Personal and Ubiquitous Computing
https://doi.org/10.1007/s00779-019-01337-7

ORIGINAL ARTICLE

SmartGrip: grip sensing system for commodity mobile devices through sound signals

Namhyun Kim[1] · Junseong Lee[1] · Joyce Jiyoung Whang[1] · Jinkyu Lee[1]

Received: 6 July 2018 / Accepted: 11 October 2019

Abstract

Although many studies have attempted to detect the hand postures of a mobile device to utilize these postures as a user interface, they either require additional hardware or can differentiate a limited number of grips only if there is a touch event on the mobile device's screen. In this paper, we propose a novel grip sensing system, called SmartGrip, which allows a mobile device to detect different hand postures without any additional hardware and a screen touch event. SmartGrip emits carefully designed sound signals and differentiates the propagated signals distorted by different user grips. To achieve this, we analyze how a sound signal propagates from the speaker to the microphone of a mobile device and then address three key challenges: sound structure design, volume control, and feature extraction and classification. We implement and evaluate SmartGrip on three Android mobile devices. With six representative grips, SmartGrip exhibits 93.1% average accuracy for ten users in an office environment. We also demonstrate that SmartGrip operates with 83.5 to 98.3% accuracy in six different (noisy) locations. Further demonstrating the feasibility of SmartGrip as a user interface, we develop an Android application that exploits SmartGrip, validating its practical usage.

Keywords Grip sensing system · Mobile device · Sound signals · Sound structure design

1 Introduction

A hand grasping for an object usually indicates its distinct intent [1, 2]; such intent is more noticeable when the object is a mobile device. For example, Fig. 1 depicts six different mobile device user grips. If the user wishes to take a photo, the user positions the mobile device horizontally and holds the top and bottom as indicated in Fig. 1a. Similarly, hand postures in Fig. 1 b, c, d, and e and f match game playing, calling, typing, and web surfing, respectively. Because users have their own distinct styles of grasping, utilizing hand postures as a user interface necessitates the detection of different user-defined hand postures. Although there have been many studies addressing the grip sensing problem, they either require additional hardware [3–8] or differentiate a limited number of grips (e.g., grasps with either the left, right, or both hands only) and operate only if there is a touch event on the mobile device's screen [9–11].

To overcome the limitations of the existing studies, this paper proposes a novel grip sensing system, called SmartGrip whose goal is as follows.

> SmartGrip senses a number of different user grips of a mobile device without (i) requiring any additional hardware and (ii) triggering a touch event on the screen.

To achieve this, SmartGrip utilizes sound signals. SmartGrip emits a series of sound signals from the speaker, records the signals using the microphone, and captures the change of the recorded signals based on different user grips. For example, Fig. 2 displays the changed recorded sound signals from two different user grips; the sound signals used for the figure are detailed in Section 3.2.

A short, preliminary version of this paper has been presented as a poster [21], which is 4 pages long.

✉ Jinkyu Lee
jinkyu.lee@skku.edu

Namhyun Kim
csci6108@skku.edu

Junseong Lee
acuworld@skku.edu

Joyce Jiyoung Whang
jjwhang@skku.edu

1 Sungkyunkwan University (SKKU), Suwon, Gyeonggi-Do, Republic of Korea

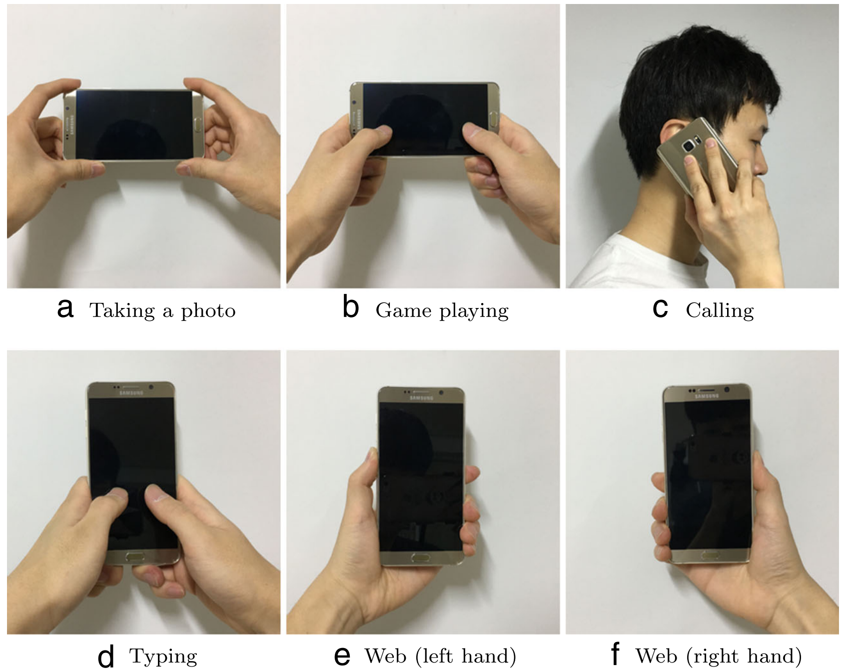


Fig. 1 Mobile device grips with different intentions

To design SmartGrip, the first step is to understand how a sound signal propagates from the speaker to the microphone of a mobile device. The emitted sound can be divided into two types: (i) *direct sound* and (ii) *reflected sound*. As the name indicates, direct sound passes directly from the speaker to the microphone. Although direct sound is independent of the mobile device location, it is affected by the shape and strength of the individual grasping hands due to their different acoustic absorption and damping/boundary conditions. This property has an important role in differentiating individual user grips. Reflected sound, on the other hand, is reflected by the surrounding environment, and thus highly dependent on the location. Therefore, the following requirements are necessary for accurate detection of individual grips regardless of locations:

R1. SmartGrip captures sufficient features from the the direct sound.

