sciendo

Proceedings on Privacy Enhancing Technologies ; 2021 (2):130–150

Yuchen Liu*, Ziyu Xiang, Eun Ji Seong, Apu Kapadia, and Donald S. Williamson

Defending Against Microphone-Based Attacks with Personalized Noise

Abstract: Voice-activated commands have become a key feature of popular devices such as smartphones, home assistants, and wearables. For convenience, many people configure their devices to be 'always on' and listening for voice commands from the user using a trigger phrase such as "Hey Siri," "Okay Google," or "Alexa." However, false positives for these triggers often result in privacy violations with conversations being inadvertently uploaded to the cloud. In addition, malware that can record one's conversations remains a significant threat to privacy. Unlike with cameras, which people can physically obscure and be assured of their privacy, people do not have a way of knowing whether their microphone is indeed off and are left with no tangible defenses against voice based attacks. We envision a general-purpose physical defense that uses a speaker to inject specialized obfuscating 'babble noise' into the microphones of devices to protect against automated and human based attacks. We present a comprehensive study of how specially crafted, personalized 'babble' noise ('MyBabble') can be effective at moderate signal-to-noise ratios and can provide a viable defense against microphone based eavesdropping attacks.

Keywords: privacy; audio; microphones; obfuscating; noise

DOI 10.2478/popets-2021-0021
Received 2020-08-31; revised 2020-12-15; accepted 2020-12-16.

***Corresponding Author: Yuchen Liu:** Indiana University Bloomington, E-mail: liu477@iu.edu
Ziyu Xiang: Stanford University (This work was conducted while at Indiana University Bloomington), E-mail: ziyxiang@stanford.edu
Eun Ji Seong: Indiana University Bloomington, E-mail: eunseong@iu.edu
Apu Kapadia: Indiana University Bloomington, E-mail: kapadia@indiana.edu
Donald S. Williamson: Indiana University Bloomington, E-mail: williads@indiana.edu

1 Introduction

There are an estimated 3.3 billion actively used smartphones around the globe today,[1] and there is a growing market for digital assistants such as Google Home and the Amazon Echo; for example, more than 100 million Alexa-enabled digital assistants have been sold to date.[2] These microphone-enabled devices feature an 'always listening' mode to support voice based commands. As a convenience over physically pressing a button, people can trigger voice commands with spoken phrases such as "Hey Alexa," "Okay Google," and "Hey Siri" for Amazon-, Google-, and Apple-based devices, respectively. (Always-on) microphones in these ubiquitous devices, however, raise significant privacy concerns. First, digital assistants are often incorrectly triggered through false positives and violate people's privacy by uploading unauthorized conversations to the cloud [37] or, worse, by sending them to contacts by incorrectly interpreting casual conversations as complex commands.[3] Second, unlike cameras, which also pose privacy risks but can be covered more 'tangibly' by users (e.g., with physical items such as clothes or stickers), microphones are not easily disabled or muted. Even if these microphones are ostensibly 'off' (e.g., using on-board mute buttons), they can potentially record conversations through eavesdropping malware [50]. Microphones, in general, pose a serious threat to the privacy of users – they are ubiquitous in people's lives, can be easily exploited, and yet have no clear, 'tangible' way of being disabled by users [3].

Although simply turning off or removing the device from one's bedroom (for example) is a viable choice, it is a heavy-handed approach. People have legitimate reasons to have their smartphones and personal assistants on hand (and 'on'), e.g., to notice and receive incoming calls. Yet, there are situations when one would prefer that the camera is obscured or the microphone disabled.

1 https://venturebeat.com/2018/09/11/newzoo-smartphone-users-will-top-3-billion-in-2018-hit-3-8-billion-by-2021/
2 https://www.theverge.com/2019/1/4/18168565/amazon-alexa-devices-how-many-sold-number-100-million-dave-limp
3 https://bgr.com/2018/05/25/amazon-alexa-recording-private-conversation/

Therefore, we seek defenses where a user can choose to 'mask' their conversations while retaining the functions and proximity of their devices. We also seek to make a more general contribution, as we explain below, where complementary jamming solutions can employ more effective noise patterns to mask conversations.

In general, defenses against eavesdropping attacks have not been adequately researched. Although much work has been done in the area of 'speech separation' [30, 60, 63] to retrieve a signal with added background noise from 'babble noise generators,' these techniques have not been studied in adversarial settings. Babble noise can mimic noisy or bustling cafes, and speech separation can sometimes retrieve speech of interest, but it is unknown to what degree signals can be recovered in the face of adversarial noise. Furthermore, adversarial settings to retrieve the signal have not considered strong adversary models where the attacker may have a trained model based on the target victim's speech. One class of microphone attacks (that could be used as a defense against eavesdropping) aims to jam microphones, e.g., using techniques that treat the microphone as an antenna and project sound from large distances [33] or by injecting ultrasonic noise [14, 49]. However, it is not known *what kind of noise* should be injected to effectively mask against eavesdropping. Another class of attack seeks to trigger or foil speech recognition by injecting adversarial noise (including ultrasonic noise) aimed at the machine learning algorithm to result in specific (but incorrect) speech transcription [13, 14]. In the latter case, however, a human attacker could potentially discern the target audio because the speech itself may not be effectively obfuscated. Therefore, defenses that obscure against both automated *and* human attackers are needed.

In this work, we devise and evaluate a method called 'MyBabble' for generating personalized, obfuscating 'babble noise' that is robust against strong adversaries who are capable of building automated speech recognition models tuned to their adversaries. Basic forms of babble noise have been shown to be effective against humans [17], but it is unknown how babble noise can perform against sophisticated attacks. We perform a series of experiments to evaluate the performance of speech separation and recognition techniques under progressively stronger adversarial models and various noise models. In addition to training adversarial automatic speech recognition (ASR) systems using 'clean' speech from the target, we studied ASR models which were trained using speech and noise mixtures at very low signal-to-noise ratios (SNR). Once we identified a reasonable scenario for a strong remote adversary using ASR, we designed our MyBabble noise mixture using a range of novel techniques. In essence, our approach mixes a large number of randomly obtained speech 'tracks' using 'voice-conversion' techniques to transform these tracks to imitate the voice characteristics of the speaker. We then show that this approach of building obfuscating noise results in poor performance for an attacker at SNR levels that, for example, can be easily achieved with headphone-style speakers placed on the device's microphone without disturbing people in the vicinity. We finally run a real-world test to confirm our simulation-based results. In addition to this use case of using one's personal headphones, our approach is general-purpose in that MyBabble can be used with other injection techniques such as remote jamming through ultrasonic methods [13, 14].

2 Related Work

We discuss related work on attacks and defenses for microphone-based eavesdropping; automated speech recognition systems; 'babble' noise; and voice conversion techniques.

2.1 Attacks and Defenses

Multiple studies have explored how different devices with microphones can be exploited to violate people's privacy. Schlegel et al. [50] designed a proof-of-concept malware called "Soundcomber," which was able to eavesdrop on a smartphone's owner and transcribe credit-card numbers and other sensitive spoken numbers in phone calls. Earlier, Zhuang et al. [67] proposed a method to obtain what users have typed on their keyboard by simply recording about 10 minutes of keyboard strokes from a nearby microphone. Malware, such as Soundcomber, could use such techniques to also eavesdrop and transcribe keystrokes heard in the vicinity. Recently, ransomware has been observed in the real world recording conversations for later blackmail.[4] In general, the threat of malware on smartphones and IoT devices

4 Forbes: Creepy New Android Malware Can Secretly Record Your Conversations. https://www.forbes.com/sites/leemathews/2018/02/28/creepy-new-android-malware-can-secretly-record-your-conversations/#2f3f6950335f

'listening in' on people's conversations is a realistic and potent threat.

Defenses against microphone-based eavesdropping attacks remain challenging. Although software-based security defenses that seek to detect such malware remain relevant, one must assume that on-device defenses can be disabled by malware. Thus, our work seeks to examine the class of *external* defenses by injecting obfuscating noise from an external device. Along these lines, a few such approaches have been reported in the literature. Kune et al. [33] demonstrated how audio can be injected into a microphone by treating them as an antenna using low-power electromagnetic waveforms. Although presented as an attack, such an approach could be used to inject noise into the microphone as a defense. For example, recent work has shown how ultrasonic jamming can be improved and made more practical for end users to disable microphones [14]. Independent of the mechanism used for jamming, it is still not known what is an effective method of generating noise to obfuscate human speech. Carlini et al. [12] showed how attackers can activate different automatic speech recognition systems without alerting their users. Such techniques could be used to foil speech-recognition systems used by adversaries. Although effective against large-scale dragnet attacks, a human attacker targeting an adversary who listens to the speech transcript will be able to easily discern the victim's speech. Thus, more work is needed on *human audible* obfuscating speech that can be injected into microphones to foil both ASR systems as well as human adversaries.

2.2 Automated Speech Recognition

In our work we assume adversaries who can use and train speech-recognition models tuned to their target victims. Today's state-of-the-art automatic speech recognition (ASR) systems can achieve impressive performance by recognizing everyday speech across a range of speakers. These ASR systems are particularly good at recognizing content of 'clean' speech but tend to degrade in performance in the presence of noise [57]. The brittleness of the ASR systems is not surprising because various studies show that even for human listeners, noise can be a hurdle for comprehending speech [24]. Adversarial attacks against ASR systems try to fool deep-learning models using malicious inputs. Several adversarial attacks have been investigated against computer vision algorithms [32, 35]. More recently, speech-based adversarial attacks have been studied [4, 13]. Such attacks, however, are 'one-to-one' attacks, i.e., one needs to devise specially crafted noise for each sentence. This makes the attack (or its use as a defense in our context) computationally expensive and cannot be executed in real time. Our proposed defense mechanism introduces an approach to generate such adversarial noise in advance to severely degrade ASR for the target speech.

2.3 Babble Noise

Our MyBabble defense mechanism improves on the concept of 'babble' noise (which combines multiple voice tracks) by tuning it for each target speaker. Basic babble noise is effective at masking speech because it is non-stationary over time and consists of several speech signals that make separating speech and recognizing speech much more difficult [44, 63]. This effectiveness partially occurs because babble noise obstructs much of the audible frequency range, which does not allow listeners (or recognition systems) to hear speech in spectral gaps that other noises produce [39]. Bronkhorst et al. [10] identified babble noise as an effective noise for masking speech and evaluated the impact of different configurations. Elliott et al. [17] performed a user study with children from 9 to 17 years old and found babble noise to be effective at masking target speech. In general, babble noise is used to test speech separation applications, e.g., to improve the performance of hearing aids in noisy environments. However, they have not been studied in adversarial settings. In most cases, generic versions of babble noise are used that are constructed from random speakers. In this study, we postulate that user-specific babble noise will better interfere with speech from the user and render speech even more unintelligible at a larger range of noise levels.

