Dual-Frequency Capacitive Sensing for Finger Identification

(指の識別のための 2 周波数帯での静電容量センシング手法の提案)

37-186978

Minghui Chen (陳明輝)

Supervisor: Assoc. Prof. Koji Yatani (矢谷浩司)

Department of Electrical Engineering and Information Systems

University of Tokyo

This dissertation is submitted for the degree of

Master of Engineering

Aug 14th, 2020

Acknowledgements

This work would never come true without the advice and assistance from many people, among whom I shall express appreciation to my supervisor, Associate Professor Koji Yatani, the first. He has always been patient, erudite, and forward-looking during the past two years when instructing me throughout this whole project.

The place where I carried out my research, the Interactive Intelligent Systems Lab (IIS-Lab), has always been a warm and helpful environment to my academic and personal life. I am extremely grateful to all members of IIS-Lab, including Arissa Janejera Sato, Asahi Takenouchi, Benio Shimada, Carla F. Griggio, Daisuke Shibato, Dimas Antony Chacon Salas, Hao Xu, Hidenori Matsui, Hiroaki Masaki, Michihiko Ueno, Minghui Chen, Misako Motooka, Kengo Shibata, Peihan Tu, Shoko Sano, Siwook Choi, Tatsuhiko Sakaguchi, Toby Chong, Yuji Sugiyama, and Zefan Sramek. All of you were so warm-hearted and eager to help with my research surveys, system design, system evaluation, and master thesis writing, let alone carrying me over the obstacles I met during the life in Japan. I really enjoy all the comments and recommendations from the lab members, which makes the IIS-Lab a creative and open-minded place.

I would also like to thanks the others who assisted me during the process of carrying out this research, including my previous colleagues from Peking University, my friends from the International Multidisciplinary Engineering program, the university staff members, and the friends I met online. Their companion encouraged me to move forward to every step in my studying abroad.

Despite the distance across countries, my family members, including my cat, have always been supportive to me. They might not fully understand what I was working on, but they never lost confidence with my decisions. I treasure every second with them.

Lastly, thanks to all the non-living objects in this study. Thank you for doing well from the very beginning of this research to the completion of my final draft. This research could not be successful without the kindness of them.

Abstract

The awareness of a specified finger touching an object expands the dimension of touch interaction design. This paper provides a novel approach that implements fast, accurate, and low-cost finger identification on touching a conductive object. Utilizing hand tissue as a radio frequency signal waveguide, and an analog process to compare the phase and magnitude difference between frequency components in the signal, we designed a signal emitting wristband and a compact detector circuit to bind the finger identification functionality to a conductor. we also show the process of retrieving finger identity from the analog data. Our system demonstrates its ability to support future interfaces on a tight screen such as a smartwatch or a wide variety of daily context.

概要

どの指が物体に触れているか識別することができれば，指によるインタラクションの幅が大きく広がる．本稿では，導電性の物体にユーザが触れたときに，どの指が触れたかを速く正確に特定する手法を紹介する．本研究において我々は，信号を出力するリストバンドと信号を検知するコンパクトな回路を設計し，導電性の物体に触れた指を特定するシステムを構築した．本システムでは，手の皮膚組織を高周波の信号を伝える導波管として用い，伝わった複数周波数帯の信号の位相や振幅を処理して比較することで，導電性の物体に触れた指を特定することが可能となっている．本稿では，複数周波数帯の信号から指を特定する手法を述べたのち，本システムがスマートウォッチなど日常で用いるデバイスに利用できる可能性について議論する．最後に，本システムを用いた将来のインターフェースの可能性について述べる．

Table of contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Background

Touching with finger is a natural and commonly accepted way to interact with objects. Such interaction applies to touch screens, touch buttons, and functional materials that capture finger movements. Especially for the touchscreen, a touch carries more information than an (x, y) coordinate and a timestamp [3]. There is also research devoted to extending the modality of touching, such as multi-touch, gestural movement [12, 15], palm-or-knuckle-touch [23], or extra finger properties, i.e. pitch or roll orientation [18].

Finger identification, finger aware interaction [15, 21], or finger specific interaction [3] is one of the methods to enhance touch interaction by determining the specific finger used for a certain touch event. In general, the term "finger identification" may refer to identifying the person or the hand used that is operating a device [13]. In this paper, however, this term is confined to recognizing the finger on a same hand used for touching. This method allows for more information to stack on a single touch(i.e. different fingers imply different functions) [3, 8].

Numerous work examined how to distinguish among fingers of the same hand by electronic hardware. Approaches include optical recognition [10, 14, 15, 21], fingerprint

recognition [13], and capacitive sensing [9, 16], yet all brought in certain drawbacks. This paper introduces a new method that achieves swift, accurate, and versatile finger identification with an additional wristband on the user.

1.2 Motivation

Despite the previous examinations on the practice of finger identification with specialized hardware, most of which being wearables on fingers, palms, or arms, in real life, seldom will people be likely to equip those optical trackers or bulky sensors on their body when operating a device. We are interested in achieving finger identification with the most traditional form of wearables: the widely accepted wristband, as the equipment for finger identification.