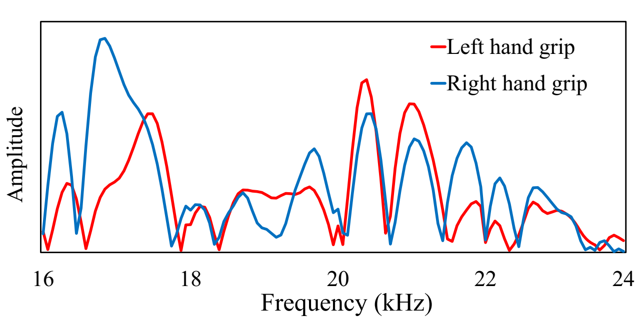


Fig. 2 Recorded sound signals for two different grasping hands

R2. SmartGrip reduces the effect of the reflected sound to the maximum possible.

Further, because SmartGrip creates sound, it must respect the following requirement for user comfort.

R3. SmartGrip must not disturb users with its sound signals.

In this paper, we attempt to achieve R1–R3 by designing and controlling the sound signals emitted by SmartGrip. In detail, we pose and address the following issues for SmartGrip's sound signals:

Q1. How to design the structures of the sound signals?
Q2. How to control the volume of the sound signals?

Regarding Q1, we carefully design the sound structure including the frequency range and amplitude, and its duration, which is intended to not only addressing R1 and R2 but also removing unusual sounds (addressing R3); this is detailed in Section 3.2. For Q2, we explore a tradeoff between achieving R1 and R2 (by higher volume) and achieving R3 (by lower volume), and then compromise the volume, as detailed in Section 3.3. By addressing Q1 and Q2, SmartGrip can extract sufficient features that distinguish different user grips. The remaining issue is as follows.

Q3. How to extract and classify features to distinguish different user grips?

For Q3, we apply a matched filter and fast Fourier transform (FFT) with hamming window technique, and then classify 172 resulting features using a support vector machine (SVM) as detailed in Section 3.4.

Based on the design, we implement SmartGrip as an Android application and validate it on three different mobile devices: Samsung Galaxy Note 5, Samsung Galaxy S8, and Google Pixel. With the six representative grips displayed in Fig. 1, SmartGrip exhibits 93.1% average accuracy for ten users in an office environment. We further demonstrate that SmartGrip operates with 83.5 to 98.3% accuracy in six different (noisy) locations.

Even though the experiments demonstrate the feasibility of SmartGrip as a user interface, we further develop a practical Android application that exploits SmartGrip; this automatically launches the target application mapped to the corresponding user grip. Please see our demo video for the application.[1]

In summary, this paper makes the following contributions.

- We propose the first grip sensing system through sound signals, which senses different hand postures of a commodity mobile device without additional hardware and touch event on the mobile device's screen.
- Based on the analysis of the sound signal propagation, we address the three technical challenges: sound structure design, volume control, and extraction and classification of features.
- We implement and evaluate SmartGrip, and demonstrate its effectiveness in detecting various hand postures with different users and environments.
- We develop an application that exploits SmartGrip, demonstrating the practical usage of SmartGrip.

The remainder of the paper is organized as follows. Section 2 discusses the related work. Section 3, the main section, presents design and implementation of SmartGrip with three technical issues: sound structure design, volume control, and extraction and classification of features. Section 4 evaluates SmartGrip via experiments. Section 5 proposes a practical application that enables SmartGrip to operate as a user interface. Section 6 discusses the limitations of SmartGrip. The paper concludes with Section 7.

2 Related work

As mobile devices have become more popular, considerable research issues regarding mobile devices have appeared. A significant research issue is to offer improved interaction between users and mobile devices; grip sensing is representative of the interaction challenges. Studies on grip sensing have been performed not only for mobile devices but also for other mobile devices. We can classify the studies into two categories as follows.

First, many researches have detected hand postures utilizing *additional hardware*. Yoon et al. [3] studied micromobility based on detection of hand postures on a tablet using a capacitive sensor. Hand Sense [4] distinguished grips by attaching a capacitive sensor to mobile or tangible devices. Hand Sense can distinguish six grips such as hold left/right, pick up left/right, and pick up top/bottom. Graspables [5] also used a capacitive sensor and pattern recognition software to sense the grips; Graspables detects the grips by applying the system to objects and suggests applications using it. iGrasp [6] used grip sensing technology to automatically change the keyboard depending on the user of the tablet. iGrasp also attached a capacitive sensor to the back of the tablet to detect the user's grip. Touch & Active [7] used acoustic signals to recognize touches and grips of target objects; a vibration speaker and piezo-electric microphone must be attached to the objects. Touch & Active has applied their system to different objects including a ceramic bowl, plastic toy, and mobile device. It has been demonstrated that they can detect touches and grips on those objects. iRotate Grasp [8] proposed an interface that changes the orientation of the mobile device automatically according to the viewing orientation of the mobile device user. To accomplish this, 32 light sensors were attached to the back of the mobile device. Although successful in differentiating individual grips, the studies belonging to the first category require additional hardware, making it impossible to utilize the studies for commodity mobile devices as they are.

Unlike the studies belonging to the first category, there have been a few studies that address grip sensing without any additional hardware. GripSense [9] and a study by Park and Ogawa [10] detected hand postures using only the built-in sensors (such as accelerometer and gyroscope) of the mobile devices. ContextType [11], a subsequent study of GripSense proposed an adaptive text entry system based on grip sensing. Although the studies in the second category succeeded in differentiating a limited number of grips without additional hardware, they required a touch event on the mobile device's screen for grip sensing. On the other hand, the grip sensing system proposed in this paper, SmartGrip, does not require additional hardware and touch event. In addition, SmartGrip is differentiated from the existing studies in that it is the first grip sensing system utilizing sound signals.

[1] https://www.youtube.com/watch?v=FvQ87wmS6kk

3 Design and implementation of SmartGrip

In this section, we design a new grip sensing system, SmartGrip, which enables mobile devices to detect different hand postures without additional hardware and screen touch events by satisfying R1–R3 mentioned in Section 1. To achieve this, we first analyze how a sound signal propagates from the speaker to the microphone of a mobile device. Based on the sound propagation analysis, we address technical issues by answering Q1–Q3, which correspond to sound structure design, volume control, and feature extraction and classification, respectively. Finally, we describe the implementation of SmartGrip.

