Your Table Can Be an Input Panel: Acoustic-based Device-Free Interaction Recognition

MINGSHI CHEN, University of Science and Technology of China, China
PANLONG YANG*, University of Science and Technology of China, China
JIE XIONG, University of Massachusetts Amherst, USA
MAOTIAN ZHANG, University of Science and Technology of China, China
YOUNGKI LEE, Seoul National University, South Korea
CHAOCAN XIANG, Chongqing University, China
CHANG TIAN, Army Engineering University of PLA, China

This paper explores the possibility of extending the input and interactions beyond the small screen of the mobile device onto ad hoc adjacent surfaces, *e.g.*, a wooden tabletop with acoustic signals. While the existing finger tracking approaches employ the active acoustic signal with a fixed frequency, our proposed system *Ipanel* employs the acoustic signals generated by sliding of fingers on the table for tracking. Different from active signal tracking, the frequency of the finger-table generated acoustic signals keeps changing, making accurate tracking much more challenging than the traditional approaches with fix frequency signal from the speaker. Unique features are extracted by exploiting the spatio-temporal and frequency domain properties of the generated acoustic signals. The features are transformed into images and then we employ the convolutional neural network (CNN) to recognize the finger movement on the table. *Ipanel* is able to support not only commonly used gesture (click, flip, scroll, zoom, *etc.*) recognition, but also handwriting (10 numbers and 26 alphabets) recognition at high accuracies. We implement *Ipanel* on smartphones, and conduct extensive real environment experiments to evaluate its performance. The results validate the robustness of *Ipanel*, and show that it maintains high accuracies across different users with varying input behaviours (*e.g.*, input strength, speed and region). Further, *Ipanel*'s performance is robust against different levels of ambient noise and varying surface materials.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing systems and tools**.

Additional Key Words and Phrases: interaction recognition, acoustic sensing, mobile application, CNN

ACM Reference Format:
Mingshi Chen, Panlong Yang, Jie Xiong, Maotian Zhang, Youngki Lee, Chaocan Xiang, and Chang Tian. 2019. Your Table Can Be an Input Panel: Acoustic-based Device-Free Interaction Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1, Article 3 (March 2019), 21 pages. https://doi.org/10.1145/3314390

*Corresponding author.

Authors' addresses: Mingshi Chen, University of Science and Technology of China, Hefei, China, cms603421@gmail.com; Panlong Yang, University of Science and Technology of China, Hefei, China, panlongyang@gmail.com; Jie Xiong, University of Massachusetts Amherst, Amherst, MA, USA, jxiong@cs.umass.edu; Maotian Zhang, University of Science and Technology of China, Hefei, China, maotianzhang@gmail.com; Youngki Lee, Seoul National University, Seoul, South Korea; Chaocan Xiang, Chongqing University, Chongqing, China; Chang Tian, Army Engineering University of PLA, Nanjing, China.

2474-9567/2019/3-ART3 $15.00
https://doi.org/10.1145/3314390

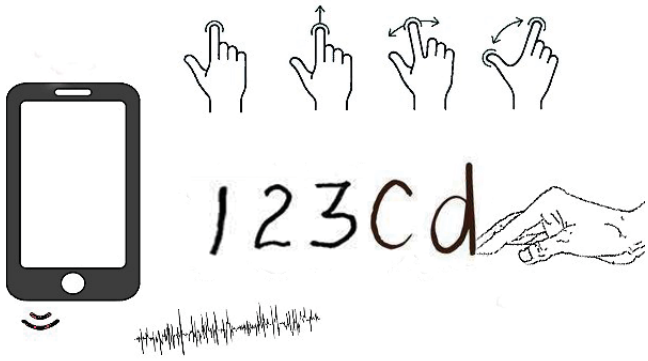


Fig. 1. Typical gestures and handwriting.

1 INTRODUCTION

Smart devices are prevailing in various forms from smartphones to smartwatches and wristbands. A core challenge for the interaction with smartwatches and wristband is the small screen compromised for portability and compactness. This issue raises a urgent need for new interaction medium beyond the small built in screen of the smart devices.

In this paper, we explore the potential of utilizing *finger input on a nearby surface (e.g., sliding or writing on a desk surface) as a new interaction modality*. A user can make various finger operations on the nearby surface to instruct the devices – for example, drawing a line from the bottom to up can indicate scrolling up a web page. This can significantly expand the area of interaction, from a small screen of the device to a larger nearby surface, making the interaction more convenient and intuitional. In addition, such capability can be utilized to enable multi-device interactions on a surface [21].

We propose *Ipanel*[1], a novel system that enables intuitive *surface-drawn interfaces* using passive acoustic sensing. For example, a desk surface can become a touch pad with *Ipanel* to achieve gestural interactions. The desk surface can also be enabled as a writing pad for user hand input. Our key observation is that the finger sliding on a surface generates acoustic signals and what is more interesting, different finger movements generate unique acoustic signatures. *Ipanel* captures the acoustic signals when fingers slide on the surface and recognizes the corresponding finger movement/gesture through a suite of proposed techniques. The main advantages of *Ipanel* are threefold: (1) *it only uses the microphone already embedded in COTS devices;* (2) *users do not need to wear or hold a device but can freely move their hands and fingers;* and (3) *different from the active tracking system, Ipanel does not require the device's speaker to continually emit acoustics signals for tracking purposes.*

There has been a rich body of prior research to enable gesture-based interfaces. However, most existing works require extra dedicated hardware or customized devices [1, 2, 24, 28, 44]. Some require users to hold a device in hand [26, 49, 58] or wear sensors [14–16, 22, 40, 53], which are burdensome for real-life applications. *Ipanel* differs from the above systems in that it does not need the user to hold or wear any device. Recently, active acoustic sensing-based techniques such as AAMouse[58] and LLAP [55] are proposed. The device continuously emits out ultrasound signals at a fixed frequency and employs the reflected signal from the finger to track the finger movement. Though the ultrasound can not be heard by most people, it is reported [27] that long-term ultrasound exposure can make people feel uncomfortable or even sick not to mention that ultrasound can still

[1]We create the word "Ipanel" which is the combination of "Input" and "Panel".

be heard by pets which is acoustic pollution to them. *Ipanel* does not need the speaker to emit any acoustic signal but relying on the weak signal generated by subtle finger sliding on the table surface for finger movement recognition. With our proposed methods, *Ipanel* could sense not just simple finger gestures such as click and scroll but also support more complex numbers and alphabets.

To realize *Ipanel* in real time on a resource-constrained mobile device, several challenges need to be tackled:

- *The acoustic signals generated by finger sliding are very weak.* For instance, the acoustic signal strength is 67dB[2] when sliding a flip gesture on a wooden tabletop with a soundmeter placed 10 cm away, while the ambient background sound volume is around 48dB in a typical office room, and 62dB in a cafe.
- *The acoustic signals generated by finger moving on a surface do not have a fixed frequency.* This makes signal analysis in frequency domain much more challenging compared to prior techniques using active sound sources emitting inaudible acoustic signals at a fixed frequency [38, 58]. According to our experiments, frequency of the acoustic signal due to a sliding operation can spread from $1Hz$ to $15kHz$. And thus, finger-generated acoustic signals cannot be cross-correlated as those fixed-frequency active acoustic signals.
- *The characteristic of an operational acoustic signal needs to be effectively captured.* To ensure accuracy recognition of various operation interactions, the feature of operational signal do matter – For example, the number '1' and alphabet 'l' are extremely similar in writing. Therefor, using commonly used frequency domain features such as *Amplitude Spectrum Density (ASD)* or its extensions is far from feasible.

To deal with the former two challenges, we first propose wavelet analysis and bandpass filtering to fully analyze information from multiple dimensions including spatial, time and frequency domains of captured acoustic signals for input detection. As for the third challenge, we combine sound spectrogram with image identification of convolutional neural network [23] to increase recognition accuracy.

We implement *Ipanel* on Android-based smartphones and we take NUBIA Z9 mini with Android 5.0 to show the prototype demo. We conduct extensive real-world experiments to evaluate its accuracy and robustness. *Ipanel* is able to recognize seven commonly seen gestures (*i.e.*, click, flip left/right, scroll up/down, and zoom in/out) and basic handwriting including numbers and lowercase alphabets in printed form as shown in Fig.1 at a 91.3% accuracy on average, and maintains the high accuracies across different devices, surface materials, users, and ambient noises. The main contributions of the paper are summarized as follows:

- We propose *Ipanel*, a novel passive acoustic sensing system to extend the human-device interaction beyond the small screen of a smart device to a large adjacent surface. Compared with existing approaches employing active acoustic signals with a fixed frequency, the signals generated by finger movement is weaker and the frequency varies significantly, making the problem in this work much more challenging.
- We develop a suit of techniques to realize *Ipanel*. We employ features from multiple dimensions including spatial, time and frequency domains of the received acoustic signals to significantly increase the identification accuracy.
- We design, implement and evaluate *Ipanel* through comprehensive experiments in various real-world environments. For our experiments, the participants were asked to perform handwriting in printed form with no ligature near[3] to the phone. The results show that *Ipanel* achieves consistently high accuracies across different users with varying input behaviours (*e.g.*, input region and strength).