To fully exploit the convenience of a wristband, we survey on the topic of capacitive sensing. The capacitive techniques allows fast and indirect observations over the electromagnetic field (EM field) around a certain space. We introduce the method of using two frequencies for signal feature detection, and hypothesize that the phase and magnitude information of such a signal could reveal the finger identity during a touch on a piece of conductive material. We then fabricate a combined system of hardware and software to detect the signal features, and describe the method to measure and project them to the corresponding finger identity.

1.3 Research Questions and Scope

Our main research questions are as follows:

- Can we identify a finger via capacitive sensing without attaching anything to the finger directly?
- What hardware and algorithm do we need for the finger identification?

- What features contribute the most accuracy to the identification?

To provide an answer to these questions, we devise a system consisting of a wristband (emitter), a detector (sensor), and an algorithm to provide the finger identity detection results. We choose the scene of touching a metal pad and measured the accuracy of finger identification in this controlled environment.

1.4 Contributions

The contributions of this work are as follows:

- Proposing a hardware system to identify fingers using capacitive sensing.
- Quantitative evaluations of the system under controlled experiments.

1.5 Overview

In this thesis, we explain our design and experiment process of a capacitive sensing based finger identification system.

Chapter 2 classifies the current finger identification methods and lists the important capacitive sensing works related to our system. Chapter 3 explains the principles of our system, and presents a pilot study to prove its feasibility. Chapter 4 introduces the first version of our system design. Chapter 5 describes out experiments and results on our first design. Chapter 6 explains the second version of our system design, based on the first version. Chapter 7 demonstrates the informal user study on the second version of the system. Chapter 8 summarizes our work and provides further discussions. Finally, Chapter 9 concludes our research work.

Chapter 2

Related Work

2.1 Finger Identification

There are two kinds of method to identify a finger: passive or active. The main characteristic of either method is that a passive method takes advantage of the information collected from the touch action itself, and an active method injects designed information to the touch action on purpose that benefits the finger identification. A passive method, also known as freehand detection, does not add extra sensor or actuator on the hand; an active method requires external instruments on the hand, which reduces mobility but may increase accuracy.

2.1.1 Passive Method

The most common and natural way for finger identification is a computer-vision-based detector, such as a depth camera. Related devices, i.e. Kinect[1] and LeapMotion[2], offer hand skeleton data. Mapping the skeleton data to the spatial coordinate of a screen or an object, we may decide which finger the user is using on a touch.

[1]Kinect, Microsoft, https://developer.microsoft.com/en-us/windows/kinect
[2]LeapMotion, ultraleap, https://www.ultraleap.com/

Glass+Skin [21] examined the finger identification during menu selection action. They used a camera to capture the fingertip positions and directly compare them to the touch events in the device. In this work, they did not mention the accuracy because the finger identification technology is only one minor stage of their user study analysis.

Touch180 [14] installed a fish-eye camera at the bottom of the touchscreen for hand shape capturing, and trained a VGG16 CNN model to map the gesture to the touch input. This method showed an accuracy of 98.56% on single-finger input over a smartphone-sized touch screen.

The geometric properties of human hands are helpful to the identification. Au et al. [1] devised a geometric information based finger identification system on a touch screen when all five fingers of one hand are on the screen. Their system had a 96.0% accuracy and 1 second reaction time over four finger commands. They also mentioned that this method is capable of detecting the left or right hand. In such a system, a user has to put fingers as predetermined shapes so that the system can discover the geometric relations.

Benko et al. [2] showed that muscle sensing on the forearm is applicable to finger identification, achieving 90% accuracy among five fingers with a latency of only 300ms. Although they attached electrodes on the forearm, the method operates by only receiving the signal from the electrodes, therefore it is also a passive method. In this work, they managed to achieve both finger identification and touch pressure sensing, and also provided several applications on user interface interaction.

Fingerprint is a unique feature of each finger that allows identification. Fiberio [13] is an example of sensing the fingerprint on a touch screen. Under a projector screen, it has a fiber optic plate as a light waveguide to illuminate the fingerprint, which is then captured by a high-resolution camera. By classifying the fingerprint, Fiberio could identify several fingers at once.

As for another branch of the methodology, the capacitive images collected from the touch screen carry plenty of features to retrieve the finger identity that touches a certain position on the screen. Le et al. [16] investigated the possibility of identifying images of fingers on a typical capacitive touch screen with CNN, with a 92% accuracy on discriminating left/right thumb. The capacitive method has vast potential of detecting finger features, including finger shape, orientation, tilt, and finger knuckles [23].

TriTap [9] applied a similar method to a smartwatch touch screen, harvesting an accuracy up to 98% over three fingers. Different from the previous work, this work used geometric information extracted from the capacitive image rather than an end-to-end model for classification. The author argued that on a smartwatch, such a method is more practical for a small screen that is hard to manipulate with more than one finger.

