Towards Device Independent Eavesdropping on Telephone Conversations with Built-in Accelerometer

WEIGAO SU, Hunan University, China
DAIBO LIU*, Hunan University, China
TAIYUAN ZHANG, Hunan University, China
HONGBO JIANG, Hunan University, China

Motion sensors in modern smartphones have been exploited for audio eavesdropping in loudspeaker mode due to their sensitivity to vibrations. In this paper, we further move one step forward to explore the feasibility of using built-in accelerometer to eavesdrop on the telephone conversation of caller/callee who takes the phone against cheek-ear and design our attack Vibphone. The inspiration behind Vibphone is that the speech-induced vibrations (SIV) can be transmitted through the physical contact of phone-cheek to accelerometer with the traces of voice content. To this end, Vibphone faces three main challenges: i) Accurately detecting SIV signals from miscellaneous disturbance; ii) Combating the impact of device diversity to work with a variety of attack scenarios; and iii) Enhancing feature-agnostic recognition model to generalize to newly issued devices and reduce training overhead. To address these challenges, we first conduct an in-depth investigation on SIV features to figure out the root cause of device diversity impacts and identify a set of critical features that are highly relevant to the voice content retained in SIV signals and independent of specific devices. On top of these pivotal observations, we propose a combo method that is the integration of extracted critical features and deep neural network to recognize speech information from the spectrogram representation of acceleration signals. We implement the attack using commodity smartphones and the results show it is highly effective. Our work brings to light a fundamental design vulnerability in the vast majority of currently deployed smartphones, which may put people's speech privacy at risk during phone calls. We also propose a practical and effective defense solution. We validate that it is feasible to prevent audio eavesdropping by using random variation of sampling rate.

CCS Concepts: • **Security and privacy** → **Mobile and wireless security**;

Additional Key Words and Phrases: Privacy Attack, Smartphone Conversation Eavesdropping, Vibration Recognition

ACM Reference Format:
Weigao Su, Daibo Liu, Taiyuan Zhang, and Hongbo Jiang. 2021. Towards Device Independent Eavesdropping on Telephone Conversations with Built-in Accelerometer . 1, 1 (November 2021), 29 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Telephone conversation is considered as one of the most concerning privacy and security issues because it involves with users' personal information, such as user identification [1], financial information [2], passwords

*Daibo Liu is the corresponding author.

Authors' addresses: Weigao Su, weigaosu@hnu.edu.cn, Hunan University, China; Daibo Liu, dbliu@hnu.edu.cn, Hunan University, China; Taiyuan Zhang, tyzhang@hnu.edu.cn, Hunan University, China; Hongbo Jiang, hongbojiang@hnu.edu.cn, Hunan University, China.

XXXX-XXXX/2021/11-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

[3], location and trajectory data [4]. Hence, the system permission level of microphone usage is set to the highest by default in most operating systems [5] to prevent privacy disclosure from malicious applications.

Over the last few years, researches have focused on audio eavesdropping by exploiting the onboard zero-permission motion sensors such as gyroscope and accelerometer. Specifically, Gyrophone [6] showed that gyroscope is sensitive enough to measure acoustic signals from an external loudspeaker to reveal speaker information. Accelword [7] used smartphone's accelerometer to extract signatures from the live human voice for hotwords extraction. Speechless [8] further tested the conditions and setups for speech leakage. Anand et al. conclude that external sound sources as long as the generated vibrations may indeed influence motion sensors can propagate along the surface to the embedded motion sensors placed on the same surface. Furthermore, Pitchin [9] presented an eavesdropping attack leveraging embedded motion sensors in an IoT infrastructure (having a higher sampling rate than a smartphone motion sensor) that is capable of speech reconstruction. And Spearphone [10] explored the possibility of revealing the speech played by the smartphone's built-in speakers from the phone's motion sensors recently. Based on above studies, AccelEve [11] further exploited deep learning to recognize and reconstruct speech information from the spectrogram representation of acceleration signals stimulated by loudspeakers.

Existing works have showed the feasibility of using motion sensors embedded in commercial off-the-shelf (COTS) smartphones to eavesdrop on speech privacy. However, all the above works failed to cover the most adverse setup, where the motion sensors are utilized as a side-channel to capture the speech signals spoken by the caller/callee who takes the phone. Under this context, we move one step forward to exploring the feasibility of using a built-in accelerometer to eavesdrop on the telephone conversation of caller/callee who takes the phone against cheek-ear area and show that the fundamental design vulnerability of the motion sensors usage (zero-permission) in currently deployed smartphones may put people's speech privacy at risk during phone calls.

The inspiration behind this work is that the speech-induced vibrations (referred to as SIV) can be transmitted through the physical contact of phone-cheek to accelerometer with the traces of voice content. However, to implement the novel attack model, several technical challenges need to be addressed. First, SIV signals usually carry much information that is highly relevant to the target device. The impact of device diversity is considerably decisive in our attack scenario, where that victim's smartphone model is probably unknown or newly issued. Thus, the ensemble of various device-relevant SIV traits makes eavesdropping on telephone conversation a formidable job and the biggest challenge to our attack scenario. Second, feature-agnostic methods prove to be a tragedy in pursuing device-independent recognition [12] even assisted by superior classification models. Naive solutions such as collecting an extensive dataset are time-consuming, inefficient, and incapable of generalizing to not yet released smartphones. Last but not least, to recognize the voice content represented by SIV signals, a headmost prerequisite is the reliable detection of SIV signals on the fly. This novel attack should also work economically to conceal its intention.

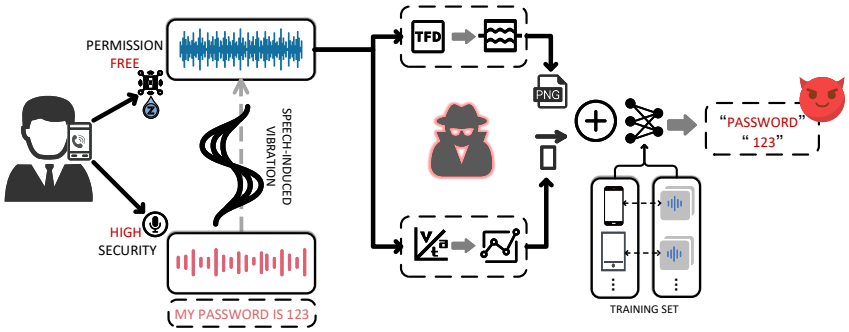


Fig. 1. Accelerometer-based telephoone conversation eavesdropping and the workflow of speech recognition

To address these challenges, in this paper, we propose Vibphone, a new side-channel attacking method exploiting a built-in zero-permission accelerometer to eavesdrop on telephone conversations as illustrated in Fig. 1. We have validated that smartphone accelerometers are sensitive to SIV signals, and the device diversity does have a decisive impact on the performance of Vibphone (see Section 3). Moreover, we conduct an in-depth investigation on SIV features to figure out the root cause of device diversity impacts. By adopting information gain for characteristic selection, we observe several critical characteristics that are highly relevant to the voice content retained in SIV signals and independent of specific devices (see Section 4). On top of these pivotal observations, we propose a combo method that is the integration of extracted critical features and deep neural network to recognize speech information from the spectrogram representation of SIV signals. Specifically, we employ an adversarial neural network in a multi-task learning style [13][12] to solve the limitation brought by device diversity and to perform high accuracy speech recognition. The adversarial neural network is fed on spectrograms, meaning the input data must be standardized. In addition, our model takes an extra input of the aforementioned features to boost its performance further.

To demonstrate the effectiveness of the above designs, we build a prototype of Vibphone with six off-the-shelf smartphone models, such as Samsung S8, Samsung S9, Samsung S10+, Xiaomi 9, OPPO R17, Huawei Nova3. We conduct extensive IRB-approved real-world experiments[1] and we act as the adversary to attack victims' call conversation on different types of smartphone models and usage scenarios. The results show that Vibphone's successful rate is 92.3% on known devices and 81.4% on unseen devices.

In summary, our contributions are listed as follows:

- We propose Vibphone, an accelerometer-based side-channel attack against telephone caller/callee. Different from previous works, Vibphone removes the impact of device diversity to achieve device-independent eavesdropping on telephone conversations. Comprehensive experiments are conducted to evaluate the feasibility of the Vibphone. To the best of our knowledge, the proposed system provides the first trial on accelerometer-based speech attack in the wild.
- We report the important observation that smartphone accelerometers are sensitive to speech-induced vibrations and the device diversity affects the performance of telephone conversations attack. The in-depth investigation on SIV features reveals the root cause of device diversity and finds out the critical features highly relevant to the voice content retained in SIV signals and independent of specific devices.
- We propose an adversarial neural network to solve the limitation brought by device diversity and to perform high accuracy speech recognition. The model takes an extra input of the critical features to further boost its performance.
- We propose a practical and effective defense solution. We validate that it is feasible to prevent audio eavesdropping by using random variation of sampling rate.

The rest of the paper is organized as follows. We introduce the principles of accelerometer and threat model in Section 2. In Section 3, we conduct empirical evaluations to demonstrate the feasibility of this work and introduce the challenges. Then, we further conduct an investigation on SIV features to find out the root cause of device diversity impacts in Section 4. On that basis, we present the design details on Vibphone in Section 5, and analyze and discuss the results of our attack in Section 6. By present the defense solution in Section 7, we disccuss the related works in Section 8 and finally conclude this work in Section 9.

2 BACKGROUND AND THREAT MODEL

In this section, we first describe the mechanism of motion sensors on current smartphones. Then we describe the threat model and overview of Vibphone (depicted in Fig. 1).

[1]Our study was IRB approved by our university. It does not raise any ethical issues.

2.1 Principles of Motion Sensors

Motion sensors are a small piece of technology that measure and record a physical, motion-relevant property. Modern smartphones typically come equipped with a three-axis accelerometer and a three-axis gyroscope. These sensors are highly sensitive to the motion of the device and have been widely applied to sense orientation, vibration, shock, etc. This measurement or reading is then utilized by an application for required purposes. Accelerometers and gyroscopes are the common motion sensors deployed on smartphones. An accelerometer is used to measure movement and orientation, and the gyroscope is used to measure angular rotation across x, y, and z axes. In practice, the information captured by a motion sensor is determined not only by its sensitivity to the surrounding environment but also by the sampling frequency.

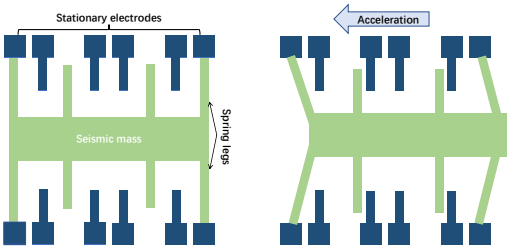


Fig. 2. Structure of MEMS accelerometer

Fig. 3. Stand-alone accelerometer for SIV collection

Accelerometer. Current built-in accelerometer sensors of modern smartphones and other smart devices such as smartwatches and smart glasses are Micro Electro Mechanical Systems (MEMS). The architecture of MEMS accelerometers consists of three main components - a seismic mass, spring legs and stationary electrodes, as illustrated in Fig. 2. The seismic mass is anchored to the substrate using two pairs of flexible spring legs. When an acceleration is applied, the seismic mass moves, which causes a change in the capacitance between the stationary fingers. This change is recorded to accurately measure the acceleration. In a three-axis accelerometer, 3 separate sets of components are employed to measure the accelerations separately.

Gyroscope. Gyroscope is a device used for measuring or maintaining orientation and angular velocity. It is a spinning wheel or disc in which the axis of rotation (spin axis) is free to assume any orientation by itself. When rotating, the orientation of this axis is unaffected by tilting or rotation of the mounting, according to the conservation of angular momentum.

We have compared the frequency response of both accelerometer and gyroscope. The accelerometer response shows us that it was able to register motion (acoustic vibrations) for the audio frequency range 100-3300Hz. Comparing with gyroscope's response, we see that the gyroscope's response is considerably weaker than accelerometer in the human speech frequency range. Owning to the limitation of the scope, we do not plot the repetitive measurement results which have been well studied [11]. Therefore, we use accelerometer in our experiments.