The rest of this paper is organized as follows. Section 2 presents the overview of *Ipanel* system. In Section 3, we describe the detailed design and algorithms. We present the implementation and experimental results in Section 4.

[2]The dB (decibel) here is the sound intensity level.

[3]From our experiments, we observed a performance degradation when the input location is more than 20 cm away from the phone. So we asked the participants to write near to the phone and we noticed that most participants actually input at the region 5-10cm away from the phone which is a conformable zone for them to input.

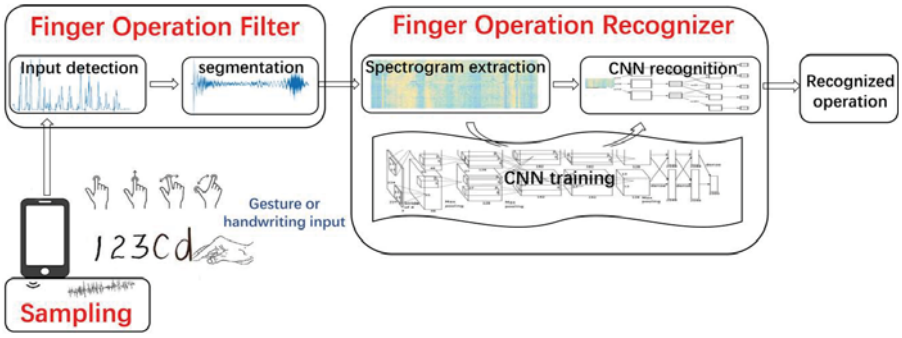


Fig. 2. *Ipanel* architecture.

We further discuss our work in Section 5, and introduce related work in Section 6. Finally, we conclude our work in Section 7.

2 SYSTEM OVERVIEW

Fig. 2 illustrates the architecture and work flow of *Ipanel*[4]. We implement *Ipanel* on a Android smartphone (NUBIA Z9mini *etc.*), and its embedded microphone continuously captures acoustic signals generated by different finger operations. After sampling, the captured signals are processed by the following two key components: *Finger operation filter* and *Finger operation recognizer*.

- *Finger operation filter* filters out signals not related to finger operations. Also, upon the detection of operations, our system segments the signals so that each segment corresponds to a single input. (See Section 3.2 for details)
- *Finger operation recognizer* recognizes the performed operations. It first extracts an extended set of time-frequency domain feature (acoustic spectrum feature), and matches the features with pre-collected training dataset using a computationally-efficient algorithm.

3 SYSTEM DESIGN AND ALGORITHMS

3.1 Data Collection

We employ the embedded microphone of the smartphone to continuously sample acoustic signals received. We call the *AudioTract()* and *AudioRecord()* APIs to capture the audio and record the audio respectively. We employ the command statements "*android.permission.RECORD_AUDIO*" and "*android.permission.WRITE_EXTERNAL_STORAGE*" to obtain microphone and storage permissions. We use the default sampling rate of the smartphone, *i.e.* 44 KHz, to collect the audio signal, and then retrieve the amplitude readings.

3.2 Finger Operation Filter

Ipanel's operation filter filters out signals not associated with an operation and passes the signals caused by operations to the next stage. The basic idea is to detect the energy peaks of received acoustic signals and surpass surrounding noises. Note that it is not possible for us to apply the commonly seen conjugate correlation method [57] with the known reference signal as we do not employ a dedicated known reference signal and the generated acoustic signal keeps changing in terms of amplitude and frequency.

[4]We release our source code at GitHub: https://github.com/chenmingshi/acoustic-sensing.

3.2.1 Input Detection. The amount of energy depends on the speed and strength of finger movements as well as the physical properties of the surfaces [8]. To understand the characteristics of the energy feature, we conducted benchmark experiments performing several finger operations (*e.g.*, flip) at a moderate speed (about $12.5cm/s$ for gesture or handwriting) using a NUBIA Z9mini smartphone. As shown in Fig. 3, the acoustic signals produced by a finger operation show a clear energy burst in the beginning. This peak occurs when the finger hits the surface to start its movement. We repeated the experiments with different speeds, varying strengths on table surfaces of different materials. We found that this energy burst always exists. *Ipanel* thus leverages such a unique energy burst to detect the start of an operation.

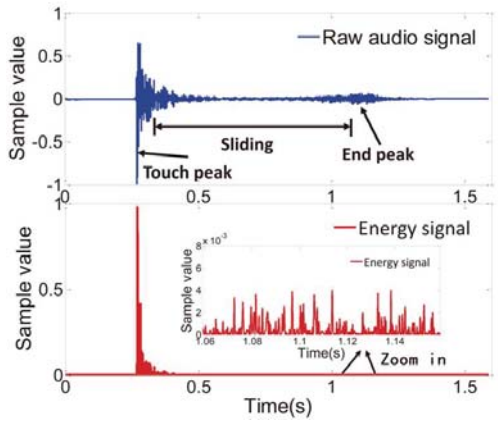


Fig. 3. Example of an operation's acoustic signal and its energy.

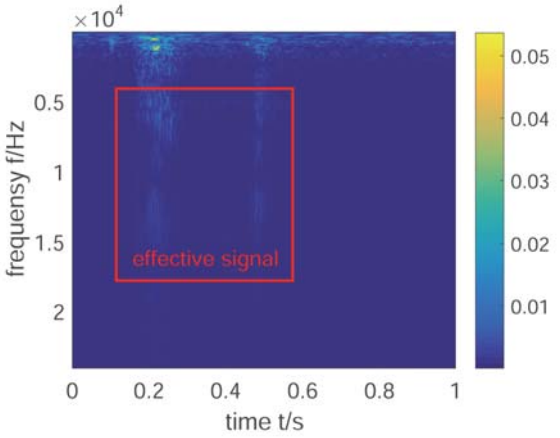


Fig. 4. Wavelet analysis to acoustic signal of a finger operation.

Formally, the energy of an acoustic signal $x(t)$ can be represented by $E(t) = kx^2(t)$, where k is a constant. We show the energy level of the acoustic signal in the lower part of Fig. 3. To find the *Touch peak*, we first do a wavelet analysis to acoustic signal, which obtains time-frequency characteristic of each operation. From Fig. 4, we observe that acoustic signal of the finger operation can spread from $1Hz$ to $15kHz$. While in low frequency band (below $5kHz$), it contains a lot of useless noise, such as human speech *etc.*. Therefore we conduct a bandpass filter (from $5kHz$ to $15kHz$) to suppress the ambient noises. $A(t)$ denotes the moving average energy level with a window size W_a:

$$A(t) = \sum_{n=t}^{t+W_a} E(n) \tag{1}$$

The acoustic signal generated by finger movement, in a short time window (10-30 ms), can be regarded as a quasi-steady state. We set the window size W_a to be $20ms$, *i.e.*, $44.1kHz \times 20ms \approx 884$ sample points with a $44.1kHz$ default sampling rate of the microphone on most Android smartphones. Inspired by the approach presented in [61], we detect the *Touch peak* (denoted by t_{touch}) as below:

$$\begin{aligned} t_{touch} &= \arg\max_t A(t) \\ s.t. &: A(t) \geq A(i) \text{ for } t - 50ms \leq i \leq t + 550ms \\ &\text{and } A(t) \geq A_\epsilon \end{aligned} \tag{2}$$

where i is the sample index and the search range is empirically set to be 600ms which is typically the maximum period of a stroke [5], and A_ϵ is an empirical threshold set as 0.02 to determine the starting point of a *Touch peak*. Note that the threshold is empirically selected as 0.02 to achieve the best performance for *Touch peak* detection in a typical office environment. The threshold is dependent on the environments and will be updated accordingly due to the change of environments [34, 61].

ALGORITHM 1: Algorithm for segmentation

Input: The acoustic signal $x_(t)$, the width of moving average window W_a, the width of signal piece W_g and three temporal threshold T_1, T_2, T_3.

Output: The segment of each operation $\overrightarrow{t_g}$.

```
1  for t in signal x(t) do
2      A(t) = sum_{n=t}^{t+W_a} E(n);
3      if A(t) > A_ε then
4          for i > t and i < t + T_1 do
5              if A(t) < A(i) then
6                  break;
7              end
8              i++;
9          end
10         if i == t + W_g then
11             t is the time of a Peak;
12             t_g = (t − 50ms, t + 950ms), record t_g;
13             t = t + W_g;
14         end
15         t++;
16     end
17 end
```

3.2.2 Segmentation. Our system can recognize both commonly seen control gestures and handwritings including numbers and letters. For those letters with two strokes such as 'i' and 'x', there exist two *touch peaks* because they are consisted of two strokes. *Ipanel* identifies a segment of acoustic signal corresponding to a single input. Algorithm 1 shows the details of the segmentation process. We set the operation width of each piece W_g as 1s[6], containing 44100 samples. The moving average window of W_a is used to calculate the accumulated energy (line 2). The *Touch peak* can be detected by Equation 2 (line 3–9). After detecting the *Touch peak*, for each operation, we take from 50ms before t_{touch} to 950ms after t_{touch} to form a segment corresponding to an operation (line 10–14). The acoutic signals in buffer are processed window by window with this algorithm.