3.1 Sound propagation analysis

To distinguish different grips using sound signals, this subsection first analyzes how a sound signal propagates, and then summarizes technical issues for the SmartGrip design. Figure 3 illustrates how a sound signal emitted by the mobile device speaker reaches the microphone. The emitted sound can be divided into two types: *direct sound* and *reflected sound*.

Directed sound (as the name indicates) goes directly from the speaker to the microphone; its sound spectrum change is depicted in Fig. 4. As indicated in the figure, the main reasons for a spectrum change are (a) hardware imperfection and (b) the resonant property of each mobile device and acoustic absorption of each user grip, which are detailed as follows.

First, when a mobile device attempts to emit a sound signal and then record it, the actual signal emitted by the speaker and recorded by the microphone is slightly different from the original signal. This is because the speaker and microphone are imperfect in terms of hardware design; when the sound is emitted by the speaker and recorded by the microphone, a loss of amplitude occurs at certain frequencies [12].

The emitted direct sound propagates through the mobile device body and air. The direct sound propagated through air is affected by the skin of the hand. Because the human hand has different acoustic absorption coefficients for each frequency [13], the directed sound propagated through air is attenuated in certain frequency ranges, which depend on the user's grip.

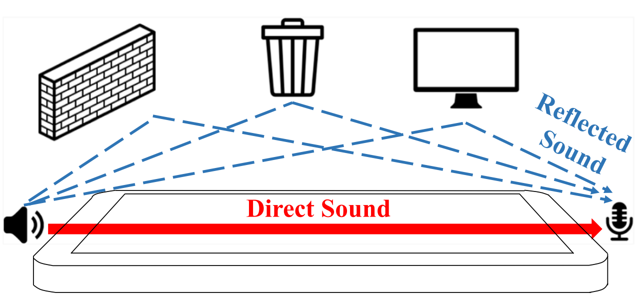


Fig. 3 Propagation of a sound signal on a mobile device

Similarly, the direct sound propagated through the body of the mobile device is changed based on the user's grip, which is related to the damping and the boundary conditions [14, 15]. Because the change of the damping and boundary conditions depends on the shape and strength of the grasping hand, the recorded directed sound propagated through mobile device's body with a specific user grip is different from that with another grip, even if the same sound signal is emitted from the speaker. This interesting property has been recently used in the mobile systems' area owing to its effectiveness in generating a unique sound spectrum [7, 16]; the property was originally studied in the architecture field.

Unlike direct sound, reflected sound is reflected by the surrounding environment. This can be an opportunity in specific situations. For example, Tung and Shin [17] created an acoustic signature that identifies the location of the mobile device using the reflected sound, resulting in fine-grained indoor localization. However, reflected sound leads to uneven attenuation in certain frequency ranges depending on the surrounding environment, and can result in a different sound spectrum even with the same grip, which makes it difficult to detect individual user grips.

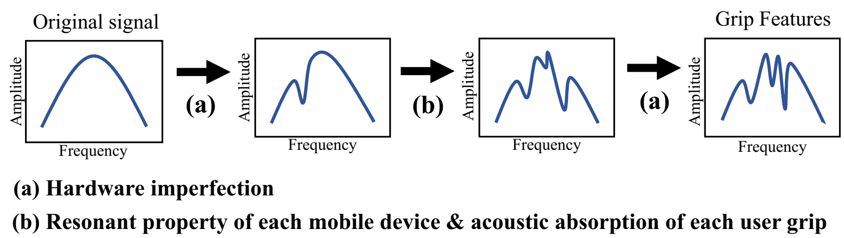
Therefore, designing SmartGrip using sound signals entails utilizing the direct sound (i.e., achieving R1) and removing the effect of the reflected sound (i.e., achieving R2). In the next subsections, we explain how to satisfy R1 and R2 for grip sensing accuracy and R3 for user comfort by addressing Q1–Q3.

3.2 Sound structure design

To distinguish different grips, we must carefully design the sound signals to achieve both accuracy of grip sensing and user comfort (meeting Q1–Q3). To accomplish this, we determine the frequency range and structure of the generated sound signals, as explained in the following.

First, we must determine the frequency range. It is well known that the maximum frequency people can hear is between 19 and 20kHz [18]. Therefore, it would be better to use a frequency higher than 20kHz to prevent the user from hearing the sound. However, through our experiments, we determined that a higher frequency sound yields reduced amplitude compared with a lower frequency sound when the sound is recorded by a microphone. The reduced amplitude makes it difficult to capture the change of sound signals for each user grip. Therefore, we inevitably include some lower frequency range to capture the grip features more clearly. Considering that Android supports up to 48kHz sampling

Fig. 4 Spectrum change of direct sound on a mobile device



rate, meaning that the maximum recordable frequency is 48kHz/2=24kHz, we select a frequency range from 16 to 24kHz, where we maximize the use of the high frequency while securing a sufficient number of features for recognizing the change of our sound signals by different user grips. Although we use a frequency range that a user could hear, sound set to a proper volume does not interfere with the user, which is discussed in Section 3.3.

Further, we must determine the structure of the sound signals. Because we have previously determined the frequency range, the simplest structure of the sound signals to be generated is a single linear chirp signal sweeping from 16 to 24kHz. However, such a unified chirp signal presents problems. The most significant problem is regarding the length of the overlapped reflected sound; as the sound signal becomes longer, the length of the overlapped reflected sound also becomes longer, meaning that SmartGrip is more affected by the surrounding environment. To solve this problem, we divide the single long chirp signal into four short chirp signals and set each chirp signal's length as 200 samples. Further, we remove the interference between different chirps by setting the distance to 500 samples between two consecutive chirps. When we extract features from the recorded sound signals, we use only “200 samples” of each recorded chirp signal. Because the speed of sound is approximately 340m/s, we can completely block the reflected sound farther than 1.4m. Moreover, fade in & fade out must be applied to each chirp signal to eliminate the phenomenon that generates unusual sounds due to abrupt energy changes [19].