2.1.2 Active Method

For the most brutal way of adding information to each finger, optical trackers are reliable for finger identification because they introduce visual differences to fingers that are easy to detect. InfiniTouch [15] created a precise association between the touch pattern on the screen and the gesture by reflective trackers of an OptiTrack [3] system. A CNN model maps the gesture model to the touch screen events, where they found that the classification error mainly depends on the finger placement. Goguey et al. [10] also used colorful markers for a computer vision detector to detect the finger identity during a touch among three fingers.

When we attach wearable sensors or actuators to a hand, there will be broader possibilities for accurate feature extraction. One way to deal with the sensor data is to measure the distance of the fingertip to the touch surface. FingMag [20] used a

[3]OptiTrack, https://optitrack.com/

magnet on the finger and a magnetometer on the device to obtain a 96% identification accuracy among three fingers. The researchers chose a smartwatch as the device, and merged the data collected from the magnetometer and the touchscreen to a CNN network. This method worked even under extreme angles of the finger touching the screen.

DualKey [11] mainly focused on using the index or middle finger for text entry on a smartwatch screen with keyboard. It made use of an IR camera on one of the fingers to measure the distance from the screen and achieved a 99.5% accuracy for a single touch among the first two fingers only.

Another side of the wearable device is to present each finger with a unique invisible signal. A device may identify the finger on touch by explaining the received signal as the associated finger. The work Beauty tech nails [25] stuck RF-ID chips on each fingernail to communicate with an RF-ID receiver when touching it. An RF-ID chip is light, non-invasive, and user-friendly enough for a beauty product while keeps the function of offering each finger a customizable identity.

FingeRing [6] illustrated that the accelerometers on fingers are suitable for finger identification. A good point of using the accelerometer is that the object you are touching will not restrict the performance of finger identification. WhichFingers [17] also used such an opinion of adding vibration sensors to the finger for identification when touching a keyboard or a trackpad.

2.2 Capacitive Touch Sensing

Sensing the touch of a finger while utilizing the human body as an electrical signal conduit has always been a research topic. Human body emits bio-electrical signals and, in the meantime, acts as an antenna that captures the surrounding electronic-magnetic fields over a broad spectrum.

Touché [22] introduced the method to detect the way how human body in interacting with an object via spectrum-based detection, which is the amplitude distribution of body conducted signal. In this work, they used a frequency sweep technique to implement an inexpensive and robust spectrum detector. The result showed that such a system can recognize a rich set of capacitive context, and the accuracy reached 99% after classifier model training.

More work that take advantage of impedance-based capacitive sensing include the work from Suzuki et al. [24] that senses the touch position on the forearm by several groups of electrodes. Another work from Zhang et al. [26] depicted that it is possible to sense high-resolution structural properties using impedance tomography, by measuring the impedance features through scanning several pairs of electrodes.

DiamondTouch [4] also exploited human body to transfer time domain encoded information. The user sits on a chair that emits a certain signal sequence, and the electrical information for recognition flows through the body to the fingertip. A capacitive detector array receives the signal and decodes it into personal identity. This work allows several users to operate a same screen without the need to wear anything while the system keeps tracking each person's touch actions.

Capacitive sensing also applies to on-body positioning technologies. These technologies map the phase, signal magnitude, and magnetic vector information to a spatial coordinate or an angular posture. In fact, FingMag [20] is the example that installed this technology to finger identification. SkinTrack by Zhang et al. [27] used an RF emitter ring and several phase detectors on the wrist to get the position of finger touch on the hand or arm. They reported a 7.6mm standard deviation for positioning error. AuraRing [19] created a magnetic emitter ring and a wristband sensor to reconstruct a 5-degree-of-freedom(DoF) motion of the finger, where the dynamic accuracy is 4.4mm.

Chapter 3

Sensing Principle and Pilot Study

3.1 Sensing Principle

Our method for finger identification relies on the difference of spatial position among fingers, and the features of EM wave propagating through human tissue. [7]

Human body may act as a waveguide for high-frequency signals. The group delay as well as the amplitude changes according to the signal traveling length [26]. If we emit one signal at the frequency f to a certain location near the hand, the phase and magnitude difference of the signal between the source and the fingertip carries the information of from which path the signal traveled. However, this requires an apparatus to be connected directly to the signal source on the body - restricting the mobility of the user's hand.

If we emit two signals at different locations on the skin, at the precise frequency of f and $2f$, as the distance from each signal source to the fingertip varies among fingers, the signal received at the fingertip should be a combination of f and $2f$ components with a phase and magnitude difference - each component carries the information of which path it chooses. By splitting and measuring the features of the signal - done at only the fingertip, we may overcome the restrictions of using a single frequency.