2.4 Voice Conversion

'Voice conversion' is a technique that 'transfers' an utterance from a source speaker to a target speaker. Following conversion, the utterance will then sound as if it were spoken by the target speaker. Of course, there are many nuances to mimicking the speech of a target speaker, and voice conversion techniques continue to evolve [27, 51]. An overview of this field can be found in the work of Mohammadi and Kain [41]. In our work, we seek to mask a victim's utterances using specialized babble noise constructed from voice-converted tracks tuned to the victim's speech. Several traditional voice

conversion systems based on Gaussian mixture models (GMM) [2, 29] require parallel corpora, which consists of the same set of recorded utterances from the source and target speakers. This is a practical limitation since, (1) such parallel corpora are difficult to obtain, and (2) the possible conversions are limited to the utterances in the corpora. Recent work has explored the use of deep neural networks to avoid the need for parallel data [52]; however, this approach is less flexible because it can only perform conversions to a single target speaker, which limits scalability. In our work, we build a convolutional variational auto-encoder (VAE) for the voice conversion system inspired by the work of Hsu et al. [25], which is a many-to-many voice conversion system that does not require parallel data from speakers. We include a convolutional network to better capture local and short-time spectral-temporal structure, which is not captured with deep neural networks. Our VAE does not require phonetic and lexical information, which simplifies the approach since it requires only audio data.

3 Method

In this section, we describe our defensive approach based on babble noise, the adversarial setting, and our metric for evaluation.

3.1 Threat Model

We assume that attackers are interested in a 'target' user of interest who owns at least one 'target device.' The attacker compromises this device to eavesdrop on the target's conversations. Likewise, we assume the target user is aware of such a potential attack and is motivated to defend against such attacks.

3.1.1 Target User Assumptions

We assume the target user does not trust the device with the embedded microphone and is worried it may eavesdrop on the target user's *face-to-face* conversations. For example, the target's smartphone may eavesdrop on a 'physical' space conversation between the target user and their spouse, or a confidential, in-person conversation between a lawyer and their client. In this model, the target user assumes that either the platform itself is adversarial (or, e.g., coerced by the government), or a remote adversary has entirely compromised the platform. Even if the user is particularly cautious in maintaining the security of the device, the user could still be worried about potential eavesdropping by the platform itself. For example, the target user may be uncomfortable with companies such as Amazon, Google, or Apple having access to the microphone at all times. At the same time, as is commonplace today, the target user finds sufficient utility in owning and using the device and, thus, is unwilling to discontinue the use of the device altogether. Such practical choices are already demonstrated in practice where people use stickers to obscure cameras on their laptops in situations where they do not trust the device with access to the camera and allow limited use of the camera when needed by temporarily removing the sticker. Likewise, we assume target users will want to enact an analogous defense against the microphone by trying to 'obscure' the microphone in various situations. Recent work has shown that people would prefer such functionality to temporarily disable microphones on mobile/IoT devices [3].

3.1.2 Adversarial Capabilities

We assume a strong adversarial model where attackers have full access to the target user's device. In particular, we make the following assumptions:

- We assume a remote attacker who is not physically proximal to the target and must thus rely on a compromised device with a microphone to eavesdrop on the target's conversations. We assume the attacker can gain (or as the platform owner, already has) full control of the target device.
- The attacker has access to the latest computational techniques for speech separation and automated speech recognition (ASR) so that they can try to isolate the target user's speech and even perform automated surveillance without the need for human-based audio transcription.
- At the same time, we assume that if ASR fails (e.g., because of any employed defenses) for the attacker, the attacker has the time to listen to the captured audio. Thus, any defense must be robust against both automated and human based attacks.
- We assume the attacker has access to clean speech samples from the target speaker, which can help them train and tune ASR models to attempt to circumvent defenses that add noise to the captured speech. In particular, we assume the attackers can tune their models to noisy environments as well.

3.1.3 Orthogonal Attacks

In this paper, we focus on designing personalized noise defenses for single microphones. One counter-attack is to use beamforming techniques that use multiple microphones to isolate speech based on direction information [6] where such attacks assume speech and noise come from different directions. We consider such attacks orthogonal for the following reason: Defenses against beamforming attacks rely on injecting noise into multiple microphones so that the direction of the noise matches the direction of the target speech [65]. These techniques are orthogonal to the question we address, which is: After employing such defenses, *what kind of noise should be injected?* Even if the noise occurs at the same location as the target speech, other sophisticated machine learning attacks may still isolate the target speech if the noise does not obscure the spectral-frequency content of the speech [7]. Any beamforming defense must thus also employ personalized noise to be effective. Our work focuses on the technical issues and effectiveness of personalized noise, evaluated in a single-microphone scenario but applicable to multi-microphone scenarios.

3.2 Defense Method

Our proof-of-concept defense is to play noise through an external device (earbuds) directly on top of the microphones of the compromised device (i.e., not at a distance). Since our experiments show that the noise does not disturb people nearby, we can assume speakers will not alter their voice via the Lombard effect (i.e., where people enhance their voices in noisy environments). The external noise generating device is assumed to be a standalone device that functions without the constant need for an internet connection, and we assume it cannot be compromised by the attacker in our threat model.

Before describing our defensive approach, we provide some background about the basic approach and identify two state-of-the-art 'baseline' defenses for comparison.

3.2.1 Baseline Defense Mechanism

Our baseline defense against eavesdroppers is to add obstructive noise to the environment to render the speech unintelligible. When noise is added to speech at a certain signal-to-noise ratio (SNR), the noisy speech mixture, $m(t)$, at time t is defined as:

$$m(t) = s(t) + 10^{\alpha/20} \cdot n(t) \tag{1}$$

where $s(t)$ and $n(t)$ are the 'clean' speech and noise signals, respectively. We multiply the noise signal by a scalar value to ensure that the noisy speech mixture has a desired SNR. α is calculated as:

$$\alpha = 20\log_{10}\left[\sqrt{\frac{\sum_{i=1}^{T}(s(i)-\mu_s)^2}{\sum_{i=1}^{T}(n(i)-\mu_n)^2}}\right] - SNR \tag{2}$$

where T is the length of the signal, SNR is the desired SNR level in decibels (dB), and μ_s and μ_n are the average values of the speech and noise signals, respectively. The SNR of a noisy speech signal is based on the relative total energy of the noise and speech components. SNR plays a crucial role in the performance of both perceptual (according to human evaluations) and automatic speech recognition. The SNR of a noisy speech signal is calculated as:

$$SNR = 10\log_{10}\left[\frac{\sum_{i=1}^{T} s^2(i)}{10^{\alpha/10}\sum_{i=1}^{T} n^2(i)}\right] \tag{3}$$

Next, we must determine if stationary or non-stationary noise should be added to the speech. The statistical properties of stationary noise remain unchanged over time as opposed to a non-stationary distribution, which varies with time. Typically, both speech recognition and speech separation techniques perform significantly worse when non-stationary noise is present [38]. Therefore, we decide to add non-stationary noise as our form of defense. Two different baseline non-stationary noises have been chosen – Babble and Cafe noises, where these noises come from the NOISEX dataset [56]. Babble noise contains many people talking simultaneously, and Cafe noise contains sounds from a cafeteria environment, which includes people talking, doors closing, and dishes clanking, to name a few. Bronkhorst et al. found that 'babble noise' is highly effective at masking speech (humans find it difficult to comprehend babble noise) [10]. Babble and cafe noise also present challenges to speech separation [63] and recognition [44]. Hence, we restrict $n(t)$ to babble and cafe noise types.

3.2.2 MyBabble: Proposed Defense Mechanism

For our proposed defense mechanism, user- (or target-) specific babble noise is generated by combining simulated speech signals of the user or the intended victim.

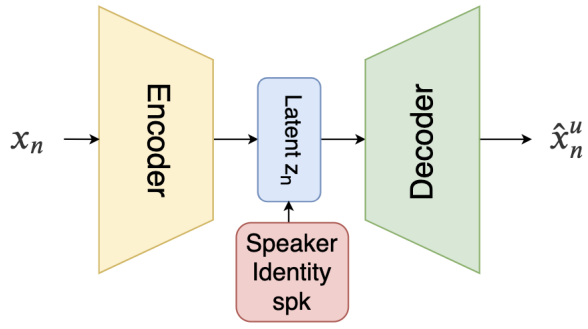


Fig. 1. A depiction of the model structure for the variational auto-encoding based voice conversion system.

User-simulated speech is generated by adapting a deep-learning-based voice conversion technique that transforms voice characteristics from one speaker to another (target) speaker [25]. This generates new utterances in the targeted speaker's voice. Multiple simulated speech signals are generated and combined to form the My-Babble noise, which complicates recognition (perceptually and computationally) for the eavesdropper because of the spectral overlap. Additionally, generating speech signals that the target user has not spoken previously acts as an extra layer of defense as the user can be assured that the attacker has not seen this data before, which weakens their potential attack.

We use a convolutional variational auto-encoder (VAE) for the voice conversion system, which is inspired by the work of Hsu et al. [25]. A VAE-based voice conversion model is chosen because training such a model is simpler since, unlike traditional Gaussian mixture model (GMM) based approaches [54], parallel speech corpora between multiple speakers are not needed. This means that time-aligned features from the source and target speakers are not needed, which broadens the possible utterances that can be generated since any speech utterance can be converted to the target speaker's voice. Secondly, this approach enables voice conversions purely from the audio data where phonetic and lexical information about the utterances are not needed. Typically, this information is often difficult to obtain as it requires experts in linguistics. VAE models have been shown to learn useful spectral and phonetic information that can be used on a broad range of input signals [40, 66].

Fig. 1 shows the model structure of our voice conversion system. The VAE-based voice conversion model contains two main parts – an encoder and a decoder. The model structure is similar to the originally proposed model [25]. However, we use convolutional networks instead of fully-connected feed forward DNNs since this allows utterance-level conversion as opposed to short-time level conversions. Convolutional neural networks (CNNs) have also been shown to better capture local spectral and temporal dependencies [55].Since speech has strong short-term correlations across time and frequency, we elect to use a convolutional based VAE instead of a DNN to better capture this information. Also, convolutional networks typically have fewer parameters, which reduces computational resources compared to fully connected feed-forward DNN models. Our implementation uses code by Hsu.[5] The network configuration and parameter values mentioned below are based on the recommendations from Hsu et al. [25], where they show good voice-conversion results. The encoder is composed of five convolutional layers with a kernel size of 7 and stride of 3. The encoder is represented by function h_ϕ. The decoder contains four convolutional layers with filter widths of 9, 7, 7, and 1025, respectively. The corresponding strides of each layer are 3, 3, and 1. Each convoluational layer is followed by a normalization layer. Leaky rectified linear (ReLU) activation functions are also used after each layer in order to apply non-linearities that help with estimation. The decoder operation is represented by function g_θ.