2.2 Threat Model

The attack studied in this paper is launched by a smartphone app that has access to motion sensor readings during phone conversations. Since platforms such as Android impose no permission restrictions on motion sensor data, any app installed on the phone would easily meet this requirement. The attacker's objective is to make inferences in the context of phone conversations. Examples of attackers who might be interested in such information include government surveillance agencies tracking suspected criminals, commercial entities seeking to illicitly pilfer trade secrets from their rivalry (e.g., advertising companies, insurance companies) or general-purpose hackers

seeking to steal passwords, social security numbers, etc. This line of attack would be appealing to these instances given the possibility, as mentioned earlier, to access sensor data stealthily.

We assume that the attacker is unaware of victim's phone model; has no direct access to the targeted smartphone; has no special privileges beyond sending messages over the internet, and cannot tempt users to take any action.

Our attack mainly focuses on Android, thanks to its prevalence as an open-source mobile operating system (not taking its variants into account). Please note that iOS users are not spared from the proposed attack because the maximum sampling rate of the accelerometer is only determined by hardware. The reason we prefer to use the accelerometer is that it is more sensitive to vibrations than gyroscope.

2.3 Attack Scenarios

The side-channel attack described in this paper allows an adversary to eavesdrop in a phone conversation on the sly to extract private information in real-time (e.g., password to bank account, social security number, home address or other untold secrets) through a pre-trained model with no prior awareness of target device.

This attack requires no compromises to the victim device and only utilizes the built-in accelerometer of a smartphone. Such an attack is workable because modern smartphones have a trending sampling rate higher than 400Hz and that the malware could be disguised as a harmless app running on the smartphone since accessing the accelerometer does not require any permission. Moreover, this model overcomes the deviation of accelerometer readings shared across different types of smartphones. For immaculate camouflage, this malicious app also develops a mechanism to detect telephone conversation, which helps it reduce battery consumption to avoid alert by OS.

3 FEASIBILITY ANALYSIS AND CHALLENGES

In this section, we first present experimental validations to show that smartphone accelerometers are sensitive to the SIV signals. Then, we explore the feasibility of using features of SIV signals in both time domain and frequency domain to eavesdrop on telephone conversations, which motivates our research in this work. On the other hand, we also conduct a series of measurements to reveal the key factors that challenge the effectiveness of Vibphone in the attack scenarios mentioned above.

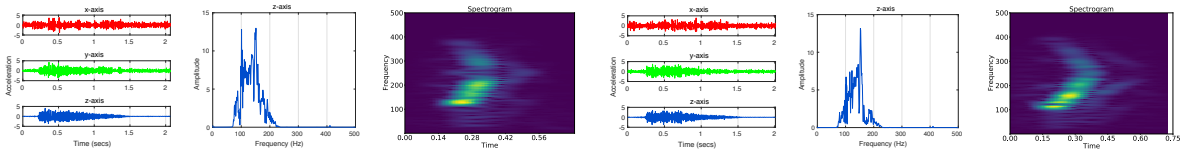
3.1 Sensitivity to Speech-Induced Vibrations

Here we conduct empirical evaluations to show the sensitivity of MEMS accelerometer to SIV signals and the robustness to external disturbance.

Sensitivity of accelerometer. During a telephone conversation, the caller or callee typically holds the phone against the positions near ear for better from the speaker on top. In this situation, the cellphone will inevitably and most of the time keep physical contact with the holder's cheek-ear area, thus naturally bringing incredible opportunity to transmit the vibration signals sourced from vocal-cord vibration to the phone surface. Because a MEMS accelerometer is highly sensitive to its motion change, which is dependent on the devices where it is mounted, the built-in accelerometer can sufficiently perceive the speech-induced vibrations in theory.

For validation, we recruited 10 volunteers, consisting of 4 females and 6 males. In each test experiment, the volunteers are instructed to first keep silent for a short while (less than 2 seconds) and then pronounce a hot word in normal tone and volume. We use a stand-alone three-axis accelerometer (referred to as SA-accelerometer) as illustrated in Fig. 3 and several modern smartphones (e.g., Samsung S9 and Samsung S10+) with built-in accelerometer (referred to as BI-accelerometer) to collect the produced response signals induced by vocal-cord vibration when they pronounce. Because SA-accelerometer has an identical structure to BI-accelerometer, which has been widely applied in modern smartphones and is largely free from the influence of other electronic gadgets,

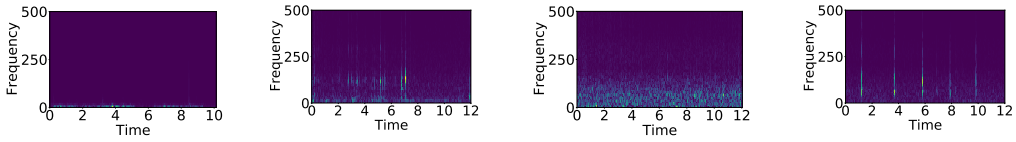
the produced response signals can represent the actual SIV processes to the maximum extent. Hence, we take the response signals recorded by SA-accelerometer to be the ground-truth data in this work.



(a) SA-accelerometer (b) Samsung S10+: BI-accelerometer

Fig. 4. The vibration profile and spectrograms of spoken word "Key" in time and frequency domain. The vibration data is respectively collected by a SA-accelerometer and Samsung S10+. For each figure, the left part denotes the response signals at x-, y-, and z-axis; the middle part denotes power spectral density, and the right part denotes spectrograms.

We plot the response signals recorded by both SA- and BI-accelerometers in Fig. 4. The left half and right half of the figure respectively show the response signals to SIV by SA- and BI-accelerometers at three-axis (i.e. "x," "y," and "z"-axis) and the corresponding spectrograms in time-frequency domain. As shown in the figure and lots of repeated measuring results, the "x"-axis and "y"-axis can indeed respond to SIV signals emitted by the recruited volunteers. However, the "z"-axis exhibits a much more intense response than the other two in the vast majority of cases. Hence, Vibphone uses the "z"-axis in accelerometer to represent the produced response for the rest of the experiments. Besides, the spectrograms present significant change and clear boundaries between the period of silence and pronunciation. The above results demonstrate that MEMS accelerometers are highly sensitive to SIV signals. Furthermore, the results of BI-accelerometer present quite interchangeable spectrogram with those of SA-accelerometer as illustrated in Fig. 4(a) and 4(b), which is consistent with the previous studies [11][14].

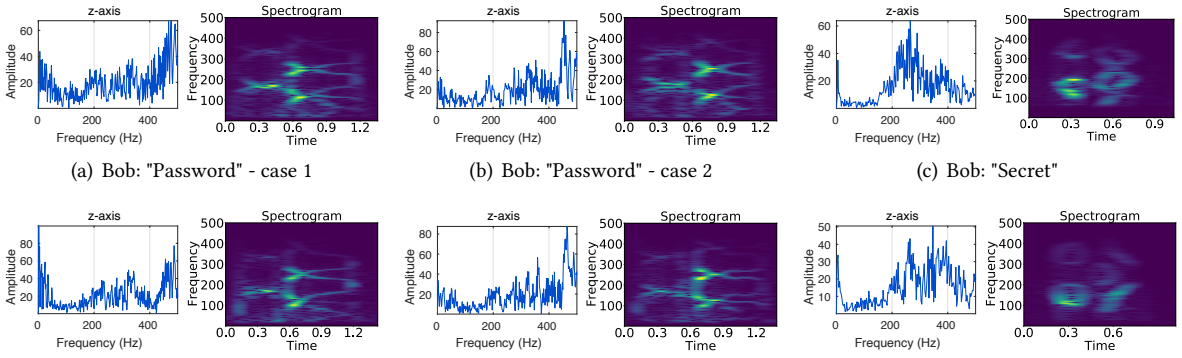


(a) Human activities (b) Acoustic noise (c) Self noise (d) Touch input

Fig. 5. The impact of external disturbances on robustness of built-in accelerometer for perceiving speech-induced vibrations.

Robustness to external disturbance. Accelerometers on smartphones are highly sensitive to external disturbances, such as body movements, touch input, environmental noise, and self-noise. We look into all these disturbances and find that they are either unlikely to affect word recognition or can be effectively erased. Body movement is a binding attribute to our attack scene. We recruit volunteers to make common telephone conversations in real-life dynamic settings, i.e., standing up, walking and going upstairs to make an assessment. The BI-accelerometer records the response signals of SIV during each experiment. Typically, we notice that each listed behavior only alters acceleration signals below 50Hz, as illustrated in Fig. 5(a). Thus, the impacts from user behavior can be eliminated through a high-pass filter. As to environmental noise, the speech signals that travel through aerial medium are not powerful enough to invoke any noticeable response on accelerometer, as illustrated in Fig. 5(b). MEMS accelerometer's insensitivity to surrounding acoustic noise has been well studied by Anand et al. [8]. Yet, self-noise (See Fig. 5(c)) is output by the smartphone's internal components at no external stimulus, making it consistent and inherent. It is not viable to completely dispose of such a random noise, but it is not necessarily causing much perplex to our word recognition system if properly handled. We put Samsung S9

on a table and record data of its SA-accelerometer to observe a constant noise pattern from all three axes. This pattern is full-ranged, but most of it lies below 80Hz (See Fig. 5(c)). Thus we can still leverage a high-pass filter to dump most of such interference. The impact of such disturbances on the recognition accuracy is evaluated in Section. 6.



(a) Bob: "Password" - case 1 (b) Bob: "Password" - case 2 (c) Bob: "Secret"

(d) Alice: "Password" - case 1 (e) Alice: "Password" - case 2 (f) Alice: "Secret"

Fig. 6. The power spectral density and spectrograms of speech-induced vibration signals for the pronunciations of words "Password" and "Secret" by volunteers "Bob" and "Alice". The upper half figures represent the power spectral density and spectrograms of the produced signals while Bob speaks the hot-word "Password" twice (Fig. 6(a) and 6(b)) and "Secret" once (Fig. 6(c)). Alice has also completed the same experiments as illustrated in the lower half figures.

3.2 Recognizability of Speech-Induced Vibrations

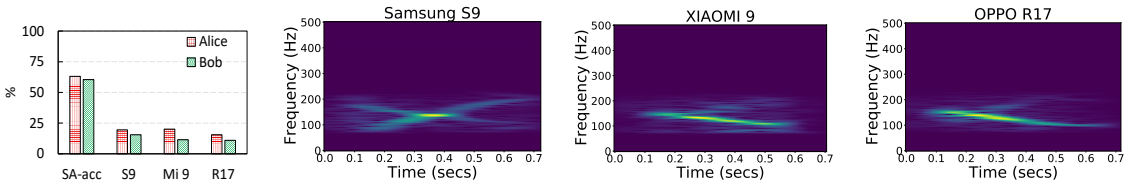
In this experiment, we study the possibility of using SIV signals to achieve eavesdropping on telephone conversations. The key observation behind this work is that SIV signals can be transmitted through the contact area of phone-cheek to render signal change on accelerometer. It is worth noting that human voice ranges typically from 80Hz to 400Hz and mostly lies in 85Hz to 255Hz. Hence, during telephone conversations, adequate information in the form of SIV corresponding to the victim's voice is well retained because of a standard sampling rate of built-in accelerometer on modern smartphones reaching up to 400Hz or even 500Hz [11] and the Nyquist theorem.

Since accelerometers can only pick up speech signals in low-frequency band, the Mel-Frequency Cepstrum Coefficients (MFCCs) being widely applied in audio-based speech recognition are not suitable for Vibphone to classify the SIV signals. Thus we directly use spectrograms as samples for signal identification and classification. The spectrogram representation illustrates the multi-dimentional information of an SIV signal in both time and frequency domain, possessing enough inherent features on voice content[15][16].

For recognition, the spectrograms of different pronunciations (i.e., hot-words like *Password, Code, Key, Secret, Account, Salary, Encoder, Bank, Number, Word*) should be distinct enough while those of the same words should be consistent. Besides, the distinct characteristics of each hot word should be the same with different sound sources, namely different victims.

We recruit 10 volunteers to pronounce each hot word at least 50 times in normal tone and volume. Also, we use the SA-accelerometer to acquire produced response signals. Without loss of generality, we randomly select two volunteers referred to as "Bob" and "Alice" and plot the SIV signals for each pronunciation in Fig. 6. The upper half figures represent the power spectral density and spectrograms of the produced signals while Bob speaks the hot-words "Password" twice (i.e. Fig. 6(a) and 6(b)) and "Secret" once (i.e. Fig. 6(c)). Alice has also completed

the same experiments as illustrated in the lower half figures. According to Fig. 6(a) and 6(b), as well as Fig. 6(d) and 6(e), we can observe that, for the same user (Bob or Alice), the profiles of two pronunciations on the same hot-word (Password) match well in both time and frequency domains, which implies the consistency of inherent characteristics. Moreover, the pronunciations on the same hotword (Password or Secret) by different volunteers (Bob and Alice) also share similar profiles of power spectral density and spectrograms. By contrast, the profiles of pronunciations on different hot-words (Password and Secret) differ significantly as illustrated in Fig. 6(b) and Fig. 6(c), as well as Fig. 6(e) and Fig. 6(f). Note that the comparisons between other volunteers reach the same conclusions (see Section 6 for further details). In conclusion, the intuitive empirical study opens up the possibility of using SIV signals to identify victim's telephone conversations via the built-in zero-permission accelerometer.



