SideSwipe: Detecting In-air Gestures Around Mobile Devices Using Actual GSM Signals

Chen Zhao[1], Ke-Yu Chen[1], Md Tanvir Islam Aumi[2], Shwetak Patel[1,2], Matthew S. Reynolds[1,2]

[1]Electrical Engineering, [2]Computer Science & Engineering, UbiComp Lab, DUB Group
University of Washington
Seattle, WA 98195 USA
{chzhao, kychen, tanvir, shwetak}@uw.edu, matt.reynolds@ee.washington.edu

ABSTRACT

Current smartphone inputs are limited to physical buttons, touchscreens, cameras or built-in sensors. These approaches either require a dedicated surface or line-of-sight for interaction. We introduce SideSwipe, a novel system that enables in-air gestures both above and around a mobile device. Our system leverages the *actual (unmodified)* GSM signal to detect hand gestures around the device. We developed an algorithm to convert the bursty reflected GSM pulses to a continuous signal that can be used for gesture recognition. Specifically, when a user waves their hand near the phone, the hand movement disturbs the signal propagation between the phone's transmitter and added receiving antennas. Our system captures this variation and uses it for gesture recognition. To evaluate our system, we conduct a study with 10 participants and present robust gesture recognition with an average accuracy of 87.2% across 14 hand gestures.

ACM Classification: H.5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

Keywords: GSM, antenna, in-air hand gesture

INTRODUCTION AND MOTIVATION

Despite an exponential increase in the computation and storage capabilities of mobile devices, the space available for touch interaction will continue to be limited by what users are willing to carry. Although capacitive touch displays have effectively made the entire front face of most mobile devices interactive, occlusion remains an inherent problem. Researchers have proposed many ways to extend interaction on small mobile devices. This includes leveraging the backs of devices [20] or the space directly in front and behind the device using cameras and microphones [8,19]. Other works utilized the space directly above the screen using Electric-field sensing like GestIC[4], the surface on which a device may reside [9], and even the human body [10].

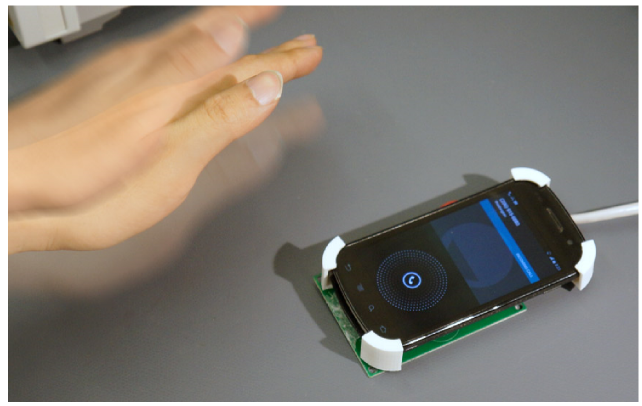

Figure 1. The SideSwipe system. Our system leverages the unmodified GSM signal to enable in-air hand gestures both above and on the side of the phone.

The spaces that currently remain "blind" from existing sensors are the areas surrounding the mobile device (*i.e.,* the sides, top, and bottom). These areas are out of the view of the front and back facing cameras and microphones. In addition, placing proximity sensors on the sides of a mobile device [2] would largely be occluded when a user holds it. An alternative method explored in the past uses magnetic-field sensing for inputs [3,11,17]. However, these techniques require the user's finger to be instrumented with a permanent magnet. Other researchers have explored RF signals radiated from the ambient environment to detect 1D hand gestures using a low-power analog receiver [12]. This work has been limited to simple 1D gestures and requires either a continuous signal source, such as an RFID emitter or a constant envelope ambient signal, like that from a digital TV tower. Our approach is similar to [12]; however, we leverage the GSM signal that already exists on a mobile phone and is modulated as a result of human motion. Instead of filtering these modulations out, we show that they can be used for enhancing user experience.

In this paper, we introduce SideSwipe, a new system that enables in-air gestures around a mobile device using the existing GSM signal (Figure 1). Specifically, when the hand moves around the phone, the reflected GSM signal is amplitude-modulated as a result of induced multipath propagation. Our system captures such variations from the re-

UIST '14, October 05 - 08 2014, Honolulu, HI, USA
Copyright 2014 ACM 978-1-4503-3069-5/14/10…$15.00.
http://dx.doi.org/10.1145/2642918.2647380

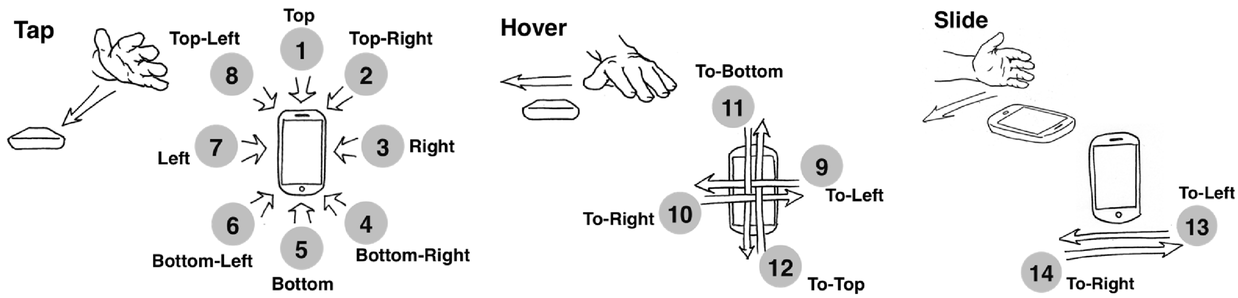

Figure 2. The gestures performed in the user study, including eight tapping gestures (**left**), four hover gestures (**middle**) and two sliding gestures (**right**). The sliding gestures (13 and 14) were performed at the bottom of the phone.

flected GSM pulses and uses them to extract hand gestures. That is, unlike prior work, we do not require a custom transmitter or an external signal source. Our approach requires little modification to the mobile device and could easily be adopted by mobile phone manufacturers. We demonstrate the robust recognition of 14 gestures: tapping at corners, tapping on the four sides of the phone, swipes over the face, and swipes at the bottom of the phone (Figure 2).