The input to the encoder contains three parts. The first part is the spectral magnitude S_i for the i^{th} frequency bin. The spectral magnitude is computed from the short-time Fourier transform (STFT) of the speech signal. The STFT is computed with the fast Fourier Transform (FFT) of size 1024, which results in 513 frequency channels (or bins). The second part of the input is the aperiodicity a_i for each frequency bin, which is the power ratio between the speech signal and the aperiodic component of the signal [42]. Lastly, the pitch, f_0, of the speech signal is used as the third component of the encoder input. Therefore, for each time frame, n, the encoder input x_n is expressed as

$$x_n = [S_1 \cdots S_{513}, a_1 \cdots a_{513}, f_0] \tag{4}$$

The input features are extracted by the WORLD vocoder [42, 43]. The encoder then generates a latent representation, z_n, as an output. This latent representation is concatenated with a variable *spk* that indicates the target converted speaker's identity. The concatenated vector, $[z_i, spk]$, then serves as the input to the decoder, which estimates the voice characteristics (e.g.,

5 https://github.com/JeremyCCHsu/vae-npvc

spectral, aperiodicity, and pitch) of the targeted speaker (e.g., user), $\hat{x}_n^u$.

$$\begin{aligned} Encoder &: z_n = h_\phi(x_n) \\ Decoder &: \hat{x}_n^u = g_\theta([z_n, spk]) \end{aligned} \tag{5}$$

From the original VAE Voice Conversion paper, the author uses 150 utterances per user to extract the voice characteristics. Therefore, we suggest at least 150 utterances from each target speaker should be used to train the voice conversion system. In our experiment, we used 720 utterances from two different speakers to separately learn the voice characteristics of the two speakers. Future efforts will assess performance with differing amounts of training utterances. The basic assumption of the VAE is that the encoder output should obey a standard normal distribution; if not, it should be penalized [31]. If we do not include the regularizer, the encoder could learn to cheat and give each datapoint a representation in a different region of Euclidean space [1]. Therefore, given encoder parameters, ϕ, decoder parameters, θ and input x_n, the objective function for training the VAE is expressed as:

$$\begin{aligned} \hat{\mathcal{L}}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \boldsymbol{x}_n\right) = &- D_{KL}\left(q_\phi\left(z_n|\boldsymbol{x}_n\right) \| p\left(\boldsymbol{z}_n\right)\right) \\ &+ \log p_\theta\left(\hat{x}_n^u|z_n, \boldsymbol{spk}\right) \end{aligned} \tag{6}$$

$D_{KL}(\cdot\|\cdot)$ calculates the Kullback-Leibler divergence (KLD) between the approximate, $q_\phi\left(z_n|\boldsymbol{x}_n\right)$, and the true, $p\left(\boldsymbol{z}_n\right)$, posterior probabilities. It serves as a regularizer to ensure that $p(z_n)$ has a standard normal distribution. $q_\phi(\cdot)$ and $p(\cdot)$ are functions that calculate the approximate and true posterior probabilities. They are both modeled as normal distributions with diagonal covariances

$$\begin{aligned} q_\phi &= \mathcal{N}\left(\boldsymbol{z}_n; \boldsymbol{\mu}_{\boldsymbol{z}_n}, \operatorname{diag}\left(\boldsymbol{\sigma}_{\boldsymbol{z}_n}\right)\right) \\ p_\theta &= \mathcal{N}\left(\hat{\boldsymbol{x}}_n^u; \boldsymbol{\mu}_{\boldsymbol{x}_n}, \operatorname{diag}\left(\boldsymbol{\sigma}_{\boldsymbol{x}_n}\right)\right) \end{aligned} \tag{7}$$

where $\boldsymbol{\mu}_{\boldsymbol{z}_n}$ and $\boldsymbol{\sigma}_{\boldsymbol{z}_n}$ are the mean and standard deviation for the latent representation, z_n; and $\boldsymbol{\mu}_{\boldsymbol{x}_n}$ and $\boldsymbol{\sigma}_{\boldsymbol{x}_n}$ are the mean and standard deviation for the input feature, x_n. The second term in Equation (6) measures the reconstruction quality. This term equals $log(1)$ if x_n is perfectly reconstructed.

3.3 Attack Method: Speech Recognition

Two state-of-art speech recognition models serve as the main form of attack. The first one is Google's Speech-to-Text automatic speech recognition (ASR) system that was developed by Google Brain [15]. The model developed by Google uses a multi-headed attention-based neural encoder-decoder architecture. The model is first trained on 12,500 hours of hand-transcribed utterances extracted from Google voice-search data. In order to improve robustness to noise, the system is then trained a second time with noisy speech data that combines clean speech utterances with noise from daily life events that are captured from YouTube videos [47]. Google's Speech-to-text ASR system is powerful because of the large training data set they use.

The second ASR model is the Deep Speech 2 ASR system from Baidu Research [5]. Compared to Google's Speech-to-Text ASR system, Deep Speech 2 is easier to customize and retrain. In this paper, the model is a Connectionist Temporal Classification (CTC) based Recurrent Neural Network (RNN) that has two convolutional layers that are followed by four recurrent layers.

Both ASR systems are end-to-end speech recognition models. Unlike traditional hybrid ASR systems that require hand-crafted input features and expert knowledge in linguistics [21], end-to-end speech-recognition systems take either an unprocessed time- or time-frequency domain input signal and jointly learn all the components of the speech recognizer without prior expert knowledge. Compared to hybrid ASR systems, however, these models normally require a large amount of data for acceptable performance. Both ASR systems use hidden Markov models (HMMs) that enforce character- and word-level language constraints to minimize errors and to ensure that the most likely word transcription is produced.

3.4 Evaluation Method

We use word error rate (WER) to evaluate ASR system performance and the attacker's ability to automatically obtain useful speech information. WER is calculated by first identifying the number of words that the ASR system correctly recognizes. Then, the total number of incorrect word substitutions W_S, deletions W_D and insertions W_I are also counted by comparing the ASR system's output transcription to the ground truth transcription. WER is computed by dividing the sum of substitutions, deletions, and insertions by the total number of words N_W in the reference transcription.

$$\text{WER} = \frac{W_S + W_D + W_I}{N_W} \times 100 \tag{8}$$

4 Experiments and Results

We now step through our evaluation of attacks and defenses in different adversarial settings.

4.1 Datasets

We use three different English speech corpora to evaluate our proposed approach. The first is the TIMIT corpus [19] that contains 6,300 sentences spoken by 630 native English speakers from eight dialect regions in the United States. The TIMIT corpus is often used for speech recognition studies [21, 22] since it is phonetically rich and useful for speaker-independent studies. The TIMIT dataset has been pre-seperated into training and test datasets that do not share the same set of speakers. The training dataset contains 462 different speakers that provide 4,620 total clean speech signals. The testing dataset has 168 speakers with 1,680 spoken utterances.

We also use the IEEE speech corpus [28], which is spoken by two different speakers, one male and one female, who each utter the same 720 sentences. The IEEE corpus provides a relatively large dataset of utterances that are spoken by one person, which can be used to simulate an attack on a single user. In this project, 520 utterances are used for training, 100 for development, and the last 100 are used for testing proposes for each speaker.

Finally, we use the LibriSpeech corpus [46], which is derived from the LibriVox project [48]. It contains 1,000 hours of speech. We use 30,000 clean speech signals (about 100 hours) to train the Deep Speech 2 ASR system since this number results in sufficient phonetic coverage while also ensuring that the system can be trained in a reasonable amount of time. We then randomly select a single speaker to use for the voice conversion task. This random male speaker generated 118 utterances that are used for training our voice conversion system.

4.2 Basic Defense: Inject Generic Noise

In this basic scenario, we assume that attackers do not have prior information about the victim or their environment. The attacker only observes the audio signal that the device (e.g., cell phone) captures. Therefore, the attacker uses an ASR system that is trained with a large and diverse speech dataset that contains speech utterances spoken by thousands of individuals. Audio from the target victim is not contained in this dataset. We use the pre-trained Google Speech-to-Text ASR system [15], which is trained on 12,500 hours of data, since it has been shown to perform well for different speakers and environments [47].

The victim is aware of potential attacks, so they can play obfuscating noise at different amplitudes as a defense mechanism. This scenario is simulated by mixing noise with speech signals from the TIMIT testing dataset at different signal-to-noise ratios (SNRs). Two different non-stationary noises, Babble and Cafe, are separately combined with the clean speech at various SNRs in order to minimize the recognition efficiency of the chosen ASR system [44]. The noisy speech signals are provided as inputs to the Google Speech-to-Text ASR system.

Table 1. WER for noisy speech, as a function of noise and SNR.

SNR (dB)	Babble	Cafe
-15	100.00%	100.00%
-13	100.00%	100.00%
-10	99.84%	99.75%
-8	99.24%	98.67%
-5	93.67%	91.93%
-3	82.46%	82.39%
0	62.34%	64.41%
5	41.75%	43.87%
10	35.74%	36.67%
Clean	15.34%	15.34%

4.2.1 ASR Results

Table 1 shows the average WER at each noise and SNR level. We can see that the error is relatively low when clean speech is provided to the ASR system, where an average WER of 15.34% is observed. At a 10 dB SNR, Babble noise achieves a 35.74% WER, while Cafe noise results in a 36.67% WER, which are about 20% higher than the clean speech case. We then notice that the word-error rate increases with decreasing signal-to-noise ratio for each noise, where the results are similar at each SNR for Babble and Cafe noises. This occurs because the noise becomes more dominant than the speech at lower SNRs since the total intensity of the noise increases beyond that of the speech. This makes word recognition much more difficult. Both Babble and Cafe noises reach 100% WER at a -13 dB SNR, which shows that Babble and Cafe noise, at the right SNR, can completely mask the target speech and prevent an

attacker from using a speaker- and environmentally-unaware ASR system for eavesdropping. Fig. 7a shows the spectrogram for a random clean speech signal from the TIMIT corpus and 7b shows the spectrogram for the noisy speech signal that combines the clean speech from Fig. 7a with Babble noise at a -5 dB SNR. The figure shows that the noise masks much of the speech, especially at low frequencies, which hinder recognition capabilities.

Table 2. Computed intelligibility (STOI) of the noisy speech signals. Lower scores indicate lower intelligible speech.