(a) Recognizability study (b) Samsung S9 (c) Xiaomi 9 (d) OPPO R17

Fig. 7. The impact of device diversity. (a) plots the recognition accuracy when test set is collected by different devices; When a volunteer uses three different smartphones (i.e., Samsung S9, Xiaomi 9, and OPPO R17) to pronounce the same voice content, (b)-(d) denote the three spectrograms of SIV signals.

Recognition analysis. To explicitly validate the feasibility of exploiting spectrogram analysis for SIV-based voice content recognition, we collect the SIV signals of both Bob and Alice and use a three-layer structured CNN to classify the gathered data and recognize hot-words. The spectrograms are normalized to 224×224 grayscale images to feed in the network. Note that the use of grayscale images instead of colored images can improve the processing efficiency without compromising the recognition performance. As shown in Fig. 7, when training and test set are collected by the same device, we can obtain average accuracy of 63.1% for Alice test and 60.5% for Bob test when SA-Accelerometer acts as both training and test device, respectively. The tiny difference between the two test sets results from different pronunciation habits, which is detailed in Section. 6.4.5.

To summarize, speech-induced vibration is identifiable underpinned by feature-rich spectrograms, motivating us to implement Vibphone under the aegis of deep neural network.

3.3 Challenges

The above empirical results demonstrate the sensitivity to SIV signals of the built-in accelerometers, motivating us to explore the novel attack on telephone conversation only by the built-in zero-permission accelerometer, and that we can effectively acquire speech information with the help of deep neural networks. However, several immense challenges stall us from applying the above observations to our attack scenarios.

(1) Telephone conversation detection and signal processing. Recognizing the voice content represented by SIV signals, the headmost prerequisite is reliable detection of SIV signals on the fly. Moreover, Vibphone should work economically to conceal itself.

This requirement poses a new challenge on separating a telephone conversation from other events. Signals caused by touch input, e.g., tapping or swiping, have analogous distribution with SIV. Immutable idiosyncrasies should determine the difference between them. Hence, to achieve efficient and reliable SIV detection with a built-in accelerometer, we should first identify key features that can be readily processed and remove all other disturbances effectively.

(2) Impact of device diversity. In practice, SIV signals carry much information that is specific to target device. This implies that the training set based on specific devices cannot be effectively extended to SIVs collected by a different device in real-time. However, in our attack scenario, the assumption is tenuous that we are able to bring in the victim's smartphone model through additional effort in advance. Hence the impact of device diversity is momentous. For instance, when Alice uses three different smartphones (i.e., Samsung S9, Xiaomi 9, and OPPO R17) to pronounce the same content 'code', we plot the three spectrograms of SIV signals in Fig. 7(b), 7(c), and 7(d), respectively. As shown, a consistent bright contour exists among three devices, which is the key to recognize the word 'code.' Nonetheless, there are other parts of the spectrograms that are inconsistent among devices, which hinders the classifier from making right decisions.

To quantitatively evaluate the impact of device diversity on recognition of SIV-based voice content, rest on the same setup of CNN-based recognition, we further use other phone models (Samsung S9, Xiaomi9 and OPPO R17) to collect SIV data. In the same way, Alice repeats the progress to record the selected 10 hot-words 50 times on each device. Then, we run a test on the established CNN model. As shown in Fig 7, the overall accuracy is sharply reduced from 63.1% to 18.4% among different devices, which is just slightly better than random guess.

The inadequate recognition accuracy in using different devices can be mainly attributed to two factors. One is that different smartphone models put accelerometer chips at different places on motherboard. The other is that due to hardware imperfections, accelerometer on different devices presents diverse traces (i.e. interference) [17] which are independent of SIV signals it senses. For example, compared with Samsung S9, there is an entirely disparate layout of electronic gadgets to induce foreign interference inside Samsung S10+, which is proven by [14]. Thus, the ensemble of various device-relevant SIV traits makes eavesdropping on telephone conversation a formidable job and also the biggest challenge to our attack scenario.

(3) Model formulation. Feature-agnostic methods prove to be a tragedy in pursuing device-independent recognition even assisted by superior classification models. Naive solutions such as collecting an extensive dataset are time-consuming, inefficient, and incapable of generalizing to not yet released smartphones. To mitigate such hindrance, the only way-out lies within feature-related study.

All technical cruxes brought by device diversity are highly correlated with the innate features of voice content in SIV signals. Hence, we need to cast out device-relevant features so that we can exploit SIV-relevant features more effectively. However, a feature-only approach turns out to be mediocre despite of its better performance in cross-device recognition. Bearing that in mind, in this paper, we propose a combo method that is the integration of extracted features and deep neural network.

Before introducing our proposals on addressing these challenges, we first conduct an in-depth investigation on SIV features to find out the root cause of device diversity impacts.

4 IN-DEPTH INVESTIGATION ON CHARACTERISTIC IMPORTANCE

Given the challenges mentioned-above, we conduct an in-depth investigation to find out the root cause of the similitude in spectrogram from same source and the divergence in different ones by a multi-characteristic based approach. Besides, we employ feature selection method to help boost prediction accuracy. For feature engineering, there has been a debate of relevance *vs* usefulness [19] [20]. In this work, we focus on building a *useful* feature subset.

4.1 Statistical exploration on available characteristics

Statistical characteristics can represent multiple characteristics of signals. Consequently, plenty of studies on signal processing employed statistical characteristics for a node or channel identification [7][22][23]. Seeing that the vibration of cranial cavity upon pronouncing different words affects the signal characteristics, we exploit some statistical characteristics to differentiate a particular word. In total, we have conceived and explored 30

scalar characteristics in both time and frequency domains with the help of LibXtract [26], a commonly used lightweight library for characteristic analysis. As a result, we found 16 of them to be highly-correlated with speech-induced vibration:

- **TotalSVM** [7]: the signal magnitude of all accelerometer signal of three axis averaged over a given time window.

$$TotalSVM = \frac{\left[\sum_{i=1}^{window} \sqrt{\sum_{s\in\{X,Y,Z\}} a_{s_i}^2} \right]}{window}$$

  In above discussion, we intend to omit contributions of x and y axis because they are rather insignificant comparing to those of z axis. However, they could still offer some useful hint.
- **Q1, Q2, Q3**: respectively denoting the first, second and third quartiles of the signal, which measure the overall distribution of accelerometer magnitude over a time period of speaking.
- **Spectral Crest** (referred to as *Spec.Crest*) shows the ratio of peak values to the effective value in spectrograms. It indicates how extreme the amplitude peaks are. Spectral Crest=1 implies no peak, such as direct current or a square wave, while higher value indicates existence of peak.

$$Spec.Crest = \left(\text{Max}\left(y_m(i)\right)|_{i\in[1,N]} \right) / C_s,$$

  where C_s refers to Spec.Centroid which is calculated by:

$$C_s = \left(\sum_{i=1}^{N} y_f(i) y_m(i) \right) / \left(\sum_{i=1}^{N} y_m(i) \right).$$

- **Other Statistical Characteristics** refer to **Mean Amplitude**, **Standard Deviation**, **Average Deviation**, **Root Mean Square Amplitude** (RMS), and **Spectral Kurtosis** (referred to as *Spec.Kurt*). RMS can be expressed as

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x(i))^2},$$

  and **Spectral Kurtosis** can be denoted as

$$Spec.Kurt = \left(\sum_{i=1}^{N} \left(y_m(i) - C_s\right)^4 * y_m(i) \right) / \sigma_s^4 - 3.$$

Besides the above-explained statistical characteristics, the rest characteristics and description are listed in Table 1.

Table 1. Descriptions on characteristics

Characteristic Name	Description
AbsMean	$\frac{1}{N} \sum_{i=1}^{N} x(i)$
Kurtosis	$\frac{1}{N} \sum_{i=1}^{N} \left(\frac{(x(i)-\bar{x})}{\sigma} \right)^4 - 3$
Skewness	$\frac{1}{N} \sum_{i=1}^{N} \left(\frac{(x(i)-\bar{x})}{\sigma} \right)^3$
Spec.Skewness	$\left(\sum_{i=1}^{N} \left(y_m(i) - C_s\right)^3 * y_m(i) \right) / \sigma_s^3$
Spec.StdDev	$\sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (x(i) - \bar{x})^2}$

4.2 Information gain-based selection

During experiments, we find that directly adopting all the provided 16 characteristics may hamper the classifier in making right decisions. Therefore, we conduct a selection as follows: First we sanitize the data by linear interpolation, high-pass filter and segmentation, which will all be provided with details in Section. 5, we calculate each characteristic value to form a vector V_i for each sample i. With a calculated V_i in hand, to measure the ability of a given characteristic to differentiate a hot-word from other ones, we use Information Gain based characteristic selection. Information gain [18] is a general approach used as a feature evaluation method that calculates the reduction or increment in entropy from transforming a sample in some way. It works by evaluating the information gain for each variable (i.e., elements in V_i), and designating the one that maximizes the information gain, which on the other hand, minimizes the entropy to build a effective classification model. Assume G represents all instances in our dataset of n samples, G_i represents the i^{th} sample where $i < n$ and $p(G_i)$ is the proportion of G_i in G. Then the entropy of our dataset is:

$$H(G) = -\sum_{i=0}^{n} p(G_i) \cdot log_2 p(G_i). \tag{1}$$

Let $IG(\tau)$ be the information gain of the characteristic τ and G_τ be the subset of G which satisfies τ. $IG(\tau)$ can be calculated as:

$$IG(\tau) = H(G) - H(G|G_\tau). \tag{2}$$

Thus, $IG(\tau)$ can be regarded as a hint to gauge the extra information that τ is capable to give to the classifier. The information gain ranges from 0 to 1 where a higher value indicates a characteristic can provide more helpful information.

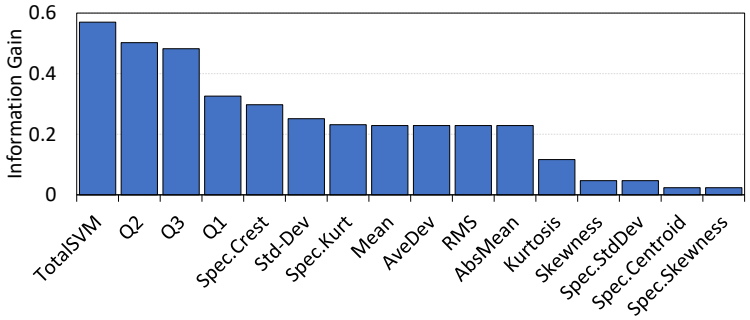


Fig. 8. Information gain ranking. Higher value indicates more contribution to the classification

As shown in Fig. 8, the histogram ranks the information gain of each provided characteristic corresponding to hot-words. Note that the rest of the 30 initial characteristics have information gain close to zero and thus are excluded. Next, we stepwise cut out characteristics from the queue and run test in order to find the best combination in a doubly nested loop way. Instead of ranking them simply by value of information gain, we traverse all the combinations to pick out the best-performing characteristic set : TotalSVM, Q1, Q2, Q3, Spectral Crest, Standard Deviation, Spectral Kurtosis, Mean Value, Average Deviation and Root Mean Square Amplitude. It is worthy noting that our method sets the goal to increase the accuracy relying on the theory that redundant variables can help each other for a performance gain [21].

4.3 Attempt On Erasing Impact of Device

Interference brought by the device is the main limitation to stall us from extracting useful information from such subtle vibrations. To fully understand the interference, we conducted another characteristic selection[22][23]

Algorithm 1 Characteristic Selection

Input: Labeled characteristic vector $\{(xi, si)\}_{i=1}^{n*m}$

initial Characteristic set: t
For τ in Features:
    For i in Samples:
        Compute characteristic value for sample i: $ti \leftarrow calculate(xi, \tau)$;
    F1-Score for SVM: $F1_S \leftarrow trainSVM(t, s)$;
    F1-Score for Decision Tree: $F1_T \leftarrow trainTree(t, s)$;

using support vector machine (SVM) and Decision Tree. Three types of devices are adopted, including Samsung S9, Samsung S10+, Xiaomi 9, and OPPO R17.