Specifically, our contributions include:

1. A technique using the *unmodified* GSM signal, which is bursty in nature, for in-air gesture sensing.
2. A simple antenna array that captures amplitude-modulated GSM signals to enable hand gesture detection around the phone.
3. A set of interaction techniques that leverage this capability.

RELATED WORK

Because of the increasing popularity of Natural User Interfaces (NUI), gesture recognition has become a very promising approach to enhance human-computer interaction. Kim *et al.* instrumented IR cameras on the wrist to capture finger gestures [13]. Commercial products, such as Kinect [18] and Leap Motion[1], use computer vision to detect in-air gestures for laptops and desktops. For mobile devices, PointGrab[2] and CrunchFish[3] are also popular commercial solutions using computer-vision based technologies for gesture recognition. Recently, the Samsung Galaxy S4 introduced in-air gestures using a front-facing IR camera. While these systems are able to detect hand or finger gestures, they require direct line-of-sight (LoS) between the device and may be brittle due to varying lighting conditions. SideSwipe, on the other hand, does not require LoS and works in various scenarios, even when the phone is in the pocket or when the gesture occurs outside camera's field-of-view.

Apart from these vision-based techniques, there are other interesting approaches that use electric field (E-fields), Doppler effect, and inertial sensing. GestIC[4] is a promising E-field based sensing technique, but it requires a comparatively larger surface area to act as an antenna ($\sim$10cm^2) and restricts hand gestures to the front and back surfaces. SoundWave [8] is an ultrasonic-based Doppler shift detection technique, but it only enables 1D gestures. Finally, there are some on-body sensing solutions using inertial sensing, but they require instrumenting the human body [6,7]. Our system addresses all these limitations, as it does not require any instrumentation on the human body.

Over the past decades, wireless signals have proven to be a promising approach in detecting a moving object [1,4]. Researchers have also used various wireless signals for localization [5,15,21]. Recently, researchers have started using wireless signals to recognize gestures. Pu *et al.* extract gestures from wireless signals like Wi-Fi at a whole-home range [16]. While these solutions are promising, they require a dedicated, continuous signal source, a fundamental limitation that inhibits their ubiquity in practical scenarios. In contrast, our system utilizes the unmodified GSM signal inherently radiated from mobile devices for gesture recognition and classification.

BACKGROUND AND THEORY OF OPERATION

The SideSwipe system leverages GSM signals to detect hand gestures. GSM networks operate in several carrier frequency bands (e.g., AT&T uses 850MHz/1900MHz, while T-Mobile uses 1900MHz). GSM is a frequency-modulated signal that uses two different frequency components to transmit logical 1's and 0's, a technique called Gaussian Filtered Minimum Shift Keying (GMSK). Comparing to amplitude-shift keying and on-off keying signals, GMSK signal maintains a nearly constant envelope. As a minimum shift modulation, GMSK encodes each bit as a half sinusoid, which reduces the power fluctuation caused by non-linear distortion. The envelope of the GMSK signal is independent of the transmitted data; that is, we can measure the magnitude of the fading propagation channel without knowing the content or encryption status of transmitted data.

[1] https://www.leapmotion.com/

[2] http://www.pointgrab.com/

[3] http://crunchfish.com/

[4] http://www.microchip.com/pagehandler/en_us/technology/gestic

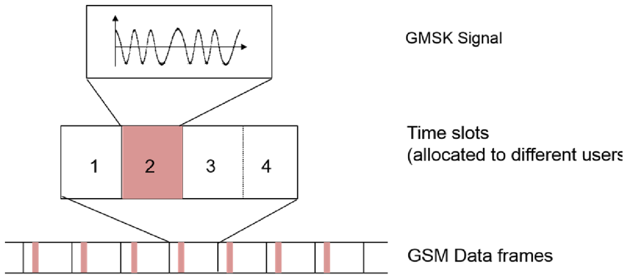

Figure 3. Time Division Multiple Access (TDMA) in the GSM system. The bandwidth is divided into frames for multiplexing. The GSM signal appears as a sequence of random bursts.

In order to allow multiple users to share the same frequency channel, GSM uses Time Division Multiple Access (TDMA), which divides the bandwidth into different slots (Figure 3). In turn, the phone's transmitted GSM signal is a sequence of bursts in certain time slots. Without knowing the modulation details, the distribution of slots appears as the outcome of a random process. In addition, the cellphone signal is normally mixed with other communication signals (e.g. WiFi, Bluetooth and even LTE). To leverage these random signal bursts for gesture recognition, our system filters out unnecessary signals and leverages the GSM pulses to sample the continuous motion of the user's hand.