SNR	Babble	Cafe
-15	0.34	0.35
-13	0.37	0.37
-10	0.42	0.42
-8	0.46	0.45
-5	0.53	0.52
-3	0.58	0.56
0	0.65	0.64
5	0.77	0.76
10	0.86	0.86

4.2.2 Intelligibility Results

We also evaluate the computational intelligibility of the noisy speech signals, as this serves as a measure of human-level intelligibility. This is done in case the attacker employs human listeners for speech recognition. We evaluate intelligibility using the short-time objective intelligibility measure (STOI) [53]. STOI outputs scores between 0 and 1, where 1 means perfect intelligibility and 0 means that the signal is completely unintelligible. STOI has been shown to have strong correlations with intelligibility that is measured by human listeners [62]. STOI computes scores by comparing the correlation between the signal of interest (e.g., noisy speech) and the clean speech signal over short-time segments of a human-inspired time-frequency representation. Table 2 shows the STOI scores for the Babble and Cafe mixtures at different SNR levels.[6] At an SNR of 10 dB, Babble and Cafe mixtures both have STOI scores of 0.86, which means that the noisy speech signals are mostly intelligible. However, as the SNR level decreases, the STOI scores also decrease, which shows that human-level intelligibility also decreases with the SNR. More specifically, at -15 dB, STOI scores drop to 0.34 and 0.35, respectively, for Babble and Cafe mixtures. Therefore, injecting Babble and Cafe noises at specific SNR levels dramatically lowers intelligibility for both human listeners and ASR systems.

4.3 Advanced Attack: The Victim's Speech Data is Obtained

In this section, we examine a stronger adversary. We now assume that the attacker has gained access to clean speech data from the user, where this data is used to retrain an ASR system to improve recognition performance. This is a common technique that is based on transfer learning, and it has been shown to improve performance in similar scenarios [34].

We simulate this scenario by using the IEEE corpus and the Deep Speech 2 ASR system [5]. We expect that a personalized attack from a system such as Google ASR would improve the attacker's performance, in general. However, this is a proprietary system that we do not have access to modify. Therefore, we choose another state-of-art speech recognition model: Deep Speech 2 from Baidu. The implemented Deep Speech 2 model contains two 2-D convolutional layers, four bi-direction recurrent layers, and one fully connected layer. We separate the 720 IEEE utterances from the male and female speakers into 520 separate utterances for retraining the Deep Speech 2 model, 100 separate signals for development, and 100 separate signals for testing. Hence, separate models are made for the male and female speakers to ensure consistency across both genders. The ASR systems are initially trained with 30,000 random clean samples from the LibriSpeech speech corpus. This trained system is then separately retrained with the 520 male utterances for the male ASR system and 520 female utterances for the female ASR system. In each case, the development signals are used for parameter tuning.

The 100 IEEE testing speech signals for each gender are combined with the Babble and Cafe noises at {10, 5, 0, -5 ,-10 , -15} SNRs. These signals are then provided to the pretained ASR system (from LibriSpeech data only), and the system is re-trained with user speech data. The re-trained system is denoted as 'user aware,' whereas the initial ASR system is denoted as 'user unaware.' The WER across each SNR and noise type are also calculated. This training and testing approach is shown in Fig. 2.

6 STOI scores of 0.5 correspond to human-level intelligibility rates between 20% and 50%. STOI scores around 0.6 correspond to 60% to 80% human-level intelligibility [53].

Table 3. WER comparison between user-aware and user-unaware ASR systems.

	Babble				Cafe			
	Male		Female		Male		Female	
SNR (dB)	user unaware	user aware	user unaware	user aware	user unaware	user aware	user unaware	user aware
-15	100.00%	99.63%	100.00%	99.88%	100.00%	99.88%	100.00%	99.83%
-10	100.00%	99.52%	100.00%	99.62%	100.00%	99.75%	100.00%	99.75%
-5	100.00%	99.26%	99.50%	98.12%	100.00%	99.88%	100.00%	99.68%
0	98.64%	99.26%	99.38%	97.63%	100.00%	98.51%	99.88%	98.50%
5	93.42%	92.31%	97.02%	96.03%	97.15%	95.16%	97.64%	97.89%
10	74.81%	71.09%	79.53%	76.43%	82.63%	80.77%	87.47%	87.72%
Clean	31.76%	27.67%	30.77%	26.55%	31.76%	27.67%	30.77%	26.55%

Table 3 shows the WER results for the noisy speech signals using the user-unaware and user-aware ASR models. Generally, the WER improves (lowers) when the ASR system is trained with user data (compare 'user unaware' vs. 'user aware' columns). The performance improvement occurs at each SNR (including clean speech), noise, and for male and female speakers. The average WER decreases by 1.56% for Babble noise and 1.13% for Cafe noise. The average WER improvement also increases with SNR. On average, the user-aware system increases recognition performance by 1.42% for the IEEE male and 1.27% for the IEEE female. These results demonstrate that the attackers can have modest performance gains when only a small amount of user speech data is obtained. It is likely that further performance gains will occur if more data is utilized, so stronger defense mechanisms may be needed. Furthermore, as we show in Section 4.4.2, Google's ASR significantly outperforms these approaches, and we need a better understanding of how user-aware models might perform as compared to user-unaware models for such a system.

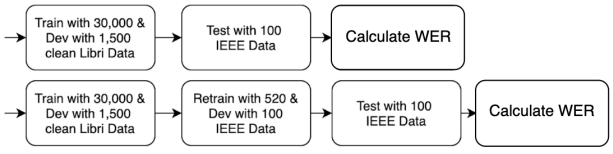


Fig. 2. Flow charts for the user-unaware (top) and user-aware (bottom) ASR systems.

In addition to this attacker scenario, we also tried to enhance the attacker by allowing them to use state-of-art speech separation techniques to remove our injected noise before recognizing the speech. The results are mixed, with WER *increasing* with speech separation. In some cases where WER decreased, it was by a small amount (1–2%). Further details can be found in Appendix C.

4.4 Proposed Defense: Specially Crafted MyBabble Noise

In this section, we show recognition results when our user-specific babble noise, based on the user's voice characteristics, is generated and combined with speech. The user-specific babble noise is generated by combining multiple outputs from the voice conversion VAE model.

4.4.1 System Setup and Result

First, we choose a random speaker from the LibriSpeech speech corpus that is not in the training and development utterances that are used to train the Deep Speech 2 ASR model. This speaker provides 118 clean speech signals. We also use the IEEE male and female speech data (720 utterances each). There are a total of three different speakers with 118, 720, and 720 utterances, respectively. We train the VAE model to learn the voice characteristics of the IEEE male and IEEE female speakers. The model then converts the 118 utterances from the chosen LibriSpeech speaker to sound like the IEEE male and the IEEE female speakers.

After the 118 converted speech signals are estimated, we produce different combinations of these utterances and add them together to serve as user-specific babble noise (one for the IEEE male and one for the IEEE female). The new noise will be tested based on the strongest attacker scenario from Section 4.3. WER is calculated and compared with the baseline noises for each gender.

Fig. 3 shows the WER differences between the user-specific babble noise that we generate and the generic noises. At low SNR levels (e.g., -15 to 0 dB), the dif-

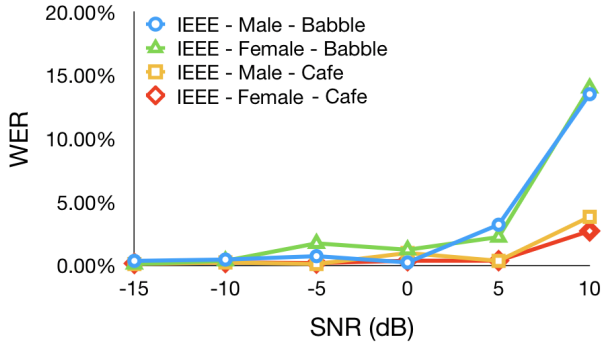


Fig. 3. WER differences between MyBabble noise and generic babble noise on an user-aware ASR system.

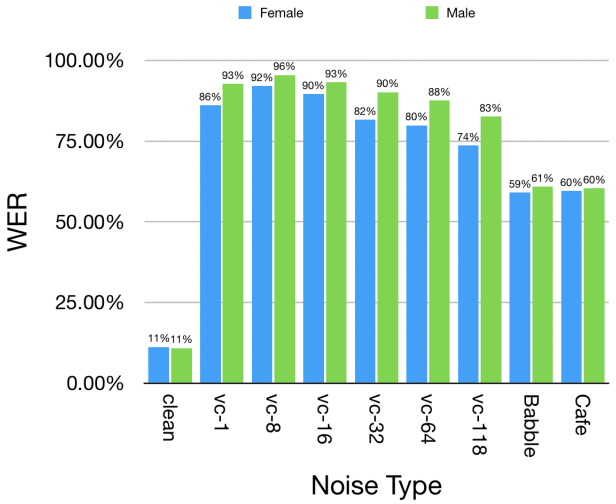


Fig. 4. WER results with different number of voice-converted utterances for 0 dB noisy speech mixtures.

ferences are small because the WERs are already near 100%. As the SNR level increases, the differences become more pronounced. Compared to generic Babble noise, the WER increases by 13.52% for male utterances and 14.02% for female utterances at 10 dB SNR. For generic Cafe noise, a 3.85% increase is shown for the male and 2.73% is shown for the female at 10 dB SNR.

By comparing our results with the baseline noises, we see that the WER increases for each gender at all SNR levels, especially at high SNR levels. The results imply that user-crafted noise is much better than the generic noise at impeding word recognition. By using the synthetic speech signals as noise, the state-of-art ASR system is mislead by which signal serves as the target one and which signals are the background noise.

Although these performance gains appear small, as we show in the next subsection, Google's ASR system (even though not trained on the target speaker) is able to significantly outperform the models tested in this subsection. The WERs at 0 dB SNR for Google's ASR go down to about 60% as opposed to nearly 100%. Thus, the next subsection demonstrates the strength of MyBabble.

4.4.2 Varying Number of Speech Tracks in MyBabble

In this section, we focus on attaining a high WER against Google's ASR, which outperforms the previous user-specific models. It is not possible for us to obtain or train Google's ASR model tuned to a target user, but our results in the previous subsection indicate that the incremental benefit is likely to be limited to a 'few percent.' Thus, we aim for WERs in the 90–95% range (compared to 60–61% for Cafe and plain Babble) in our approach against Google's ASR. We choose a 0 dB SNR as this represents a reasonable SNR at which the noisy speech starts to become unintelligible to human attackers as well [23, 24]. In practice, using noise at -5 dB SNR would be highly effective at this baseline.