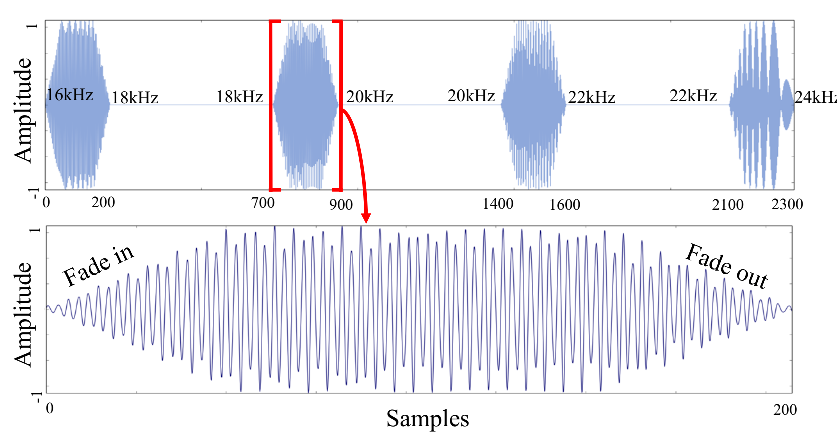
Note that we empirically determined the number of samples for each chirp signal as 200 samples. If the number is overly small, the signal is not sufficient for extracting features for grip classification. Conversely, with a greater number of samples, the effect of the reflected sound becomes larger, making it difficult to differentiate individual grips.

To summarize, we generate sound signals as indicated in Fig. 5. We use four sound signals to sense a single grip. Each of the sound signal is a linear chirp signal sweeping from 16 to 18kHz, from 18 to 20kHz, from 20 to 22kHz, and from 22 to 24kHz. The length of each chirp signal is 200 samples and the interval between each chirp is 500 samples. Therefore, the total time to sense a single grip is approximately $(200\times4+500\times3)\times48\text{kHz}=47.9\text{ms}$. Further, the fade in & fade out was applied to each chirp as indicated in the lower subfigure of Fig. 5.

3.3 Volume control

The proposed sound structure design allows SmartGrip to not only collect sufficient features from direct sound but also block reflected sound from the environment within 1.4m of the mobile device. However, despite its effectiveness, the sound structure design does not completely address R1–R3, omitting following issues. First, it is possible for the reflected sound of a chirp to overlap with the observed “200 samples” for the chirp. Second, people can hear the sound signals based on the proposed sound structure design owing to the use of frequencies

Fig. 5 Sound structure design for SmartGrip



lower than 20kHz. To address these two issues, we should carefully control absolute volume of the sound signals.

For the first issue, we performed experiments as follows. We emitted and recorded sound signals with different volumes and measured the amplitude of the direct sound and reflected sound through a band-pass filter and matched filter. We observed that the ratio between amplitude of the direct sound and that of the reflected sound increased as the volumes increased. For example, the ratio was 4:1 with 20% volume, while the ratio changed to 10:1 with 50% volume. Therefore, we can reduce the effect of reflected sound using a high volume.

When it comes to the second issue, we performed a usability study. We exposed ten participants to sound signals with different volumes and asked them if they were audible. Their age ranges from 20 to 32 years. When the volume was greater than 50%, eight out of ten participants could hear the sound signal. However, at 50% volume, eight out of ten participants could not recognize the sound signal, and the other two participants stated that the sound was negligible.

Therefore, we chose to use 50% volume (for Samsung Galaxy Note 5).[2] With 50% volume, SmartGrip not only reduced the effect of the reflected sound to 1/10th of the direct sound but also exhibited no or minimal disturbance to the users.

3.4 Feature extraction and classification

Although a buffer for the recorded sound signals has useful information regarding the current grasping hand, the signals not only have noise but also are expressed as a time domain, which must be transformed into a frequency domain. Thus, SmartGrip must extract the features from the recorded sound signal buffer and make the sound signals a feature set to define the current grip. To accomplish this, we first apply a matched filter. The matched filter removes the noise from the recorded sound buffer and searches for the starting point where the sound signal is recorded. After applying the matched filter, we set the sample index with the peak value as the starting point of the sound signal. Then, FFT is performed based on the starting point of the sound signal. A hamming window is used for FFT where the window size is 1024. We fill the first 200 samples of the window with the actual data of the chirp signal, and the remaining 824(=1024–200) samples with zeros to prevent unnecessary reflected sound from being transformed together. Because the sound signal is composed of four chirps, the matched filter and FFT are performed four times, once for each chirp. After eliminating the unnecessary parts from FFT results of each chirp, we combine them as one. Consequently, we extract 172 features[3] in total, which are defined as an individual grip.

Using the extracted 172 features, we classify the grips into six predefined classes via SVM [20], which is a well-known machine learning method for the classification task. We build personalized classification models, where the data from a user is applicable to the user only. This is because different users exhibit different characteristics of grasping hands (e.g., size, the amount of hair, thickness of skin); therefore, the same grip from different users yields different extracted features. We trained the SVM using 4800 samples and tested the classification performance on 1200 samples (we collected 6000 samples for each user). To tune the parameters for the SVM, we used 5-fold cross validation and selected the parameters that yielded the highest accuracy. (Hence, we used the linear kernel and set the penalty parameter $c = 0.5$ for SVM.)

3.5 Implementation

We implemented SmartGrip as an Android application. Except for certain libraries such as LIBSVM and FFT, we used Android APIs only. Because SmartGrip requires neither any kernel modification nor Android-specific APIs, the system can be easily extended to other mobile operating systems, e.g., iOS or Windows.

4 Evaluation

For evaluation, we experimented three different mobile devices: Samsung Galaxy S8, Samsung Note 5, and Google Pixel. The evaluation consists of three parts. First, we present grip sensing accuracy of SmartGrip. We then demonstrate that SmartGrip operates within various environments in terms of noise level and the size of the location. Finally, we evaluate the effect of the concurrent use of multiple mobile devices employing SmartGrip.