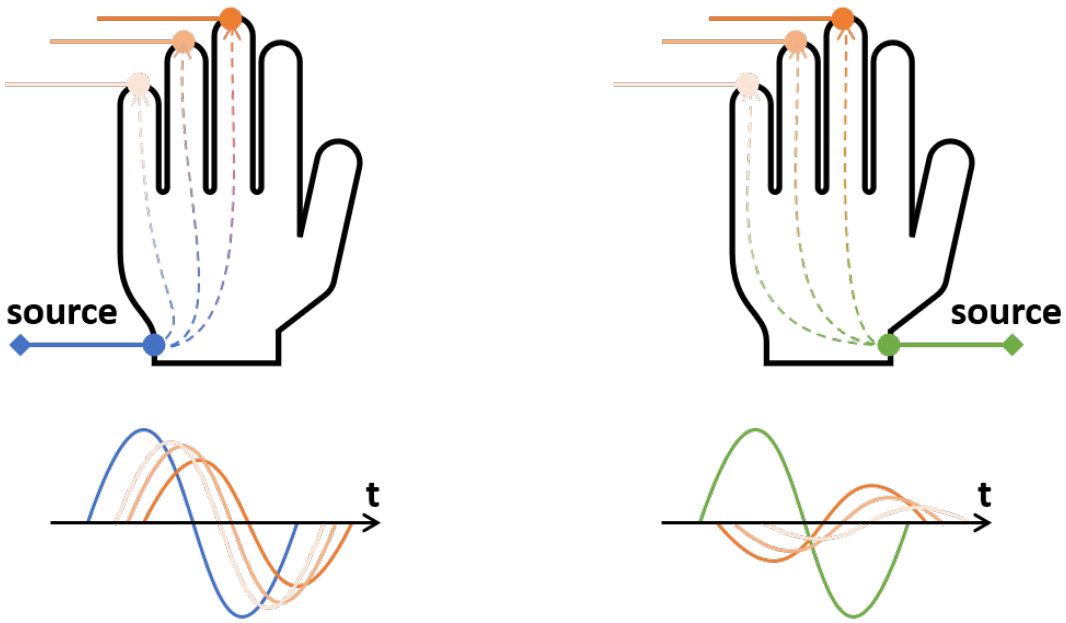


Fig. 3.1 The schematic of signal propagating through the hand. Different routes result in different phase difference (group delay) and attenuation

The f value should be high enough to show apparent phase and magnitude difference among different signal routes, but not so high that the hand will act as an antenna that escapes the signal.

We did a theoretical calculation on whether such a technique produces detectable signal features at the dimensions of a hand.

According to electrophysiological measurings from human body tissues [7], the bone cortical and fat has the lowest relative permittivity ε_r of about 15 to 20 at the frequency f of 100 MHz. The conductivity of the finger tissue σ at this frequency is around 0.07 S/m. We assume that human body tissues possess a permeability of the water, which is near the absolute permeability constant. This gives an EM wave propagating attenuation constant α and phase constant β of:

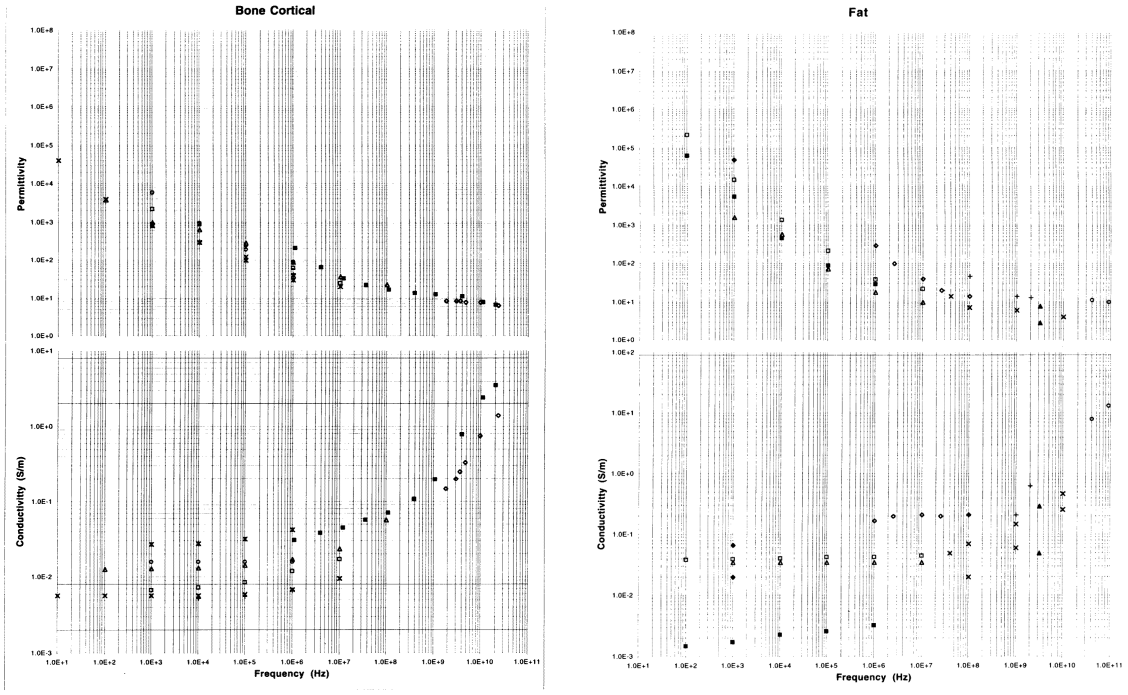


Fig. 3.2 The permittivity and conductivity values of bone cortical (left) and fat (right) at different frequencies.[7]