We would also like to determine how varying the number of user-generated utterances impacts recognition performance. We test six different babble noise configurations. The user-specific babble noise is combined with different IEEE speech signals at a 0 dB SNR. The number of generated utterances varies between 1, 8, 16, 32, 64, and 118.

Fig. 4 shows the WER results for the different noise realizations. For both IEEE male and female signals, the MyBabble noise that is generated from 8 voice-converted signals (vc-8) produce the highest WERs. The WER gradually decreases as more converted speech signals are added. For IEEE male signals, voice-converted noise with 8 signals has a 95.53% WER. This is much higher than the baseline Babble noise, which produces a 60.96% WER, even though the babble noise consists of speech from 20 different spearkers. The Cafe noise achieves a 60.41% WER. For IEEE female signals, voice-converted noise with 8 signals (vc-8) outperforms all other noises with a 92.10% WER. The generic Babble and Cafe noises perform similarly to the same IEEE male scenario, with 59.07% and 59.54% WERs, respectively.

We surmise that vc-8 noise has the best performance because as more soundtracks are added, each soundtrack creates too much overlap and eventually the noise will sound like generic babble noise (poorer defense). If too

few soundtracks are used, there will be gaps revealing the target's speech. Although we make a first step, more work is needed to study defending multiple speakers simultaneously.

Finally, we checked the STOI scores at 0 and -5 SNRs respectively. MyBabble attains scores of 0.62 and 0.52 respectively. Thus, we recommend, in practice, MyBabble be used at a -5 dB SNR. (We later verify real human unintelligibility with a user study as described in Section 4.6.)

4.5 Real-World Experiment and Results

We further test the performance of our approach in a real-world home environment using three different phone devices. We assumed that the third device was hacked by a malicious attacker. The first device plays the speech signals. The second device then plays MyBabble noise (vc-8), while the third device records both sounds.

The conversation speaker and noise speaker were 20 cm and 2 cm from the recording device, respectively. This setup mimics how one might use our noise generator in the real world since this distance and chosen SNR ensures that the user will not be disturbed by the noise that is played. The noise and conversation signals are both played at 50 dB (SPL) as measured at the recording device, which results in a 0 dB SNR level. Then, the IEEE male vc-8 MyBabble noise is played simultaneously with 100 sequential-played IEEE male test samples. The same setup occurs with the IEEE female vc-8 noise and the 100 IEEE female test samples. After these sounds are recorded, they are provided as inputs to Google's Speech-to-Text ASR system.

The WER for both genders is 100%, which confirms our simulation results and shows that our specially crafted MyBabble noise is effective in practice. We omit the -5 dB results since 0 dB signals produce a 100% WER.

4.6 Human Intelligibility Study

Next, we conducted an ethics-board approved user study to evaluate how well the generated noise obstructs speech against human attackers.

Table 4. No. of Participants by condition in the human intelligibility study

Condition	Female	Male
Clean speech	17	24
Generic babble at 0 dB	15	16
MyBabble at -5 dB	13	20
MyBabble at 0 dB	17	15
MyBabble at 5 dB	20	17
MyBabble at 10 dB	18	18

4.6.1 Participants

We recruited 210 participants from Amazon's Mechanical Turk online recruitment system. Participants were required to be 18 years or older and have been living in the United States for a minimum of five years. They were asked to use headphones for this survey and perform the study in a quiet environment. They were also asked to confirm that they had normal hearing and were native or bilingual, professional-level English speakers. Detailed demographics can be found in Appendix B.

4.6.2 Procedure

All procedures were carried out in accordance with a protocol approved by our institution's review board for the conduct of human research. After completing the informed consent form, participants were presented with 20 audio clips. The audio clips were the same as the test data described in previous sections and were selected from the 100 test utterances from IEEE data corpus. Participants were compensated $3. Through a pilot study, we confirmed this was 'fair compensation' according to the participants (through a free-text question specifically asking about fair compensation) based on the amount of work, which was approximately 15 minutes per participant (the median time for the full user study was 13 min 28 sec).

Each audio clip was mixed with the baseline noise and our proposed noise at different SNRs. Each participant was randomly assigned to one of the mixture conditions where they were asked to type in the words that they heard. Detailed participation distribution can be found in Table 4.

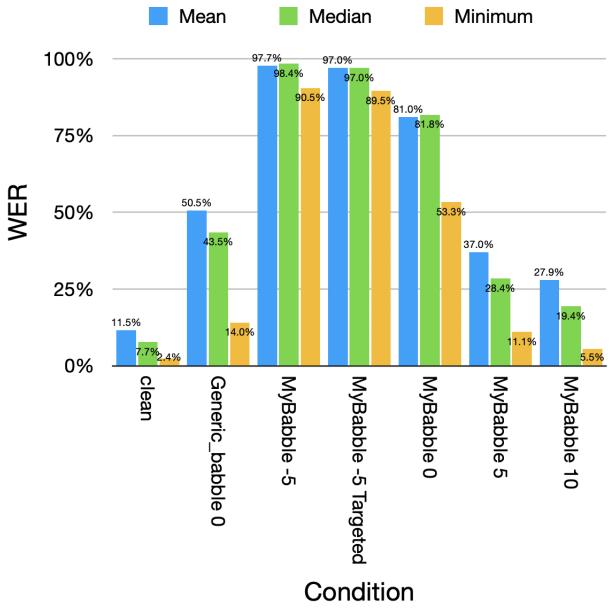


Fig. 5. WER results for the user listening study

4.6.3 Results

Fig. 5 shows the word error rate (WER) results for the participants in our user study. The study shows similar trends as observed in Section 4.4.2 where MyBabble noise provides a stronger defense than the baseline generic babble noise. For the voice signals, participants performed well at transcribing clean speech (11.5% mean and 7.7% median WER). For the noisy speech mixtures, the 8-soundtrack MyBabble mixture reached a WER of 81% mean and 81.8% median at a 0 dB SNR level, which is much higher than the baseline Babble noise that reached a 50.5% mean and 43.5% median WER. The WER increased to a 97.7% mean and 98.4% median if we lower the SNR level to -5 dB. However, if we use a SNR level higher than 0 dB, which indicates that the speech energy is higher than the noise energy, human listeners can easily recognize the speech. The mean WER at a 5 dB SNR drops to 37% and the median WER drops to 28.4%.

Even though we reach an average of 98% WER at a -5 dB SNR, we examined which words the human listeners identified correctly. From the transcripts of the participants who obtained the best score at -5 dB, we found that normally the words they got correct were prepositions such as “to,” “in,” and “before,” and articles such as “the” and “a.” For example, the sentence “a streak of color ran down the left edge” is transcribed as “the edge.” The sentence “crouch before you jump or miss the mark” is transcribed as “before.” In conclusion, compared to other defensive approaches, MyBabble noise mixed with speech at -5 dB SNR is a strong defense mechanism against human-based eavesdropping attacks.

4.6.4 Followup Study: Human Attackers with Prior Knowledge

We conducted a followup user study by adding an extra condition to the previous study with 61 participants to simulate attackers who are more familiar with the target speech. The requirements were the same as the previous study, and the demographics can be found in Table Appendix B. Unlike the previous experiment, we asked the participants to first transcribe 20 clean speech samples from the target speaker to familiarize them with the target speaker’s voice. They were then asked to transcribe 20 different sentences mixed using MyBabble noise at -5 dB SNR as with the previous study. The ‘MyBabble -5 Targeted’ column in Fig. 5 shows the results for this experiment. A high WER is maintained even when the attacker is familiar with the target’s voice – the mean, median and minimum WERs are 97.0%, 97.0% and 89.5%, respectively. The results show that our approach remains robust against strong human adversaries.

4.7 Disturbance Level Experiments

In this section, we conduct realistic experiments to determine if MyBabble noise is disturbing to nearby users, where headphones are used to ‘inject’ noise into the microphone of the compromised device.

4.7.1 Experiment Setup

For this experiment, we play MyBabble noise through a pair of consumer headphone buds (‘earpods’) to demonstrate the practicality of injecting MyBabble while not disturbing people in the vicinity. We then use a professional NIST-certified sound level meter[7] to measure the real-time sound level in dB.

First, we measured the sound level when speech and noise are not present. This serves has a baseline control

7 REED Instruments R8060 Sound Level Meter with Bargraph, Type 2, 30 to 130 dB

for our experiment. Then, the earpods were placed at 5 different distances from the meter to measure the sound level that would be perceived by people. We place the sound meter at 0 cm (e.g., directly connected), 10 cm, 25 cm, 50 cm, and 100 cm from the earpods. Those distances simulate different situations when users stay in the room with their phone. In practice, we would expect people to keep their phone 'at a distance' while applying the MyBabble defense. Although we expect the phone to be kept 'a couple of meters away' in general, we pick shorter distances to cover situations when one uses the defense 'bedside,' e.g., next to one's alarm clock at night. The 0 cm condition is also used to enforce the desired sound-pressure level (SPL) when we play MyBabble noise and allows us to pick the appropriate SNR as heard by the microphone of the device we are protecting. The other four distances simulate what a human listener might hear when they are at the different distances from the noise producer.

If the sound level at a certain distance is close to the sound level of the quiet room, we can say that the disturbance level is low and that the user will not be affected by the noise. Both male and female noise are played for 90 seconds during each condition. The sound level is then measured for each condition every 10 seconds. The mean sound level for each condition is reported. Conversational speech is generally between 60 and 70 dB [8]. Therefore, in order to obtain a -5 dB SNR, we constrain the sound level at 0 cm to 75 dB so that the resulting mixture has a -5 dB SNR at best.

4.7.2 Results

Fig. 6 shows the disturbance-level results. As the distance between the noise source and sound level meter increases, the corresponding sound level decreases accordingly. The mean sound level for MyBabble noise at 10 cm, 25 cm, 50 cm, and 100 cm are 42.03 dB, 36.12 dB, 33.17 dB, and 31.91 dB, respectively. Compared to the sound level of the quiet room, which is 30.21 dB, the difference is negligible if the distance is 50 cm or higher. Therefore, we can conclude that users will not be disturbed if they are 'a couple of feet' away from the noise producer.