4.1 Accuracy

In this subsection, we measured the accuracy of SmartGrip for different people. We included ten participants (P1–P10), eight males and two females; their ages range from 20 to 32 years. The characteristics of their hands were considerably different. For example, a male participant had hard, thick leathery hands, whereas a female participant had

[2] Note that the criterion volume depends on mobile devices. For Samsung Galaxy S8 and Google Pixel, the criterion volume corresponds to 60%. It is not difficult to calibrate the criterion volume for target mobile devices.

[3] After applying FFT, we have 512 features with 0–24kHZ. We remove features with 0–16kHZ as well as additional features related to fade in & fade out, yielding 172 features.

soft, smooth hands. Another participant had significant hair on his hands, considerably more than the other participants.

Our experiments were conducted in an office environment displayed in Fig. 6a. For each participant, we collected the data for the evaluation as follows. They were required to view and grasp the six different grips indicated in Fig. 1. For each grip, we recorded 100 measurements. However, if each participant grasped the mobile device 100 times in the same position and location, the accuracy would be expected to be high because of the constant reflected sound and constant sound delivery. For more realistic scenarios, we had each participant repeat the following process 20 times for each grip. First, each participant grasped the mobile device with the target grip, and we measured the grip five times. Then, s/he set down the mobile device and changed its location. We collected $20{\times}5 \times 6 = 600$ grip data of the six grips from each participant. This data collecting procedure was independently repeated on the Samsung Galaxy S8, Galaxy Note 5, and Google Pixel mobile devices.

SmartGrip used the personalized classification model for each participant and each mobile device. We applied 5-fold cross validation to measure the accuracy of SmartGrip. Figure 7 a displays the accuracy of classifying the six grips according to the participants. Among the ten participants P1–P10, P1 exhibited the highest accuracy, an average of 96.5% for the three mobile devices; the person with the lowest accuracy was P6, an average of 86.7% for the three mobile devices.

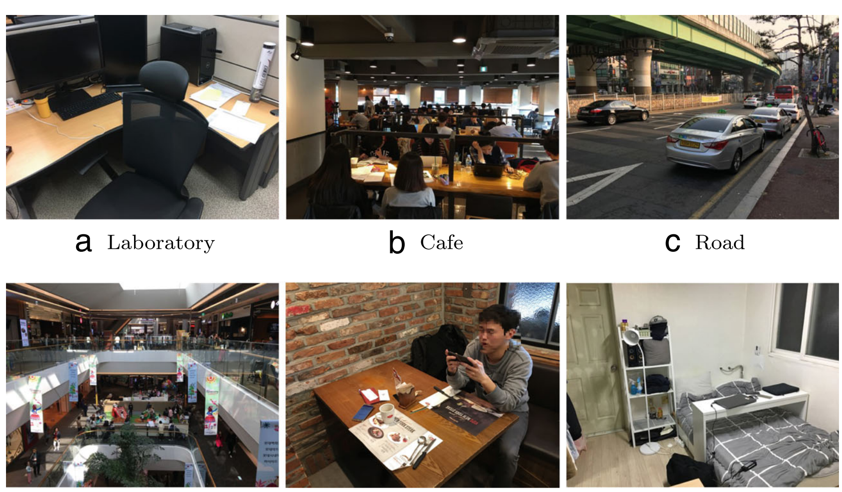
Figure 7 b indicates the accuracy for the different mobile devices. In Fig. 7b, we display the average accuracy of the ten participants with the corresponding mobile device. The average accuracy was 95.7% for Galaxy S8, 91.5% for Pixel, and 92.2% for Galaxy Note 5 in an office environment. The standard deviation was 4.5, 2.4, and 3.5 for Galaxy S8, Pixel, and Galaxy Note 5, respectively. The reason for the low accuracy at the Pixel is that the microphone of the Pixel did not thoroughly record the high frequency range (22–24kHz), compared with the other mobile devices. In total, the average accuracy of the three mobile devices was 93.1%.

Figure 8 displays the confusion matrices of the six grips of participant P3, whose accuracy is the closest to the average accuracy. The *y*-axis represents the actual grip assumed by the participant; the *x*-axis indicates the predicted grip by SmartGrip. Thus, denser diagonal blocks indicate improved prediction performances. We can observe that SmartGrip with the Galaxy Note 5 evenly differentiated the six grips accurately, whereas the lowest accuracy was with Pixel. In particular, among the six grips, SmartGrip with Pixel had difficulty in distinguishing between Grip #1 and #2, corresponding to Fig. 1 a and b, respectively.

4.2 Noise and environmental disturbances

In this subsection, we evaluated SmartGrip in different locations to demonstrate how accurately SmartGrip operated. We performed experiments using SmartGrip in the six locations in Fig. 6 to measure the effect of the reflection sound and external noise of everyday life. Note that we did not intentionally control the environments of the locations. For example, there were six people working on their own studies and some running equipment in the laboratory in Fig. 6a, where the degree of noise was 63dB. At the cafe in Fig. 6b, there were approximately 20 people talking or studying, with an average of 76dB noise. At the mall in Fig. 6d, a number of people were enjoying the shopping with music playing; the average noise was approximately 85dB.

Note that the previous subsection indicates that the standard deviation of the ten participants was an average 3.5, which means the accuracy for SmartGrip does not depend highly on participants. Therefore, we performed experiments for a single user in the six different places



a Laboratory b Cafe c Road

d Mall e Restaurant f Home

Fig. 6 Locations for evaluation

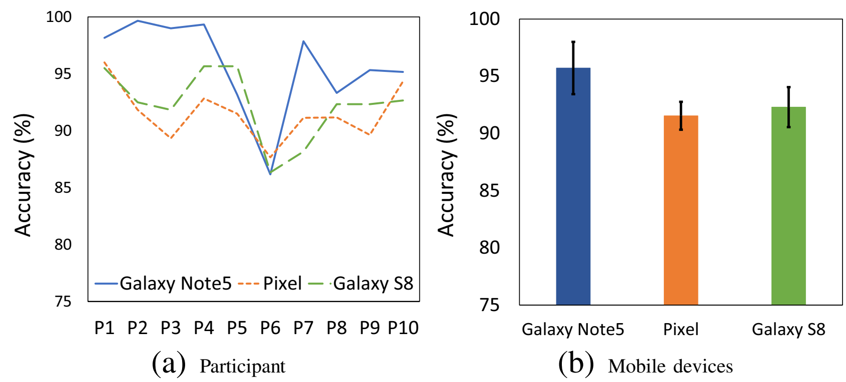


Fig. 7 Accuracy depending on different participants and mobile devices

and observed the accuracy according to the variation in the locations and noise levels.