Detecting Hand Gestures

When the user is making a call, the phone transmits GSM pulses to communicate with the cellular station. Around the mobile phone, there are a number of propagation paths. Such paths start from the transmitting antenna of the mobile phone and end up at each of our receiving loop antennas. When the user moves their hand around the mobile phone, their skin, muscle, and bones affect the character of the propagation path by absorbing or reflecting part of the signal. The absorption reduces the signal intensity, while the reflection generates Doppler shift. As a combination of the signals from all propagation paths, the envelope of the receiving signal will be changed by the absorption and Doppler shift. The outcome of these effects is an amplitude-modulated envelope. Our system captures this envelope modulation and uses it for gesture recognition. We will detail our algorithm in the following section.

The GSM pulses non-uniformly sample the propagation channels at staccato time intervals. Since the average frequency of the pulse is nearly 80 Hz, which is more than twice the rate of hand gestures (on the order of 10 Hz), we can reconstruct the propagation channel variation by the received GSM pulses [14].

IMPLEMENTATION DETAILS AND ALGORITHM

In this section, we describe the SideSwipe system in detail, including the hardware design and gesture detection algorithm.

Hardware Design

To build a system for detecting hand gestures, we designed a circuit board with four antennas (Figure 4). These four antennas (i.e., top, down, left, and right) allow the system to capture the signal fluctuations caused by hand movements from all directions around the phone. Each of these 4 receiving channels has a loop antenna and a RF power detector (Analog Devices AD8361). We designed four directional loop antennas pointing to distinct directions. The receiving antenna array is attached to the back of the phone and close to the transmitting antenna. The radius of the loop antennas is 1 cm and its circumference is close to a quarter of the GSM wavelength. It should be noted that we chose to design a custom antenna array to allow for hand gestures from all directions around the devices; however, the technique presented here applies to existing antenna systems in current mobile devices as well, with the limitation that they will allow only 1-D gesture recognition.

In our prototype, we put four antennas on the four edges of the PCB board and separately connect them to RF power detectors located at the center of the PCB. Since every antenna has a unique radiation pattern, we are able to get different signal intensity fluctuations from distinct antennas when the user is performing gestures. In addition, we placed a ground plane on the back of the PCB to enhance the difference in radiation patterns of the antennas.

Antenna Simulation

We use CST Microwave Studio to simulate our antennas and PCB board as well as calculate the radiation patterns (Figure 5). The simulation is done at 850MHz (*i.e.,* the same frequency in AT&T's GSM network). These figures show distinct radiation patterns for each of the four antennas. Because the sensitivity range of each antenna is determined by the direction and gain of the main lobe, each antenna is able to receive the signal through a unique propagation path. In our simulation, we only consider parameters from the PCB board such as antenna size and distance. Although modeling an accurate radiation pattern requires thorough consideration of all environmental factors, including

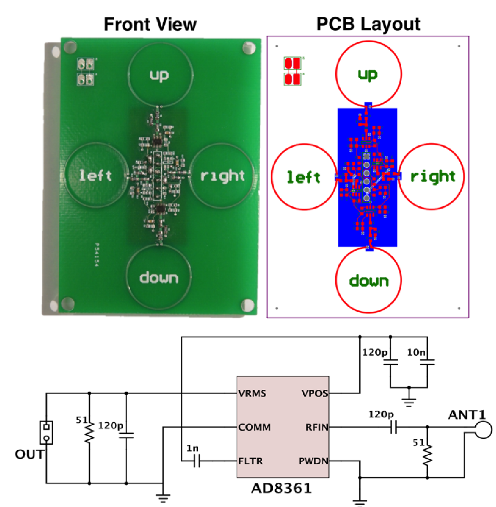

Figure 4. The PCB design of SideSwipe, including the front view of circuitry **(left)**, the layout **(right)** and the schematic view of the up antenna **(bottom)**.