4.7.3 Disturbance-level User Study

Finally, to verify our experimental results, we conduct an ethics-board approved disturbance-level user study

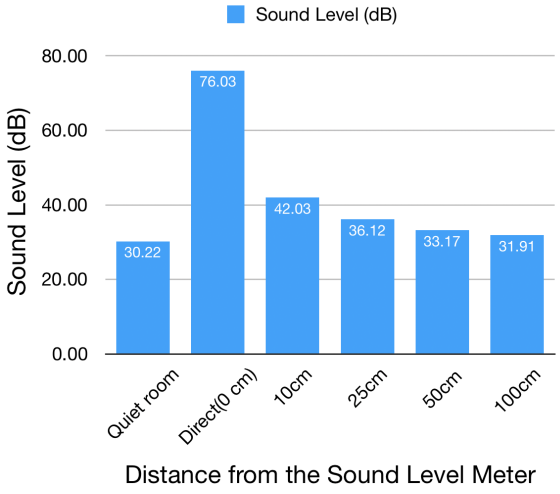


Fig. 6. Mean Sound Level (dB) by distance to user

with 134 participants to understand how noticeable and disturbing MyBabble noise is to users. The requirements of the participants are the same as the previous two user studies, and the detailed demographics of the participants can be found in Appendix B. Participants on Amazon Mechanical Turk were paid $1.50 based on an estimated duration of six minutes for our study.

Although it would have been ideal to conduct this experiment in person, due to the COVID-19 pandemic, we could not perform in-person experiments. To best simulate a realistic environment, we mixed the MyBabble noise with five different clean daily conversations at {35,40,45,-5} dB SNRs to simulate how participants would hear a conversation with corresponding MyBabble noise at a distance. These mixtures were carefully calibrated; the three SNRs cover all possible distances between the target speaker and the noise producer based on the previous study.

We use two controls: The -5dB mixture is what the attacker receives. Our participants should show low comprehension for this mixture. We use clean speech as the other control. Participants should show high comprehension for this condition. Participants in our study were assigned randomly to one of the noise conditions (clean, 35dB, 40dB, 45dB, or -5dB mixtures), and then presented with five one-minute conversations at that noise level. For each of the five conversations, participants were asked five questions. The first two questions ask participants about the content of the conversation to make sure they were paying attention (e.g., a multiple choice question about the topic of the conversation). The next three questions asked participants how

strongly they agreed or disagreed with the following statements:

Q1: The conversation was easy to understand.
Q2: There was noticeable background noise.
Q3: The background noise was disturbing.

Each question was answered with a 5-point Likert item (1 - Strongly disagree, 2 - Disagree, 3 - Neither agree or disagree, 4 - Agree, and 5 - Strongly agree).

Table 5 shows detailed results of how participants answered these questions across the conditions. In summary, we find that the mean values for Q1, Q2, and Q3 of clean speech and mixtures at {35,40,45} dB are very similar (our statistical tests also show no statistically significant differences). The clean speech received average scores of 4.19, 2.48, and 1.94 for the three questions. 35dB noisy mixture scores are really similar, which are 4.32, 2.33, and 1.87. The 40dB and 45dB mixtures yielded similar results: 4.2, 2.28, and 1.98 for the 40dB mixture, and 4.27, 2.29, and 2.00 for 45dB mixture. For the noisy control group, the mean values of the -5dB mixture were 1.81, 4.8, and 4.7 showing that participants did not find the conversation easy to understand, could definitely notice the background noise, and found it disturbing. A t-test shows that there is a statistically significant difference between the clean speech and -5dB noisy mixture.

To conclude, if the noise producer is more than 10 cm away, people employing our defense will not find the setup disturbing any more than during a conversation *without* any added noise.

Table 5. Disturbance Level User Study Result

	Mean	Std	t-value	t Critical	P-value
Q1					
Clean	4.19	0.86			
35dB	4.32	0.74	-1.37	+/- 1.9681	0.17
40dB	4.20	0.81	-0.08	+/- 1.9694	0.94
45dB	4.27	0.62	-0.86	+/- 1.9692	0.39
-5dB	1.81	1.19	17.20	+/- 1.9709	0.00
Q2					
Clean	2.48	1.17			
35dB	2.34	1.17	0.99	+/- 1.9680	0.32
40dB	2.28	1.17	1.32	+/- 1.9694	0.19
45dB	2.29	1.16	1.26	+/- 1.9692	0.21
-5dB	4.80	0.51	-18.49	+/- 1.9709	0.00
Q3					
Clean	1.94	0.85			
35dB	1.87	0.86	0.67	+/ -1.9691	0.51
40dB	1.99	1.14	-0.34	+/ -1.9694	0.73
45dB	2.00	1.07	-0.48	+/ -1.9692	0.63
-5dB	4.71	0.62	26.98	+/ -1.9709	0.00

4.8 Two-Speaker Scenario

We conducted additional experiments that consider a two-speaker conversational scenario to examine how MyBabble might simultaneously protect *two* speakers in a conversational setting. We use the same methodology as previous sections but with speech data alternating between two speakers. We generated and combined multi-track MyBabble noise for each speaker and tested the effectiveness against the strongest adversary. The eight soundtracks used by MyBabble contains four specialized soundtracks for each speaker. We found this method of combined noise as a defense is still effective – the WER is about 25% at 10 dB SNR, 50% at 5 dB SNR, 92% at 0 dB SNR and close to 100% at -5 dB SNR.

5 Discussion

Better microphone designs for 'tangible privacy'. Smartphones and other IoT devices are currently poorly designed by having embedded microphones. We follow the view that defense approaches should be more 'tangible,' i.e., users should have confidence in their effectiveness [3]. In that sense, ultrasonic jamming does not provide users with any tangible notion that a defense is in place. Our work thus explores tangible microphone jamming approaches where 'babble noise' available to the user can provide such assurances about their privacy.

In the longer term, better hardware designs are needed so that microphones can be easily – and 'tangibly' – disabled instead of relying on external defenses. For example, these microphones may have a physical switch that convincingly breaks the physical circuit to disable recording [3]. Apple has started enforcing a "hardware disconnect"[8] of the microphone in their most recent designs, e.g., when a laptop is closed or the iPad case is closed. These approaches are a step in the right direction, although some users may still be unconvinced whether the microphone is 'really' disabled. For example, in the past, a 'hardware controlled' LED indi-

8 Hardware microphone disconnect in Mac and iPad: https://support.apple.com/guide/security/hardware-microphone-disconnect-mac-ipad-secbbd20b00b/1/web/1

cator for the Macbook laptop's camera was nevertheless hacked through its firmware [9]. Absent *convincing* approaches, one may have to rely on MyBabble-style noise generators to be assured of privacy. We note that it is not sufficient to 'just not use the device' or 'just turn it off.' In many cases, people will want to have their smart personal assistants or smartphones operational and in their vicinity with only the microphone disabled. Compare this scenario to the case where many people cover their laptop cameras instead of abandoning use of their laptops (or cameras) altogether.

Prototype and scalability considerations. We built a physical 'MyBabble Box' prototype of our envisioned defense (see Figure 8 in the Appendix), although we were unable to conduct a user study with this prototype because of the ongoing COVID-19 pandemic. We built a wooden housing to serve as a 'bedside cradle' for a smartphone. This approach maintains the usability of the phone screen and speaker (e.g., to serve as a music player and allow the user to observe notifications) yet allows for application of the MyBabble defense on the microphones. The box included an Adafruit Audio FX Sound Board to play the MyBabble defense through two uxcell 1.5W 8 Ohm Mini speakers into the smartphone's microphones. Although this prototype is effective at obfuscation, we note, for example, the 16MB limitation of the soundboard memory, which may not be practical for statically loaded MyBabble tracks as a long-term defense. An internet connection could be used for downloading and converting new tracks; however, future work should explore the application of generative adversarial networks (GANs) to automatically generate synthetic noise to foil the best possible ASR systems [20]. Non-GAN approaches, such as a regression-based approach by Donahue et al. [16], also provide insights and should be considered in future work.

Limitations. Although we studied sophisticated adversarial approaches with models trained on the victim's speech under noisy conditions, these adversaries did not perform better than Google's ASR system. If one is to assume powerful adversaries, e.g., where companies such as Google are legally forced to apply their models toward targeted adversaries, it is yet unknown how effectively noise-based defenses might perform against these models tuned to the victim's speech. Yet, based on our experiments, we predict small incremental gains in the face of adversarial noise. As another limitation of this work, we study only the single microphone setting. Application of our technique in practice should also examine how to defend against adversaries with access to multiple microphones (as is now available on many smartphone models), which can allow adversaries to better isolate a victim's speech.

6 Conclusions

We evaluate obfuscating noise as a defense against microphone-based eavesdropping attacks. Through a comprehensive evaluation of various attacker capabilities, we find that our personally crafted 'MyBabble' defense performs best and renders automated speech recognition (ASR) attacks (including Google's ASR system, which performed the best) as well as human-based attacks ineffective. Our proposed MyBabble noise mechanism uses voice-conversion techniques to generate synthetic speech signals that can be combined to form a user-specific 'babble' mixture as noise. These synthetic signals have the same voice characteristics as the target speaker, which further confuses ASR systems. Because of the type of 'babble' noise used, our approach is also effective against human attackers, which we validate through an accepted human-intelligibility metric as well as a user study. Furthermore, MyBabble can be injected into devices using commodity headphones at volumes barely noticeable to users and no more disturbing than background noise (i.e., environmental noise without MyBabble) as verified in a user study.

Although we take a first step toward effective noise-based defenses, we believe much more exploration is needed. Further research is needed in evaluating attacks that can perform better in the presence of noise-based defenses. In particular, it is important that researchers study not only 'adversarial noise' that can trick ASR systems but also those that retain unintelligibility against human adversaries – the resultant noise-speech mixture should be hard to discern by both humans as well as ASR systems.

7 Acknowledgments

This material is based upon work supported by the National Science Foundation under grant CNS-1814513. We would also like to thank Sai Teja Peddinti and our anonymous reviewers for their feedback.

References

[1] Tutorial - What is a variational autoencoder? https://jaan.io/what-is-variational-autoencoder-vae-tutorial/. Accessed: 2019-07-30.

[2] Masanobu Abe, Satoshi Nakamura, Kiyohiro Shikano, and Hisao Kuwabara. Voice conversion through vector quantization. *Journal of the Acoustical Society of Japan (E)*, 11(2):71–76, 1990.

[3] Imtiaz Ahmad, Rosta Farzan, Apu Kapadia, and Adam J. Lee. Tangible privacy: Towards user-centric sensor designs for bystander privacy. *Proceedings of the ACM Journal: Human-Computer Interaction: Computer Supported Cooperative Work and Social Computing (CSCW '20)*, 4(CSCW2):116:1–116:28, October 2020.

[4] Moustafa Alzantot, Bharathan Balaji, and Mani Srivastava. Did you hear that? Adversarial examples against automatic speech recognition. *arXiv preprint arXiv:1801.00554*, 2018.

[5] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International Conference on Machine Learning*, pages 173–182, 2016.

[6] Xavier Anguera, Chuck Wooters, and Javier Hernando. Acoustic beamforming for speaker diarization of meetings. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(7):2011–2022, 2007.

