combined impact of α, β on the distortion level of the output voice. To estimate it, we need to first quantify the *distortion strength* of the warping function h. An intuition is that the closer h is to the identity function (*i.e.*, $h(\omega, \alpha, \beta) = \omega$), the less distortion h brings to the output voice. Take Fig. 2(a) as an example, the closer the bilinear function is to the identity function, the closer α is to 0, and the less distortion it produces on the output voice. This way, we can measure the distortion strength of h by the area between the curves of itself and the identity function.

DEFINITION 1 (DISTORTION STRENGTH). *The distortion strength (denoted by dist) of a warping function $f(\omega, \mathbf{a})$ is defined as the area between the curves of itself and the identity function, i.e. $dist_f(\mathbf{a}) = \int_0^\pi |f(\omega, \mathbf{a}) - \omega|$, where $\mathbf{a}$ represents the warping factor(s).*

We first find the proper range of α in basic voice conversion (§4.2), then compute the distortion strength's proper range, and finally use it to deduce the proper range of α, β in the function h. We will present more details in §4.3.

3.4 Add a Flavor of Differential Privacy

If the attacker correctly guesses the values of α and β for a sanitized utterance, though the chance is almost zero, it will be able to reverse the compound function and recover the true voice. Here, we utilize differential privacy to improve VoiceMask's robustness to the reversing attack when the attacker accidentally knows the warping factors.

Differential privacy [16] was initially proposed to prevent the attacker from inferring individual records in a dataset based on the released aggregate results. Because of its strictness and generality, it has been widely applied for various scenarios [12, 19, 38]. We will apply differential privacy after the IFFT step of voice conversion (see Fig. 5). IFFT converts the warped frequency-domain signal to the time-domain signal, which is then post-processed and released as the new voice. The attacker can recover the frequency-domain signal by FFT and then attempt to reverse the VTLN procedure, so the goal of differential privacy is to protect the frequency-domain signal. For an audio frame of length N, let $\boldsymbol{X} = [X_0, X_1, \ldots, X_{N-1}]$ be a sequence of uniformly-spaced samples of the frequency-domain signal yielded from VTLN, and $\boldsymbol{Y} = [Y_0, Y_1, \ldots, Y_{N-1}]$ be an equivalent-length sequence of samples of the time-domain signal. $\boldsymbol{X}$ are all complex numbers. For $\forall 0 < j < N$, X_j and X_{N-j} are complex conjugates of each other. $\boldsymbol{Y}$ is the IDFT (inverse discrete Fourier transform) of $\boldsymbol{X}$, and $\boldsymbol{Y}$ are all be real numbers. Specifically,

$$Y_j = \text{real}\left(\frac{1}{N}\sum_{k=0}^{N-1} X_k \cdot e^{\frac{i2\pi kj}{N}}\right), 0 \le j < N. \quad (3)$$

To make an analogy, we can treat each of the N functions above as a query function F_j, that is, $Y_j = F_j(\boldsymbol{X})$. We see $\boldsymbol{X}$ as a dataset consisting of $(N+1)/2$ data points (the rest are their complex conjugates). Neighboring datasets are any $\boldsymbol{X}, \boldsymbol{X'}$ that differ at only one element and accordingly its complex conjugate (if not the first point). We need to release $\boldsymbol{Y}$ (that is, to answer N queries) while protecting each X_i from being inferred. We have the following theorem (proved in Appendix A).

THEOREM 1. *The sensitivity of F_j for $0 \le j < N$ is $\Delta F = \frac{2\Delta_m}{N}$, where Δ_m is the maximum possible variation of X_j.*

Our mechanism: We only modify the VTLN-based voice conversion procedure at the end of IFFT, by setting the amplitude at time j to $\hat{Y}_j = Y_j + \text{Lap}(\frac{2\Delta_m}{\epsilon})$ for all $j = 0, \ldots, N-1$, where $\text{Lap}(\cdot)$ is Laplace noise. This guarantees $(\epsilon, 0)$-differential privacy by the following theorem (see proof in Appendix B).

THEOREM 2. *Our mechanism satisfies $(\epsilon, 0)$-differential privacy.*

Optimization: reduce ΔF. In the worst case, $\Delta_m = \frac{N}{2} \cdot 2 = N$ if the audio is normalized, so $\Delta F = 2$. However, this is definitely exaggerated. Recall neighboring datasets $\boldsymbol{X}, \boldsymbol{X'}$ differ at only one element, say X_k. Because of the continuity of the frequency domain signal of the human voice, X_k is usually very close to X_{k-1} and X_{k+1}. The attacker can guess the possible range of X_k based on the context, so Δ_m is much smaller than N in fact. Accordingly, we can set $\Delta_m = \lambda N$ ($\lambda < 1$) to decrease the sensitivity. The smaller ϵ/λ is, the more noise is added to the speech signal.

By incorporating the three techniques aforementioned, warping factor randomization, compound warping functions, and differential privacy, our *robust voice conversion* mechanism is much securer than basic voice conversion.

4 EVALUATION

We conduct a detailed evaluation via emulation over three speech datasets and implementation on mobile devices. In addition, we also show the effects of multiple external factors (such as noise, motion, human and device diversity) on our proposed system.

Table 1: Statistics of the datasets in our evaluation.

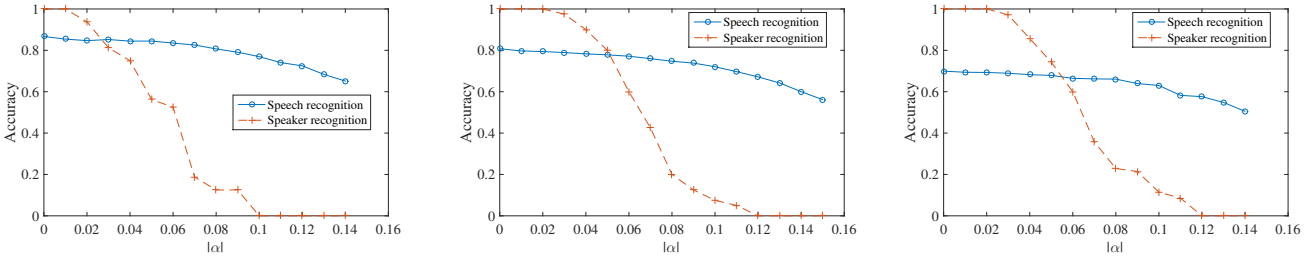
Dataset	#Speakers	#Speeches	Hours	English accents
PDA	16	836	1.8h	Mostly native
LibriSpeech	251	27.7k	100h	All native
Volunteers	14	240	0.7h	Various accents