3.3 Finger Operation Recognizer

We expect our system to recognize not only the user's simple gestures (*e.g.*, flip and zoom), but also the user's handwriting input such as numbers and alphabets. Therefore, we need unique feature to describe each individual gesture/number/alphabet operated by the user. The classification method is also important as the number of

[5]Similar to the stroke of Chinese characters, we define each desktop touch of user's operation as a stroke. For example, '1' and 'a' *etc.* contain one stroke, while '4', '5' and 'x' *etc.* contain two strokes.

[6]Through benchmark experiments, we observe that one operation or one letter/number writing usually can be completed within 1s.

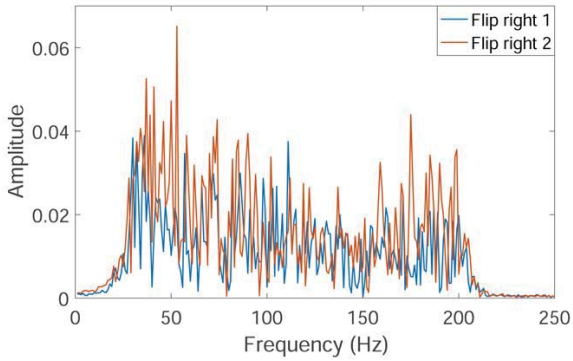


Fig. 5. ASD of Flip Right.

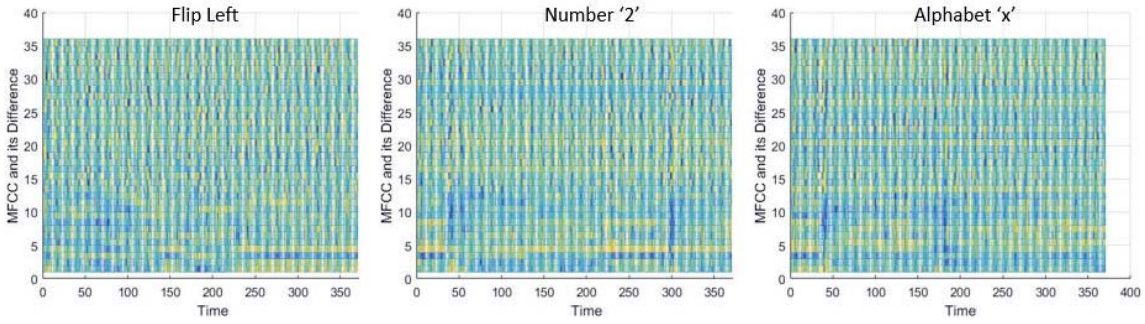


Fig. 6. MFCC of different operations.

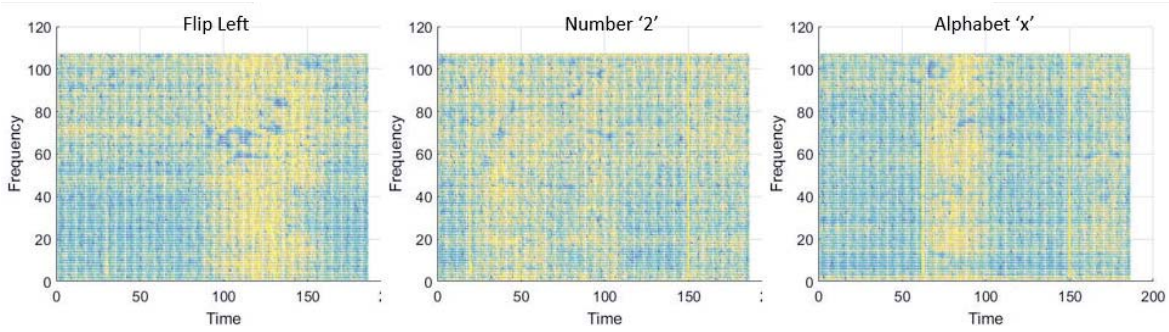


Fig. 7. Spectrograms of different operations.

classification category has reached 43 (10 numerical digits, 26 alphabets and 7 gestures). We want the recognition to be accurate and at the same time the latency is small so we can achieve realtime recognition.

Table 1. Accuracy of different combinations.

KNN and ASD	SVM and ASD	SVM and MFCC	CNN and MFCC	CNN and Spectrogram
14.57%	46.53%	62.16%	82.17%	92.25%

3.3.1 Feature Extraction. On one hand, we first study the frequency domain profiles of the acoustic signal which is also called amplitude spectrum density (ASD). ASD is the square root of the PSD (power spectral density). PSD represents the power distribution in frequency domain for a continuous signal. ASD of a discrete acoustic signal $x(t)$ is given by[7] $FFT\big(x(t)\big)$. However, we observed that ASD is unstable for the same operation among different samples. As shown in Fig.5, the frequency distribution (ASD) of the same gesture varies. So ASD can not be directly employed as the unique feature for our recognition purposes. On the other hand, time domain MFCC (Mel-Frequency Cepstral Coefficients) information is widely used in voice recognition. It mimics the human ear structure for feature extraction, which is more sensitive to the low-frequency (below 5kHz) signal. G.Luo *et al.*. [20] combine MFCC with KNN (K-Nearest Neighbors) and achieve 92% recognition accuracy for simple gesture. Inspired by previous studies, we want to exploit features in both frequency domain and time domain for recognition. To obtain credible features in frequency and time domain, we first segment each effective signal into fragments with a short time window (20 ms). MFCC is time domain feature which consists of the cepstral coefficients of each fragment, its first-order difference and second-order difference. In our experiment, the differences of MFCC between different operations are tiny as shown in Fig.6. This means MFCC is still not better enough to be employed to distinguish the 43 types of finger operations (10 numerical digits, 26 alphabets and 7 gestures). Another feature spectrogram is a visual representation of the spectrum of frequencies as they vary with time. It is used extensively in the fields of music, sonar, radar, seismology and speech processing. In Fig.7, we can see that spectrogram shows great discriminability between different finger operations.

Thus, we perform spectrum analysis on the signal and save it as a spectrogram. Our algorithm sets the spectrogram size to 64*64 which is a size most graphics cards can easily handle. To extract the full features, we set the size of time window, overlap window and number of FFT points as 1024, 512, and 1024 respectively.

3.3.2 Operation Recognition. There are many popular machine learning algorithms for classification purposes, such as KNN (K-Nearest Neighbours), SVM (Support Vector Machine) and CNN (Convolutional Neural Network). We tried KNN, SVM, CNN and CNN achieved the best performance on the spectrogram feature. From previous research in handwriting recognition [23, 31, 43], the performance of CNN is shown to be robust against the writing speed and writing strength. Similarly, in our research, the acoustic signals of the same operation generated by different handwriting speeds and strengths result in difference in the generated acoustics signals. Furthermore, the two dimensional feature can be viewed as an image and CNN is well known to work well in image identification. As shown in Table 1, we can see that CNN with spectrogram works best, outperforming the other combinations. We thus adopt CNN in this work.

We did try using the raw signal for classification. The results show that the performance with raw signal is poor and the performance is severely affected by factors such as the sliding duration, strength and variation of different people. We thus propose to employ the sound spectrogram, which extracts features both in time and frequency domains for classification. When the feature matrix is transformed into an image, we find that the features of the same operation are consistent and stable. To verify this, we construct a sample set that contains the raw signals of all operations (10 numerical digits, 26 alphabets and 7 gestures). We evaluate the recognition performance of different combinations (different machine learning techniques + different feature types) on this

[7]We use the first half of FFT results, since the acoustic signals are real numbers and the ASD is symmetric with respect to the half-frequency.

Table 2. Structure flow of our hybrid CNN

layer	name	configuration
1	Image Input	64x64x3 images with 'zerocenter' normalization
	Convolution	64 11x11 convolutions with stride [1 1] and padding [2 2]
	ReLU	ReLU
	Normalization	cross channel normalization with 5 channels per element
	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
2	Convolution	128 5x5 convolutions with stride [1 1] and padding [2 2]
	ReLU	ReLU
	Normalization	cross channel normalization with 5 channels per element
3	Convolution	256 3x3 convolutions with stride [1 1] and padding [2 2]
	ReLU	ReLU
	Max Pooling	2x2 max pooling with stride [2 2] and padding [0 0]
4	Dropout	50%dropout
	Fully Connected	256 fully connected layer
	ReLU	ReLU
5	Fully Connected	10 fully connected layer
	ReLU	ReLU
	Softmax	softmax
	Classification Output	cross-entropy

sample set, and the results are shown in Table 1. Note that ASD only applies the FFT operation on the raw signal and is considered as raw signal.