[7] Jon Barker, Shinji Watanabe, Emmanuel Vincent, and Jan Trmal. The fifth 'chime' speech separation and recognition challenge: Dataset, task and baselines. In *Proc. Interspeech*, pages 1561–1565, 2018.

[8] Braxton Boren, Agnieszka Roginska, and Brian Gill. Maximum averaged and peak levels of vocal sound pressure. In *135th Audio Engineering Society Convention 2013*, pages 692–698, United States, 2013. Audio Engineering Society.

[9] Matthew Brocker and Stephen Checkoway. iSeeYou: Disabling the Macbook webcam indicator LED. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 337–352, San Diego, CA, August 2014. USENIX Association.

[10] Adelbert W Bronkhorst. The cocktail party phenomenon: A review of research on speech intelligibility in multiple-talker conditions. *Acta Acustica united with Acustica*, 86(1):117–128, 2000.

[11] Douglas S Brungart, Peter S Chang, Brian D Simpson, and DeLiang Wang. Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. *The Journal of the Acoustical Society of America*, 120(6):4007–4018, 2006.

[12] Nicholas Carlini, Pratyush Mishra, Tavish Vaidya, Yuankai Zhang, Micah Sherr, Clay Shields, David Wagner, and Wenchao Zhou. Hidden voice commands. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 513–530, Austin, TX, 2016. USENIX Association.

[13] Nicholas Carlini and David Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 1–7. IEEE, 2018.

[14] Yuxin Chen, Huiying Li, Shan-Yuan Teng, Steven Nagels, Zhijing Li, Pedro Lopes, Ben Y. Zhao, and Haitao Zheng. Wearable microphone jamming. In *2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*, April 2020.

[15] Chung-Cheng Chiu, Tara Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J. Weiss, Kanishka Rao, Katya Gonina, Navdeep Jaitly, Bo Li, Jan Chorowski, and Michiel Bacchiani. State-of-the-art speech recognition with sequence-to-sequence models. 2018.

[16] Chris Donahue, Bo Li, and Rohit Prabhavalkar. Exploring speech enhancement with generative adversarial networks for robust speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5024–5028. IEEE, 2018.

[17] Lois L Elliott. Performance of children aged 9 to 17 years on a test of speech intelligibility in noise using sentence material with controlled word predictability. *The Journal of the Acoustical Society of America*, 66(3):651–653, 1979.

[18] Hakan Erdogan, John R Hershey, Shinji Watanabe, and Jonathan Le Roux. Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 708–712. IEEE, 2015.

[19] John S Garofolo. Timit acoustic phonetic continuous speech corpus. *Linguistic Data Consortium, 1993*, 1993.

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[21] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE workshop on automatic speech recognition and understanding*, pages 273–278. IEEE, 2013.

[22] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

[23] William Hartmann, Arun Narayanan, Eric Fosler-Lussier, and DeLiang Wang. A direct masking approach to robust asr. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(10):1993–2005, 2013.

[24] Eric W Healy, Sarah E Yoho, Yuxuan Wang, and DeLiang Wang. An algorithm to improve speech recognition in noise for hearing-impaired listeners. *The Journal of the Acoustical Society of America*, 134(4):3029–3038, 2013.

[25] Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min Wang. Voice conversion from non-parallel corpora using variational auto-encoder. In *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–6. IEEE, 2016.

[26] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Joint optimization of masks and deep recurrent neural networks for monaural source separation. *IEEE/ACM Transactions on Audio, Speech, and Language*

Processing, 23(12):2136–2147, 2015.

[27] Wen-Chin Huang, Tomoki Hayashi, Shinji Watanabe, and Tomoki Toda. The sequence-to-sequence baseline for the voice conversion challenge 2020: Cascading asr and tts. *arXiv preprint arXiv:2010.02434*, 2020.

[28] IEEE. IEEE recommended practice for speech quality measurements. *IEEE Trans. Audio Electroacoust.*, 17:225–246, 1969.

[29] Alexander Kain and Michael W Macon. Design and evaluation of a voice conversion algorithm based on spectral envelope mapping and residual prediction. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221)*, volume 2, pages 813–816. IEEE, 2001.

[30] Gibak Kim, Yang Lu, Yi Hu, and Philipos C Loizou. An algorithm that improves speech intelligibility in noise for normal-hearing listeners. *The Journal of the Acoustical Society of America*, 126(3):1486–1494, 2009.

[31] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[32] Jernej Kos, Ian Fischer, and Dawn Song. Adversarial examples for generative models. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 36–42. IEEE, 2018.

[33] Denis Foo Kune, John Backes, Shane S Clark, Daniel Kramer, Matthew Reynolds, Kevin Fu, Yongdae Kim, and Wenyuan Xu. Ghost talk: Mitigating EMI signal injection attacks against analog sensors. In *2013 IEEE Symposium on Security and Privacy*, pages 145–159. IEEE, 2013.

[34] Julius Kunze, Louis Kirsch, Ilia Kurenkov, Andreas Krug, Jens Johannsmeier, and Sebastian Stober. Transfer learning for speech recognition on a budget. *arXiv preprint arXiv:1706.00290*, 2017.

[35] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[36] Ning Li and Philipos C Loizou. Factors influencing intelligibility of ideal binary-masked speech: Implications for noise reduction. *The Journal of the Acoustical Society of America*, 123(3):1673–1682, 2008.

[37] Nathan Malkin, Joe Deatrick, Allen Tong, Primal Wijesekera, Serge Egelman, and David Wagner. Privacy attitudes of smart speaker users. *Proceedings on Privacy Enhancing Technologies*, 2019(4):250–271, 2019.

[38] Kotta Manohar and Preeti Rao. Speech enhancement in nonstationary noise environments using noise properties. *Speech Communication*, 48(1):96–109, 2006.

[39] George A Miller and Joseph CR Licklider. The intelligibility of interrupted speech. *The Journal of the Acoustical Society of America*, 22(2):167–173, 1950.

[40] Hiroyuki Miyoshi, Yuki Saito, Shinnosuke Takamichi, and Hiroshi Saruwatari. Voice conversion using sequence-to-sequence learning of context posterior probabilities. *arXiv preprint arXiv:1704.02360*, 2017.

[41] Seyed Hamidreza Mohammadi and Alexander Kain. An overview of voice conversion systems. *Speech Communication*, 88:65–82, 2017.

[42] Masanori Morise. D4C, a band-aperiodicity estimator for high-quality speech synthesis. *Speech Communication*, 84:57–65, 2016.

[43] Masanori Morise, Fumiya Yokomori, and Kenji Ozawa. World: a vocoder-based high-quality speech synthesis system for real-time applications. *IEICE TRANSACTIONS on Information and Systems*, 99(7):1877–1884, 2016.

[44] Arun Narayanan and DeLiang Wang. Ideal ratio mask estimation using deep neural networks for robust speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 7092–7096. IEEE, 2013.

[45] Arun Narayanan and DeLiang Wang. Investigation of speech separation as a front-end for noise robust speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(4):826–835, 2014.

[46] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[47] Rohit Prabhavalkar, Tara N Sainath, Yonghui Wu, Patrick Nguyen, Zhifeng Chen, Chung-Cheng Chiu, and Anjuli Kannan. Minimum word error rate training for attention-based sequence-to-sequence models. *arXiv preprint arXiv:1712.01818*, 2017.

[48] Kishore Prahallad. Automatic building of synthetic voices from audio books. *Diss. Nagoya Institute of Technology, Japan*, 2010.

[49] Nirupam Roy, Sheng Shen, Haitham Hassanieh, and Romit Roy Choudhury. Inaudible voice commands: The long-range attack and defense. In *15th $USENIX$ Symposium on Networked Systems Design and Implementation (NSDI 18)*, pages 547–560, 2018.

[50] Roman Schlegel, Kehuan Zhang, Xiao-yong Zhou, Mehool Intwala, Apu Kapadia, and XiaoFeng Wang. Soundcomber: A stealthy and context-aware sound trojan for smartphones. In *NDSS*, volume 11, pages 17–33, 2011.

[51] Joan Serrà, Santiago Pascual, and Carlos Segura Perales. Blow: a single-scale hyperconditioned flow for non-parallel raw-audio voice conversion. *Advances in Neural Information Processing Systems*, 32:6793–6803, 2019.

[52] Lifa Sun, Kun Li, Hao Wang, Shiyin Kang, and Helen Meng. Phonetic posteriorgrams for many-to-one voice conversion without parallel data training. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2016.

[53] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136, 2011.

[54] Shinnosuke Takamichi, Tomoki Toda, Graham Neubig, Sakriani Sakti, and Satoshi Nakamura. A statistical sample-based approach to GMM-based voice conversion using tied-covariance acoustic models. *IEICE TRANSACTIONS on Information and Systems*, 99(10):2490–2498, 2016.

[55] Ke Tan and DeLiang Wang. A convolutional recurrent neural network for real-time speech enhancement. In *Interspeech*, pages 3229–3233, 2018.

[56] Andrew Varga and Herman JM Steeneken. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise

on speech recognition systems. *Speech communication*, 12(3):247–251, 1993.

[57] Emmanuel Vincent, Jon Barker, Shinji Watanabe, Jonathan Le Roux, Francesco Nesta, and Marco Matassoni. The second 'CHiME' speech separation and recognition challenge: Datasets, tasks and baselines. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 126–130. IEEE, 2013.

[58] Deliang Wang. On ideal binary mask as the computational goal of auditory scene analysis. In *in Speech Separation by Humans and Machines*, pages 181–197. Kluwer, 2005.

[59] Yuxuan Wang, Arun Narayanan, and DeLiang Wang. On training targets for supervised speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 22(12):1849–1858, 2014.

[60] Yuxuan Wang and DeLiang Wang. Towards scaling up classification-based speech separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(7):1381–1390, 2013.

[61] Felix Weninger, Hakan Erdogan, Shinji Watanabe, Emmanuel Vincent, Jonathan Le Roux, John R Hershey, and Björn Schuller. Speech enhancement with lstm recurrent neural networks and its application to noise-robust ASR. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 91–99. Springer, 2015.

[62] Donald S Williamson, Yuxuan Wang, and DeLiang Wang. Reconstruction techniques for improving the perceptual quality of binary masked speech. *The Journal of the Acoustical Society of America*, 136(2):892–902, 2014.

[63] Donald S Williamson, Yuxuan Wang, and DeLiang Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(3):483–492, 2016.

[64] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee. A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(1):7–19, 2014.

[65] Xueliang Zhang and DeLiang Wang. Deep learning based binaural speech separation in reverberant environments. *IEEE/ACM transactions on audio, speech, and language processing*, 25(5):1075–1084, 2017.