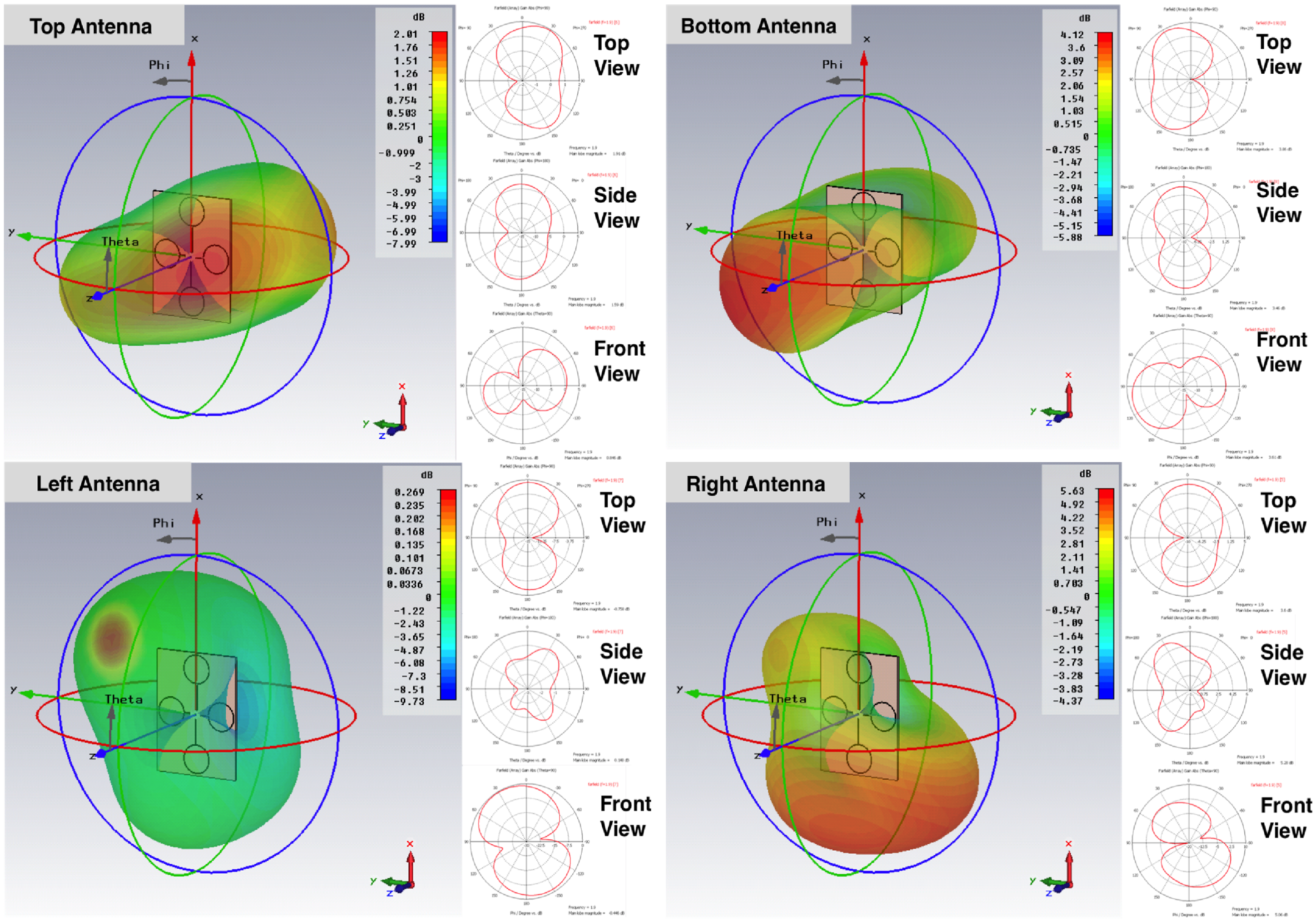

Figure 5. Simulation results for the top, bottom, left and right antenna. Each antenna has a distinct sensitive range and is able to perceive signals through their respective propagation path. The received signals from the four antennas yield unique signal patterns that can be used for gesture detection and classification.

the human body, clothes, and cellphone, the simulation provides guidance for the antenna design. Given this simulation, we are able to understand the sensitivity of sensing a gesture from all directions around the board and the best instrumentation of this board to a mobile device. That is, it describes the *interaction space* and provides guidance in designing a new gesture.

Gesture Recognition Algorithm

SideSwipe leverages unmodified GSM bursts that inherently exist when the user is using GSM communication (*e.g.*, making a call). When one performs a hand gesture, the antennas pick up the fluctuation in their respective propagation paths. By combining the signals from four antennas, we are able to identify a unique pattern for different gestures. In our prototype, we used a sliding window of 2.5 seconds to capture a hand gesture. Figure 6 shows the block diagram of our gesture recognition algorithm, including four major steps: (1) Bursts to Continuous Wave interpolation, (2) Smoothing and Segmentation, (3) Feature Extraction and (4) Gesture Recognition.

Bursts to Continuous Wave interpolation

As described earlier, the unmodified GSM signal appears as random bursts in communication. We need to first convert these bursts to a continuous signal before they can be used to extract any features for gesture recognition. To do this, we developed an algorithm to detect GSM pulses. Intuitively, we identify the GSM pulses by suppressing undesired wireless signals (*e.g.*, WiFi or Bluetooth) and interpolating these pulses to a continuous wave. Based on our observation, the duration length of each GSM pulse is approximately 0.67 ms (*i.e.,* 12 samples at our system's sampling rate of 18 KHz) and WiFi and Bluetooth signal both have relatively shorter pulse widths (*i.e.*, less than 8 samples). Therefore, we apply a threshold on the pulse width at 8 samples to filter these undesired signals. Furthermore, we use a fixed threshold (80 mV for our hardware design) to determine the rising and falling edges of the pulses.

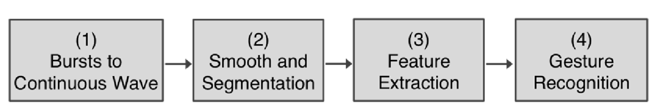

Figure 6. The gesture recognition algorithm. The system converts the actual GSM bursts to a continuous wave, which can be used for gesture recognition.

Output voltage of receiving channels (V)				V_{det} (V)
Top	Down	Left	Right	
3.22	3.22	3.22	3.22	**1.49**
3.22	3.22	3.22	3.22	**1.26**
3.25	3.22	3.22	3.22	**0.99**
2.11	3.22	3.22	3.22	**0.78**
1.56	3.22	2.81	3.22	**0.60**
1.57	3.22	2.80	3.22	**0.49**
1.32	2.66	2.24	2.91	**0.38**
0.99	2.09	1.75	2.23	**0.31**
0.58	1.47	0.95	1.74	**0.25**
0.47	1.18	0.77	1.41	**0.21**
0.37	0.94	0.62	1.12	**0.18**
0.30	0.75	0.49	0.89	**0.15**
0.32	0.64	0.51	0.69	**0.13**
0.24	0.49	0.39	0.52	**0.12**
0.20	0.41	0.32	0.44	**0.11**
0.16	0.35	0.25	0.36	**0.10**

Table 1. The profile of the five channels on an NI DAQ (see Fig 10). V_{det} is the voltage referring to the phone's transmitting power. By normalizing the receiving signals, our system eliminates the undesired fluctuation due to varying transmitting power.