4.1 Emulation Setup

We run the emulation on three **datasets**. *PDA*: PDA is a speech database from CMU Sphinx Group [4], which contains 16 speakers each speaking over 50 short sentences. The number of female and male speakers are well balanced. The majority of them are in their 20's. They are all native English speakers, and 12 of them speak American English. *LibriSpeech*: LibriSpeech corpus is from OpenSLR [30]. The dataset contains a set of 100 hours speech, clearly read by 251 native speakers of American English without background noise. *Volunteers*: This dataset was collected by ourselves from 14 college students all in their 20's. They are made of 11 males and 3 females. Twelve of them are not native speakers and they have various accents. The ethnic diversity can be seen in Fig. 12(c). Each volunteer read ten long sentences in the lab office. Ten sentences last about 100 s, which is enough for extracting a person's voice print (30 s is adequate). There was noise from desktop computers and the air conditioning. We also recorded one volunteer's voice in a variety of scenarios to study the impact of ambient noise, speaker's motion, and phone brand on the performance. So, there are 240 speeches in this dataset in total. Tab. 1 gives the datasets' statistics. We format all the audio to 16 kHz 16-bit PCM-encoded mono-channel WAV.

We utilize the Speaker Recognition API and the Bing Speech API from Microsoft Cognitive Services, which provided the state-of-the-art algorithms for spoken language processing when we were working on this paper [3, 51]. For PDA and LibriSpeech, we use 10 speeches for training and the rest for testing. For Volunteers, which is smaller, we use the 5 speeches for training and the rest for testing. The **steps of evaluation** go as follows. First, we create a speaker recognition system and use the training set to train a voice model for every speaker in the three datasets, which represents each speaker's voice characteristics. Second, we process the utterances in the test set using our VoiceMask. We perform voice conversion differently for male and female speakers. Specifically, we deepen female voices (by setting $\alpha < 0$) and sharpen male voices (by setting $\alpha > 0$) to make their voices closer in pitch so as to increase the difficulty for the adversary in distinguishing different speakers, which was validated by experiment. Third, we use the trained voice models to identify the speakers of the sanitized utterances and evaluate the accuracy of speaker identification. Finally, we perform speech recognition on the sanitized utterances and evaluate the accuracy as well.

Metrics: We quantify privacy with the accuracy of speaker recognition, *i.e.*, the fraction of correctly identified utterances. By default, we measure the accuracy of identifying a speaker from a pool of 50 candidates, but we also study the cases where there are more candidates (Fig. 7). We measure the performance of speech recognition using word accuracy (WAcc), which is calculated by one minus word error rate (WER). WER is similar to edit distance: it is the fraction of the number of editings (substitutions, insertions, deletions) made when comparing the true transcript and the prediction of the speech



(a) PDA (b) LibriSpeech (c) Volunteers

Figure 6: The impact of α on the accuracy of speaker/speech recognition when we use bilinear warping function only. When $|\alpha| \in [0.02, 0.10]$, the accuracy of speaker recognition goes down significantly.

content. The computation overhead of VoiceMask is measured by *real-time coefficient*, the ratio between the CPU time for processing the audio and the duration of the audio.

4.2 Basic Voice Conversion

Though insecure, basic voice conversion is still worth studying, as it will help us determine the proper range of the warping factors in robust voice conversion (§4.3). First of all, we study the case of using only the bilinear function in the voice conversion. Both speech recognition and speaker recognition have accuracy degradation when the utterances are processed by voice conversion, but the extent of degradation with respect to a specific $|\alpha|$ is different for them. We can observe that in Fig. 6. Speakers can be correctly identified with a 100% chance when $\alpha = 0$, *i.e.* on original utterances. Speech recognition on the original utterances achieves an accuracy of around 80%. It is lower than the accuracy claimed by Microsoft (93.7% [3]) probably because our test was done on datasets that are less clean and with a larger vocabulary. Recall that greater $|\alpha|$ induces greater distortion of the processed speech data. In Fig. 6, when voice conversion is applied with $|\alpha| \in [0.02, 0.10]$, the accuracy of speaker recognition degrades sharply with growing $|\alpha|$ while that of speech recognition is barely influenced. For the PDA dataset, speaker recognition accuracy is substantially decreased to only 6.8% when $|\alpha| = 0.10$ but speech recognition still has the accuracy of over 79.1%. The huge gap provides us an opportunity to find a value range of $|\alpha|$ to overcome our key challenge, that is, to suppress the speaker recognition possibility while preserving the speech recognition utility. There is a tradeoff between security and the accuracy of voice input. We may leave the choice of $|\alpha|$ to voice input users themselves. If they prioritize privacy more, they can set a greater $|\alpha|$. In our experiment, the proper range of $|\alpha|$ is set to $[0.08, 0.10]$. Thus, the speaker recognition accuracy is restricted to be smaller than 0.13 while the speech recognition accuracy is in $0.77 \sim 0.81$. For LibriSpeech, if we choose $|\alpha|$ from $[0.08, 0.10]$, the accuracy of speaker recognition on the output speeches would be restricted within 0.20 but the speech recognition accuracy is still maintained at $0.72 \sim 0.75$.

The accuracy of speaker recognition is also partially up to the number of candidates from which the cloud identifies the target speaker, as shown in Fig. 7. It is straightforward that the accuracy is lower when there are more candidates when the voice is sanitized.

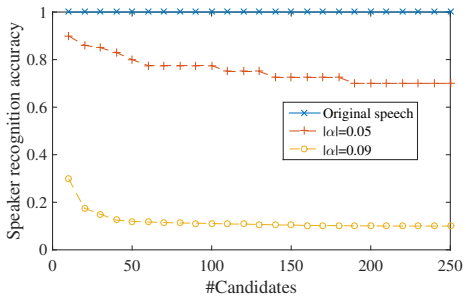


Figure 7: Speaker recognition accuracy *vs* $|\alpha|$ and the number of candidates (on LibriSpeech). The accuracy of speaker recognition decreases when $|\alpha|$ increases or when there are more candidates.