Figure 9 a shows the average accuracy of the three mobile devices in different locations. While the average accuracy in the six locations was 91.6%, the home and the road exhibited the highest and lowest accuracy of 98.3% and 83.5%, respectively. As indicated in the figure, external noise is inversely proportional to accuracy, and the accuracy difference between the highest and lowest noise level was about 14.8%(=98.3–83.5). Figure 9 b demonstrates that the accuracy of the different locations did not significantly rely on the target mobile devices. Of the three mobile devices, the Galaxy Note 5 and Pixel exhibited the highest and lowest overall accuracy, respectively, which agrees with the results of the previous subsection.

One may wonder why SmartGrip works effectively with Samsung Galaxy Note 5, compared with Samsung Galaxy S8 and Google Pixel. Such performance difference comes from many factors, and it is difficult to figure out which factors and how much the factors affect the performance of SmartGrip. We conjecture that the major factors are (i) the quality and placement of the speaker and microphone, and (ii) the material and structure of the mobile device body through which the sound propagates. In the future, it would be interesting to figure out the factors and the degree of effect thereof.

4.3 Disturbances between multiple users

SmartGrip records and analyzes sound signals in the frequency range from 16 to 24kHz. Therefore, if there is another user using SmartGrip at the same time in the same location, there could be interference between the two. We conducted the following experiment to identify the separation distance required to prevent interference. We let one mobile device emit the sound signal designed for SmartGrip with 50% volume (irregularly) four or five times per second, while a second mobile device measured the accuracy of SmartGrip with different distances between the two mobile devices (30cm, 60cm, 90cm, 150cm, 180cm, and 210cm) as illustrated in Fig. 10a. As shown in Fig. 10b, the accuracy increased as the distance between the two mobile devices increased. With a distance of 180cm, we can achieve similar accuracy to the accuracy with a single user, implying that multiple users can use SmartGrip simultaneously assuming their distance is no closer than 180cm. Note that the distance of 180cm is a safe, pessimistic bound for concurrent

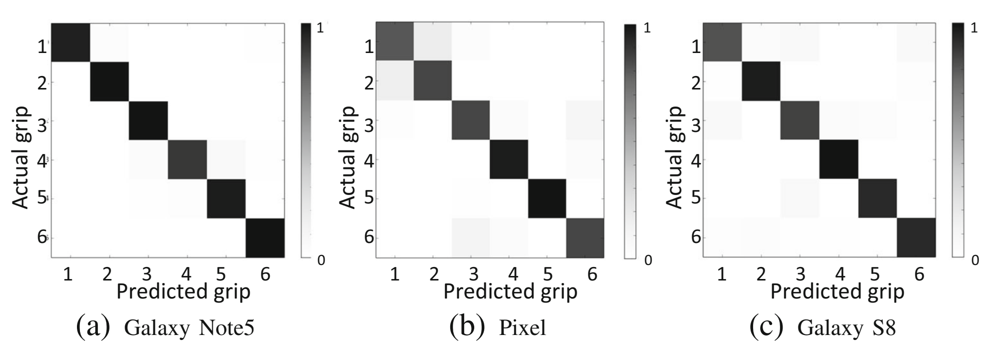
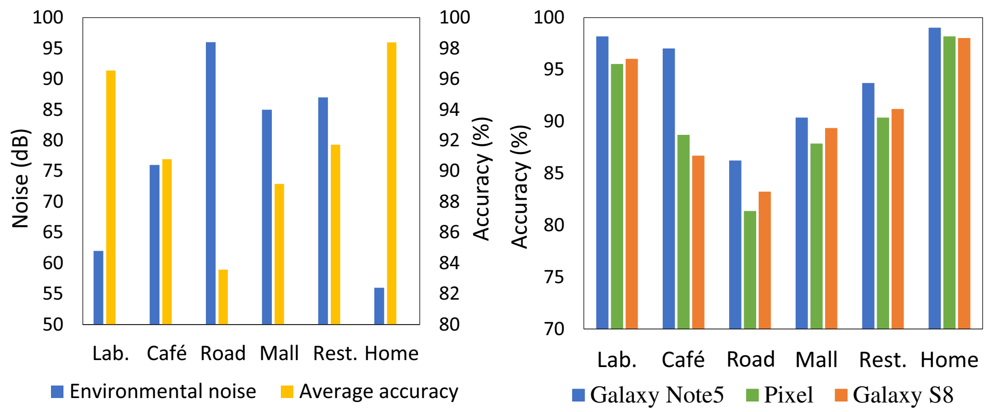


Fig. 8 Confusion matrix of P3



(a) Noise & average accuracy (b) Mobile devices

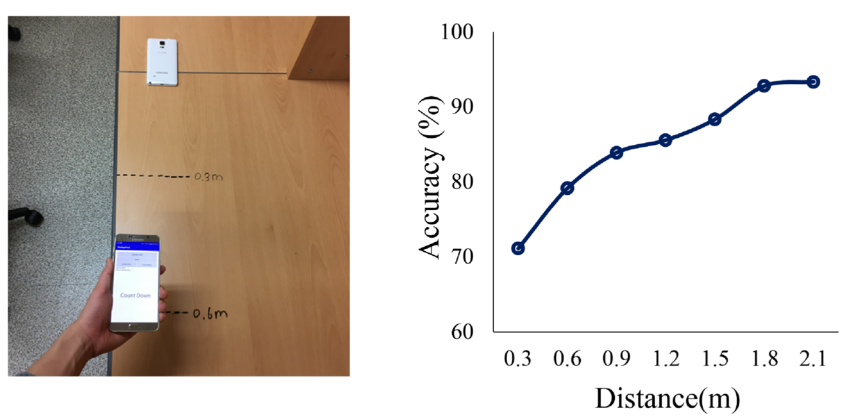
Fig. 9 Accuracy depending on locations

execution of SmartGrip. In reality, multiple mobile devices employing SmartGrip do not continuously generate sound signals, yielding minimal opportunity of interference even within a 180-cm placement; we present an example in Section 5.