$$\alpha = \omega \sqrt{\frac{\mu_0 \varepsilon}{2} \left(\sqrt{1 + \left(\frac{\sigma}{\omega \varepsilon} \right)^2} - 1 \right)}$$

$$\beta = \omega \sqrt{\frac{\mu_0 \varepsilon}{2} \left(\sqrt{1 + \left(\frac{\sigma}{\omega \varepsilon} \right)^2} + 1 \right)}$$

where μ_0 is the permeability constant, ε is the absolute permittivity of the tissue, and ω is the angular frequency of the signal. We have:

$$2.82\,\mathrm{Np/m} \leq \alpha \leq 3.17\,\mathrm{Np/m}, \qquad 8.71\,\mathrm{rad/m} \leq \beta \leq 9.77\,\mathrm{rad/m}$$

which means an attenuation between -0.25 to -0.28 dB/cm and a phase difference of 5.0 to 5.6 °/cm [7, 5], desirable for measuring with feasible hardware. This frequency

gives a half wavelength of around 35 cm, much longer than an ordinary palm so that the signal will not radiate efficiently into the air. Of course, due to the individual difference of electrophysiological properties between the hands, and the limitations of measuring conditions, we mainly focus on whether there will be a detectable difference for our finger identification. We first used existing equipment to examine our idea.

3.2 Pilot Study for Sensing Principle Validation

To evaluate the feasibility of our theory, we designed an experiment to test the estimated phase difference at the fingertip. We chose to examine the phase part because it is the most resistant to environmental noise. The magnitude measuring involves too much exterior interference, and we suppose that if the phase detection works, the magnitude detection will work as well, which allows us to enter further studies. This experiment involves a signal emitting wristband as the wearable hardware part. If our theory mentioned in the previous section works, emit two signals at different sides of the wrist, and we will be able to get a compound signal at every certain fingertip. We may collect the signal waveform at the fingertip by instruments, and calculate the phase difference from the waveform mathematically.

3.2.1 Hardware Design

According to the calculations in the previous section, and the signal conditioning approaches we have, we experimented under the condition of $2f = 98\,\text{MHz}$. We made a wristband generating rising-edge-aligned 49 MHz and 98 MHz signals at a peak-to-peak voltage of 5 V.

The emitter is a wristband to be tightened at the place outside the carpal bones over the skin. The two electrodes connected to the two output ports of the signal

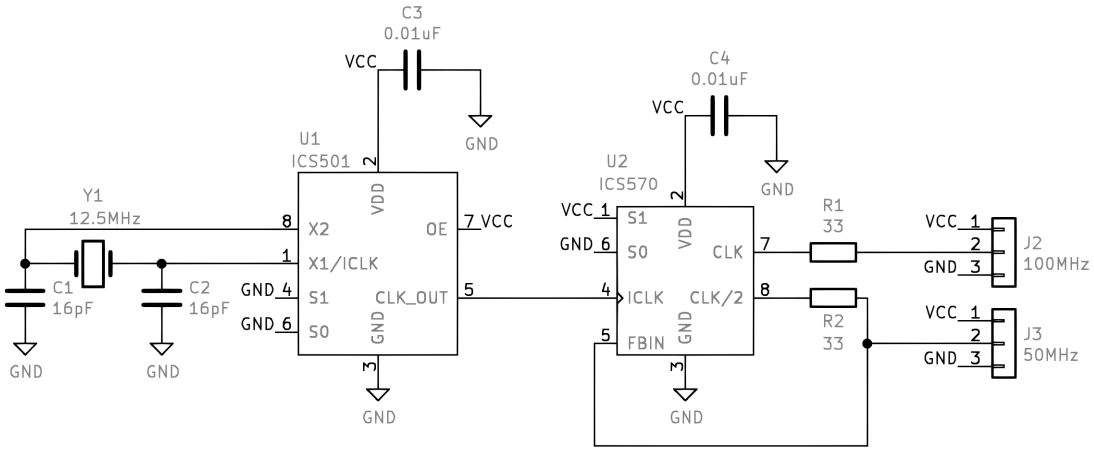


Fig. 3.3 The circuit diagram of the wristband used in the pilot study

generator are placed over the skin so that each electrode is on either left or right side of the wrist.