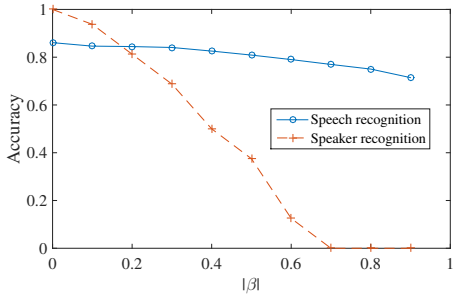
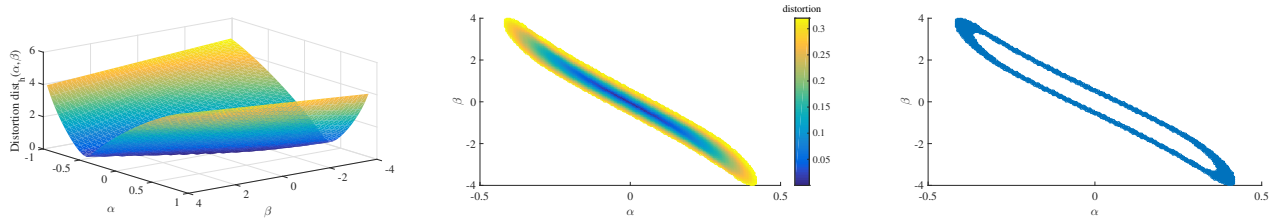


Figure 8: The impact of β on the accuracy of speaker/speech recognition when we use quadratic warping function only.

However, if we do not apply voice conversion, the accuracy is still 1 when we identify the speaker from 250 candidates. We could not try more candidates because of the limit of datasets, but it may decrease extremely slowly when there are more candidates. This figure shows that our method can reduce the speaker recognition accuracy by up to 90% when $|\alpha| = 0.09$. Additionally, we evaluate the effect of voice conversion with quadratic warping function separately in Fig. 8. The impact of the parameter β on the speaker/speech recognition accuracy is similar to that of α mentioned above.



(a) Distortion *vs* α, β (b) When $dist_h \leq 0.40$ (c) When $0.32 \leq dist_h \leq 0.40$

Figure 9: The impact of α, β on voice distortion and their proper range.

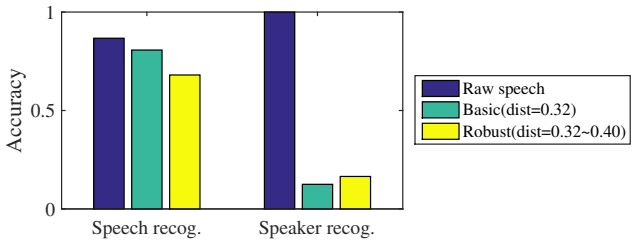
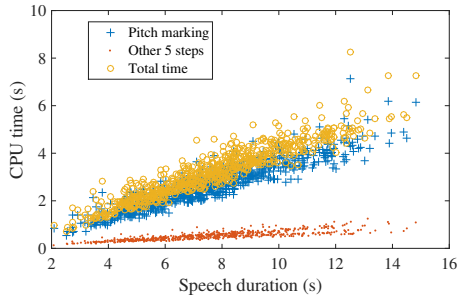
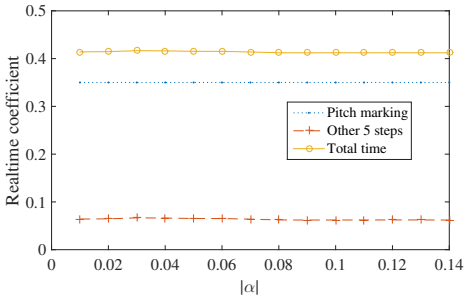


Figure 10: Comparison of different voice conversion techniques. Though the performance of robust voice conversion is not as good as basic voice conversion, it is much securer.



(a) CPU time *vs* duration



(b) Realtime coefficient *vs* α

Figure 11: Computational cost of voice conversion *vs* utterance duration and α. It demonstrates the linear relation between the CPU time and the duration of the speech being processed.

4.3 Robust Voice Conversion

Based on the results above, now we try to find the proper range of α, β for the compound function $h(\omega, \alpha, \beta)$. By Definition 1, the distortion strength of our compound function $h(\omega, \alpha, \beta)$ can be estimated, as plotted in Fig. 9(a). Now the question is what is the boundary of a *proper* distortion strength. In §4.2, we have set the proper range of $|\alpha|$ in the bilinear function f to $[0.08, 0.10]$, with corresponding distortions, $dist_f(0.08) = 0.32$, $dist_f(0.10) = 0.40$. We adopt these two values as the boundary of a proper distortion: $dist \in [0.32, 0.40]$. Fig. 9(c) shows the proper range of α, β for h such that $0.32 \leq dist_h(\alpha, \beta) \leq 0.40$. By randomly selecting α, β from this range every time we convert the voice of a speech, we can achieve our goal of impeding speaker recognition while preserving speech recognition, plus the goal of preventing the reducing attack. An experiment on PDA demonstrates the efficacy of this technique, as presented in Fig. 10. The test results on the unsanitized speech (dark blue bars) and weakly sanitized speech (green bars, bilinear function with $\alpha = 0.08$) are also given as a contrast. After the speeches are perturbed by voice conversion with the compound warping function, the speech/speaker recognition accuracy decreases to 0.68 and 0.16 respectively, as indicated by the yellow bars. We can improve speech recognition accuracy by relaxing the privacy level.