The GSM system usually adjusts its transmitting power to maintain a stable gain in a fluctuating environment and in turn affects the receiving power density perceived by our antenna array (Figure 7). Such fluctuation is undesirable in gesture recognition, as two signal patterns of the same gestures may appear completely different. To adapt to this fluctuation, we normalize the received data corresponding to the transmitting power. For this, we recorded 20 minutes of data without any gestures being performed and saved the transmitting power and receiving gain in a table (Table 1). When receiving a GSM pulse, the system measures the transmit power from the phone and uses it to find the corresponding receiver gain in Table 1. Next, we divide the mean voltage of the GSM pulse with this gain. This approach eliminates possible power variations caused by fluctuating transmitting power. It is noted that there are only 16 different levels of transmitting power, so maintaining such a table in the memory will be low-cost. Although we built this signal profile manually, this calibration can be easily done using an automatic scanning process.

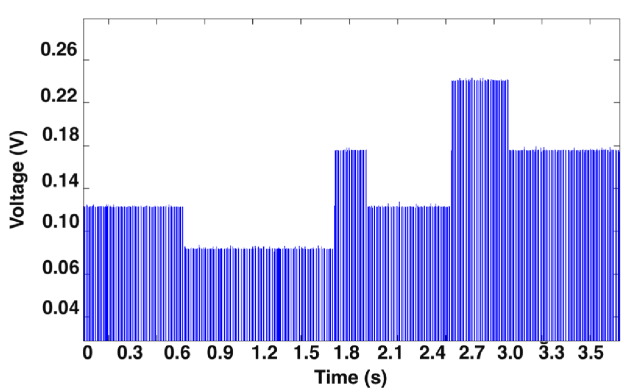

Figure 7. The fluctuation of the transmitting power.

After a GSM pulse is recognized, we calculate the mean of the middle six points and use the mean to represent the received magnitude in the respective antenna. Finally, we interpolate this discrete sequence to construct the continuous wave. Figure 8 shows our approximated continuous signal using actual GSM pulses.

Smoothing and Segmentation

We next identify and segment the hand gesture from this continuous wave. The reconstructed wave is noisy and difficult to be used for gesture detection and therefore, we apply signal processing techniques to extract the gesture segment. Figure 9 demonstrates the process of gesture detection. For each of the four antennas' data, we first apply a Savitzky-Golay (SG) filter with a window size of 301 to smooth the curve (red line, Figure 9 (1)). The SG filter is able to de-noise the signal while maintaining the shape of its original curve. Next, we take the 1st derivative of the smoothed curve to capture the significant signal variations caused by a hand gesture.

Since a gesture may cause signal changes in any of the four receiving antennas, we sum up the absolute value of the derivative curves from all of the channels (Figure 9 (2)). In turn, we are able to find an observable peak, which represents a possible hand gesture. Finally, we take another 1st derivative on the cumulative-derivative curve and use a global threshold to identify a hand gesture (Figure 9 (3)). The threshold was selected to be the equal error rate using all of the data from our user study. The green circles in Figure 9 denote the starting and end point of a hand gesture segment.

Feature Extraction

After recognizing a hand gesture, we extract features from gesture segmentations for classification. The system calculates the midpoint of the start and end of a gesture segment and truncates a window of 4500 data points (*i.e.*, 2.5 sec) centered at the midpoint. We use these data points as the feature set, which describes the shape of the receiving signal. These data points are further demeaned to decrease possible environmental influences, such as putting the

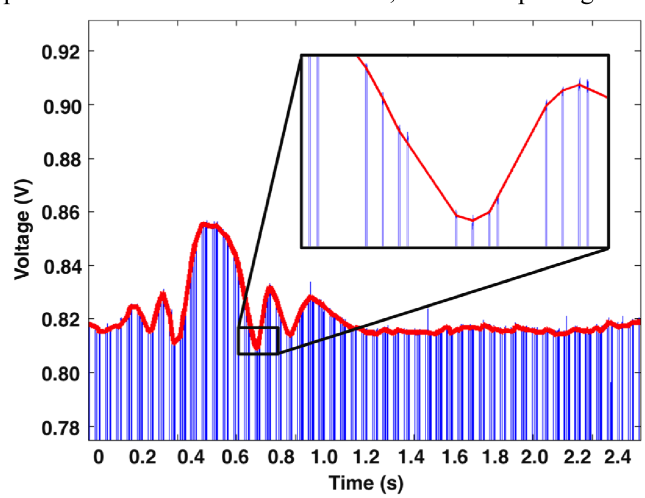

Figure 8. Converting actual GSM bursts (blue stems) to a continuous wave (red curve).

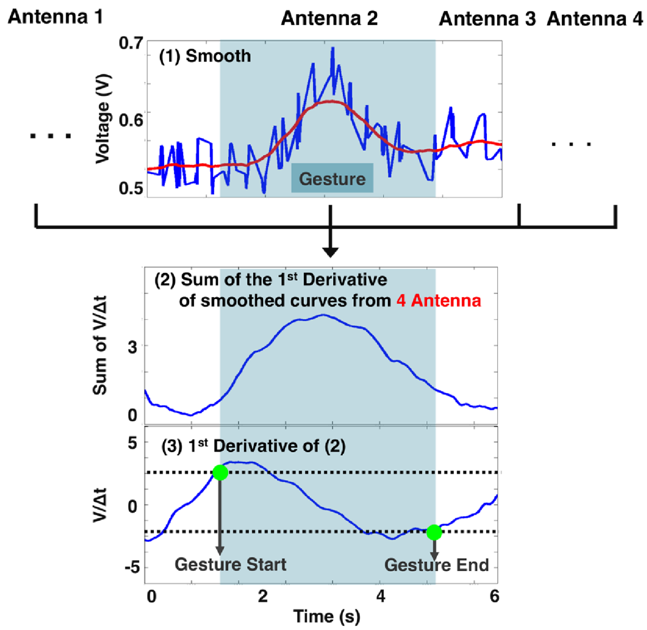

Figure 9. Gesture detection algorithm. In step (2), we sum up the absolute value of 1st derivative of the smoothed curve (**red** line in (1)) from four antennas. This captures significant energy variation in any of our four antennas. In (3), we used a global threshold to identify the start and end of a gesture (**green circles**).