The selection constitutes several classifications, of which each begins with a training phase and follows by a test. During training, for n samples, only one characteristic is computed, and the n sets of the characteristic are used to train the classifier. For m words, $n \times m$ sets of characteristic can be used to train the classifier altogether. The detailed selection process is shown in Algorithm.1, where characteristic set refers to the previously selected 16 characteristics.

Table 2 shows our test results. We can see that some of the characteristics are so hugely affected by device diversity that they can even be used to classify samples by device type at high accuracy. Therefore, we traverse all combinations of the preliminarily selected characteristics and to reach the optimal set of characteristics. As shown in Table 3, we obtain the best-performing characteristic set by discarding Mean Amplitude, Standard Deviation, Average Deviation, Root Mean Square Amplitude and Spectral Kurtosis in order to erase the impact of device. The result is encouraging as it raises the recognition rate from 17.3% of CNN to 34.2% on average, implying we could make use of the characteristic set to assist our neural network.

Table 2. F1-Score Ranking. With a higher F1-score, a variable performs better classification tasks

Mean F1-Score(%)	Feature Name									
	RMS	Spec.Kurt	Ave.Dev	Std.Dev	Spec.Crest	Mean	Q3	Q1	TotalSVM	Q2
SVM	94.7	97.9	78.0	74.1	62.3	57.5	48.9	37.1	35.1	33.2
Decision Tree	89.4	80.4	85.2	79.4	69.3	60.9	39.2	41.3	40.1	37.6

Table 3. Cross-device performance on the computed characteristics

Accuracy (%)	**Test device**			
Training device	Samsung S10+	Samsung S9	Xiaomi 9	OPPO R17
Samsung S10+	61.4	34.5	31.7	33.6
Samsung S9	37.5	62.3	34.2	34.8
Xiaomi 9	32.3	33.9	62.5	36.3
OPPO R17	33.3	33.3	35.5	60.3

5 ATTACK DESIGN

Vibphone utilizes speech-induced vibration to eavesdrop on targeting victim's phone conversations silently. Since attackers are unaware of the victim's smartphone model in our attack scenarios, the key to a successful attack is to effectively remove the impact of device diversity on the recognition of SIV-based voice content. In

particular, after pre-processing, Vibphone has to first exploit the produced response of SIV to reliably detect phone conversations and then uses the aforementioned characteristics that are highly correlated with the conversation content but independent of the involved device to boost the recognition model to some extent. In the following subsections, we discuss the details of Vibphone, which mainly includes three modules, i.e., preprocessing module, SIV detection module, and the recognition model.

5.1 Preprocessing

Because MEMS accelerometer is highly sensitive to its own motion change, it can produce different frequency-response in time domain depending on the source of vibration which results in the motion change. Hence, once obtaining motion sensor output from the malicious application, Vibphone should first normalize the acceleration data (with random intervals) possibly recorded by an unknown device for facilitating the next signal processing and combat the noise caused by user's mobility as introduced in Section 3.

Interpolation. Since the data are not sampled at a fixed interval, the derived values making the frequency domain characteristics difficult to compute. Hence, Vibphone employs a linear interpolation to construct new data points at sampling rate of 1000Hz so that samples are equally-spaced. This interpolation process does not increase the speech information retained in the SIV signal. Its primary purpose is to generate acceleration signals with a fixed sampling rate.

High-pass filtering. Noting that human mobility could only induce interference signal below 50Hz, a high-pass filter can be used to filter out the interference from the SIV signal. We can do that without sacrificing helpful information because the fundamental frequency of adult males and females is usually around 85Hz to 255Hz. We apply an 80Hz high-pass filter to reduce the influence of user mobility. We first convert the SIV signals along z-axis to frequency domain with Short-Time Fourier Transform (STFT), which divides the whole signal sequence into units of equidistance (with overlaps) and calculates the Fourier transform on each segment separately. Then it sets the amplitude value of all frequency components below the cut-off frequency to zero and converts the signal back to the time domain using inverse STFT. After that, the output signals mainly consist of targeted high-frequency SIV information and self-noise from the accelerometer.

(a) Pre-processed acceleration signal of real speech.

(b) Segmentation result on the raw signal

Fig. 9. Signals are processed with interpolation, high-pass filtering, and segmentation in order to remove the influences of user mobility and to acquire SIV clips. The magnitude sequences are then directly calculated from the filtered signals since they do not involve intense human movements.

Segmentation. To acquire vibration information for subsequent classification, we need to pick up potential SIV segments resulting from speaking slots in the phone conversations from those silent ones because vibration signals are continuous in the time domain. To locate these segments therein and use the obtained boundary to slice the filtered acceleration signal along z-axis, firstly, we use another round of high-pass filter at 120Hz. Note that such an operation will not influence speech recognition because it is only used for segmentation. Secondly, before we smoothen the obtained sequence with two rounds of exponential moving average, we carry out the

absolute value of the filtered signal. The sliding windows are 200 and 60 respectively. Lastly, we plot a line to label out the audible segments by calculating a cut-off threshold $0.9A_{max} + 0.15A_{min}$ where A_{max} and A_{min} denotes maximum and minimum of the amplitude sequence as shown in Fig. 9, whose data is sampled from real speech. Points between intersections with the line are classified as audible segments. Note that moving average will narrow down the scope of sampled sequence. We then modify points' position 200 units to both sides along horizontal axis.

5.2 SIV Detection

Previous processes standardize data by filtering out the interference caused by body movements and self-noise but still keeps what caused by the use of touch input, i.e. tapping or swiping the screen, with a dominating frequency component between 80Hz and 200Hz[40].

As illustrated in Fig. 10, the vibration users' tapping and swiping operations has significant frequency overlapping with SIV. Hence, before moving to the next stage for SIV-based voice content recognition, Vibphone needs reliable detection on SIV signals on the fly. On that basis, Vibphone can work in an economic way for better disguise.

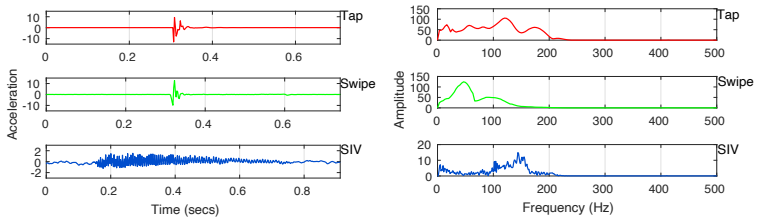


Fig. 10. Comparison between touch input and SIV. Though sharing similar frequency range, they are fairly different in distribution patterns.

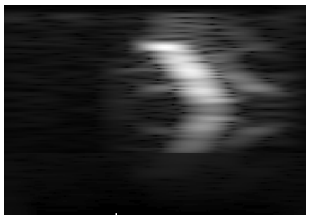


Fig. 11. Gray-scale image "Key"

Time series analysis. As shown in Fig. 10, in spite of similar frequency ranges, SIV signals are evidently unique in distribution. Therefore, we can clarify their difference by scrutinizing both time and frequency traits. Important findings are that tapping and swiping signals are "sharper" than SIV signals and that in frequency domain, they appear to be "smoother". Given that, in exchange for energy consumption, we only select 3 traits in each time window:

- Spectral Smoothness[17][24]: describes how smooth a spectrum is. It is determined by the degree of amplitude difference between adjacent partials in the spectrum over a time window. It can be calculated as follow:

$$Spec.Smoothness = \sum_{i=2}^{N-1} \left| 20 \cdot \log\left(y_m(i)\right) - \frac{\left(20 \cdot \log\left(y_m(i-1)\right) + 20 \cdot \log\left(y_m(i)\right) + 20 \cdot \log\left(y_m(i+1)\right)\right)}{3} \right|$$

- Kurtosis: a robust metric for impulsive content; measures peakedness in distribution of signal segments. Its description is in Table 1.
- Range: measures the difference between the highest and lowest amplitude in a segment. Both tapping and swiping lead to larger ripples in waveform which is not commensurate with that of an SIV signal.

Lightweight SIV detection. By taking advantage of the above characteristics in time and frequency domain, Vibphone can identify the SIV segmentations accurately and timely. We denote that SIV typically marks a phone conversation. However, it is against our intention to constantly extracting and calculating the characteristics to

identify an SIV signal, which is energy-consuming. To address this issue, Vibphone will only respond to signals higher than 100Hz.

Upon detecting high-frequency signals, Vibphone moves onto the confirming stage, i.e., to distinguish SIV from touch input, which uses calculated characteristics from all types of touch-induced vibration signals from different subjects to train a one-class SVM classifier. Given the clip of unknown signals, the detection module would output a normalized score, and the clip is only considered as an SIV signal if it is lower than a pre-configured threshold.

5.3 Speech Recognition

As demonstrated in Section. 3 and Section. 4, both a feature-agnostic deep learning model or a feature-only machine learning model have an unpromising generalization ability. In this section, we employ a feature-related adversarial neural network in a multi-task learning style to solve the limitation brought by device diversity and to perform high-accuracy speech recognition. The adversarial neural network is fed on spectrograms, meaning the input data must be standardized. In addition, our model takes an extra input of the aforementioned characteristics to further boost its performance. Next, we present the details on the design of our speech recognition model.

5.3.1 SIV signal transformation. When detecting segmented SIV signals, Vibphone should convert the SIV signals to a standardized spectrogram.

Signal-to-Spectrogram Conversion. To generate a spectrogram of an SIV segment, we first divide the signal into multiple small grid with a fixed overlap. Then, let a window pass through the sequence to calculate its spectrogram through STFT, which generates a series of complex coefficients for each grid. We set the overlap as 250 units and a hamming window size as 256 units. The signal along z-axis is now converted into a STFT matrix that reflects the magnitude and phase for each time and frequency as follow:

$$spectrogram\{x(n)\}(m,w) = |STFT\{x(n)\}(m,w)|^2,$$

where $x(n)$ and$|STFT\{x(n)\}(m,w)|$ respectively represent z-axis acceleration signal and the magnitude of its corresponding STFT matrix. Then, a 2D spectrogram can be generated for the subsequent processing.

Spectrogram Grayscale Images. For the neural network to effectively make use of our signal, the spectrogram is better converting to grayscale image because this single-channeled format keeps helpful information- the scale of magnitude- and discards the others, lessening training overhead. Let Bz be the value matrix of spectrogram and $Bz[i,j]$ be the point at i^{th} row and j^{th} column on Bz. We note that values in the matrix are mostly lying around $[10^{-7}, 10^{-5}]$, directly mapping them to $[0, 255]$ may result in data overflow. Therefore, we take the square root of all points in spectrogram.

$$Bz[i,j] = \sqrt{Bz[i,j]}$$

Next, we linearly map all Bz to integers between 0 and 255 (i.e., normalization). The reason of taking square root is that most elements in the original matrix are close to zero. Directly mapping these elements will result in considerable information loss. At last, we store Bz in portable network graphics (PNG) format. In the obtained spectrogram-image, the brighter region indicates that the acceleration signal has stronger energy at that frequency range during that time period. Fig. 11 is an example of word "Key". Upon obtaining gray scale images, it is necessary to resize them to fit in neural network. Given the original size, we transform images to $200 \times 200 \times 1$.

5.3.2 Characteristics Processing and Integration. Previous studies [11][10][14] are trapped with the highly device-dependent features retained in the spectrograms. As validated by experimental studies in Section 4, we find that some critical characteristics are decisive to ease up the stress of the network. Here, we detail the further processing of these critical characteristics.

Alignment. Signal segmentation can only achieve a rough estimate of the starting and end point of the SIV signal. A defective estimate segment boundary is catastrophic for characteristic extraction which needs accurate

time domain information. Vibphone thereupon leverages Cross-Correlation to measure time delay of different signals [46]. We design an iterative adaptive implementation of Cross-Correlation analysis. For each iteration, we estimate all relative shifts by maximizing the Cross-Correlation of each signal and a template signal. Initially, the first signal is the template signal. From the relative shifts, we compute the average of the shifted signals. The updated template signal is set equal to this signal average.