The well-known CNN structure AlexNet with 8 layers achieves high accuracies in the classification of 1000 categories of images. Our target pool with 43 operations (10 numerical digits, 26 alphabets and 7 gestures) has a much smaller size, so simpler structure with 5 layers of CNN proposed in Table 2 is enough to achieve good performance. During the design of CNN structure, we tried different layers (6, 7 and etc) of CNN structure. From the result, we find that increasing the number of layers more than 5 only slightly improves the performance. With 7 layers, the accuracy is only increased by 2.3% but the computational load is much higher. So for our system, we adopt 5-layer CNN, which achieves a good balance between accuracy and computational load. In our implementation, the first three layers are used to extract the characteristics, and the last two layers are used for classification. The parameter settings of the first three layers are similar to that in reference [23].The sizes of three convolutional and two max pooling kernels are 11×11, 5×5, 3×3, 3×3 and 2×2 respectively. The parameters of the last two layers are set based on the parameters of the first two layers and the number of output gesture types. For our experiment, we divide the dataset into two parts in the ratio of 8:2, where the former is the training set and the latter is the test set.

4 EVALUATION

We evaluate the performance of *Ipanel* in terms of recognition accuracy and robustness, with varying impact factors such as the input strength, the input region, the surface materials, the type of mobile devices *etc.*. For the experiments, we recruit a total of 57 student volunteers (25 females and 32 males) from our university aged

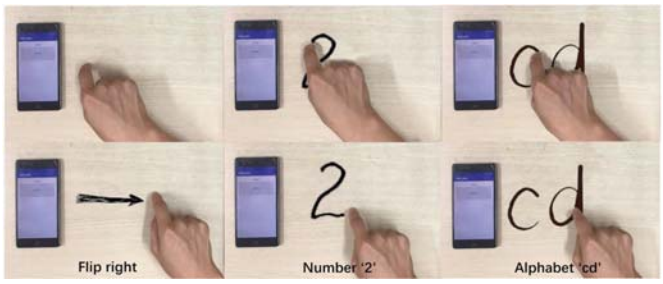


Fig. 8. Experiment setup.

between 20 to 27 to test the performance of *Ipanel*. The experiments are conducted in different environments, including an office room, a library and a cafe, to evaluate the robustness and feasibility of *Ipanel*.

4.1 Implementation and Experimental Setting

We implement *Ipanel* system on several smartphones of different brands all runing Android 5.0. We use the bottom microphone on the smartphone for *Ipanel*. The audio sampling frequency is set as 44.1kHz by default. The samples are put into an acoustic buffer and then the operation detection and recognition algorithm are applied on these samples. The prototype system is implemented in matlab which is capable of working off-line for accuracy evaluation.

In the experiments, we use several different smartphones to evaluate *Ipanel*, and the mobile devices are placed on surfaces made of different materials. Participants are provided a brief instructions of *Ipanel*. As the default setting, a NUBIA Z9mini smartphone is placed on top of a coated-wood table in an office room, and the user inputs gestures or operations at the right side of the smartphone, as shown in Fig. 8. The microphone at the bottom of the smartphone captures and buffers the acoustic signals.

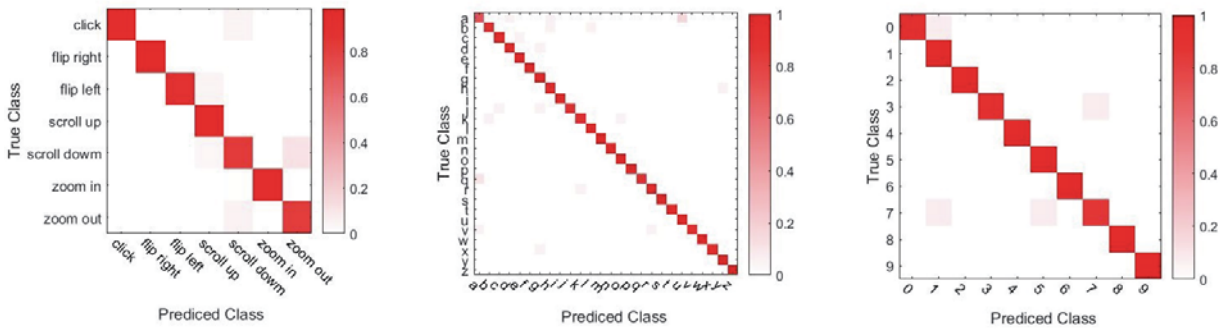
We arrange participants to conduct experiment in different environments and then collect data to evaluate our system. At first, the participant practices operations on the surface for one minute. During the training period, the *Ipanel* App demonstrates the operations to help the participants. The experimenter records the operation inputs of each participant as the ground truth. We perform off-line analysis with collected data using *Ipanel* prototype. We apply several restrictions when the participants perform the operations.

- The participants are asked to perform the writing operation at the region around 10cm away from the smartphone. Currently, we require the user to perform the operations not too far away from the smartphone because the signals generated become too weak to be used for recognition. We did notice that when the distance was larger than 20cm, the recognition accuracy decreased.
- The participants are asked not to ligature (write continuously without a stop between letters). Different people have different ligature pattern and thus it is very difficult to segment each letter and run recognition.
- For the essay writing, we asked the users to write all the letters in lower case and in printed form. The capital letters are currently not supported.

We do want to point out that we did not apply restriction on the size, strength and speed of the writing. The participants can write naturally in a comfortable manner they like. When we evaluate the effect of factors such as writing strength and speed, the participants are then asked to vary their strength and speed.

4.2 Accuracy

In this subsection, we conduct experiments to comprehensively validate the performance of *Ipanel*.



(a) Gesture (b) Alphabet (c) Number

Fig. 9. Confusion matrix of different operations

Table 3. Gesture input detection rates in different environments

Environment	Library	Office	Cafe
Noise level	40.2dB	47.5dB	61.5dB
P_{mis}	0.29%	0.57%	0.86%
P_{fls}	0.0%	0.29%	1.15%

4.2.1 Input Detection Rates in Different Environments. At first, we evaluate the input detection in different environments, *i.e.*, a library (12 users), an office room (25 users), and a cafe (10 users). At each environment, the participants repeat each operation 50 times. The collected acoustic signals are fed into our system for signal processing and recognition.

The mis-detection (P_{mis}, false negative) and false-alarm (P_{fls}, false positive) rates are used to measure the operation detection accuracy. Table 3 shows the experimental results. In all three scenarios, both P_{mis} and P_{fls} are fairly low (less than 1%). Even in a noisy cafe, our system can detect about 95% operations. When the noise level is lower than 70dB, the operation detection rate of *Ipanel* is very high. If there are bursty noises, *e.g.*, in cafe, the detection error rates slightly increase. The main intuition for this high rate is that although the noise level is relatively high, the operation is performed around the device so the signal level is still high enough to be detected. Meanwhile, we filter out the most noise frequency band (below 5 kHz) by bandpass filter, such as human speech is commonly distributed in between 200 Hz and 5 kHz, except the frequency of some sopranos can reach 1.1 kHz.

4.2.2 Accuracy of Different Operations. We let the participants perform operation recognition experiments in a typical office room with an ambient noise level of 47.5dB. For each operation, the participants repeat 50 times, which construct a sample set with 122550 samples in total (57 participants and 43 kinds of operations). In calculation of accuracy, to each operation, we calculate the average accuracy among different users. Fig. 9 illustrates the accuracy achieved by *Ipanel* for each operation. The overall recognition accuracy is above 85%. It shows that our algorithm could accurately recognize the 43 operations (7 gestures, 10 numbers and 26 alphabets). While the accuracy of some operations is relatively low (about 85.4%), for example, '4', 'a', 'j', 'w' and flip left *etc.*. On one hand, this might be because the acoustic signals of '4', 'a', 'j' and 'w' have too many peaks which bring excessive information, while flip left only contains one peak which is prone to be missed. It is easy to make mistakes between flip left and flip right, scroll down and scroll up for their similar structure.

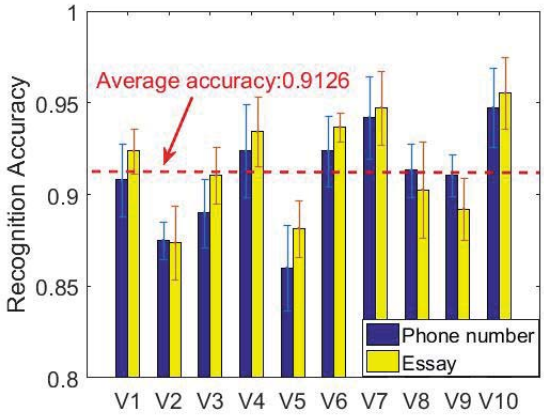


Fig. 10. Recognition accuracy among different users.