phone in the pocket or on different surfaces. Combining the data points from the four antennas, we generate a feature vector with 18,000 elements. This feature vector represents a unique pattern for different gestures. We further reduce the feature space by down-sampling the feature vector to 80 elements (*i.e.*, 20 elements per channel). In this study, we evaluate 14 hand gestures (Figure 2) performed at a natural speed. These gestures have no high frequency components, and therefore, it is sufficient to describe the signal pattern using only the time domain data.

Gesture Recognition

We next feed the data set into Weka for gesture classification. In Weka, we perform 14-fold cross validation using a Support Vector Machine with PUK kernel. The PUK kernel has the flexibility to vary among Gaussian, Lorentzian, and other shapes, and therefore can be used as a universal kernel in our case. Since the signal pattern of a gesture may vary from person to person (*e.g.*, waving the hand at different speeds), this PUK kernel is able to better adapt to signals that have various shapes.

EVALUATIONS

To evaluate the performance of our system, we setup an experiment in a lab-controlled environment and conducted a 10-user study (3 females). The data collected from all participants are used to empirically decide a global threshold in our gesture recognition algorithm. The analysis shows the optimal threshold of 2.5 V, yielding a perfect event detection rate.

Experimental Setup

Figure 10 shows the experimental setup for data collection. The PCB antenna array was attached on the back of the mobile phone (Samsung Nexus S). To extract the magnitude of transmitting power, we connect a wire to the output pin of the power detector on the TX Front-End module of the Nexus S ("V_{det}" in Figure 10). It is noted that the power information could also be retrieved by tweaking the device driver without modifying any hardware when the driver's source code is accessible. To record the GSM bursts, we utilized a National Instruments USB 6259 ("NI DAQ" in Figure 10) data acquisition unit. The unit takes 16-bit voltage samples at 18 KS/s and streams the 5-channel data to the computer through a USB cable. To get ground truth, we installed proximity sensors on four sides of the phone.

Gesture sets

Figure 2 shows 14 gestures performed in this study, within three categories: **Tap**: top tap (TT), top-right tap (TRT), right tap (RT), bottom-right tap (BRT), bottom tap (BT), bottom-left tap (BLT), left tap (LT) and top-left tap (TLT); **Hover**: hover-to-left (H2L), hover-to-right (H2R), hover-to-bottom (H2B) and hover-to-top (H2T); **Slide**: slide-to-left (S2L) and slide-to-right (S2R). This gesture set covers both gestures being performed above and around the mobile phone. We regard these gestures as the basic building blocks that can be easily combined to form a more complicated vocabulary.

User Study Design

In the beginning of the study, we introduce the gestures to our participants and inform the appropriate speed in performing each gesture. All participants can practice the gestures for three to five times. Although we controlled the gesture speed to avoid unnecessary false detections, participants were still allowed to perform gestures at arbitrarily fast or slow speeds. During data collection, they were required to perform each gesture 13 to 15 times and in total, 1746 gestures were collected. All gestures were performed at a distance within 30cm.

Results

Using a global threshold and the hand gesture detection algorithm described earlier, we obtained 100% detection rate across all of our user data. Using the same global threshold, we only detect 3 false positives over 10 minutes