To form a stable signal generator, we used an ICS501 clock generator IC oscillating at a stable frequency at around $f = 12.25\,\text{MHz}$, and an ICS570 clock buffer IC to multiply the frequency to 98 MHz. The phase-locked loop (PLL) structure assures the rising edges of the 49 MHz and 98 MHz rails to occur at the same time, as Fig.3.3 shows. A crystal determines the clock frequency and allows us to tune the performance of the clock generator. We can observe a waveform of generated signal as Fig.3.4, which shows a raw 5 V peak-to-peak output voltage swing (although there show some attenuations according to the limited oscilloscope bandwidth of 50 MHz).

We power the emitter part with a Li-ion battery to separate it from noises introduced from the AC power line. To get a better separation between the two signal rails, we modified the source termination resistor recommended in the ICS570 datasheet from 27ohm to a direct short, which reduced the load effect of either signal rail to another. Notice that, in fact, this emitter configuration generates a square wave on each rail with certain distortions among harmonic frequency components due to the equivalent capacitive load of the hand tissues. This does not violate our target of

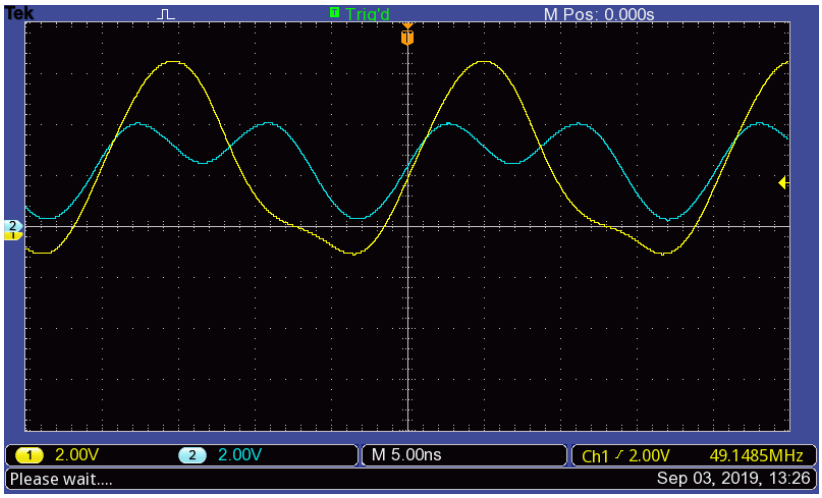


Fig. 3.4 The output waveform of the wristband measured by an oscilloscope. Yellow: 49 MHz signal. Cyan: 98 MHz signal.

generating two signals of f and $2f$, as a square wave only consists of odd-order harmonics so that the two rails contains entirely different frequency components, and the harmonics over order 3 (frequency over 147 MHz) could not pass the filters in the detector part, which we will explain later in the detector chapter.

3.2.2 Experiment Design

We recruited 1 participant as the tester of the wristband part in this pilot study. In this section, we will describe how do we evaluate the feasibility of the phase-measuring idea using an oscilloscope.

3.2.3 Experiment Process

We use a 2 cm × 2 cm PCB board with copper foil as the touchpad to collect signal from the fingertip in this experiment. The copper foil is AC coupled to the oscilloscope probe and ground at the impedance of 100 pF + 1 kOhm (turning frequency at 1.6 MHz), which efficiently filters the AC power line noise.

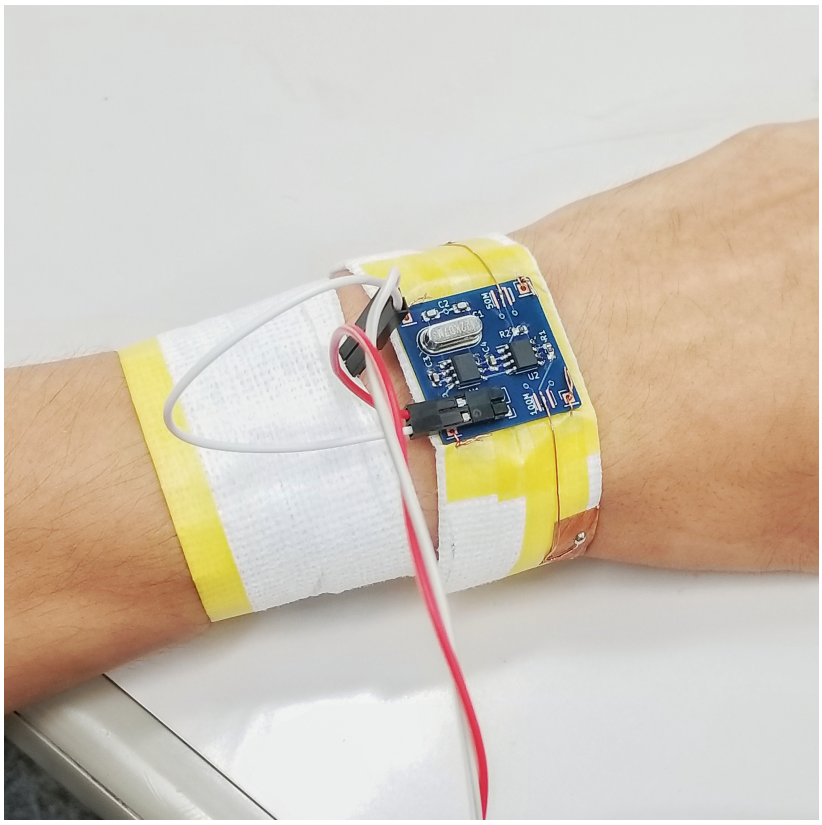


Fig. 3.5 The participant is wearing the wristband. The blue PCB is the signal generator, connecting to two electrodes (only one shown in the figure) by copper wire. The signal generator is DC powered by the red-white cable pair.