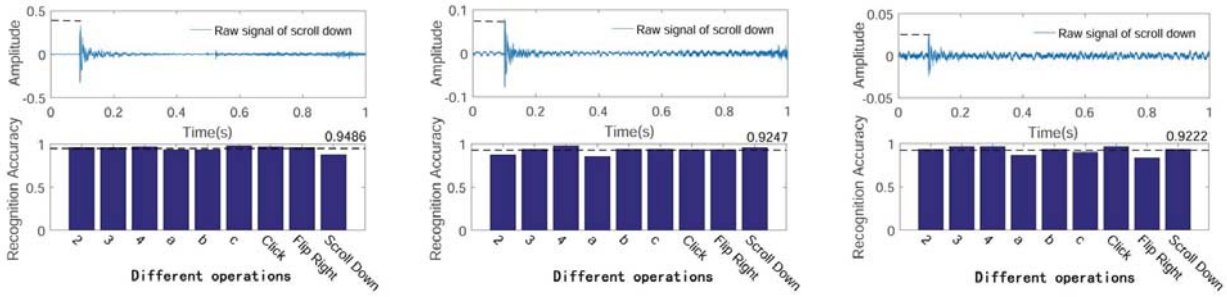
On the other hand, in addition to differences induced by various stroke structures, different writing directions can slightly generate different frequency shifts. However, with the strong feature classification capability of CNN, the recognition performance is not affected.

4.2.3 Accuracy Achieved by Different Users. We emphatically evaluate the system performance among different users. In this experiment, we invite 10 volunteers (denoted as V1 to V10) who are asked to write an essay of about 100 words and input 10 series of mobile phone numbers for 10 times respectively in a typical office room. During the experiment, each volunteer independently performs operations under the same conditions (operation region, devices, *etc.*). The recorded signals are collected and sent to *Ipanel* for recognition. After that, we compare the recognition result with the ground truth and calculate the accuracy for each volunteer's handwriting.

Fig. 10 shows the experimental result, in which the average handwriting recognition accuracy among all users is about 91.26%. The highest recognition accuracy reaches 95% while the lowest accuracy is still above 85%. Therefore, although the recognition accuracy varies slightly across different users, it stays around 90%. The rationale behind this is that the unique features are extracted from both frequency domain and time domain of the acoustic signals, while it has little to do with the operating habits of users [8]. Compared with the recent work WordRecorder [23], *Ipanel* has better performance even in more challenging circumstances.

4.3 Impact Factors

Multiple factors have been taken into consideration from the algorithm, the writing manner (strength, speed and region) to the desk material, which we think could affect the recognition accuracy of the system. For the algorithm part, the CNN structure design and parameter setting of CNN are essential which will affect the system performance. For writing, we evaluate the effect of input strength, input speed and input region on the performance of our system. Input strength is closely related to the signal strength of the generated acoustic signal and thus we believe it affects the input recognition accuracy. The input region relative to the mobile also affects the performance as we employ the microphone at the bottom for sensing. So if the input region is on top of the mobile, then the signal strength is weaker as the propagation distance is larger. We also evaluate the effect of writing speed on the performance which we believe the faster speed may change the frequency domain



(a) Strong input strength (b) Medium input strength (c) Weak input strength

Fig. 11. Impact of input strength on the recognition accuracy. The upper part depicts raw signals of an example gesture.

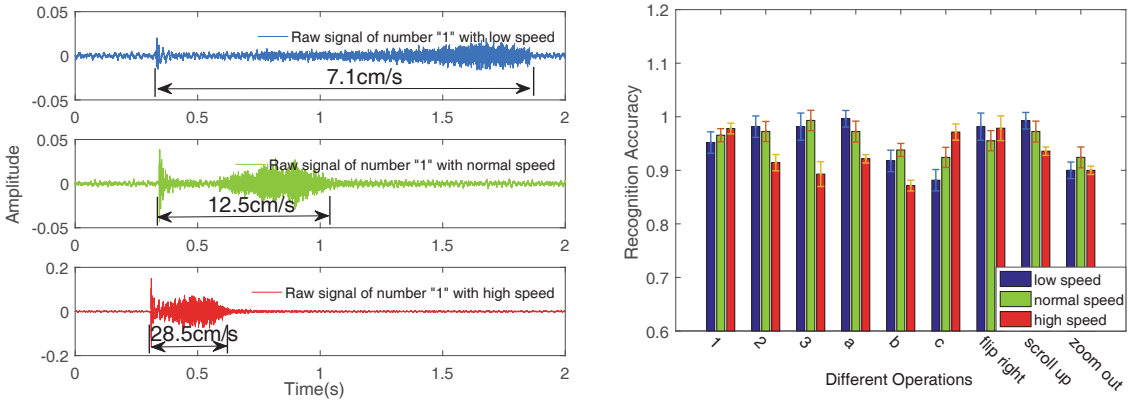
information we employed for classification. We also consider different surface material types as the material greatly affects the strength of the signal generated.

4.3.1 Impact of Input Strength. We evaluate the impact of input strength on the recognition accuracy in this section. Intuitively, the input strength could be strong, medium and weak, which will greatly affect the strength of generated acoustic signals. Before the actual experiment, we instruct each participant how to control the input strength to have three different input strengths. With three type input strengths, each participant repeat 50 times to different operations in a common office. We employ a digital weighting scale with $0.01g$ resolution to measure the input strength. The average strength of strong, medium and weak inputs are $3.13N$, $1.72N$ and $0.75N$ respectively. We note that these results are derived by $F = mg$, where m is the measured weight, g is the acceleration of gravity, and N is the unit of force.

We plot the raw signal of an example gesture and the corresponding recognition accuracy in Fig. 11. The peak amplitude indicates the input strength clearly. For convenient displaying, we employ several representative operations to show the accuracy of recognition. The accuracies are above 90% with strong, medium and weak input strengths as shown in Fig. 11. We could see that *Ipanel* still achieves relatively good performance when the operation is soft. This is because different strengths simply change the amplitude of the feature but not feature itself, while this effect is eliminated by normalization. However, in practice, very weak input strength does decrease the recognition accuracy because very small acoustic signals can be easily buried in surrounding noise. With the results above, we can conclude that the input strength makes little effect on the performance of *Ipanel* and *Ipanel*'s performance is stable with different input strengths.

4.3.2 Impact of Input Speed. Next, we evaluate the effect of finger sliding speed on input recognition accuracy. In this experiment, we asked the participants to perform input operations on the wood surface in the office environment, where each operation was repeated 50 times. The finger-writing speeds were about 7.1cm/s, 12.5cm/s, 28.5cm/s respectively and the participants were given simple instructions and exercises.

The raw signals of one gesture at three speeds are shown in Fig. 12a, which clearly shows the effect of finger speed on the signal. We show the recognition accuracy for 9 typical operations at different writing speeds in Fig. 12b. We find that the accuracy slightly decreases as writing speed increases. However, the accuracy remains above 85% at a speed of 28.5cm/s (twice the normal writing speed). The reason behind is that the faster writing speed results in an overlapping of frequency domain information, which thus reduces the chance of discrimination between different operation inputs.



(a) Raw signal of number '1' at different speeds.

(b) The recognition accuracy of 9 typical operations with different writing speed.

Fig. 12. Impact of input speed on recognition accuracy.

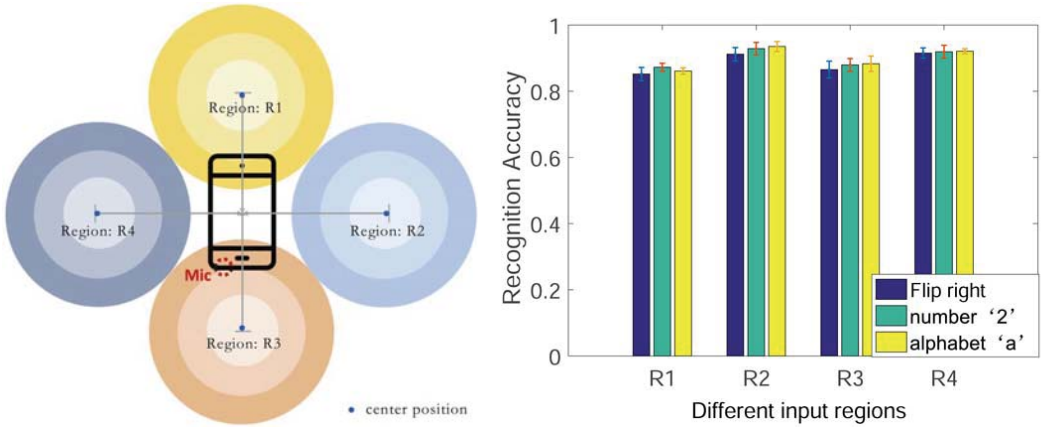
Table 4. Impact of surface materials

Material	Wood	Board	Metal	Steingut	Paper
Accuracy	92.1%	93.2%	91.7%	89.3%	83.5%

4.3.3 Impact of Surface Material. Additionally, we evaluate *Ipanel* on surfaces of five typical materials types, namely, wood, cardboard, metal, steingut and paper. Obviously, the material of the surface the users operate on is closely related to the generated acoustic signals, and thus greatly affects the recognition performance. In this experiment, we ask each participant to repeat operations for 50 times on surfaces of each material and we obtain 21500 samples (10*43*50) for each material type.