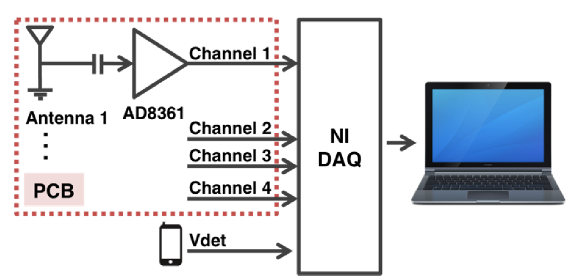

Figure 10. Partial schematic of the PCB board and experimental setup for data collection. National Instruments USB 6259 (NI DAQ) records the GSM bursts and streams the data to a laptop.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Top Tap - 1	66.9	23.0	0.7	0.0	0.0	0.0	0.7	8.6	0.0	0.0	0.0	0.0	0.0	0.0
Top-Right Tap - 2	3.4	81.2	0.9	0.0	0.0	0.0	0.0	13.7	0.0	0.0	0.0	0.0	0.0	0.9
Right Tap - 3	0.8	1.5	86.2	9.2	0.0	0.8	0.8	0.8	0.0	0.0	0.0	0.0	0.0	0.0
Bottom-Right Tap - 4	0.0	0.0	8.0	90.6	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.7
Bottom Tap - 5	2.1	0.0	0.0	4.2	90.9	0.0	0.7	1.4	0.0	0.0	0.0	0.0	0.7	0.0
Bottom-Left Tap - 6	5.8	0.7	0.0	0.0	0.0	87.1	0.0	2.9	1.4	1.4	0.7	0.0	0.0	0.0
Left Tap - 7	0.7	0.0	0.0	0.0	0.0	1.5	90.3	6.7	0.0	0.0	0.0	0.0	0.0	0.7
Top-Left Tap - 8	2.8	1.8	0.0	0.0	0.0	0.9	0.0	94.5	0.0	0.0	0.0	0.0	0.0	0.0
Hover-to-Left - 9	2.2	0.0	0.0	0.0	0.0	2.2	0.0	0.0	91.4	2.9	0.7	0.7	0.0	0.0
Hover-to-Right - 10	0.0	0.0	0.0	1.7	0.0	0.0	0.0	0.0	5.8	88.4	0.0	4.1	0.0	0.0
Hover-to-Bottom - 11	0.0	0.0	0.0	0.7	1.4	0.0	2.1	0.0	2.9	0.7	86.4	5.7	0.0	0.0
Hover-to-Top - 12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.7	0.0	3.5	94.8	0.0	0.0
Slide-to-Left - 13	0.0	0.0	0.0	0.8	0.0	0.8	0.0	0.8	0.0	0.0	0.0	3.9	88.4	5.4
Slide-to-Right - 14	0.8	0.0	0.0	0.0	0.0	0.8	0.0	0.8	1.7	0.0	0.0	2.5	7.6	85.7

Figure 11. Confusion matrix for 14 gestures.

of data where no gestures were performed. It shows that our algorithm is able to reliably detect all hand gestures while maintaining low false alarms.

For gesture classification, we trained a 14-class (*i.e.*, representing the 14 hand gestures) SVM classifier and tested the model by 14-fold cross validation using all user data, as shown in Figure 11. We obtained an average accuracy of 87.2% (chance=7.14%). The results show that our algorithm is able to robustly differentiate in-air hand gestures using the actual GSM signal. In addition, although we randomly chose the folding number of 14, we did not see significant difference in accuracy when using other folding numbers (86.77% for 10-fold and 85.69% for 5-fold).

From Figure 11, it can be seen that there is confusion between the top-right tapping (TRT), top tapping (TT) and top-left tapping (TLT). These gestures are relatively further from the transmitting antenna and therefore have weaker signals. Also, our simulation results (see Figure 5) show that the top-right corner of PCB has a null between lobes, and therefore there is a bigger confusion between top-right (TR) and top tapping (TT). Furthermore, the confusion between bottom-right tapping (BRT) and right tapping (RT) resulted from the fact that participants tend to use their right hand to perform BR tapping and R tapping. In such case, the whole arm causes much stronger fluctuations than that caused by the palm. Such user behaviors resulted in similar signal patters of these two gestures and cause confusion. As shown earlier, we are able to classify all 14 gestures with 87.2% accuracy; however, using a reduced gesture set of 13 gestures (i.e., removing top tap) yields 90% (chance=7.7%).

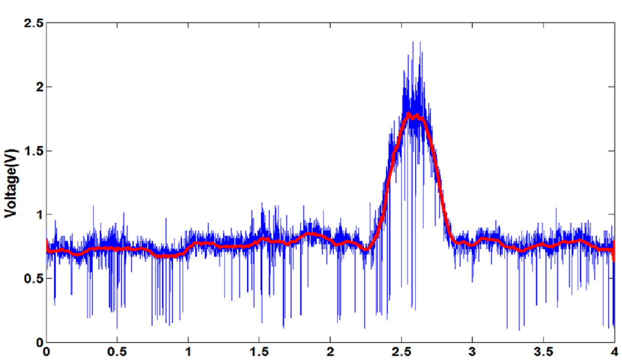

Figure 12. Fluctuation of LTE signals (**blue**) when user performs a bottom-tap gesture. The similarity in observed signals shows that our algorithm can easily adapt to future network standards.

POTENTIAL INTERACTION AND APPLICATIONS

Incoming Call Management

Our system can be beneficial when user receives phone call in a non-appropriate situation (*e.g.,* during a meeting). Instead of taking the phone out of her pocket, she can just use hand gesture to respond to the incoming call. We envision three gestures to control the three modes: enable silent mode with a downward gesture towards the phone, send predefined text with a right swipe gesture, and decline incoming calls by tapping on the phone.

Phone Navigation from Distance

In some situations, it becomes difficult for the user to touch the screen in order to navigate through the phone. To accommodate user inputs in such cases, a user can perform an upward gesture to enable scrolling while following a recipe. In addition, a user can perform in-air gestures to control their music listening experience in a variety of touch unfriendly situations. Besides accommodating touch unfriendly situation, this navigation system can also accommodate user's need in different situational impaired scenarios such as driving. The user can easily switch between map, music, and messaging app with simple left and right swipe gestures without having to touch specific location of her phone ever. Once selected, any app specific feature (e.g., map zoom in/out, turn GPS on/off) can then be navigated using inward or outward swipe gestures. These eyes-free gestures could reduce users' secondary task burden and allow them to focus more on primary driving task.

DISCUSSION AND FUTURE WORK

Our current prototype uses reflected GSM pulses to sample the gesture induced variation in propagation channels and hence, the rate of GSM pulses may limit the detectable hand gestures. As described earlier, the hand gesture usually occurs on the order of tens of Hz, which is smaller than the rate of GSM pulses (i.e., 80 Hz). However, for faster gestures like quick flicking, the high frequency components of the gesture may not be reconstructed using GSM pulses.