Computation of Critical Characteristics. Instead of directly extracting characteristics from a raw signal, we should first pre-process the signal to obtain a set of intermediate data. One is the value of acceleration through all three axes, and the other one is the frequency domain representation of acceleration signal along z-axis because it is the dominant axis. Let $\{s_x(k), s_y(k), s_z(k)\}$ be the k-th acceleration along x-, y- and z-axes respectively, $\{Z_f, Z_m\}$ be the magnitude and bin frequencies of z-axis respectively, and $T(k)$ be the timestamps.

As mentioned in Section 4, we identify 5 critical characteristics that are highly relevant to the voice content retained in the SIV signals and is independent of the devices. Hence, we can make use of them in our recognition module. First we compute the magnitude of accelerometer signal on all three axis averaged over time (i.e., TotalSVM) using $s_x(k), s_y(k)$ and $s_z(k)$. Next, we calculate the three quarters amplitude with $s_z(k)$. At last, we use Z_f and Z_m to compute spectral crest on z-axis spectrogram. We thereof acquire a vector of selected characteristics: {TotalSVM, Q1, Q2, Q3, Spec.Crest}

Characteristic-Image Integration To resolve the limitation resulting from device diversity, we have to dilute the divergence that exists between two devices' response to the same vibration in furtherance of easing the burden for our neural network by providing useful information. Therefore, we integrate each set of calculated critical characteristics with a gray scale image to form a new tensor: Let $p \in \Omega_p$ be an input $200 \times 200 \times 1$ image, $\tau \in \Phi_\tau$ be a $5 \times 1 \times 1$ characteristic vector where Φ_τ indicates the set of all possible statistic values τ can take. We then concatenate p and τ along the first dimension after zero padding τ to build tensor x. In the end, we resize x to $224 \times 224 \times 1$ as the new input.

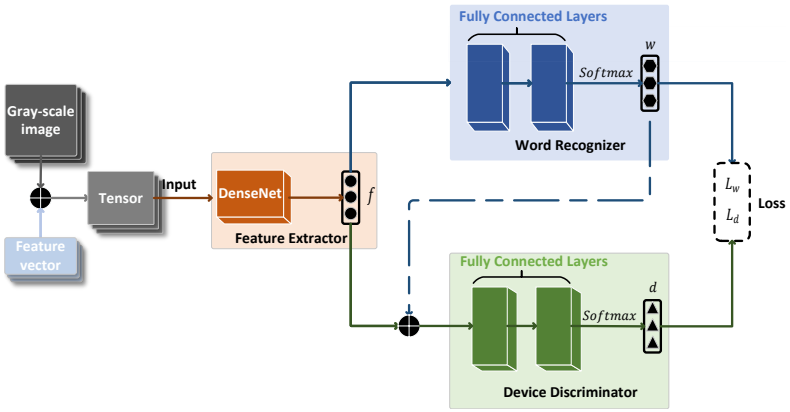


Fig. 12. Network outline.

5.3.3 Network Outline. Our model incorporates an adversarial network to predict the label of single words with the ability to remove the uniqueness of each domain (defined as a distinct device, i.e., BI-accelerometer), and extract commonness shared across different domains. Consequently, it can be used to recognize spoken words recorded through unseen devices. An overview of the proposed deep learning model is shown in Fig. 12.

Let $x \in \Omega_x$ be the concatenated input tensor, $y \in \Omega_y$ as an output label and $s \in \Omega_s$ denote an auxiliary label that refers to the device of a specific input tensor x.

In our application, the above notation translates into the following: the input sample x is pre-processed and transformed while the output label y is a predictive word and s is the class of different devices. Note that we assign each input x an auxiliary device label s .

The network has three components: A feature extractor $\mathbb{F}$, a word recognizer $\mathbb{W}$, and a device discriminator $\mathbb{D}$. Our model is set up as a game, where the feature extractor $\mathbb{F}$ plays a cooperative game with the word recognizer $\mathbb{W}$ to allow it to recognize the correct words by the extracted features. The feature extractor $\mathbb{F}$ also plays a min-max game against the device discriminator $\mathbb{D}$ to prevent it from discriminating the device label from the features representation. When it works, $\mathbb{F}$ extracts features from image x , noted as f. Then $\mathbb{W}$ calculates its results w based on f. The input of $\mathbb{D}$ is the concatenation of f and w because some dimensions in f, though being device-specific, are helpful to word recognition task. Thus, we still need to keep those. The goal of domain discriminator is to maximize the performance of device label prediction, which in turn help the word recognizer to grow robust through the following: the feature extractor tries its best to cheat the domain discriminator (i.e., minimize its predictive accuracy), and at the same time, boost the performance of the activity recognizer. Through the min-max game, the feature extractor $\mathbb{F}$ can finally learn the common features for all the hot-words in spite of device diversity.

5.3.4 Network Construction. The structure of an adversarial network consists of a feature extractor, a word recognizer, and a device discriminator. The details of them are as follow:

Feature Extractor: We employ DenseNet [25] as the backbone to extract features for our recognition tasks. Unlike conventional CNNs, it connects each layer to every other layer in a feed-forward fashion. For L layers in this network structure, there are $L(L+1)/2$ direct connections. For each of L layers, the feature-maps of all preceding ones are used as inputs, and its own feature-maps are used as inputs into all subsequent layers. In the proposed approach, we build the feature extractor $\mathbb{F}$ with a convolution layer followed by four dense blocks. Each of the dense blocks is headed by a bottleneck and then a convolution layer. Between each dense block, we set a transition layer consisting of one convolution and one pooling layer. Set θ_F as parameters in $\mathbb{F}$. Given input data X, we get the output

$$f = X \times \theta_F.$$

Word Recognizer: According to the outputs of feature extractor $\mathbb{F}$ (i.e., f), we design two neighboring fully connected layers to be our word recognizer $\mathbb{W}$. We choose $Softmax$ as our activation function. So given input f and parameters θ_W, the output of word recognizer is:

$$w = f \times \theta_W.$$

3. *Device Discriminator:* Similar to word recognizer $\mathbb{W}$, we set two fully connected layers as device discriminator $\mathbb{D}$. However, $\mathbb{D}$ has $f \oplus w$ as input. So given parameters of $\mathbb{D}$ as θ_D, the output d is:

$$d = (f \oplus w) \times \theta_D.$$

5.3.5 Training Process. We implement the extended three-player game with iterative updates of the players. Firstly, we forward the network to compute the cross entropy loss L_w for $\mathbb{W}$ and L_d for $\mathbb{D}$. Secondly, we compute overall loss :

$$L = L_w - \lambda L_d \quad (\lambda > 0).$$

Then back propagate the network to update θ_F and θ_W using L. As to $\mathbb{D}$, we do not wish the min-max game collapse because F and W, might be over-trained against a nonoptimal D, and thus they will try to maximize L_d at the cost of increasing L_f. So that an inner loop is adapted to update θ_D by $-L$ and iteratively compute L_d. Yet, it is not supposed to influence θ_F, so during this round of back propagation, gradient along $\mathbb{F}$ is stopped.

6 EVALUATION

In this section, we conduct a series of experiments to evaluate the effectiveness and robustness of Vibphone. We first introduce the implementation and experimental setup in Section 6.1. Then, by analyzing the overall performance of Vibphone in Section 6.2, we respectively discuss the effectiveness of SIV detection in Section 6.3 and the impact of different interference factors on the performance of Vibphone in Section 6.4. At last, we use Vibphone to achieve the eavesdropping on cellphone conversation in the wild in Section 6.5.

6.1 Implementation and Experimental Setup.

We have implemented Vibphone of its prototype version on Android in this work. Vibphone utilizes the built-in accelerometer in a smartphone to perform crafty eavesdropping on telephone conversations.

Since typical hot-words, digits and letters are usually quite short in length, and most victims can speak them in less than 2 seconds, Vibphone buffers 2 seconds of accelerometer data in a FIFO queue. Note that this can be adjusted based on the typical time taken to speak the hot word. In each run of the feature calculation, Vibphone first filters the data using a high-pass filter with cut-off frequency of 80 Hz. Then the calculated features are compared with the extracted hot-word signature. We set the time interval between each feature calculation to be a variable and test with different interval lengths.

We evaluate our attack using experiments with common smartphone models. Next, We describe our experimental setup and data collection.

Experiment setup. To evaluate the performance of Vibphone, we conduct classification tests with 10 volunteers (4 female and 6 male college students) and 7 distinct devices (SA-Accelerometer, Samsung S8, Samsung S9, Samsung S10+, Xiaomi 9, OPPO R17, Huawei Nova3). Table 4 presents the information of listed smartphones. We mainly evaluate our proposed system on signals collected from a Samsung S9. The details of data collection scenes are illustrated in Fig. 13 and introduced as follows. We instruct volunteers to pronounce a series of hot-words, numbers, and letters on one or multiple devices to collect SIV signals through Vibphone running in the background for each specific setting.

Table 4. Device information.

Device	Sampling Rate (Hz)	Year of Production	Size
SA-Accelerometer	500 Maximum	2020	/
Samsung S8	408	2017	148.9 * 68.1 * 8.0 mm
Samsung S9	418	2018	147.7 * 68.7 * 8.5 mm
Samsung S10+	425	2019	157.6 * 74.1 * 7.8 mm
Xiaomi 9	420	2019	157.5 * 74.7 * 7.6 mm
OPPO R17	415	2018	157.5 * 74.9 * 7.5 mm
Huawei Nova3	225	2017	157.0 * 73.7 * 7.3 mm

Data Collection. When collecting data, a volunteer speaks as a data collector attaching to his cheek-ear area where it senses strong vibration. However, for the stringency of our evaluation, we hereby specify three types of data collection scenes.

For the first data collection scene, hereafter referred to as *Scene 1*, we leverage a MEMS three-axis accelerometer as illustrated in Fig. 3. To acquire SIV signals, a user holds it to the front area of his ear, where people usually pick up a phone call (see Fig. 13(a)). When ready, before repeating a certain word 10 times at a one-second interval, the user remains silent for 3 seconds while data collecting program is running in the background.

For the second one, hereafter referred as *Scene 2*, we make use of a phone holder, keeping the phone at a fixed position to minimize the impact of body movements. Then, likewise, after a three-second silence, the user says a hot-word repeatedly in a normal tone with the phone, fixed, clinging on cheek-ear area where people usually pick up a phone call (i.e., microphone towards mouth while speaker towards ear as shown in Fig. 13(c)).

(a) Scene 1 (b) Phone holder (c) Scene 2 (d) Scene 3-case1 (e) Scene 3-case2 (f) Scene 3-case3

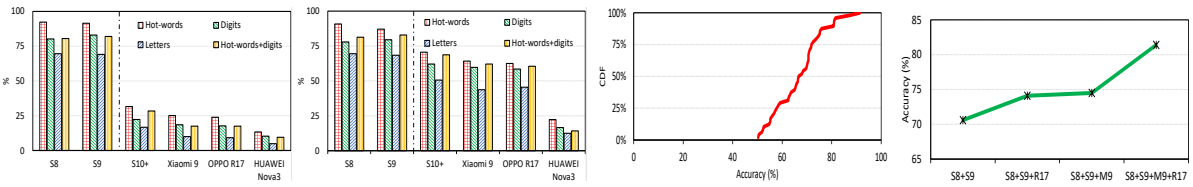
Fig. 13. Experimental setup.

In terms of the third one (i.e., *Scene 3*), we ask participants to take a variety of angles and gestures in the way they hold the devices:

- (1) Phone tilts at an angle of 45° along the user's cheek-ear area at which also points to the user's mouth as illustrated in Fig. 13(d) and denoted as *Scene 3-case1*;
- (2) Phone placed horizontally from the front area of ear towards the center of the user's face as illustrated in Fig. 13(e) and denoted as *Scene 3-case2*;
- (3) Phone placed vertically where speaker is at the front area of ear and microphone points to ground as illustrated in Fig. 13(f) and denoted as *Scene 3-case3*.

We have designed a set of 10 hot-words {*Password, Code, Key, Secret, Account, Salary, Encoder, Bank, Number, Word.*}, 10 numbers {*0 to 9*} and 26 letters {*A to Z*} and instructed volunteers to pronounce them as mentioned above.

Metrics. Word recognition is a multi-class classification problem. We use Confusion matrix, Accuracy false positive rate (FPR) and false negative rate (FNR) to comprehensively measure the overall performance of Vibphone.



(a) Densenet (b) Vib-Net (c) CDF (d) Impact of dataset size

Fig. 14. Report on overall performance.