Table 4 shows the average recognition accuracy achieved by *Ipanel*. In general, solid surfaces, such as wood, cardboard and metal perform better than lithoid surfaces (*e.g.*, paper). The main reason could be that the acoustic energy of friction is related to the material type [8], and solid surfaces emit larger detectable audio signals [22].We can see from the results that cardboard presents the highest accuracy while paper surface presents the lowest accuracy. We believe the strength of the generated signal is the main determining factor of the performance difference here. As for our experiments, five different surface materials cause around 10% accuracy variation. In light of these experimental results, we suggest applying our system on solid surfaces such as wood and cardboard for best performance. Though the material of the surface the user input on is related to the component of the acoustic signal, different operations maintain well discrimination under the same material.

4.3.4 Impact of Input Region. Although the operations are performed beneath the device by default, we evaluate the system performance when we perform the finger operations at other regions with respect to the device. Fig. 13a depicts the four regions where we can operate around the smartphone. For example, the center of region R1 is 10*cm* away from the center of the smartphone, while R2, R3 and R4 are the same distance from the mobile at different directions respectively. For simplicity, we take flip-right, number '2' and alphabet 'a' as examples and the results are shown in Fig. 13b. The users repeat the operations 50 times at each region in a typical office room.



(a) Different positions and regions of operation input.

(b) Impact of input region.

Fig. 13. Impact of input region on recognition accuracy. We take flip-right, number '2' and alphabet 'a' as an example.

Table 5. Impact of different kinds of mobile devices.

Device	NUBIA	MOTOROLA	SAMSUNG
Accuracy	92.1%	89.5%	90.5%

As shown in Fig. 13b, the accuracy achieved in the two side regions are higher than the top and below regions. R1 is the farthest position from smartphone's microphone, leading to lower recognition accuracy. As for the region R2 and R4, the reason why they achieve higher recognition accuracy is that they are near to the microphone than region R1, and poses more abundant multi-path than region R3. Meanwhile, compared to region R3, R2 and R4 have stronger Doppler effect, which generates higher frequency variations and accordingly more unique spectrograms features. The recognition accuracy for the 4 regions are all above 85%, demonstrating a larger sensing area than the active acoustic sensing [55, 58].

4.3.5 Accuracy of Different Devices. Additionally, we use the other two mobile devices, SAMSUNG G3568V and MOTOROLA MT887 to evaluate the performance of *Ipanel*. During the experiment, we let each participant to repeat operations for 50 times on surfaces of each material and we obtain 21500 samples (10*43*50) for each mobile devices. When running *Ipanel*, each device uses its own training set. Table 5 shows that the recognition accuracy of three smartphones are all around 90% and there are only small variations among different mobile devices. The small accuracy difference could be caused by varying microphone qualities. In future, we plan to test *Ipanel* on other type of smart devices such as smart watches and tablets.

4.3.6 Impact of Training Epoch. Pattern matching and classification of operations rely on parameters (training epoch and learning rate *et al.*) setting in the training as described in Section 3.3.2, where training epoch is the most important factor. In this test, we stepwise adjust the training parameters across the same sample set. Fig. 14 shows that the recognition accuracy achieved by *Ipanel* ascends with the number of training epoch increasing. With just one training epoch, the recognition accuracy is around 6%. The accuracy escalates to above 90% when the number of training epoch increases to 10. Marginal improvement is achieved by further increasing the number

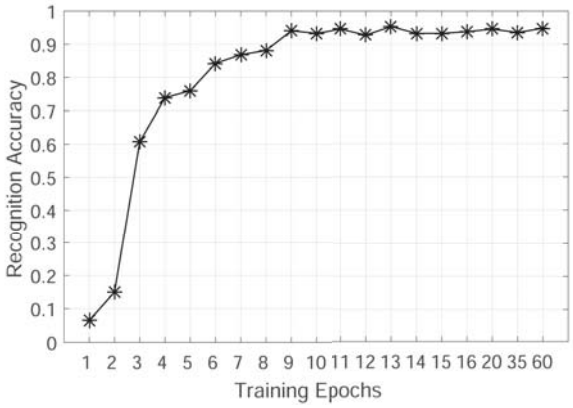


Fig. 14. Recognition accuracy with different training epochs.

of training epoch beyond 10. We also note that the recognition accuracy has a fluctuation when the training epoch is greater than 10. The setting of training epoch depends on the scale of the training set. Too many training epochs may have a negative impact on accuracy because of over-fitting. Under this experiment, we also tried to change the learning rate, drop factor and drop period. We find that once these parameters are set, it is not suggested to change them as the recognition performance can be greatly affected. The training epochs can be appropriately adjusted with the increase of samples.

5 DISCUSSION

Ipanel advances the research on passive input recognition. We discuss some issues and the directions we can further explore in the future.

Multiple microphones. Most smartphones are equipped with two microphones, and support stereo recording. Recently, dual-microphone has been employed for keystroke localization [34, 54]. *Ipanel* can either choose the closer microphone with respect to the input position or combine the information from both microphones to improve the recognition accuracy. Additionally, the time-difference-of-arrival (TDoA) of the signals at the two microphones maybe be employed to accurately track the fingers' position to recognize more complicated gestures. We keep this interesting problem for further exploration in our future work.

Why *device-free* is important? In our design, we use only one smartphone device to achieve device-free operation input. We believe that, smart devices such as smartphone, tablet, smartwatch, *et al.* are pervasive. Incorporating them for system design will not bring in much burden to users. For device-free interactions, the first advantage lies in the ease of control when the device is not in hand. For example, users could answer the call by sliding on the surface of a nearby desk. The second advantage is that it provides users with more natural interactions with smart devices. For instance, device-free interactions will also support future smart table-gaming and augmented reality applications [47].

Training. According to the results in Subsection 4.3.6, ten training epochs per operation is enough to achieve considerably high accuracies. We are exploring novel schemes to get rid of the bondage of initial training. The physical distinction of each operation is investigated for further enhancement.

Fast input and ligatures. Inputting very fast ($< 300ms$ per gesture) and ligatures degrade the recognition accuracy. The reason lies in that the features of fast input gesture have big differences with those in the training set. It is suggested that the user keeps consistent input speed.

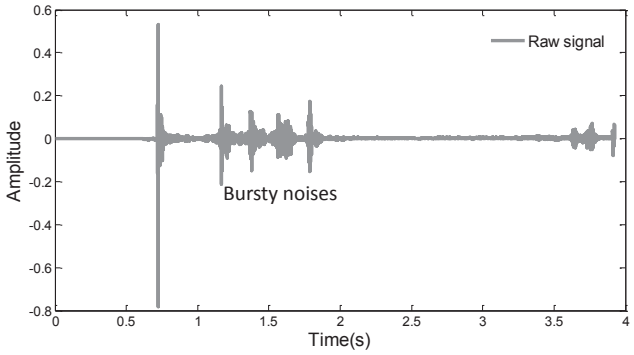


Fig. 15. An example of disturbance by the bursty noises.

End-to-end latency and user experience. The current average end-to-end latency of our recognition process (From the end of an operation to output recognition results) is about 63ms. We believe this latency can be further reduced. In our current system, the latency is composed of three parts: signal segmentation (3ms), feature extraction (15ms), and recognition (45ms). The recognition time increases with the size of the database. With the pre-processing method such as the Local Binary Pattern method [37], we can reduce the database size by 30%. Another key factor deciding the processing time is the graphical processing unit employed in the phone. For our current implementation, we employ a NUBIA Z9 mini phone with an Adreno 405 GPU. The latest Apple and Samsung flagship are embedded with a much more powerful GPUs and we believe this latency can be further reduced with these latest mobiles. Recently, NPU (Neural Processing Unit) is commonly used in mobile phones, which will further reduce the latency.

Sensing range. One limitation of our system we believe is the sensing range. We found that when the fingers are 20 cm away from the mobile, the recognition accuracy significantly decreases because the signals become too weak to be used for recognition. How to increase the sensing range is one important direction we would like to input more efforts in the future.

Disturbed by bursty noises. The bursty noise, *e.g.*, a sequence of clicks, disturb the energy detection of operation input, as shown in Figure 15. The noise may override audio signal of the operation. We are investigating the benefits of multi-dimensional sensing using more sensors. The motion sensors, such as the accelerometer and gyroscope, may help combat bursty noises [54] and provide complementary information for interaction recognition.

6 RELATED WORK

Gesture recognition systems. Many commercial products have motivated gestural input for mobile devices. Xbox Kinect [3], AN580[1], Leap Motion [4] and Google Project Soli [2] are practical examples. However, the price of these products is expensive and the dedicated hardware usually need to be installed properly.