4.4 Implementation & Overhead

We first ran emulations on a MacBook Pro and recorded the CPU time of voice conversion on every utterance. As depicted in Fig. 11(a), the time of pitch marking and other 5 steps, and the total CPU time are all proportional to the duration of the utterance. We find that pitch marking consumes about 5/6 of the total time. The real-time coefficient of voice conversion is 0.42 on average with standard deviation 0.05, which means we can use VoiceMask on non-mobile devices without feeling an extra latency. As shown in Fig. 11(b), the warping parameter α has little influence on the computational cost. Likewise, β, ϵ have little influence on the computational cost, too. Then, we implemented VoiceMask on Android and used it to sanitize each speech in the PDA dataset on several mobile devices. Tab. 2 lists the run time and power consumption. It shows VoiceMask is most efficient on MEIZU Pro 6, with real-time coefficient 2.42 on average. This work is the first attempt to implement voice input sanitization on mobile devices, which sacrifices user experience for voiceprint security. The most expensive step is pitch marking, so designing a

Table 2: Computation overhead. Power consumption is induced by both VoiceMask and the background Android system.

Device	Memory	OS	Real-time coef.	Power cons.
Google Nexus 5	16 GB	Android 5.1.1	3.93 ± 0.45	0.70 W
Google Nexus 6	32 GB	Android 5.1.1	4.90 ± 0.15	0.78 W
Google Nexus 7	16 GB	Android 5.1.1	4.11 ± 0.03	0.83 W
MEIZU Pro 6	32 GB	Android 6.0	2.42 ± 0.05	0.49 W

more efficient pitch marking method will greatly reduce the latency. The overhead of our method is almost as much as that of basic voice conversion, because we only modify the VTLN step and add a noise (see Fig. 5) and they cause little extra overhead. According to a study conducted on 47 participants [43], the acceptable latency is 4 s and the acceptable accuracy is 0.70.

Offline voice input: We enabled offline speech recognition in Google voice typing, disconnected the internet, used a laptop to play the audio of each speech in the PDA dataset, let Google voice typing transcribe it into text, and recorded its performance. For the voice typing app on Nexus 7, the accuracy is 0.63 with a real-time coefficient 3.02. For Nexus 5 and 6, the accuracy is at most 0.74 and 0.75 respectively, but the app stops working frequently, especially when the user speaks fast. When the app stops, the user has to disable and re-enable it and repeat the missed speech. As offline voice input does not need to send the audio to the cloud and wait for the response, it can be faster than its online counterpart. Indeed, offline voice typing can totally avoid the privacy risk, but the cloud lacks incentives to fully support offline speech recognition because these companies crave users' speech data. At present, Google offline voice typing is only available when the user is disconnected to the Internet, which greatly limits the applications of voice input as many services like chat apps and email apps cannot function without the Internet. Additionally, the local speech models are updated less frequently. Thus, we believe it is very unlikely that offline voice input will completely replace its online counterpart. As a result, VoiceMask is a usable solution for *online* voice input users to protect their voiceprints and even identity privacy in some cases.

4.5 Security Analysis

Resistance to the reversing attack: The attacker can reverse the voice conversion (with quality loss) if she knows both the parameters α, β, because the warping functions are invertible:

$$f(g(h(\omega, \alpha, \beta), -\beta), -\alpha) = f(f(\omega, \alpha), -\alpha) = \omega. \quad (4)$$

Since we randomly choose α, β from the proper range as in Fig. 9(c) for every utterance, the probability that the attacker can correctly guesses their values is almost zero. Assume the value of α is fixed, the proper range of β would become very small, so it would be easy to guess it. However, in our method, α is randomly selected from a large range for each utterance so it is not fixed. Two distinct α, β combinations may produce the same *amount* of distortions but the directions and distributions of the distortion along the frequency axis are different, so using a different α, β combination to reverse the distortion is infeasible. Additionally, differential privacy makes it difficult to reverse the voice conversion in case the attacker knows α, β. Even if the attacker perfectly recovers an utterance, it is still difficult, though possible, for her to conduct voice cloning with a single training sample. Therefore, our robust voice conversion scheme is resistant to the reversing attack.

Resistance to the reducing attack: The reducing attack aims to weaken the overall distortion strength of a voice conversion scheme by partially reversing voice conversion with the expected values of the warping factor(s) (see an example in §3.2). In robust voice conversion, $\mathbb{E}(\alpha) = \mathbb{E}(\beta) = 0$, so there is no way to reduce the overall distortion strength. If the attacker attempts to reduce the distortion strength of a specific utterance, which was sanitized with α_1, β_1, she partially reverses voice conversion with α_2, β_2 as follows:

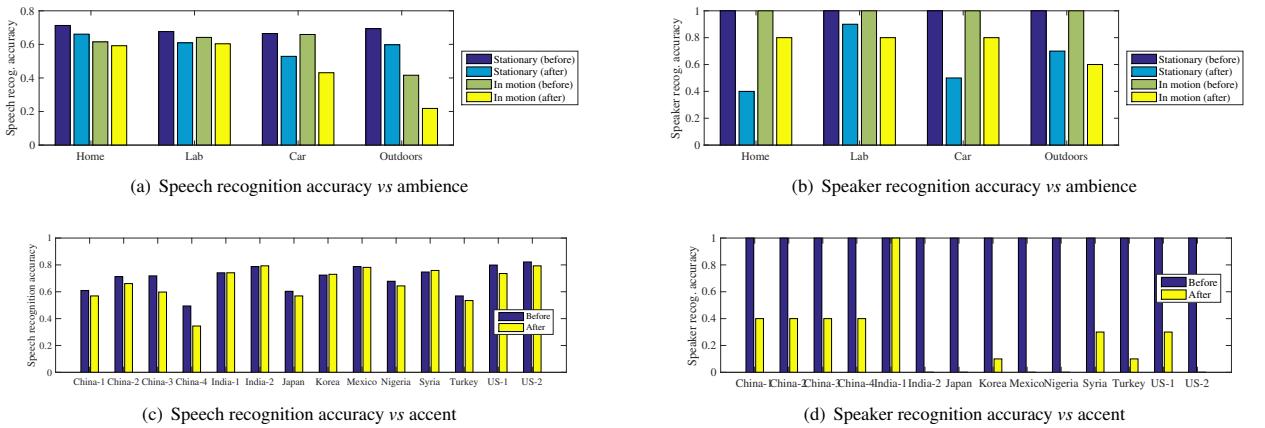
$$f(g(h(\omega, \alpha_1, \beta_1), \beta_2), \alpha_2) = f(g(f(\omega, \alpha_1), \beta_1 + \beta_2), \alpha_2). \quad (5)$$

Since α_1, β_1 are random and unknown to the attacker, it is much more likely that the operation above actually increases the distortion strength rather than reduces it. An example of the brute force attack is to uniformly choose 100×100 α_2, β_2 values and try "reducing" the voice conversion with each of them. The attacker produces 10000 utterances, one of which is very close to the original utterance. However, the attacker does not know which one it is. If the attacker randomly guesses it, the success rate is only 1/10000. Hence, our scheme is resistant to the reducing attack.