6.2 Feasibility Results

In this section, we evaluate the overall speech recognition accuracy by Vibphone in terms of three cases: (1) trained phone for test, (2) untrained phones for test, and (3) importance of training with diverse devices.

Training and test settings. We altogether recruited 10 volunteers (4 females and 6 males) to participate in our experiments. To ensure the reliability and validity of experiment results, we have endeavored to eliminate potential influence factors on performance. During the experiment, we mount smartphones on a phone-holder with a fixed pose as illustrated in Fig. 13(c), where microphone towards mouth and speaker towards ear. Volunteers are instructed to pronounce 46 classes of contents, consisting of 10 digits (0-9), 26 letters (A-Z), and 10 hot words. Each class contains 3000 samples which are collected from 10 volunteers using 6 different types of smartphones.

We take data collected by SA-Accelerometer, Samsung S8 and Samsung S9 as the basic training set to construct an SIV-based word recognition model. The other data collected from the rest of the devices will be used as test set.

Recognition performance with trained devices. We first exploit established word recognition model on known devices (i.e., the phones Samsung S8 and Samsung S9 contributing to training set) to rate its primary performance in comparison with normal Densenet [25], which is widely used in computer vision tasks. As shown by the two items (S8 and S9) on the left side of dotted lines in both Fig. 14(a) and Fig. 14(b), by treating trained devices as test devices, both Densnet and VibphoneNet are with high and parallel recognition rate, which is up to 92.3% for hot-words, 83% for digits, 69.5% for letters, and 82.9% for the mixture of hot-words and digits. On the whole, the recognition rate of Densnet is slightly higher than that of VibphoneNet by 3.5%, which is a rational result because during training process, VibphonNet has considered latent impact from different devices by reducing their weights in back propagation but those elements may as well contain information that are helpful to word recognition task.

Recognition performance with untrained devices. To further evaluate Vibphone's potential for erasing the impacts of device diversity, we employ unknown devices as test set. See the items on the right side of dotted lines in both Fig. 14(a) and Fig. 14(b). By bringing unseen devices, i.e. S10+, Xiaomi 9, OPPO R17, and Huawei Nova3, to test, we observe a lopsided result- a huge performance gap between Densenet and VibphoneNet. To eavesdrop on an untrained device, the recognition accuracy of all types of voice contents (e.g. hot-words and digits) of Densnet drops sharply to as low as 17.5%. By contrast, the recognition accuracy of VibphoneNet still maintains at least 60.5% for hot-words, digits, and the mixture of hot-words and digits. The recognition accuracy of letters is about 47%, slightly lower than that of the other three types of contents. It is mainly attributable to the uniform intonation and fewer syllables.

Generally, our model is far superior to the benchmark by a large margin in overcoming device diversity.

One exception is the Huawei Nova3. The recognition accuracies of both VibphoneNet is less than 23%, well below the average of 68%. The reason lies in its low sampling rate (i.e., 225 Hz). In fact, owing to feature loss, eavesdropping via accelerometer with low sampling rate is always afflicted by poor recognition performance, which is consistent with former study [14] and the evaluation results on the impact of sampling rate in Section 6.4.

Except for Huawei Nova3, we plot the cumulative distribution function (CDF) of the average recognition accuracy of each of the 46 class contents in Fig. 14(c). More than 73% content samples can be recognized by VibphoneNet with the accuracy higher than 60%.

Importance of training with diverse devices. From another perspective, we study Vibphone's ability to consolidate its generalization ability by increasing the size and diversity of training set. The test set is now taken over by hotwords data only from Samsung S10+ while other phone models will be added to training set step by step except for Huawei Nova3.

As reported by the results in Fig. 14(d), it indicates that the generalization ability of feature extractor unit in the adversarial network is further enhanced being fed with more diversified devices and we conjecture that the difference in results between known and unknown devices can be fully eliminated if the model is trained by enough data from varied phone models.

6.3 Efficacy of Telephone Conversation Detection

The threat model requires Vibphone to escape the surveillance of smartphone operation systems. In spite of the zero-permission vulnerability, we still need a SIV detection mechanism in the interest of reducing Vibphone's battery consumption to an inconspicuous level. To measure the contribution of our SIV detection mechanism, we take false positive rate (FPR) and false negative rate (FNR) as metrics where:

- False positive rate (FPR): refers to the case in which Vibphone considers a "silent" move as calling.
- False negative rate (FNR): refers to the case in which it determines a calling action as a "silent" move.

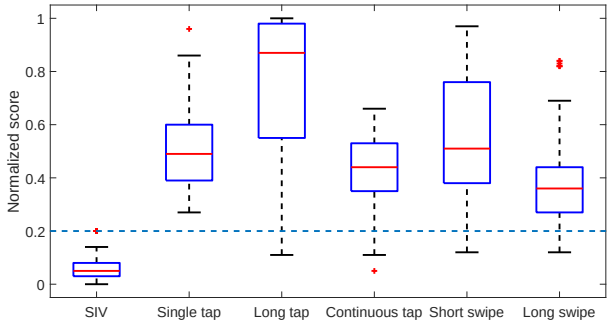


Fig. 15. Evaluation on SIV detection unit.

We instruct 10 participants hold a cell phone to collect varied types of gesture data, i.e., 10 subjects hold the cell phone in hands and perform all types of touch input moves that they would use in real life (i.e., single tap, long tap, continuous tap, short swipe and long swipe). The exact trajectory of gestures is not specified.

We conduct an experiment to measure the ability of the one-class SVM classifier to distinguish touch input and SIV. 10 volunteers are asked to tap and swipe on the screen of Samsung S9 and Samsung S10+ each for 50 times. Plus 1000 SIV samples randomly selected from the previous database, we make use of 2000 samples altogether. Then, we take 50 samples from touch input database respectively to train a one-class SVM classifier and then run a test on the model. As shown in Fig. 15, the FNR is 0%, and FPR is 5.3% in average, meaning no misses on any given SIV signals. In other words, Vibphone successfully detects phone conversations in all settings. This result further supports our discovery of dichotomous traits between SIV and touch input.

The detection is a success, while the higher FPR stems from the manual modification we made on the output threshold of the classifier to assure FNR is anchored to 0%. Such modification sacrifices FPR, but we believe FNR should take precedence over FPR because the attacker does not want to miss any potentially useful conversations.

6.4 Impact of Different Factors on Recognition Performance

In this section, we study the impact of different factors on the performance of SIV-based voice content recognition. The factors contain body movements, phone-holding posture, interference of top speaker, gender, and different sampling rate.

6.4.1 Impact of Body Movements. Although vibration signal induced by user mobility is considered to be low-frequency which can be erased through filtering, it may still weaken the features in acceleration signals because when a user moves, the smartphone might temporarily detach from the user's cheek. From Section. 3, we know motion sensors are susceptible to even tiny changes, so such behavior resulting in device displacement could be a negative factor on recognition accuracy.

In order to assess the impact of body movements, we list some common real-life situations that people pick up phone calls: standing up, walking and going upstairs. Note that these moves only corrupt spectrogram below 50Hz by themselves. We design an experiment that involves the above actions and 4 volunteers, in which the users start by holding up a device attaching to their cheeks. After that, they are instructed to pronounce the hot-words while performing the listed actions. Each hot-word is repeated 10 times by each participant to construct a thorough test set. Table 5 shows the top1 and top3 (testing) accuracy.

Good results in standing up and slow movements indicate that most of the interruptions are wiped out with the help of high-pass filter. However, we also notice that distortions incurred by strenuous exercises have some negative influences. Specifically, the distortions mainly trouble the segmentation and alignment unit because

Table 5. Evaluation on impact of body movements.

Action		Top1 acc (%)	Top3 acc (%)
Standing up		89.4	95.1
Walking	Slow	86.6	93.8
	Fast	75.7	88.4
Going upstairs	Slow	82	90.5
	Fast	71.2	87.3

the signal turns so irregular that sometimes we need to manually adjust the threshold value, which lower the precision of both segmentation and alignment.

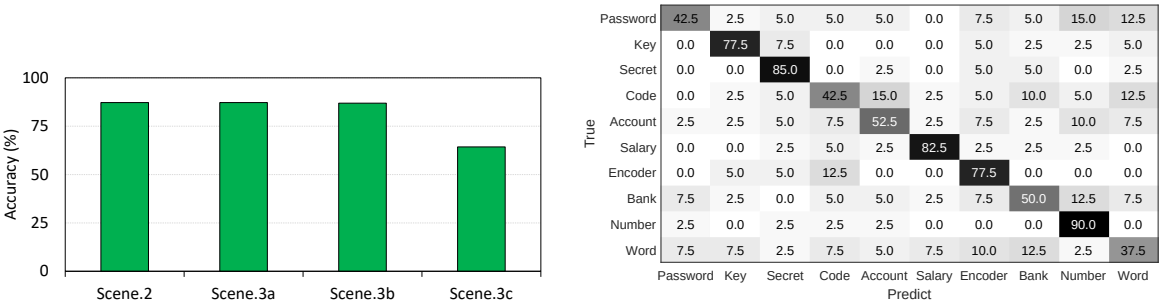
6.4.2 Impact of Phone Holding Posture. The posture and angle of cell phone at which people make contact with the screen is critical. So as to evaluate this variable, we also list 3 normal situations as illustrated in Fig. 13(d), 13(e) and 13(f) to take into consideration. For each posture, 4 volunteers are instructed to record data using Vibphone on a Samsung S9, as illustrated in Fig. 13(d) (i.e. *Scene 3-casse(1-3)*). We use the data collected from volunteers when mounting smartphone on the phone holder (see Fig. 6.2, namely *Scene 2*) as training set. The results are in Fig. 16.

From the results in 16, we have validation set at Scene.2 as control group, which provides an accuracy of 87.2%. For Scene 3-case 1, we have an accuracy of 87.2%, which is equivalent to the benchmark because the only difference between Scene.3(a) and Scene.2 is that it does not use the phone holder. Such result fits fine with our conclusion on the impact of body movements. For Scene.3(b), the accuracy is 87.0% , which is close to what we have in the first place. This is because when user holds the phone vertically, the contact area between screen and the upper jaw is almost identical to that of Scene.3(a) and that we only adopted data along z axis so that the vibration pattern changes little.

For Scene.3(c), the accuracy drops to 64.6% (See Fig. 16). In this setting, some hot words remain their identifiability while others do not. To be specific, *Key, Secret, Salary, Encoder, Number* have an average of 83.3% accuracy, but the others (i.e., *Account, Password, Code, Bank, Word*) only have 45.9%. Such result is accountable as we observe a change on the contact area and that the bright contours on spectrograms of the latter 5 are hardly modified but have relatively lower energy level. When the user holds the phone horizontally, there is less vibration conducting to the phone panel because roughly half of the contact area is at zygoma, which resonates less intensively with cranial cavity. As consonant generally has lower vibration energy itself, its influence is further dented, increasing the difficulty to distinguish hot words with the same syllable number. Nevertheless, we barely see anyone using such posture in real life so that it does not diminish the performance of Vibphone.

The above three generalize the most commonly seen situation where people talk through microphone, suggesting the model is relatively robust but slightly influenced by deviation caused by a rarely-used hand-grip posture.

6.4.3 Impact of Built-in Top Speaker. The caller/callee on another side of a phone conversation could also induce signal change in built-in accelerometer as the top speaker plays sound. Consequently, there is a chance that the accelerometer of the victim side will be affected. To study the impact of such interference, we recruit the 4 volunteers to participate in recording data at *Scene 2* as illustrated in Fig. 13(c) using Samsung S9 with the top speaker playing white noise that has equal intensity at different frequencies. Note that top speaker generally gives out sound with a much lower volume than bottom speaker. We set the victim smartphone at 5 different volume levels corresponding to (0%,100%] so that the dataset for each level consists of 400 gray-scale images from 4 volunteers.



(a) Results on different phone-holding posture

(b) Result confusion matrix of test on *Scene 3*

Fig. 16. Evaluation on impacts of phone-holding posture

As shown in Fig. 17, we can conclude that the mid-low volume has little impact on the performance of Vibphone because sound power level is much lower than vibration. There is a downturn on Vibphone's recognition ability when the volume is set as "high" and "highest" as the sound power largely increases the signal noise ratio (SNR). However, by surveying questionnaires from 100+ adults, we argue that most people, for the sake of privacy concerns and hearing health, do not set volume as "high" or "highest" when having a sensitive information-related phone conversation. Moreover, in most cases, two sides of a phone conversation, especially when speaking important information, do not speak simultaneously but one would wait until the other finishes.