[66] Huadi Zheng, Weicheng Cai, Tianyan Zhou, Shilei Zhang, and Ming Li. Text-independent voice conversion using deep neural network based phonetic level features. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2872–2877. IEEE, 2016.

[67] Li Zhuang, Feng Zhou, and J Doug Tygar. Keyboard acoustic emanations revisited. *ACM Transactions on Information and System Security (TISSEC)*, 13(1):3, 2009.

A Additional Figures

See Figure 7 (Spectrogram) and Figure 8 (MyBabble Box Prototype).

B Demographics

We provide demographic information of the participants from Sections 4.6.4, 4.6.1 and 4.7.3 (see Table 6).

C Speech Separation Attack

The eavesdropper may employ methods to remove injected noise to present a stronger attack that helps decrease WER and improve recognition at lower SNRs. This is simulated by training multiple speech-separation methods that serve as noise-reduction front ends to the ASR system. The noise-reduced signal is then provided as an input to the ASR system.

We use two different high-performing speech separation techniques to improve performance for the attacker. Wang et al. [58] introduced the ideal binary mask (IBM) approach, which assigns all noise-dominant units with a value of 0 and speech-dominant units with a value of 1, based on the SNR at each time (t) and frequency (f).

$$IBM(t,f) = \begin{cases} 1 & \text{if} \quad SNR(t,f) \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

$$SNR(t,f) = 10\log_{10}(X(t,f)/N(t,f)) \tag{10}$$

where $X(t,f)$ and $N(t,f)$ are the instantaneous energies of the speech and noise, respectively. When an IBM is applied to a noisy speech spectrogram, all points where noise dominates will be removed, while all speech-dominant points will be retained. This separation strategy is human inspired, as it is consistent with the processing that occurs within the human auditory system [39]. The IBM approach has also been shown to result in highly-intelligible speech [11, 36].

We also use an ideal ratio masking (IRM) approach as the front-end to the ASR system [44]. The IRM is a smoothed version of the IBM, where continuous values between 0 and 1 define the mask. Eq. (11) shows how the IRM is calculated from the speech and noise that are within a noisy speech signal.

$$IRM(t,f) = \frac{10^{(SNR(t,f)/10)}}{10^{(SNR(t,f)/10)}+1} \tag{11}$$

The IRM can be interpreted as the percentage of speech energy at a particular time-frequency point, so when it is applied to the noisy speech spectrogram, it results in an estimate of the true clean spectrogram. The IRM has been shown to be one of the top performing speech separation approaches [59].

Table 6. User Study Demographics table

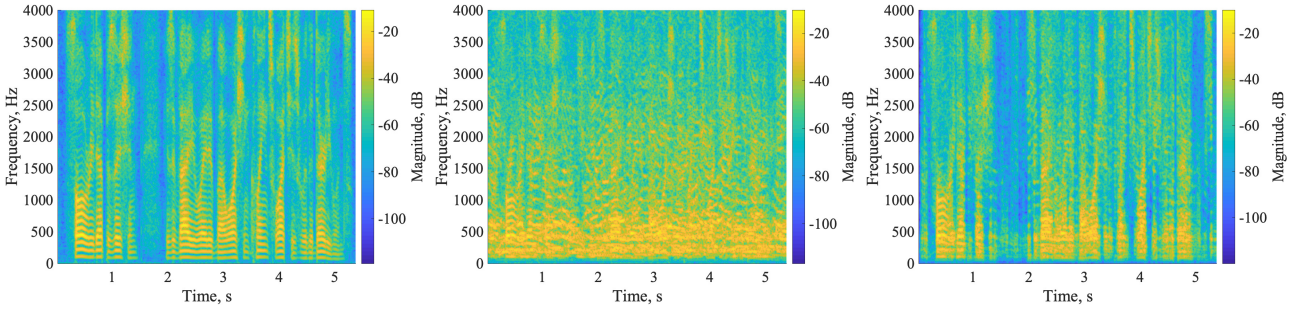
Condition	User study	User Study w/ Human Adv	Disturbance Level
Total	210	61	134
Gender			
Male	114 (54.29%)	31 (50.82%)	84 (62.69%)
Female	93 (44.29%)	28 (45.90%)	49 (36.57%)
Other	3 (1.43%)	2 (3.28%)	1 (0.75%)
Age			
18-29	74 (35.24%)	16 (26.23%)	40 (29.85%)
30-49	108 (51.43%)	38 (62.30%)	75 (55.97%)
50-64	27 (12.86%)	6 (9.84%)	16 (11.94%)
65+	1 (0.48%)	1 (1.64%)	3 (2.24%)
Education			
No High School	0 (0.00%)	0 (0.00%)	0 (0.00%)
High School	42 (20.00%)	20 (32.79%)	27 (20.15%)
Underguraduate	131 (62.38%)	31 (50.82%)	70 (52.24%)
Master's Degree	35 (16.67%)	9 (14.75%)	34 (25.37%)
Professional (MD, JD/PhD)	2 (0.95%)	1 (1.64%)	3 (2.24%)
Race			
Hispanic or Latino	15 (7.14%)	3 (4.92%)	7 (5.22%)
American Indian or Alaska Native	3 (1.43%)	3 (4.92%)	1 (0.75%)
Asian	51 (24.29%)	8 (13.11%)	22 (16.42%)
Black or African American	12 (5.71%)	6 (9.84%)	12 (8.96%)
Native Hawaiian or Other Pacific Islander	1 (0.48%)	0 (0.00%)	2 (1.49%)
White	128 (60.95%)	40 (65.57%)	90 (67.16%)
Other	0 (0.00%)	1 (1.64%)	0 (0.00%)

The IBM and IRM are oracle masks that must be estimated in real-world scenarios. Separate deep neural networks (DNNs) are used to estimate these masks from the given noisy speech mixtures, since DNNs have outperformed other estimation approaches [26, 61, 64]. The DNNs have 4 hidden layers with 1024 sigmoid units in each layer. The output layer of each DNN uses a softmax function to output values between 0 and 1. Other parameters and training strategies are as defined in [59]. The 4260 samples from the TIMIT training set are combined with Babble and Cafe noises at {10, 5, 0, -3, -5, -8, -10, -13, -15} SNRs, to create noisy speech mixtures that are used to train the DNNs (one for the IBM and one for the IRM). The speech and noise components of each mixture are used to generate the IBMs and IRMs (see Eqs (9) and (11)), which serve as training targets for the respective DNNs. The estimated masks are generated from the trained DNNs and the masks are subsequently applied to noisy speech testing signals to obtain estimated clean speech. As before, the Google Speech-to-Text ASR system is used for word recognition. In order to make a reasonable comparison, the dataset, noise and SNR levels remain the same as the previous case. Babble and Cafe noises are mixed with TIMIT corpus test set at {10, 5, 0, -3, -5, -8, -10, -13, -15} SNRs.

C.1 Recognition Results

Fig. 9 shows the recognition results for the noisy speech mixtures after speech separation is applied. For Babble noise, Fig. 9a, the estimated IBM only slightly improves ASR performance as compared to the unaltered noisy speech mixtures at SNRs between -8 and -15 dB. The average WER degrades after performing speech separation with the estimated IBM when the SNR is between -5 and 10 dB. On average, the WER for an estimated IBM increases by 3.83%. The estimated IRM approach offers more noticeable WER improvements at each SNR, where the average WER decreases by 1.52%. With Cafe noise, both speech separation approaches do not improve performance over the baseline noisy speech mixtures, except for a small improvement at -13 dB, see Fig. 9b. The WER *increases* by 8.81% for the estimated-IBM approach and 2.75% for the estimated-IRM approach when compared to noisy speech.

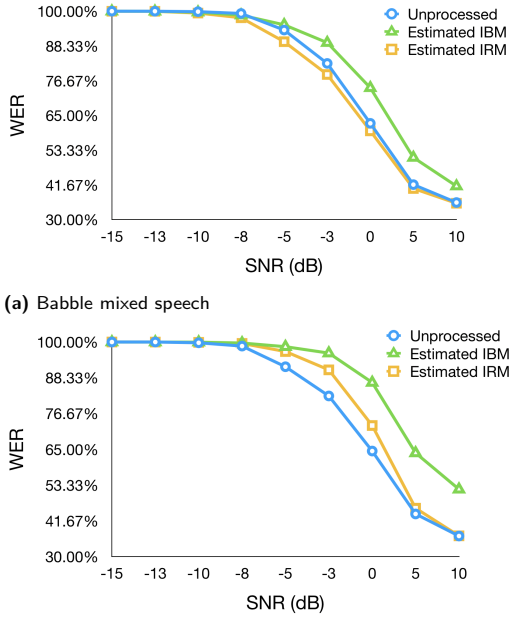
These WER results differ from other approaches that use time-frequency masks to remove unwanted noise [18, 45], but this is expected since the Google ASR system is not trained on IBM- or IRM enhanced noisy speech signals, unlike the prior approaches. This mismatch in data is exacerbated by the separation techniques, which contain estimation errors that result in



(a) Clean Speech Spectrogram **(b)** Noisy Speech Spectrogram **(c)** Estimated Clean Speech Spectrogram

Fig. 7. a) Random clean speech spectrogram from the TIMIT corpus. b) Noisy speech spectrogram by combining a) and Babble noise at -5 dB. c) An estimated clean speech spectrogram after applying an estimated IBM to b).

Fig. 8. Our MyBabble prototype consists of a wooden encasing to house a smartphone. Miniature speakers in the prototype are positioned at the microphones of the smartphone. Included circuitry plays MyBabble into the microphones through these speakers.



(a) Babble mixed speech

(b) Cafe mixed speech

Fig. 9. WER after speech separation is performed on noisy speech signals for a) Babble and b) Cafe.

the partial removal of speech and the partial retention of noise. This is shown in Fig. 7c, which shows the spectrogram of a noisy speech signal after IBM-based speech separation is performed. By comparing the noise-reduced spectrogram to the clean speech spectrogram, we can see that most noise is removed along with some of the speech when the estimated IBM is applied. For example, if we focus on the 5 second mark and between seconds 2 and 3 in the first and third spectrograms, we can easily find that part of the speech signal is eliminated and some of the noise is erroneously retained by the speech separation process. This loss of information negatively impacts word recognition.

C.2 Intelligibility Results

For the STOI result of Babble mixtures, an average improvement of 0.13 is observed when an estimated IBM is used for separation, whereas an average improvement of 0.14 is observed for IRM-based separation. For Cafe mixtures, both separation approaches have a 0.06 average increase in STOI results. The results imply that human-level intelligibility is only slightly increased after using these speech separation techniques. However, due to only minor intelligibility improvements and poor recognition performance, the attackers will not likely use speech separation on noisy speech signals before recognition.