4.6 Analysis of External Factors

In this subsection, we measure the effect of our approach in various situations. Speaker recognition accuracy is computed as the success rate of identifying a speaker among 14 candidates in the Volunteer dataset if not otherwise specified.

4.6.1 Impact of ambient noise & speaker's motion. We recorded a volunteer's voice in various scenarios, which is comprised of two dimensions: location and speaker's motion. The four locations are home, lab office, outdoors (windy weather), and the inside of a car. In each location, the speaker is either stationary or under motion (walking at home, walking in the lab, running outdoors and driving the car). Fig. 12 displays their impact on the accuracy of speech recognition and speaker recognition. As shown in Fig. 12(a), speech recognition accuracy declines after the audio clip is processed by VoiceMask. This figure reveals that ambient noise degrades the accuracy of recognizing the recorded speeches as expected. Yet, the degradation is not significant, especially when the speaker is stationary, even when the voice is recorded outdoors with slight wind noise. Walking and driving do not have a huge influence either. However, the accuracy drops greatly when the speaker is running outdoors. That is because the wind noise is magnified when running, and the speaker's heavy breathing noise and disfluency also increase the difficulty of speech recognition. For speaker recognition, we try to identify this volunteer among all the 14 volunteers. Notice, the voice of the other 13 volunteers were recorded in a sitting position



(a) Speech recognition accuracy *vs* ambience

(b) Speaker recognition accuracy *vs* ambience

(c) Speech recognition accuracy *vs* accent

(d) Speaker recognition accuracy *vs* accent

Figure 12: The impact of ambient noise (Fig. (a-b)) and speaker's accent (Fig. (c-d)) on the accuracy. In Fig. (a-b), the motions for the 4 places are walking, walking, driving, and running, respectively.

in a quiet surrounding, which makes the target speaker outstanding among them. The impact of ambient noise on the accuracy of speech recognition is presented in Fig. 12(b). The figure suggests that ambient noise does not influence the accuracy of identifying original speeches but impacts the accuracy of identifying sanitized speeches.

4.6.2 Impact of human voice characteristics. We plot the accuracy of speech recognition and speaker recognition for the two genders respectively in Fig. 13. The results are obtained from the PDA dataset because it has more speakers from both genders than Volunteers does; the accuracy of speaker recognition is obtained by identifying a speaker among 10. There is no significant difference in recognizing male and female speakers' voice. As depicted in Fig. 13(a), females experience slightly more performance degradation of speech recognition after voice conversion is conducted on their utterances. The accuracy of speaker recognition has a slightly greater drop for males than it does for females, as revealed by Fig. 13(b). Despite the mild gender differences, VoiceMask can effectively protect the voice privacy of both genders. The influence of the speaker's accent is also studied. As plotted in Fig. 12(c-d), the accuracy of speech recognition is a bit lower after voice conversion is done for most speakers. Compared to it, the accuracy of speaker recognition is reduced by a much larger margin, which validates the effectiveness of our approach. The accuracy for several volunteers is as high as 40%, which is because the number of candidates is small and the speakers have a great diversity in accent and gender (China-1, India-1, US-1 are females, the rest males).

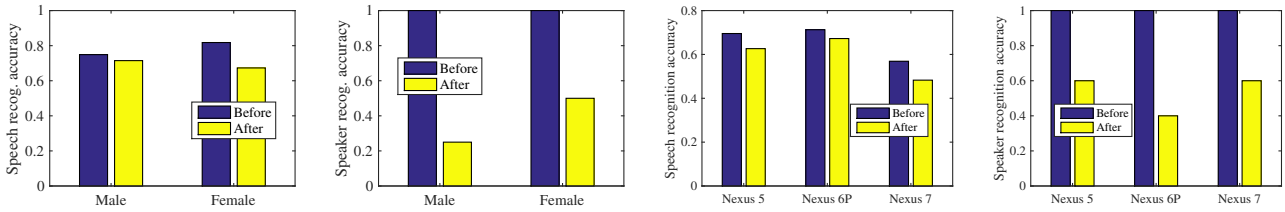
4.6.3 Impact of device brands. We also study the influence of device brand on the performance, by recording a volunteer's voice using three different Android smart devices, Nexus 5, Nexus 6P, and Nexus 7. The results are given in Fig. 13(c)(d). Fig. 13(c) shows that speech recognition accuracy is lowered for all the three different devices as expected. As for speaker recognition, we tried identifying the target volunteer using his speeches recorded with different devices. Notice that all the other 13 volunteers voice are recorded by the same device Nexus 6P. As demonstrated by Fig. 13(d), the accuracy of speaker identification declines after voice conversion. Previously some researchers have demonstrated the possibility of fingerprinting smart devices based on hardware idiosyncrasies, *e.g.* microphones [14]. Sanitizing audio signals using our method may make our devices more resistant to such fingerprinting attacks.

5 DISCUSSION

Voice input vs virtual assistants: Besides hiding voiceprint, another goal of our work is to protect the user's identity from being disclosed to *voice input* service providers. It is worth noting that voice input is different from *virtual assistants* like Siri or Google Now. A virtual assistant is a giant integration of many functions/apps. It usually works in three steps: voice input, command understanding, and command execution. They are carried out by different departments of the virtual assistant company; whether they share user data with each other is unknown. Some functions of it are personalized and require authentication of the user, *e.g.*, creating an event on the calendar, which inevitably exposes PII to the virtual assistant. Since the voice input function is dispensable for virtual assistants, we suggest it be decoupled for better user privacy, or at least, the company should prohibit different departments from sharing user data with each other. For example, Apple's data analysts are prevented from seeing the user accounts associated with the data[1]. Meanwhile, when using a virtual assistant, we can choose to click the input box and use a keyboard app like VoiceMask for voice input, instead of directly using the virtual assistant's own voice input, which can also prevent the linkage between our speech data and our identities. Even in the worst case when they are linked, our work still manages to protect users' voiceprints at least. Another interesting feature of virtual assistants is speaker authentication, which relies on comparing users' voiceprints. Fortunately, it is possible to achieve voiceprint protection and authentication simultaneously, *e.g.*, [32, 33].