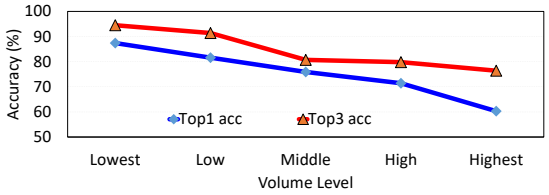


Fig. 17. Impact of top speaker. The recognition rate decreases as the top speaker volume goes higher

6.4.4 Impact of Gender Difference. Since women's voice is generally thinner and higher in pitch, it may be harder for accelerometer to preserve female voice feature. To study the extension of Vibphone with respect to gender, we design several experiments with 4 volunteers consisting of 2 females and 2 males. Table 6 shows the result.

We first ask the female participants to record data at *Scene 2* with Samsung S9 and obtain the average accuracy of 84.5% on the established model. In comparison, we have a result of 88.8% accuracy from other 2 adult male participants. Note that the original database encompasses 6 males and 4 females' data. Next, we build six training sets that hold varied numbers of female and male data only for this test. All of the training sets contain $2 \times 20 \times 10$ gray-scale images for 10 hot-words. We also form two test sets that exclusively contain female and male data, respectively.

From the results, we notice that the performance of training sets with data of same gender is under par but not vice-versa. Therefore, we conclude that the impact of gender difference to this model is quite narrow and can be wiped out via incorporating both genders into training set. This outcome could be explained by the fact that the typical fundamental frequency for an adult male and adult female respectively lie in the range 85-180Hz and 165-255Hz [44][45], both mainly staying in the range that an modern built-in accelerometer of smartphone can pick up owing to Nyquist theorem.

	Female test		Male test	
Dataset	Top1 acc (%)	Top3 acc (%)	Top1 acc (%)	Top3 acc (%)
0F2M	51.7	72.3	81.5	89.3
1F1M	65.8	79.7	68.2	80.5
1F2M	70.6	82.3	78.2	86.5
2F0M	83.3	92.8	54.1	75.5
2F1M	81	87.5	79.9	86.6
2F2M	80.6	88.5	81.3	90.8
4F6M	84.5	95.3	88.8	96.5

Table 6. Impact of gender difference.

Training set size	**Top1 acc (%)**	**Reference acc(%)**
1	75.6	89.6
2	83.5	90.1
3	86.5	91.5
4	90.6	92.2
5	91.2	92.5
6	91.7	92.5

Table 7. Impact of individual difference.

6.4.5 Impact of Individual Difference. In Section. 3, we notice a slight bias on the accuracy of un-trained speakers and argue it is people's pronunciation habits/accents to be blamed. However, it could be a major problem if Vibphone performs poorly on strangers. Next, we seek to examine the impact of this intrinsic nature.

Firstly, we invite another 5 male college students as volunteers to record hot-words data in *Scene 2* with Samsung S9. Each hot word is repeated 10 times. It is worth noting that S9 is also included in training set, and we chose only males because gender difference is somewhat influential in small-size training sets, as mentioned in Section 6.4.4.

After that, we re-train the model with data collected from 1 to 6 male participants, who contribute to the dataset in Section 6.2. We draw the result in Table 7. Results in right column are drawn from 10-fold cross validation as a reference. We observe that individual difference negatively influences Vibphone's performance when the training set is small because it is undeniable that people have different accents or pronunciation habits. However, it is also clear that the accuracy tends to align with the reference as the size grows, which proves such impact can be diluted through enlarging training set. To summarize, user diversity only has a negligible effect on Vibphone's performance and can be eliminated by covering more volunteers.

6.4.6 Impact of Sampling Rate. In Section. 3, we demonstrate that built-in accelerometers are pretty sensitive to our speech. However, technically, it is primarily determined by the maximum sampling rate of how much information an accelerometer can capture from speech-induced vibration. Experiments of past research have proven that smartphones before 2014 are ineligible for eavesdropping via built-in acceleromter because of their low sampling rate [8][6]. Our work is underpinned by the trend that modern smartphones are equipped with more advanced sensors that promote the sampling rate up to 425Hz, covering most of the frequency band of adult speech.

From this point, a new issue raised that whether sampling rate affects the accuracy of word recognition. In Table 4, we list some common smartphone models made by large phone manufacturers and their sampling rate. We notice that most of them lies in 410Hz-420Hz. To avert influences caused by device diversity, we leverage an

SA-accelerometer as the only data collector in this experiment. First, we set sampling rate of SA-accelerometer to 500Hz to collect data from an adult male volunteer (i.e., *Scene 1* in Fig. 13(a)). In total, we gather 500 raw samples to comprise a hot-word dataset, of which 20% are divided into test data. Then, we respectively down-sample the signals and group them by frequency. Next, after a normal workflow, we run tests on every combination of the groups.

The classification results are presented in the form of a confusion matrix in Fig. 18. As shown, the impact of discrepancy in sampling rate is limited in a small range, which covers most of real-life situation, guarding the robustness of Vibphone. However, as the discrepancy grows, the damage would be too much to endure. We contend that despite low accuracy under significant disparity situation, Vibphone still works only if the victim device satisfies minimum requirement (i.e., higher than 350Hz). In an actual attack, the sampling rate of target device is perceivable as Vibphone knows the number and timespan of gathered sequence. In this way, we can add devices with high sampling rates into training set and for victims that have a sampling rate far lower than those in training set, Vibphone can simply down-sample and re-train the recognition module to achieve a reliable attack.

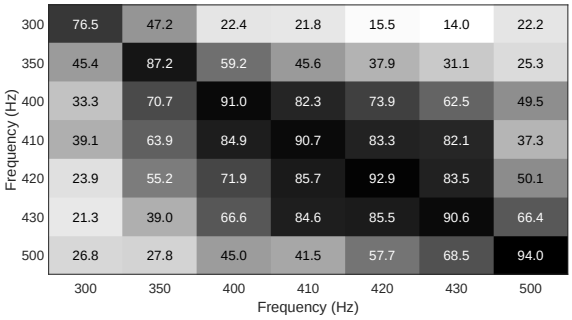


Fig. 18. Impact of sampling rate

6.5 End-to-end Case Study

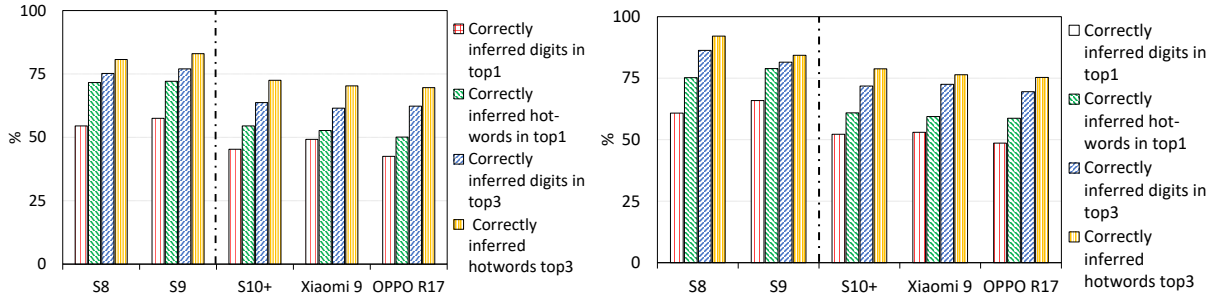
In this subsection, we explore Vibphone's capability of stealing confidential information in a real-world scenario where devices involved in training set inhere are SA-accelerometer, Samsung S8 and Samsung S9.

First, we conduct a full-scale experiment to measure its general ability to extract information from victims. 10 volunteers are asked to pronounce some hot words and digits. This is a type of indiscriminate attack where Vibphone plays a role as an " interpreter" that absorbs and tries to decode all distilled speech-induced vibration signals. In this setting, each of 10 volunteers is required to read 10 randomly selected hot-words and digits (i.e., 20×10 gray-scale images to form a test set). Also, we take device diversity into consideration. We leverage Samsung S8 and Samsung S9 as known devices to compare with three unknown devices of which the sampling rate is higher than 350Hz (i.e., Samsung S10+, Xiaomi 9 and OPPO R17).

As shown in Fig. 19(a), we observe a small discount on the accuracy because the interval between each pronunciation is unfixed and much shorter, which puts burden on segmentation and alignment.

However, in most cases, we do not wish to extract all information of victims but the valuable ones (e.g., password to the bank account). For simplicity, we presume that bank account only encompasses 10 digits and that the victim would give out their password during a phone call immediately after pronouncing *bank, account* and *password* unnecessarily in order. To be specific, volunteers are instructed to read out a hot-word sequence including *bank, account, password* and other 3 random ones preceding 10 digits (i.e., 20×10 gray-scale images to form a test set). It is worth noting that we do not build another special database for this type of attack, meaning

the training set is still 10 hot-words + 10 digits (0-9). Hence, the objectives of Vibphone can be summerized as recognizing the specified hot-words (trigger words) with a finite state machine and later switching to "digit" mode for the extraction of the succeeding 10-digit password. In Fig. 19(b), we locate enhancement in both known and unknown devices. In average, targeted recognition outperforms indiscriminate recognition by 8% accuracy. An explanation for the result is that the well-directed, practical assumption narrows down the scope of classes for recognition task as Vibphone will spare the trivial digits appear before trigger words and focus on the important ones.



(a) Indiscriminate attack

(b) Targeted attack

Fig. 19. End-to-end case study

Table 8. A naive solution of defense.

Sampling rate (Hz)	170	160	150	100	50
Accuracy (%)	54.2	51.9	50.4	39.6	32.7

Table 9. An adaptive approach for defense.

Period (s)	10	8	6	4	2
Accuracy (%)	37.5	29.2	16.4	12.6	10.8

7 DEFENSE

We now discuss possible directions to defend against the proposed attack. As mentioned above, sampling rate and security permission are the two keys to crackdown malicious apps like Vibphone.

The lowest frequency of typical human speech is around 85Hz, suggesting operation system set a constraint on BI-accelerometer to limit its sampling rate below 170Hz during telephone conversation. However, if fed with targeted data (i.e., training data has the same sampling rate with victim device), our model still presents adeptness in drawing speech information from the victim. We conduct an experiment leveraging an SA-accelerometer at *Scene1* as shown in Fig. 13(a). We set the sampling rate of SA-accelerometer at 170Hz, 160 Hz, 150 Hz, 100 Hz and 50Hz, respectively, to get the hot-words SIV. The test results are listed in Table 8. One interesting finding is that we still, at least have more than 30% accuracy on hot-word classification, meaning users are not safe even with 50Hz of sampling rate. We cannot lower the limit any longer because some gaming apps run at 50Hz [43].

Therefore, we propose a defense approach that OS would periodically regulate the sampling rate. In this case, though Vibphone can acquire real-time sampling rate of the targeted device, much of the useful information is damaged, causing trouble for model training. Likewise, we conduct an experiment at *Scene 1* to evaluate this adaptive approach by manually setting different periods for sampling rate to change from 50Hz to its maximum. As shown in Table 9, the recognition rate drops as time period decreases, and it even plummets to 10.8% when the period is set as 2 seconds, which is no better than guessing. Such method greatly secures modern smartphone users.

Another effective defense is to place restrictions on applying accelerometer permissions so that the user can decide whether to allow an app to collect accelerometer readings with a high sampling rate.

8 RELATED WORKS

In this section, we review the existing works on side-channel attacks on smartphones using motion sensors and acoustic attack related to Vibphone.

8.1 Side-channel Attack via Motion Sensors

The embedded motion sensors (i.e., accelerometer and gyroscope) are useful in supporting various mobile applications that require motion tracking or motion-based command. However, they also bring potential risks of leaking user's private information. Due to the nature of the motion sensors, they can measure the change in motion as well as the orientation of the devices. This could cause sensitive information leakage on mobile devices [11] [14] [17], . For instance, Keystroke inference using the motion sensors of mobile devices is another highly researched threat. TouchLogger [3], TapLogger [28], Accessory [30], TapPrints [29] and (sp)iPhone [27] utilize the accelerometer and gyroscope embedded on smartphones to infer keystroke sequence or passwords when the user inputs on a physical or smartphone's keyboard. This attack pattern is primarily because of the criticality associated with the informationer while accessing the Internet [38], [31]. Location tracking based on built-in motion sensors is one class of threats, focusing on tracking users' location and movements by using inertial navigation models [35] [4], [36]. Narain et al. [37] highlighted the privacy issues related to calculating a user location and his/her travel patterns using the side-channels (i.e. non-location sensitive sensors). Device fingerprinting attacks that enable an adversary to identify and track a user's device over time have also been proposed [32], [17]. Recently, researchers have successfully demonstrated more complex attacks using motion sensors, such as, inferring a target user's handwriting [39], factory floor secrets [33] and objects printed on nearby 3D printers [34].