5 Application of SmartGrip: AppLauncher

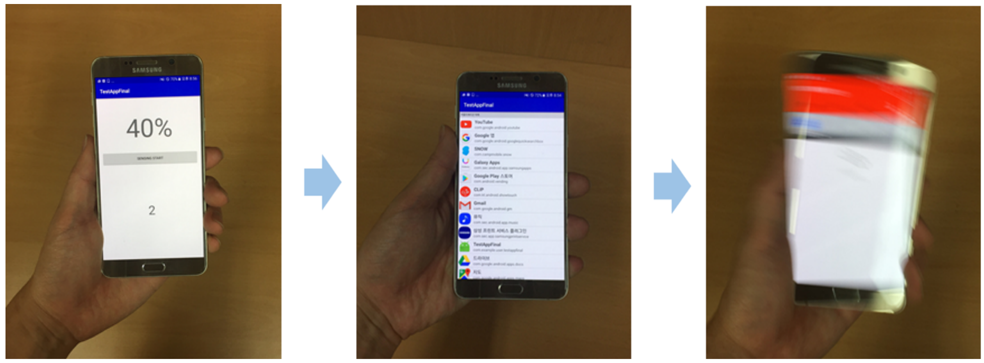
In this section, we propose an application called AppLauncher to demonstrate the practical use of SmartGrip as a user interface. As an Android background service, AppLauncher detects and differentiates a user grip (by the feature of SmartGrip), and launches the target application matched with the grip. Figure 11 presents the setup and operation of AppLauncher. First, a user is required to grasp the mobile device five times to register a user grip. Note that SmartGrip uses a personalized classification, and therefore this registration process is necessary. For each grasp, AppLauncher emits a predefined sound signal, and records the sound signal using the microphone (by the feature of SmartGrip). Then, the user selects the application to be triggered by the corresponding grip. Finally, whenever the user shakes the mobile device with the preregistered grip, AppLauncher automatically launches the mapped application. Because a user can initiate the target application simply by grasping and shaking the mobile device, AppLauncher delays a short time to launch the target application. Further, because the user continues using the grasping hand for the target application, we can significantly reduce the number of screen touches for running the target application. See our demo video for the application: https://www.youtube.com/watch?v=FvQ87wmS6kk.

Figure 12 displays the overall system architecture for AppLauncher, consisting of three parts: initialization, shake detection, and grip estimation. In the initialization step, AppLauncher generates a series of sound signals that are carefully designed for detecting different grips. For reducing the effect of reflected sound, it controls volume by the feature of SmartGrip. We store the generated signals in a buffer, which enables AppLauncher to emit the generated signals quickly upon a user's grip sensing request (by the shake event). After the initialization step, the shake detection step initiates as a background service



(a) Experiment setting (b) Accuracy

Fig. 10 Accuracy according to distances between two mobile devices using SmartGrip at the same time



(a) Register grip (b) Select application (c) Launch application

Fig. 11 Setup and operation of AppLauncher

for performance optimization, which operates as a trigger. Without any sign for detecting the current grip, we could periodically generate sound signals for the detection, which could waste mobile device's power and computing resources; using this trigger, it is probabilistically safe for multiple users within 180cm to use AppLauncher. To achieve this, we use a mobile device built-in sensor, the accelerometer; if the current acceleration exceeds a predefined threshold, AppLauncher regards this situation as a shake event and allows the speaker to generate the predefined sound signals. To avoid multiple triggers within a single shake event, AppLauncher pauses the shake event detection for a predefined duration once it generates sound signals. We empirically determine two thresholds: 2.7 m/s^2 for shake event outbreak and 2 s for pausing the detection process, both of which are easily adjusted for reflecting user behaviors. After the shake event is detected, AppLauncher emits and records sound signals, and then enters the grip estimation process (according to SmartGrip). Once the user grip is identified, AppLauncher executes the mapped application.

6 Discussion and limitations

While we demonstrated the feasibility of SmartGrip in sensing smartphone grips, the technology proposed in this paper can be potentially utilized to other applications. First, it is possible for SmartGrip to help user authentication for mobile devices. That is, after registering a series of grips, a user can be authenticated by matching the series of grips with a predefined order. Considering SmartGrip does not match a grip with a hand, with the same grip with a different hand, SmartGrip is useful for user authentication. Second, it is impossible or very costly to add a new interface (such as a button) to COTS-based (commercial off-the-shelf-based) devices. SmartGrip allows users to implement a new interface to COTS-based devices such as AI speakers and smart watches, as long as the devices are equipped with a speaker and microphone. The abovementioned potential applications need to tailor the technology used in SmartGrip, which is our future work.

Now, we discuss limitations of SmartGrip as follows. First, although we have reduced the impact of reflected sound through sound structure design and volume control, we cannot completely eliminate the impact, yielding an opportunity for further improvement. Second, the accuracy of SmartGrip is influenced by external noise. Third, SmartGrip cannot distinguish a grip covering the speaker. This is because the grip that directly covers the speaker produces a strong and unpredictable reflection sound by the hand. Finally, it is possible for multiple users using SmartGrip to interfere with each other if their distance is closer to 180cm.