We ask the participant to wear the wristband during the experiment. In order to maintain the environmental conditions to be stable during each round, we bound the wristband prototype tight at the wrist, which may cause certain uncomfortableness. During the experiment process, the participant presses a certain finger on the touchpad, keeping the shape of the hand relatively stable, while we save the waveform on the oscilloscope into a USB disk, and record the corresponding finger identity (thumb to little finger). The order of the finger is decided by the participant solely, but has to be in total 20 times for each finger, 100 waveforms in total. Each waveform is a list of normalized voltage values in time sequential order.

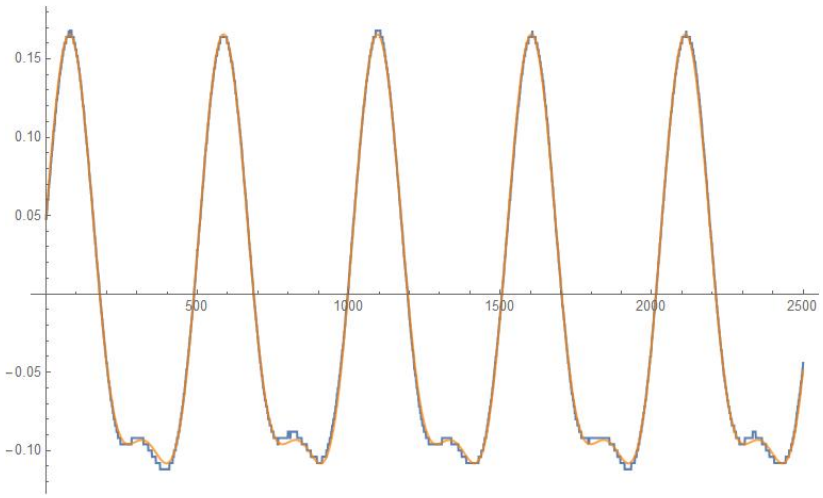


Fig. 3.6 The data recorded by the oscilloscope (blue) and the function fitting result (yellow)

3.2.4 Data Analysis

To extract the phase difference information from the waveform, we assume that the voltage signal follows the model of:

$$f(t) = a\sin(\omega t + c) + b\sin(2\omega t + d) + q$$

where ω is the normalized angular frequency of the signal. a and b are the magnitudes of the two basic frequency components. c and d are the phase of the two basic frequency components. q is the DC component of the signal. Notice that this model omits the harmonic frequency components over 3rd order because they can be treated as noises during the model fitting and that uncorrelated components would not contribute to the model parameter.

We use `FindFit` function in Mathematica to fit the model to each waveform sequence. `FindFit` function calls a least squares fitting process, and leaves the orthogonal components in the data as residuals. The phase difference of the two basic frequencies has the form of $2c - d$, modulo- 2π to clamp the value into a single period.

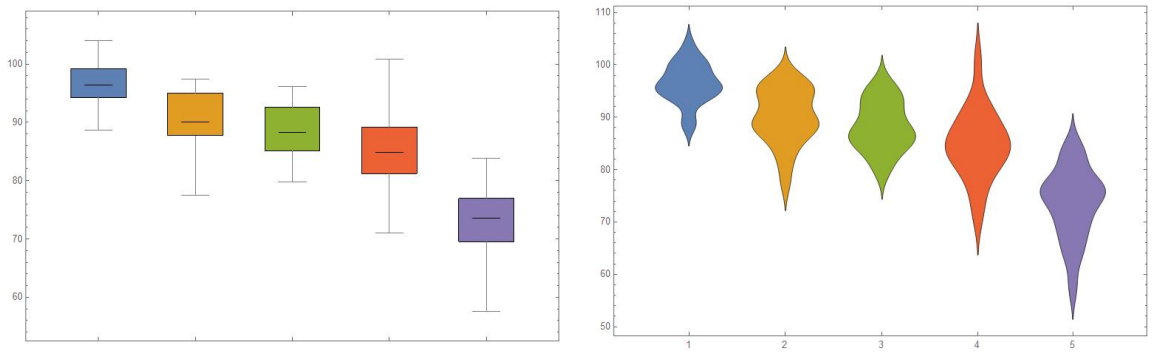


Fig. 3.7 The whisker box chart (with 1one standard variance error bar) and histogram for the data of each finger identity. Each chart from left to right: thumb to little finger. The Y-axis is the $2c - d$ value with an amplification factor, without a unit.