Recently, RF signals have been employed for gesture recognition. WiSee[44], AllSee [28], WiVi[6] and RF-IDraw [53] are several representative systems. WiVi used surrounding wireless signals and radar technique to recognize the gestures through the wall. WiSee leveraged the Doppler effect to extract gesture information. Besides, AllSee explored the amplitude change of wireless signals to distinguish different gestures. However, these wireless systems require dedicated hardware or customized devices, which limits their applicability in the wide public. Some systems also use inertial sensors for gesture recognition [7, 22, 42]. For example, Gummeson *et al.* [22] presented a ring-form device containing accelerometer, force sensing resistor and other sensors to enable

energy-efficient gesture input. Paradiso *et al.* [41] proposed a system that can localize the position of knocks and taps on a large sheet of glass, which however required dedicated hardware deployment. Besides, it could not provide complex interaction gestures such as flip and zoom. Finexus [16] instrumented the fingertips with electromagnets to track fine fingertip movements in real time. The majority of these systems demand customized hardware. Lopes *et al.* [35] proposed a system that recognized touch gestures on a touchscreen with acoustic sensing. Compared to this work, *Ipanel* is able to recognize more gestures and our system can be used directly as a smartphone application. More recently, BeyondTouch [59] used built-in sensors of smartphone to extend the input to areas off the screen. It only supported single-finger gesture, such as tap and slide. By contrast, *Ipanel* supports gestures with multiple fingers, such as zoom in and out. In addition, visible light has been employed for human posture recognition in LiSense system [33], which required dedicated hardware setups. Camera-based systems have also been leveraged for gesture or object recognition [17, 45].

Handwriting recognition systems. In 1998, LeCun *et al.*. [31] propose Convolutional Neural Network (CNN) and establish its modern structure LeNet-5 for the first time. Meanwhile, CNN was first introduced in the field of image recognition. Subjected to the limited samples and computing power of the time, LeNet-5 could not deal with complex problems. Until 2012, AlexNet [29] was proposed by Krizhevsky *et al.*, which is an improved version with adding 2 fully connected layers to avoid over-fitting. In subsequent development, VGG [48] was proposed in 2014, and ResNet [25] was introduced in 2015. VGG's main idea is to improve the accuracy of recognition by increasing the number of convolutional layers. ResNet inherits idea of VGG, while too many layers limit its training, it has to come up with the concept of residual networks. CNN is now widely utilized for almost every perceptual task related to image and video data because of its good performance. In the field of word recognition, Arik Poznanski and Lior Wolf [43] employ CNNs directly over raw pixel values, instead of using SVMs over Fisher Vectors of SIFTs [9]. Based on the layout of VGG [48], they rebuilt a network structure of CNN and obtained 96.88% accuracy on the test set of the synthetic data. Currently, handwriting recognition is dominated by the use of Recurrent Neural Networks (RNNs) [11] and its extensions such as Long-Short-Term-Memory (LSTM) networks [19], Hidden Markov Models (HMMs) [10], and various combinations of these methods [5, 12, 18].

Acoustic sensing approaches with mobile devices. Acoustic sensing has enabled varied innovative techniques with mobile devices, such as social interactions [32], mobile games [60] and security [34, 61]. For example, Spartacus [49] enabled spatially-aware neighboring device interaction leveraging acoustic sensing and Doppler effects. Zhu *et al.* [61], presented the context-free attacks using keyboard acoustic emanations. Liu *et al.* [34], proposed to integrate TDoA with mel-frequency cepstral coefficients (MFCC) to locate the origin of keystrokes. Acoustic sensing could also be used for recommendation systems [39], ranging and localization systems [30, 50, 52] and meeting systems[51]. In particular, a few papers have used acoustic signals to estimate the distance based on time-difference of arrival [34, 56, 61] or the Doppler effect [26, 49, 58]. However, these localization systems could not achieve fine-grained tracking to make gesture recognition come true. Recently, AAMouse [58] was proposed to employ a smartphone as a mouse to accurately tracks the hand movements. It also leverages acoustic sensing and the Doppler effect for tracking. A disadvantage of AAMouse is that the user needs to hold the mobile device in hand and an initial calibration process is needed. Another input system using acoustic sensing is UbiK [54], which explores the multipath fading to localize the keystrokes. However, the input position for each keystroke is fixed in a favor of a printed keyboard.

Additionally, Braun *et al.* [13] proposed to track hand activities on a surface, which however requires careful deployment of multiple microphones. Scratch Input [24] operated by listening to the sound of "scratching" that is transmitted through the surface material, which needs a customized stethoscope sensor to help capture the sound. SurfaceLink[21] used a sensor fusion method to enable multi-device interaction on a surface. FingerIO[38] proposed using active sonar for fine-grained finger tracking without instrumenting the finger with sensors. More recently, Ruan *et al.* [46] proposed a device-free gesture recognition system for in-air gestural interactions, which transformed the device into an active sonar system. LLAP [55] and CAT [36] utilized acoustic phase-based distance

measurement and FMCW(Frequency Modulated Continuous Waveform) respectively to develop fine-grained motion tracking systems. However, they cannot support gestures jointly performed by two or more fingers, such as zoom in and out.

7 CONCLUSION

In this paper, we have presented *Ipanel*, an accurate and device-free operation recognition system that employs passive acoustic signals to enable interactions with COTS smart devices. *Ipanel* leverages the in-built microphone to capture acoustic signals when fingers slide on the table surface, and then extracts acoustic features to recognize the operations. We have implemented *Ipanel* on COTS smartphones and conducted comprehensive experiments to validate the effectiveness and robustness of the system. Our results demonstrate high input recognition accuracies across different users under varying parameters in different environments. *Ipanel* is robust against ambient noise and surface materials.

ACKNOWLEDGMENTS

This research is partially supported by National key research and development plan 2017YFB0801702, NSFC with No. 61625205, 61872447, 61632010, 61772546, 61751211, 61772488, 61520106007, 61672038, 61602067 Key Research Program of Frontier Sciences, CAS, No. QYZDY-SSW- JSC002, NSFC with No. NSF ECCS-1247944, and NSF CNS 1526638. NSF of Jiangsu For Distinguished Young Scientist: BK20150030. Natural Science Foundation of Chongqing: No.CSTC2018JCYJA1879.

REFERENCES

[1] 2015. AN580: Ifrared gesture recognition by Silicon Labs. https://www.silabs.com/Support%20Documents/TechnicalDocs/AN580.pdf
[2] 2016. Google Project Soli. https://atap.google.com/soli/
[3] 2017. Kinect for Xbox One. https://www.xbox.com/en-US/xbox-one/accessories/kinect-for-xbox-one
[4] 2017. Leap Motion. https://www.leapmotion.com/
[5] Haikal El Abed and Volker Mirgner. 2011. ICDAR 2009-Arabic handwriting recognition competition. In *International Journal on Document Analysis and Recognition*. 3–13.
[6] Fadel Adib and Dina Katabi. 2013. See through walls with WiFi!. In *Proceedings of the 2013 ACM conference on SIGCOMM*. ACM.
[7] Sandip Agrawal, Ionut Constandache, Shravan Gaonkar, Romit Roy Choudhury, Kevin Caves, and Frank DeRuyter. 2011. Using mobile phones to write in air. In *Proceedings of the 9th international conference on Mobile systems, applications, and services (MobiSys)*. ACM, 15–28.
[8] Adnan Akay. 2002. Acoustics of friction. *The Journal of the Acoustical Society of America* 111, 4 (2002), 1525–1548.
[9] J Almazĺćn, A Gordo, A Fornĺęs, and E Valveny. 2014. Word Spotting and Recognition with Embedded Attributes. In *Pattern Analysis and Machine Intelligence IEEE Transactions on*. 2552–2566.
[10] Sherif Abdel Azeem and Hany Ahmed. 2013. Effective technique for the recognition of offline Arabic handwritten words using hidden Markov models. In *International Journal on Document Analysis and Recognition*. 399–412.
[11] Anne Laure Biannebernard. 2012. The A2iA French handwriting recognition system at the Rimes-ICDAR2011 competition. In *Document Recognition and Retrieval XIX*. 51.
[12] Thĺęodore Bluche, Hermann Ney, and Christopher Kermorvant. 2014. A Comparison of Sequence-Trained Deep Neural Networks and Recurrent Neural Networks Optical Modeling for Handwriting Recognition. In *International Conference on Statistical Language and Speech Processing*. 199–210.
[13] Andreas Braun, Stefan Krepp, and Arjan Kuijper. 2015. Acoustic tracking of hand activities on surfaces. In *Proceedings of the 2nd international Workshop on Sensor-based Activity Recognition and Interaction*. ACM, 9.
[14] Liwei Chan, Rong-Hao Liang, Ming-Chang Tsai, Kai-Yin Cheng, Chao-Huai Su, Mike Y Chen, Wen-Huang Cheng, and Bing-Yu Chen. 2013. FingerPad: private and subtle interaction using fingertips. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 255–260.
[15] Ke-Yu Chen, Kent Lyons, Sean White, and Shwetak Patel. 2013. uTrack: 3D input using two magnetic sensors. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 237–244.
[16] Ke-Yu Chen, Shwetak Patel, and Sean Keller. 2016. Finexus: Tracking Precise Motions of Multiple Fingertips Using Magnetic Sensing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1504–1514.