Speech segmentation and randomization: There is an alternative to prevent the adversary from performing voice conversion on the speech reversely. VoiceMask executes speech segmentation and

[1] Apple's privacy policy on data analytics, https://www.apple.com/privacy/approach-to-privacy/



(a) Speech recog. *vs* gender (b) Speaker recog. *vs* gender (c) Speech recog. *vs* phone (d) Speaker recog. *vs* phone

Figure 13: The impact of gender (Fig. (a-b)) and phone brand (Fig. (c-d)) on the accuracy

separately converts the voice of each segment with randomly selected parameters. The question is how to segment a sentence into pieces such that the perturbed speech can still be well recognized into text. We observe by experiment that if a word or phrase is split into two halves and they are transformed to two different voices, they can hardly be recognized by the cloud, causing loss of speech recognition accuracy. We may utilize accurate word segmentation (or intent segmentation) to split a sentence into words (or keyphrases) to avoid this issue. We can also randomize our segmentation process, *i.e.*, randomly partition a sentence into *sequences* of words, so as to prevent the adversary from executing the same word segmentation algorithm.

Sensitive speech content sanitization: To further strengthen users' privacy, we will study how to sanitize the speech content as the future work. It is more challenging to identify and hide private information contained in the voice input than hiding the voiceprint. Text-based content privacy preservation is already a difficult problem [20], let alone speech-based. The major reason is that there is no explicit definition for content privacy, as privacy is user-specific and context-specific. For instance, the word "doctor" might be sensitive in the context "I have to see my doctor ASAP" but less sensitive in another context like "I dreamed to be a doctor when I was young". Also, some people think seeing a doctor is a private affair for them but for others it doesn't matter to let strangers know that. A tentative idea is to ask the user to list a few sensitive words/keyphrases, harness *keyword spotting* [47] to efficiently detect them and replace them with insensitive words before sending the speech to the cloud, and reverse the substitution after receiving the recognized text.

6 RELATED WORK

Security and privacy on voice data have been concentrated on in previous works.

Spoofing attack to speaker verification systems has received considerable attention in the area of speaker recognition. The simplest method is replaying a speech sample previously recorded or created by concatenating basic speech segments of the target speaker. It has been shown that this method can effectively spoof text-independent speaker verification systems [27]. Besides, human impersonation is also an effective way, where the attacker mimics a target speaker's voice by deliberately adapting his speed, accent, intonation and choice of lexicons and grammars [50]. More powerful techniques include speech synthesis [15] and voice conversion [50]. One of the challenges of speech synthesis is that it needs a great amount of training utterances from the target speaker to train a voice model of her before generating a specific speech to bypass a text-dependent speaker verification system [15]. On the contrary, voice conversion is much easier than speech synthesis because it only requires a few speeches from the source speaker and the target speaker, whether they have different content or not [50]. The core of voice conversion is to learn a mapping function from the features of source speeches to those of target speeches with the given speech samples, which is then used to convert the speech of the attacker (source speaker) to the voice of the target speaker without changing the linguistic content [25].

Additionally, *privacy learning* has also been a focus in spoken language analysis. The utterances convey a rich amount of underlying information about the speaker due to the speaker-dependent characteristics contained. Some of the information might be considered as private for the speaker. For example, Dan Gillick [18] showed that word use preference can be utilized to infer the speaker's demographics including gender, age, ethnicity, birthplaces, and social-status. Mairesse *et al.* [28] designed classification, regression and ranking models to learn the Big Five personality traits of the speaker including "extraversion vs. introversion". Other sensitive information contained in the utterance such as emotions [31, 40, 41] and health state [29, 42] could also be inferred by speaker classification methods. Another line of research works is about fingerprinting smartphone users via various features including voice features, language preferences, ambient noises, interaction patterns [14, 39]. In addition to voice data, de-anonymization has been well studied for other data types including relational database and social network [21, 36, 37].

Meanwhile, privacy protection measures have emerged. Wu *et al.* [48, 49] proposed robust speaker verification techniques to defend against spoofing attacks. Pathak *et al.* [32, 33] adopted secure multi-party computation (SMC) to implement and achieve speaker verification in a privacy-preserving fashion. The speaker's utterance is not leaked to any entity, which greatly prevents the malicious party from gathering utterances for training spoofing attack models. Smaragdis *et al.* [44] were the first to design an SMC-based secure speech recognition, though it is said to be a rudimentary system and the cryptographic protocols are computationally expensive. Another related work by us [35] studied and quantified the privacy risks in speech data publishing and proposed privacy-preserving countermeasures. In addition, there is a body of work on distributed speech recognition *e.g.* [52] for mobile devices. It enables the device itself to extract spectral features from the speech and then send them to the cloud where the features are converted to text. Distributed speech

recognition is not commonly used yet and the main purpose is to reduce the communication cost. Though the user's voice is not directly exposed, still the features like MFCC can be used to extract voice print and reconstruct the voice accurately.

7 CONCLUSION

Voice input has been tremendously improving the user experience of mobile devices by freeing our hands from typing on the small screens. In this work, we present a light-weight voice sanitization scheme for voice input fanatics to achieve a good protection of their voice biometric information and identity privacy. Our scheme, VoiceMask, is built upon voice conversion, which we point out is prone to the reversing attack and the reducing attack. By incorporating several heuristics, our robust voice conversion mechanism is resistant to these attacks. The experimental results demonstrate that VoiceMask indeed protects users' voiceprints and impedes voiceprint-based speaker de-anonymization, at the cost of only minimal degradation in the user experience of voice input.