8.2 Acoustic Attack with Motion Sensors

Acoustic eavesdropping using motion sensors has also received significant attention in the literature [6], [41]. Recently, a significant amount of research in the literature focused on how to eavesdrop on a user's phone call by exploiting the on-board zero-permission motion sensors such as gyroscope and accelerometer. Specifically, Gyrophone [6] showed that gyroscope is sensitive enough to measure acoustic signals from an external loudspeaker to reveal speaker information. Accelword [7] used smartphone's accelerometer to extract signatures from the live human voice for hotwords extraction. Speechless [8] further tested the necessary conditions and setups for speech to affect motion sensors for the speech leakage. It concluded that motion sensors may indeed be influenced by external sound sources as long as the generated vibrations are able to propagate along the surface to the embedded motion sensors of the smartphone, placed on the same surface. Furthermore, Pitchin [9] presented an eavesdropping attack using embedded motion sensors in an IoT infrastructure (having a higher sampling rate than a smartphone motion sensor) that is capable of speech reconstruction. On the basis of the above studies, Spearphone [10] explored the possibility of revealing the speech played by the smartphone's built-in speakers from the phone's own motion sensors recently. This setting is related to a large number of practical instances, whose privacy issues are still unexplored. Based on Spearphone, AccelEve [11] further employed deep learning to recognize and reconstruct speech information from the spectrogram representation of acceleration signals stimulated by loudspeaker. Recently, Isaac et al. [14] exploited the movements of the facial musculature during the pronunciation of words to attack smartphone call privacy.

The above studies, however, focus on studying the feasibility and necessary conditions for using motion sensors embedded in smartphones to eavesdrop on speech privacy. However, all existing works failed to cover the most adverse setup, where the motion sensors are utilized as a side-channel to capture the speech signals spoken by the caller/callee who takes the phone in hand. Different to exisitng works, Vibphone is the first accelerometer-based side channel attack against telephone caller/callee.

9 CONCLUSION

This paper revisits the threat of zero-permission motion sensors to speech privacy and proposes a highly practical side-channel attack against telephone caller/callee. We first present two fundamental observations that smartphone accelerometers are sensitive to SIV signals, and the device diversity has a decisive impact on the performance of telephone conversations attack. Moreover, we conduct an in-depth investigation on SIV features to determine the root cause of device diversity impacts and find out the critical features that are highly relevant to the voice content retained in SIV signals and independent of specific devices. On top of these pivotal observations, we propose Vibphone, a learning-based smartphone eavesdropping attack that could recognize SIV signals sourcing from vocal folds' vibrations. Specifically, Vibphone employs a feature agnostic adversarial neural network in a multi-task learning style to solve the limitation of device diversity and perform high accuracy speech recognition. The adversarial neural network is fed on spectrograms, meaning the input data must be standardized. In addition, our model takes an extra input of the aforementioned features to further boost its performance.

ACKNOWLEDGMENTS

We thank the editors and anonymous reviewers for their valuable comments and the participants involved in our experiments. This study is supported in part by NSFC Nos. 61902122, U20A20181, 61732017.

REFERENCES

[1] Xuanang Feng, Hiroki Shimokubo, Eisuke Kita: Personal Identification Through Pedestrians' Behavior. Rev. Socionetwork Strateg. 12(2): 237-252 (2018)
[2] Ivars Blums, Hans Weigand: A Financial Reporting Ontology for Market, Exchange, and Enterprise Shared Information Systems. PoEM 2019: 83-99.
[3] Liang Cai, Hao Chen: TouchLogger: Inferring Keystrokes on Touch Screen from Smartphone Motion. HotSec 2011.
[4] Sashank Narain, Triet D. Vo-Huu, Kenneth Block, Guevara Noubir: Inferring User Routes and Locations Using Zero-Permission Mobile Sensors. IEEE Symposium on Security and Privacy 2016: 397-413.
[5] Youngtae Yang, Byunggyu Lee, Jun Soo Cho, Suhwan Kim, Hyunjoong Lee: A Digital Capacitive MEMS Microphone for Speech Recognition With Fast Wake-Up Feature Using a Sound Activity Detector. IEEE Trans. Circuits Syst. II Express Briefs 67-II(9): 1509-1513 (2020).
[6] Yan Michalevsky, Dan Boneh, Gabi Nakibly: Gyrophone: Recognizing Speech from Gyroscope Signals. USENIX Security Symposium 2014: 1053-1067.
[7] Li Zhang, Parth H. Pathak, Muchen Wu, Yixin Zhao, Prasant Mohapatra: AccelWord: Energy Efficient Hotword Detection through Accelerometer. MobiSys 2015: 301-315.
[8] S. Abhishek Anand, Nitesh Saxena: Speechless: Analyzing the Threat to Speech Privacy from Smartphone Motion Sensors. IEEE Symposium on Security and Privacy 2018: 1000-1017.
[9] Jun Han, Albert Jin Chung, Patrick Tague: Pitchln: eavesdropping via intelligible speech reconstruction using non-acoustic sensor fusion. IPSN 2017: 181-192.
[10] S. Abhishek Anand, Chen Wang, Jian Liu, Nitesh Saxena, Yingying Chen: Spearphone: A Speech Privacy Exploit via Accelerometer-Sensed Reverberations from Smartphone Loudspeakers. arXiv preprint:1907.05972, 2019.
[11] Zhongjie Ba, Tianhang Zheng, Xinyu Zhang, Zhan Qin, Baochun Li, Xue Liu, Kui Ren: Learning-based Practical Smartphone Eavesdropping with Built-in Accelerometer. NDSS 2020.
[12] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, Wenyao Xu, Lu Su: Towards Environment Independent Device Free Human Activity Recognition. MobiCom 2018: 289-304.
[13] Mingmin Zhao, Shichao Yue, Dina Katabi, Tommi S. Jaakkola, Matt T. Bianchi: Learning Sleep Stages from Radio Signals: A Conditional Adversarial Architecture. ICML 2017: 4100-4109.
[14] Isaac Griswold-Steiner, Zachary LeFevre, Abdul Serwadda: Smartphone speech privacy concerns from side-channel attacks on facial biomechanics. Comput. Secur. 100: 102110 (2021).
[15] Soren Becker, Marcel Ackermann, Sebastian Lapuschkin, Klaus-Robert Müller, Wojciech Samek: Interpreting and Explaining Deep Neural Networks for Classification of Audio Signals. arXiv preprint:1807.03418, 2018.

[16] Jian Liu, Chen Wang, Yingying Chen, Nitesh Saxena: VibWrite: Towards Finger-input Authentication on Ubiquitous Surfaces via Physical Vibration. CCS 2017: 73-87.
[17] Sanorita Dey, Nirupam Roy, Wenyuan Xu, Romit Roy Choudhury, Srihari Nelakuditi: AccelPrint: Imperfections of Accelerometers Make Smartphones Trackable. NDSS 2014.
[18] Frederico Soares Cabral, Hidekazu Fukai, Satoshi Tamura: Feature Extraction Methods Proposed for Speech Recognition Are Effective on Road Condition Monitoring Using Smartphone Inertial Sensors. Sensors 19(16): 3481 (2019).
[19] Ron Kohavi, George H. John: Wrappers for Feature Subset Selection. Artif. Intell. 97(1-2): 273-324 (1997).
[20] Avrim Blum, Pat Langley: Selection of Relevant Features and Examples in Machine Learning. Artif. Intell. 97(1-2): 245-271 (1997)
[21] Isabelle Guyon, Andre Elisseeff: An Introduction to Variable and Feature Selection. J. Mach. Learn. Res. 3: 1157-1182 (2003).
[22] Anupam Das, Nikita Borisov, Matthew Caesar: Do You Hear What I Hear?: Fingerprinting Smart Devices Through Embedded Acoustic Components. CCS 2014: 441-452.
[23] Kyungho Joo, Wonsuk Choi, Dong Hoon Lee: Hold the Door! Fingerprinting Your Car Key to Prevent Keyless Entry Car Theft. NDSS 2020.
[24] Stephen McAdams: Perspectives on the Contribution of Timbre to Musical Structure. Comput. Music. J. 23(3): 85-102 (1999).
[25] Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger: Densely Connected Convolutional Networks. CVPR 2017: 2261-2269.
[26] "LibXtract". http://libxtract.sourceforge.net/
[27] Philip Marquardt, Arunabh Verma, Henry Carter, Patrick Traynor: (sp)iPhone: decoding vibrations from nearby keyboards using mobile phone accelerometers. CCS 2011: 551-562.
[28] Zhi Xu, Kun Bai, Sencun Zhu: TapLogger: inferring user inputs on smartphone touchscreens using on-board motion sensors. WISEC 2012: 113-124.
[29] Emiliano Miluzzo, Alexander Varshavsky, Suhrid Balakrishnan, Romit Roy Choudhury: Tapprints: your finger taps have fingerprints. MobiSys 2012: 323-336.
[30] Emmanuel Owusu, Jun Han, Sauvik Das, Adrian Perrig, Joy Zhang: ACCessory: password inference using accelerometers on smartphones. HotMobile 2012: 9.
[31] Chen Wang, Xiaonan Guo, Yan Wang, Yingying Chen, Bo Liu: Friend or Foe?: Your Wearable Devices Reveal Your Personal PIN. AsiaCCS 2016: 189-200.
[32] Anupam Das, Nikita Borisov, Matthew Caesar: Tracking Mobile Web Users Through Motion Sensors: Attacks and Defenses. NDSS 2016.
[33] Avesta Hojjati, Anku Adhikari, Katarina Struckmann, Edward Chou, Thi Ngoc Tho Nguyen, Kushagra Madan, Marianne Southall Winslett, Carl A. Gunter, William P. King: Leave Your Phone at the Door: Side Channels that Reveal Factory Floor Secrets. CCS 2016: 883-894.
[34] Chen Song, Feng Lin, Zhongjie Ba, Kui Ren, Chi Zhou, Wenyao Xu: My Smartphone Knows What You Print: Exploring Smartphone-based Side-channel Attacks Against 3D Printers. CCS 2016: 895-907.
[35] Jun Han, Emmanuel Owusu, Le T. Nguyen, Adrian Perrig, Joy Zhang: ACComplice: Location inference using accelerometers on smartphones. COMSNETS 2012: 1-9.
[36] Sarfraz Nawaz, Cecilia Mascolo: Mining users' significant driving routes with low-power sensors. SenSys 2014: 236-250.
[37] Sashank Narain, Triet D. Vo-Huu, Kenneth Block, Guevara Noubir: Inferring User Routes and Locations Using Zero-Permission Mobile Sensors. IEEE Symposium on Security and Privacy 2016: 397-413.
[38] He Wang, Ted Tsung-Te Lai, Romit Roy Choudhury: MoLe: Motion Leaks through Smartwatch Sensors. MobiCom 2015: 155-166.
[39] Tuo Yu, Haiming Jin, Klara Nahrstedt: WritingHacker: audio based eavesdropping of handwriting via mobile devices. UbiComp 2016: 463-473.
[40] Xiangyu Xu, Jiadi Yu, Yingying Chen, Qin Hua, Yanmin Zhu, Yi-Chao Chen, Minglu Li: TouchPass: towards behavior-irrelevant on-touch user authentication on smartphones leveraging vibrations. MobiCom 2020: 24:1-24:13.
[41] Nirupam Roy, Romit Roy Choudhury: Listening through a Vibration Motor. MobiSys 2016: 57-69.
[42] Sensor Overview: https://developer.android.com/guide/topics/sensors/.
[43] Uses-permission: https://developer.android.com/guide/topics/manifest/ uses-permission-element.
[44] Ingo R. Titze and Daniel W. Martin: Principles of voice production. Acoustical Society of America Journal, vol. 104, p. 1148, 1998.
[45] Ronald J. Baken and Robert F. Orlikoff: Clinical measurement of speech and voice. Cengage Learning, 2000.
[46] Kevin J. Coakley, Paul D. Hale: Alignment of noisy signals. IEEE Trans. Instrum. Meas. 50(1): 141-149 (2001).