In our prototype, we take advantage of using GSM bursts when the user is making (or during) a phone call. It is possible to tweak the built-in GSM module to activate the GSM communication, generating a low gain-level, always-existing signal for sensing gestures. Furthermore, our approach, unlike prior Doppler-effect-based approaches, does not require high-speed ADC and FFT computation, and thus reduces the power consumption.

To explore the compatibility of our system with more modern wireless standards, we use the same experimental setup as shown in Figure 10, to capture LTE data uplink signal on an Apple iPhone 5s (AT&T version). Figure 12 shows the signal variation caused by a bottom-tap gesture. As shown in Fig 12, the duty cycle of LTE signal is much higher than that of GSM and therefore, the first step in Figure 6 is unnecessary. In addition, as the smoothed signal in Fig 12 (red line) is similar to the continuous signal in Fig 8, we believe that our methods could be easily adapted to future 4G standards such as LTE, which uses OFDMA / SC-FDMA in the uplink. We will further explore this in the future.

We also consider a couple of alternatives to improve our system. As shown in Fig 5, the right antenna has obvious null at the top-right corner, which reduces the accuracy of gesture classification. To enhance the performance, we could modify of the design of the antenna array to change the radiation patterns. Also, instead of using AD8361, it is possible to use different RF power detectors such as Analog Device's ADL5903 to retrieve the gestures from CDMA, WCDMA, TD-SCDMA and LTE signal.

CONCLUSION

In this work, we present SideSwipe, a system that enables in-air gestures for mobile devices using unmodified GSM signals. Our novel algorithm is able to interpolate the reflected GSM pulses into a continuous waveform and use it for gesture detection. In our prototype, we designed and built a receiver with four antenna elements. Since each antenna has a unique propagation path, hand movements around the phone perturb the GSM signal and these propagation paths. By combing the received signals from the four antennas, our system is able to build a unique pattern for various gestures. In our evaluation, we present 100% gesture detection and are able to recognize 14 gestures with 87.2% accuracy. Lastly, we show that SideSwipe can be easily adapted to future standards such as the LTE network.

REFERENCES

1. Adib, F. and Katabi, D., See Through Walls with Wi-Fi!. In *Proc SIGCOMM 2013*
2. Butler, A., Izadi, S., and Hodges, S. SideSight: multi-touch interaction around small devices., In *Proc UIST 2013*, pp. 201-204.
3. Chen, K., Lyons, K., White, S., and Patel, S. uTrack: 3D input using two magnetic sensors. In *Proc UIST 2013*, pp. 237–244.
4. Chetty, K., Smith, G., Guo, H., and Woodbridge, K. Target detection in high clutter using passive bistatic WiFi radar. *2009 IEEE Radar Conference*, pp. 1–5.
5. Chintalapudi, K., Padmanabha Iyer, A., and Padmanabhan, V.N. Indoor localization without the pain. In *Proc MobiCom 2010*, 173.
6. Cohn, G., Morris, D., Patel, S., and Tan, D. Your noise is my command: sensing gestures using the body as an antenna. In *Proc CHI 2011*, pp. 791–800.
7. Cohn, G., Morris, D., Patel, S., and Tan, D. Humantenna: using the body as an antenna for real-time whole-body interaction, In *Proc CHI 2012*, pp. 1901-1910
8. Gupta, S., Morris, D., Patel, S., and Tan, D. Soundwave: using the doppler effect to sense gestures. In *Proc CHI 2012*, pp. 1911-1914.
9. Harrison, C. and Hudson, S. Scratch input: creating large, inexpensive, unpowered and mobile finger input surfaces, . In *Proc UIST 2008*, pp. 205-208
10. Harrison, C., Tan, D., and Morris, D. Skinput: appropriating the body as an input surface, In *Proc CHI 2010*,, pp. 453-462
11. Hudson, S.E. Abracadabra : Wireless , High-Precision , and Unpowered Finger Input for Very Small Mobile Devices, In *Proc UIST 2009*, pp. 121-124.
12. Kellogg, B., Talla, V., and Gollakota, S. Bringing Gesture Recognition To All Devices. In *Proc NSDI 2014*,
13. Kim, D., Hilliges, O., Izadi, S., and Butler, A. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor, In *Proc UIST 2012*, pp. 167-172
14. Marvasti, F. *Nonuniform sampling: theory and practice*. Springer, 2001.
15. Minka, T. and Choudhury, R.R. Spot, Precise Localization using PHY Layer Information, In *Proc HotNets-X 2012*
16. Pu, Q., Gupta, S., Gollakota, S., and Patel, S. Whole-Home Gesture Recognition Using Wireless Signals, In *Proc MobiCom 2013*, pp. 27-38.
17. Ring, M.F., Ashbrook, D., Baudisch, P., and White, S. Nenya : Subtle and Eyes-Free Mobile Input with a magnetically-tracked finger ring, In *Proc CHI 2011*, pp. 2043–2046.
18. Shotton, J., Fitzgibbon, A., Cook, M., et al. Real-time human pose recognition in parts from single depth images. In *Proc CVPR 2011*, pp. 297–1304.
19. Starner, T., Auxier, J., Ashbrook, D., and Gandy, M. The gesture pendant: a self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *Proc ISWC 2000*, pp. 87–94.
20. Wigdor, D., Forlines, C., Baudisch, P., Barnwell, J., and Shen, C. LucidTouch : A See-Through Mobile Device, In *Proc UIST 2007*, pp. 269-278.
21. Xiong, J. and Jamieson, K. Towards Fine-Grained Radio-Based Indoor Location Categories and Subject Descriptors, In *Proc HotMobile 2012*.