A PROOF OF THEOREM 1

For any $0 \le j < N$,

$$\begin{aligned}
\Delta F_j &= \max \ |F_j(\boldsymbol{X}) - F_j(\boldsymbol{X'})| \\
&= \max_{k\neq 0} \ \text{real}\left(\frac{1}{N}\left(\Delta X_k e^{\frac{i2\pi kj}{N}} + \Delta X_{N-k} e^{\frac{i2\pi (N-k)j}{N}}\right)\right) \\
&= 2/N \cdot \max_{k\neq 0} \ \text{real}\left(\Delta X_k e^{\frac{i2\pi kj}{N}}\right) \\
&\le 2/N \cdot \max_{k\neq 0} \left|\Delta X_k e^{\frac{i2\pi kj}{N}}\right| \\
&= 2/N \cdot \max_{k\neq 0} |\Delta X_k| \\
&= 2\Delta_m/N.
\end{aligned} \tag{6}$$

B PROOF OF THEOREM 2

Because of the $\text{Lap}(\frac{2\Delta_m}{N}/\frac{\epsilon}{N})$ noise, publishing each $\hat{Y}_j$ alone satisfies $(\frac{\epsilon}{N}, 0)$-differential privacy according to the Laplace mechanism [16]. By the sequential composition theorem [16], the publishing of a whole sequence $\hat{\boldsymbol{Y}}$ satisfies $(\epsilon, 0)$-differential privacy. The waveform of a frame is produced by post-processing $\hat{\boldsymbol{Y}}$, which preserves differential privacy [16]. Thus, publishing a frame satisfies $(\epsilon, 0)$-differential privacy. The VTLN-based voice conversion is performed at the frame level, and the output audio is generated from a series of disjoint frequency-domain frames. By the parallel composition theorem [16], publishing the audio satisfies $(\epsilon, 0)$-differential privacy. An utterance is published to a data consumer *only once*, so the privacy level would not degrade over time. Hence, our voice conversion mechanism satisfies $(\epsilon, 0)$-differential privacy.

ACKNOWLEDGMENTS

Xiang-Yang Li is the corresponding author.

This work is partially supported by the National Key R&D Program of China 2018YFB0803400, China National Funds for Distinguished Young Scientists with No. 61625205, Key Research Program of Frontier Sciences, CAS, No. QYZDY-SSW-JSC002, NSFC with No. 61520106007, 61751211, and NSF CNS 1526638. We appreciate the assists of Mr. Yanbo Deng from Illinois Institute of Technology, Dr. Yu Wang from the University of North Carolina at Charlotte, and Dr. Yiqing Hu from the University of Science and Technology of China.

REFERENCES

[1] 2013. Apple stores your voice data for two years. https://goo.gl/6hx1kh.
[2] 2014. Faked Obama speech. https://goo.gl/pnR3VK.
[3] 2016. Microsoft achieves speech recognition milestone. https://goo.gl/FsPLrJ.
[4] 2017. CMU PDA database. www.speech.cs.cmu.edu/databases/pda/.
[5] 2017. Google stores your voice inputs. https://goo.gl/7w5We1.
[6] 2017. The Invisible Internet Project. geti2p.net/en/.
[7] 2017. Phone scam. https://goo.gl/T4xMxM.
[8] 2017. Tor Project. www.torproject.org.
[9] Alex Acero and Richard M Stern. 1991. Robust speech recognition by normalization of the acoustic space. In *ICASSP*. IEEE, 893–896.
[10] Sercan O Arik, Jitong Chen, Kainan Peng, Wei Ping, and Yanqi Zhou. 2018. Neural Voice Cloning with a Few Samples. *arXiv preprint arXiv:1802.06006* (2018).
[11] Yu-Shuo Chang, Shih-Hao Hung, Nick JC Wang, and Bor-Shen Lin. 2011. CSR: A cloud-assisted speech recognition service for personal mobile device. In *ICPP*. IEEE, 305–314.
[12] Linlin Chen, Taeho Jung, Haohua Du, Jianwei Qian, Jiahui Hou, and Xiang-Yang Li. 2018. Crowdlearning: Crowded Deep Learning with Data Privacy. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–9.
[13] Jordan Cohen, Terri Kamm, and Andreas G Andreou. 1995. Vocal tract normalization in speech recognition: Compensating for systematic speaker variability. *The Journal of the Acoustical Society of America* 97, 5 (1995), 3246–3247.
[14] Anupam Das, Nikita Borisov, and Matthew Caesar. 2014. Do you hear what i hear?: Fingerprinting smart devices through embedded acoustic components. In *CCS*. ACM, 441–452.
[15] Phillip L De Leon, Michael Pucher, Junichi Yamagishi, Inma Hernaez, and Ibon Saratxaga. 2012. Evaluation of speaker verification security and detection of HMM-based synthetic speech. *IEEE Transactions on Audio, Speech, and Language Processing* 20, 8 (2012), 2280–2290.
[16] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *FnT-TCS* 9, 3–4 (2014), 211–407.
[17] Ellen Eide and Herbert Gish. 1996. A parametric approach to vocal tract length normalization. In *ICASSP*, Vol. 1. IEEE, 346–348.
[18] Dan Gillick. 2010. Can conversational word usage be used to predict speaker demographics?.. In *Interspeech*. Citeseer, 1381–1384.
[19] Jiahui Hou, Xiang-Yang Li, Taeho Jung, Yu Wang, and Daren Zheng. 2018. CASTLE: Enhancing the Utility of Inequality Query Auditing Without Denial Threats. *IEEE Transactions on Information Forensics and Security* 13, 7 (2018), 1656–1669.
[20] Bernard J Jansen. 2006. Search log analysis: What it is, what's been done, how to do it. *Library & information science research* 28, 3 (2006), 407–432.
[21] Shouling Ji, Weiqing Li, Mudhakar Srivatsa, and Raheem Beyah. 2014. Structural data de-anonymization: Quantification, practice, and implications. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 1040–1053.
[22] Taeho Jung, Xiang-Yang Li, Wenchao Huang, Jianwei Qian, Linlin Chen, Junze Han, Jiahui Hou, and Cheng Su. 2017. Accounttrade: Accountable protocols for big data trading against dishonest consumers. In *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*. IEEE, 1–9.
[23] Taeho Jung, Xiang-Yang Li, Wenchao Huang, Zhongying Qiao, Jianwei Qian, Linlin Chen, Junze Han, and Jiahui Hou. 2019. AccountTrade: Accountability Against Dishonest Big Data Buyers and Sellers. *IEEE Transactions on Information Forensics and Security* 14, 1 (2019), 223–234.
[24] Peter F King. 2003. Server based speech recognition user interface for wireless devices. US Patent 6,532,446.
[25] Tomi Kinnunen, Zhi-Zheng Wu, Kong Aik Lee, Filip Sedlak, Eng Siong Chng, and Haizhou Li. 2012. Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech. In *ICASSP*. IEEE, 4401–4404.
[26] Xiang-Yang Li, Chunhong Zhang, Taeho Jung, Jianwei Qian, and Linlin Chen. 2016. Graph-based privacy-preserving data publication. In *INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE*. IEEE, 1–9.
[27] Johan Lindberg, Mats Blomberg, et al. 1999. Vulnerability in speaker verification-a study of technical impostor techniques.. In *Eurospeech*, Vol. 99. 1211–1214.
[28] François Mairesse, Marilyn A Walker, Matthias R Mehl, and Roger K Moore. 2007. Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of artificial intelligence research* 30 (2007), 457–500.
[29] Iosif Mporas and Todor Ganchev. 2009. Estimation of unknown speaker's height from speech. *International Journal of Speech Technology* 12, 4 (2009), 149–160.