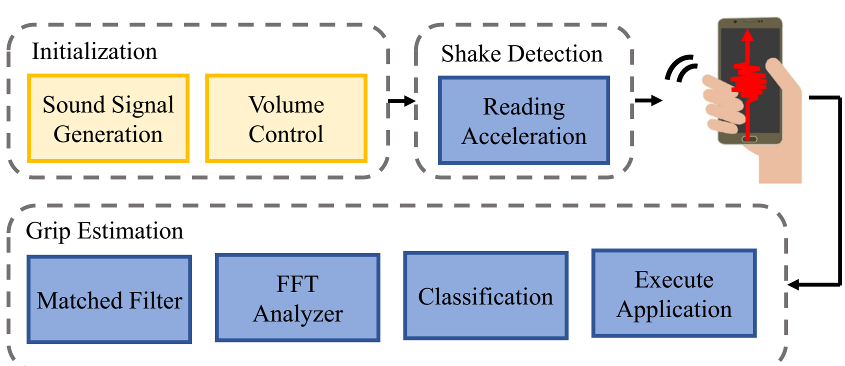


Fig. 12 System overview of AppLauncher

We may address the abovementioned limitations at the expense of some inconvenience. That is, we can block the microphone pinhole as indicated in Fig. 13. By blocking the microphone pinhole, we can prevent reflected sound from recording through the microphone. The sound recorded by the microphone is primarily direct sound propagated through the mobile device body. Consequently, SmartGrip with a blocked microphone pinhole is resilient to reflection sound and external noise. Based on our experiments, this adaptation yields at least 96% accuracy of all six locations displayed in Fig. 6. Further, it is possible to distinguish a grip that directly covers the speaker, and multiple users can use SmartGrip even within a distance of 30cm. One may think that it is not user-friendly to block the microphone pinhole. In this case, we may improve the accuracy of SmartGrip by applying adaptive parameter settings tailored to (i) each user and/or (ii) noise and environmental disturbances. As future work, we would develop personalized and/or noise-specific sound signal generation and volume control.

In addition to the above limitations, SmartGrip has one more limitation. That is, since SmartGrip relies on emitting and recording sounds, it is inevitably affected by other applications that use sounds on the target mobile device. If one of the applications emits sounds at the same time as SmartGrip emits sounds, SmartGrip may not distinguish grips due to the sound interference. As of now, a simple solution is to disable SmartGrip when other applications use sounds. In the future, it would be interesting to develop solutions on how to make SmartGrip work with those applications.

Fig. 13 Method of blocking microphone pinhole

7 Conclusion

In this paper, we presented a novel grip sensing system SmartGrip, which detects different hand postures of a mobile device through sound signals, without any additional hardware and screen touch event. By analyzing how a grasping hand on a mobile device influences the sound signal delivered from the speaker to the microphone, we addressed the technical issues of SmartGrip design, which are sound structure design, volume control, and feature extraction and classification. Our evaluation results confirmed that SmartGrip classified six widely used grips with 93.1% average accuracy for ten different users in an office environment, and 83.5–98.3% accuracy in six different (noisy) locations. Further, we proposed a practical application that utilizes SmartGrip. This sufficiently demonstrated the feasibility of utilizing hand postures as a user interface.

Funding information This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2019R1A2B5B02001794). Jinkyu Lee is the corresponding author.

References

1. Creem SH, Proffitt DR (2001) Grasping objects by their handles: a necessary interaction between cognition and action. J Exp Psychol Hum Percept Perform 27(1):218
2. MacKenzie C, Iberall T (1994) The grasping hand. advances in psychology, vol 104. New York: North Holland 26:1–32
3. Yoon D, Hinckley K, Benko H, Guimbretière F, Irani P, Pahud M, Gavriliu M Sensing tablet grasp + micro-mobility for active reading, in: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, ACM, pp 477–487
4. Wimmer R, Boring S (2009) Handsense: discriminating different ways of grasping and holding a tangible user interface. In: Proceedings of the 3rd International Conference on Tangible and Embedded Interaction, ACM, pp 359–362
5. Taylor BT, Bove VM Jr (2009) Graspables: grasp-recognition as a user interface. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, pp 917–926
6. Cheng L-P, Liang H-S, Wu C-Y, Chen MY (2013) iGrasp: grasp-based adaptive keyboard for mobile devices. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, pp 3037–3046
7. Ono M, Shizuki B, Tanaka J (2013) Touch & activate: adding interactivity to existing objects using active acoustic sensing. In: Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology, ACM, pp 31–40
8. Cheng L-P, Hsiao F-I, Liu Y-T, Chen MY (2013) iRotateGrasp: automatic screen rotation based on grasp of mobile devices, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, pp 3051–3054
9. Goel M, Wobbrock J, Patel S (2012) Gripsense: using built-in sensors to detect hand posture and pressure on commodity mobile

phones, in: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology, ACM, pp 545–554
10. Park C, Ogawa T (2015) A study on grasp recognition independent of users' situations using built-in sensors of smartphones. in: Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, pp 69–70
11. Goel M, Jansen A, Mandel T, Patel SN, Wobbrock JO (2013) Contexttype: using hand posture information to improve mobile touch screen text entry. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, pp 2795–2798
12. Zhou Z, Diao W, Liu X, Zhang K (2014) Acoustic fingerprinting revisited: generate stable device ID stealthily with inaudible sound. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, ACM, pp 429–440
13. Ackerman E, Oda F (1962) Acoustic absorption coefficients of human body surfaces. Tech. rep., DTIC Document
14. Ewins D (2000) Modal testing: theory, practice, and application, Research Studies Press,
15. Schwarz BJ, Richardson MH (1999) Experimental modal analysis. CSI Reliability week 35(1):1–12
16. Tung Y-C, Shin KG (2016) Expansion of human-phone interface by sensing structure-borne sound propagation. In: Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services, ACM, pp 277–289
17. Tung Y-C, Shin KG (2015) Echotag: accurate infrastructure-free indoor location tagging with smartphones. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, ACM, pp 525–536
18. Plack CJ (2005) The sense of hearing. Lawrence Erlbaum Associates Publishers
19. Lazik P, Rowe A (2012) Indoor pseudo-ranging of mobile devices using ultrasonic chirps. In: Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems, ACM, pp 99–112
20. Chang C-C, Lin C-J (2011) Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2(3):27
21. Kim N, Lee J (2017) Towards grip sensing for commodity smartphones through acoustic signature. In: Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, pp 1–4