[17] Tiffany Yu-Han Chen, Lenin Ravindranath, Shuo Deng, Paramvir Bahl, and Hari Balakrishnan. 2015. Glimpse: Continuous, Real-Time Object Recognition on Mobile Devices. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. ACM, 155–168.
[18] Patrick Doetsch, Michal Kozielski, and Hermann Ney. 2014. Fast and Robust Training of Recurrent Neural Networks for Offline Handwriting Recognition. In *International Conference on Frontiers in Handwriting Recognition*. 279–284.
[19] J. G Fiscus. 1997. A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER). In *IEEE Workshop on Automatic Speech Recognition and Understanding*. 347–354.
[20] Luo Gan, Chen Mingshi, Yang Panlong, and Li. Ping. 2017. SoundWrite II: Ambient Acoustic Sensing for Noise Tolerant Device-Free Gesture Recognition. In *ICPAD*.
[21] Mayank Goel, Brendan Lee, Md Tanvir Islam Aumi, Shwetak Patel, Gaetano Borriello, Stacie Hibino, and Bo Begole. 2014. SurfaceLink: using inertial and acoustic sensing to enable multi-device interaction on a surface. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*. ACM, 1387–1396.
[22] Jeremy Gummeson, Bodhi Priyantha, and Jie Liu. 2014. An energy harvesting wearable ring platform for gesture input on surfaces. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services (MobiSys)*. ACM, 162–175.
[23] Du Haishi, Yang Panlong, luo Gan, and Li. Ping. 2017. WordRecorder: Accurate Acoustic-based Handwriting Recognition Using Deep Learning. In *Infocom*.
[24] Chris Harrison and Scott E Hudson. 2008. Scratch input: creating large, inexpensive, unpowered and mobile finger input surfaces. In *Proceedings of the 21st annual ACM symposium on User interface software and technology (UIST)*. ACM, 205–208.
[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.
[26] Wenchao Huang, Yan Xiong, Xiang-Yang Li, Hao Lin, Xufei Mao, Panlong Yang, and Yunhao Liu. 2014. Shake and walk: acoustic direction finding and fine-grained indoor localization using smartphones. In *INFOCOM, 2014 Proceedings IEEE*. IEEE, 370–378.
[27] CEI IEC. 1985. Integrating-averaging sound level meters. (1985).
[28] Bryce Kellogg, Vamsi Talla, and Shyamnath Gollakota. 2014. Bringing gesture recognition to all devices. In *Usenix NSDI*, Vol. 14.
[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *International Conference on Neural Information Processing Systems*. 1097–1105.
[30] Patrick Lazik and Anthony Rowe. 2012. Indoor pseudo-ranging of mobile devices using ultrasonic chirps. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems (SenSys)*. ACM, 99–112.
[31] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*. 2278–2324.
[32] Youngki Lee, Chulhong Min, Chanyou Hwang, Jaeung Lee, Inseok Hwang, Younghyun Ju, Chungkuk Yoo, Miri Moon, Uichin Lee, and Junehwa Song. 2013. Sociophone: Everyday face-to-face interaction monitoring platform using multi-phone sensor fusion. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services (MobiSys)*. ACM, 375–388.
[33] Tianxing Li, Chuankai An, Zhao Tian, Andrew T Campbell, and Xia Zhou. 2015. Human sensing using visible light communication. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom)*. ACM, 331–344.
[34] Jian Liu, Yan Wang, Gorkem Kar, Yingying Chen, Jie Yang, and Marco Gruteser. 2015. Snooping Keystrokes with mm-level Audio Ranging on a Single Phone. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom)*. ACM, 142–154.
[35] Pedro Lopes, Ricardo Jota, and Joaquim A Jorge. 2011. Augmenting touch interaction through acoustic sensing. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*. ACM, 53–56.
[36] Wenguang Mao, Jian He, and Lili Qiu. 2016. CAT: high-precision acoustic motion tracking. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. ACM, 69–81.
[37] Raafat Salih Muhammad and Mohammed Issam Younis. 2017. The Limitation of Pre-processing Techniques to Enhance the Face Recognition System Based on LBP. *Iraqi Journal of Science* 58, 581B (2017), 355–363.
[38] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. FingerIO: Using Active Sonar for Fine-Grained Finger Tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1515–1525.
[39] Shahriar Nirjon, Robert F Dickerson, Qiang Li, Philip Asare, John A Stankovic, Dezhi Hong, Ben Zhang, Xiaofan Jiang, Guobin Shen, and Feng Zhao. 2012. Musicalheart: A hearty way of listening to music. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems (SenSys)*. ACM, 43–56.
[40] Shahriar Nirjon, Jeremy Gummeson, Dan Gelb, and Kyu-Han Kim. 2015. TypingRing: A Wearable Ring Platform for Text Input. In *Proceedings of the 9th international conference on Mobile systems, applications, and services (MobiSys)*. ACM.
[41] Joseph A Paradiso, Che King Leo, Nisha Checka, and Kaijen Hsiao. 2002. Passive acoustic sensing for tracking knocks atop large interactive displays. In *Sensors, 2002. Proceedings of IEEE*, Vol. 1. IEEE, 521–527.
[42] Taiwoo Park, Jinwon Lee, Inseok Hwang, Chungkuk Yoo, Lama Nachman, and Junehwa Song. 2011. E-gesture: a collaborative architecture for energy-efficient gesture recognition with hand-worn sensor and mobile devices. In *Proceedings of the 9th ACM Conference on Embedded*

Networked Sensor Systems (SenSys). ACM, 260–273.
[43] Arik Poznanski and Lior Wolf. 2016. CNN-N-Gram for HandwritingWord Recognition. In *Computer Vision and Pattern Recognition*. 2305–2314.
[44] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking (MobiCom)*. ACM, 27–38.
[45] James M Rehg and Takeo Kanade. 1994. Visual tracking of high dof articulated structures: an application to human hand tracking. In *Computer VisionąłECCV'94*. Springer, 35–46.
[46] Wenjie Ruan, Quan Z Sheng, Lei Yang, Tao Gu, Peipei Xu, and Longfei Shangguan. 2016. AudioGest: enabling fine-grained hand gesture detection by decoding echo signal. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 474–485.
[47] James She, Alvin Chin, Feng Xia, and Jon Crowcroft. 2015. Introduction to: Special Issue on Smartphone-Based Interactive Technologies, Systems, and Applications. *ACM Transactions on Multimedia Computing, Communications, and Applications* 12, 1s (2015), 11.
[48] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Computer Science*.
[49] Zheng Sun, Aveek Purohit, Raja Bose, and Pei Zhang. 2013. Spartacus: spatially-aware interaction for mobile devices through energy-efficient audio sensing. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services (MobiSys)*. ACM, 263–276.
[50] Zheng Sun, Aveek Purohit, Kaifei Chen, Shijia Pan, Trevor Pering, and Pei Zhang. 2011. PANDAA: physical arrangement detection of networked devices through ambient-sound awareness. In *Proceedings of the 13th international conference on Ubiquitous computing (Ubicomp)*. ACM, 425–434.
[51] Sanjib Sur, Teng Wei, and Xinyu Zhang. 2014. Autodirective audio capturing through a synchronized smartphone array. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services (MobiSys)*. ACM, 28–41.
[52] Yu-Chih Tung and Kang G Shin. 2015. EchoTag: Accurate Infrastructure-Free Indoor Location Tagging with Smartphones. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom)*. ACM, 525–536.
[53] Jue Wang, Deepak Vasisht, and Dina Katabi. 2014. RF-IDraw: virtual touch screen in the air using RF signals. In *Proceedings of the 2014 ACM conference on SIGCOMM*. ACM, 235–246.
[54] Junjue Wang, Kaichen Zhao, Xinyu Zhang, and Chunyi Peng. 2014. Ubiquitous keyboard for small mobile devices: harnessing multipath fading for fine-grained keystroke localization. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services (MobiSys)*. ACM, 14–27.
[55] Wei Wang, Alex X Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. ACM, 82–94.
[56] Robert Xiao, Greg Lew, James Marsanico, Divya Hariharan, Scott Hudson, and Chris Harrison. 2014. Toffee: enabling ad hoc, around-device interaction with acoustic time-of-arrival correlation. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*. ACM, 67–76.
[57] Jie Xiong and Kyle Jamieson. 2013. ArrayTrack: a fine-grained indoor location system. In *USENIX NSDI*.
[58] Sangki Yun, Yi-Chao Chen, and Lili Qiu. 2015. Turning a Mobile Device into a Mouse in the Air. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*. ACM, 15–29.
[59] Cheng Zhang, Anhong Guo, Dingtian Zhang, Yang Li, Caleb Southern, Rosa I Arriaga, and Gregory D Abowd. 2016. Beyond the Touchscreen: An Exploration of Extending Interactions on Commodity Smartphones. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 6, 2 (2016), 16.
[60] Zengbin Zhang, David Chu, Xiaomeng Chen, and Thomas Moscibroda. 2012. SwordFight: enabling a new class of phone-to-phone action games on commodity phones. In *Proceedings of the 10th international conference on Mobile systems, applications, and services (MobiSys)*. ACM, 1–14.
[61] Tong Zhu, Qiang Ma, Shanfeng Zhang, and Yunhao Liu. 2014. Context-free Attacks Using Keyboard Acoustic Emanations. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 453–464.

Received May 2018; revised November 2019; accepted January 2019