[30] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an ASR corpus based on public domain audio books. In *ICASSP*. IEEE, 5206–5210.
[31] Pavitra Patel, Anand Chaudhari, Ruchita Kale, and MA Pund. 2017. Emotion recognition from speech with Gaussian mixture models & via boosted GMM. *IJRISE* 3 (2017).
[32] Manas Pathak, Jose Portelo, Bhiksha Raj, and Isabel Trancoso. 2012. Privacy-preserving speaker authentication. In *International Conference on Information Security*. Springer, 1–22.
[33] Manas A Pathak and Bhiksha Raj. 2013. Privacy-preserving speaker verification and identification using Gaussian mixture models. *IEEE Transactions on Audio, Speech, and Language Processing* 21, 2 (2013), 397–406.
[34] Michael Pitz and Hermann Ney. 2005. Vocal tract normalization equals linear transformation in cepstral space. *IEEE Transactions on Speech and Audio Processing* 13, 5 (2005), 930–944.
[35] Jianwei Qian, Feng Han, Jiahui Hou, Chunhong Zhang, Yu Wang, and Xiang-Yang Li. 2018. Towards privacy-preserving speech data publishing. In *INFOCOM*. IEEE.
[36] Jianwei Qian, Xiang-Yang Li, Chunhong Zhang, and Linlin Chen. 2016. De-anonymizing social networks and inferring private attributes using knowledge graphs. In *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on*. IEEE, 1–9.
[37] Jianwei Qian, Xiang-Yang Li, Chunhong Zhang, Linlin Chen, Taeho Jung, and Junze Han. 2017. Social network de-anonymization and privacy inference with knowledge graph model. *IEEE Transactions on Dependable and Secure Computing* (2017).
[38] Jianwei Qian, Fudong Qiu, Fan Wu, Na Ruan, Guihai Chen, and Shaojie Tang. 2017. Privacy-preserving selective aggregation of online user behavior data. *IEEE Trans. Comput.* 66, 2 (2017), 326–338.
[39] Giorgio Roffo, Marco Cristani, Loris Bazzani, Ha Minh, and Vittorio Murino. 2013. Trusting Skype: Learning the way people chat for fast user recognition and verification. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 748–754.
[40] Klaus R Scherer, Judy Koivumaki, and Robert Rosenthal. 1972. Minimal cues in the vocal communication of affect: Judging emotions from content-masked speech. *Journal of Psycholinguistic Research* 1, 3 (1972), 269–285.
[41] Björn Schuller, Ronald Müller, Florian Eyben, Jürgen Gast, Benedikt Hörnler, Martin Wöllmer, Gerhard Rigoll, Anja Höthker, and Hitoshi Konosu. 2009. Being bored? Recognising natural interest by extensive audiovisual integration for real-life application. *Image and Vision Computing* 27, 12 (2009), 1760–1774.
[42] Björn Schuller, Stefan Steidl, Anton Batliner, Florian Schiel, and Jarek Krajewski. 2011. The Interspeech 2011 Speaker State Challenge.. In *Interspeech*. 3201–3204.
[43] James Scovell, Marco Beltman, Rina Doherty, Rania Elnaggar, and Chaitanya Sreerama. 2015. Impact of accuracy and latency on mean opinion scores for speech recognition solutions. *Procedia Manufacturing* 3 (2015), 4377–4383.
[44] Paris Smaragdis and Madhusudana Shashanka. 2007. A framework for secure speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing* 15, 4 (2007), 1404–1413.
[45] David Sundermann and Hermann Ney. 2003. VTLN-based voice conversion. In *ISSPIT*. IEEE, 556–559.
[46] Hélene Valbret, Eric Moulines, and Jean-Pierre Tubach. 1992. Voice transformation using PSOLA technique. In *ICASSP*, Vol. 1. IEEE, 145–148.
[47] Philip Weber, Linxue Bai, SM Houghton, P Jančovič, and Martin J Russell. 2016. Progress on phoneme recognition with a continuous-state HMM. In *ICASSP*. IEEE, 5850–5854.
[48] Zhizheng Wu, Nicholas Evans, Tomi Kinnunen, Junichi Yamagishi, Federico Alegre, and Haizhou Li. 2015. Spoofing and countermeasures for speaker verification: a survey. *Speech Communication* 66 (2015), 130–153.
[49] Zhizheng Wu, Sheng Gao, Eng Siong Cling, and Haizhou Li. 2014. A study on replay attack and anti-spoofing for text-dependent speaker verification. In *APSIPA Annual Summit and Conference*. IEEE, 1–5.
[50] Zhizheng Wu and Haizhou Li. 2014. Voice conversion versus speaker verification: an overview. *APSIPA Transactions on Signal and Information Processing* 3 (2014), e17.
[51] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig. 2016. Achieving human parity in conversational speech recognition. *arXiv preprint arXiv:1610.05256* (2016).
[52] Weiqi Zhang, Liang He, Yen-Lu Chow, RongZhen Yang, and YePing Su. 2000. The study on distributed speech recognition system. In *ICASSP*, Vol. 3. IEEE, 1431–1434.