For the calculated phase difference values of a same finger identity, we gather them and plot a labeled histogram for each finger.

3.2.5 Results

The histogram of finger identity vs. phase difference distribution in this experiment is as Fig.3.7. We can see that there is a distinguishable average phase difference fluctuation among finger identities, and that the difference value follows the spatial position of the finger - whether it could be utilized for discriminating the identity remains to be checked by more decent hardware.

However, we also notice that there exist some "double peaks" in the distribution. This may result from the wristband slipping around the skin near the wrist during data collection, which causes unstable signal route. This requires us to design a more solid wristband that doesn't slip too much on the skin.

Due to the result of phase difference checking, we may also infer that magnitude difference shall follow similar rules, that provide us information to classify the finger identity results.

3.3 Summary

We assumed that we may use the relative phase and magnitude difference of the RF signal to identify which finger the user is touching a touchpad with. Through a pilot study, we discovered that the phase difference of two frequencies in a signal is capable of determining the identity of the finger where the signal is emitted.

Chapter 4

System Design Version 1

4.1 Detector Configuration

The detector determines the finger by checking the phase and magnitude differences of frequency components in the incoming signal. On receiving the signal from a finger, the detector part first split the signal into the f and $2f$ frequency rails. A multiplier stage then doubles the frequency of the f rail into $2f$ for comparison. Afterwards, a phase-magnitude detector may measure the phase and magnitude difference of the original $2f$ rail against the doubled f rail. By performing analog-digital conversion on the phase and magnitude detection result, we may further examine the features of the signal source, which indicates the corresponding finger.

According to the properties of the Lissajous figures, to fully utilize the precise twice-the-frequency relation between two frequency components, we have to look into the working principle of the detector. The phase-magnitude detector works by an analog multiplier core. Suppose we have two sinusoidal signals of frequency ω_1 and

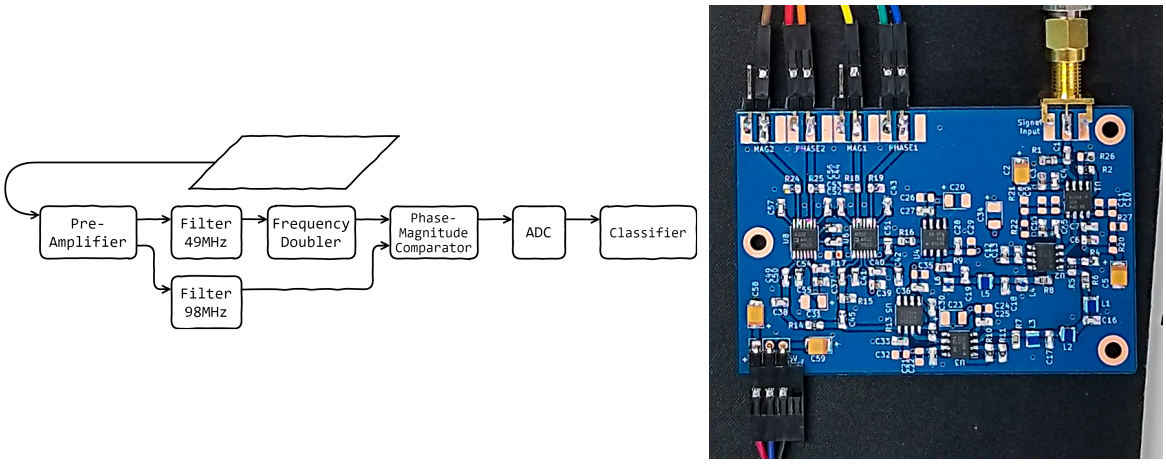


Fig. 4.1 Left: the flow chart of the detector functional blocks. Right: the photo of the detector PCB.

ω_2, while the latter is shifted with a phase of φ, the detector generates a following analog function:

$$\sin(\omega_1 t)\sin(\omega_2 t+\varphi) = \frac{1}{2}\cos[\varphi-(\omega_1-\omega_2)t] - \frac{1}{2}\cos[\varphi+(\omega_1+\omega_2)t] \tag{4.1}$$

Notice that the second cosine part is a signal of very high frequency, which can be efficiently filtered out by a capacitor. The first part is stable when, and only when, $\omega_1 = \omega_2$, so that it is unrelated with time variant t. By doubling the f rail, we may get a signal with the precise $2f$ component, thus providing the analog function with two signals of exactly the same frequency, suitable for phase detection.

The detector hardware consists of 5 stages: pre-amplification, filtering, multiplication(frequency doubler), phase-magnitude detection, and analog digital conversion (ADC) stage. Through careful circuit design, the bandwidths of all stages extend to the range that fully covers the input signal specifications to preserve signal integrity. The input interface of the detector is a 50 Ohm terminated RF connector. The previous 4 stages are assembled on a piece of PCB, powered by a ± 5 V DC power module. In the following paragraphs, we will introduce the implementation of each stage.

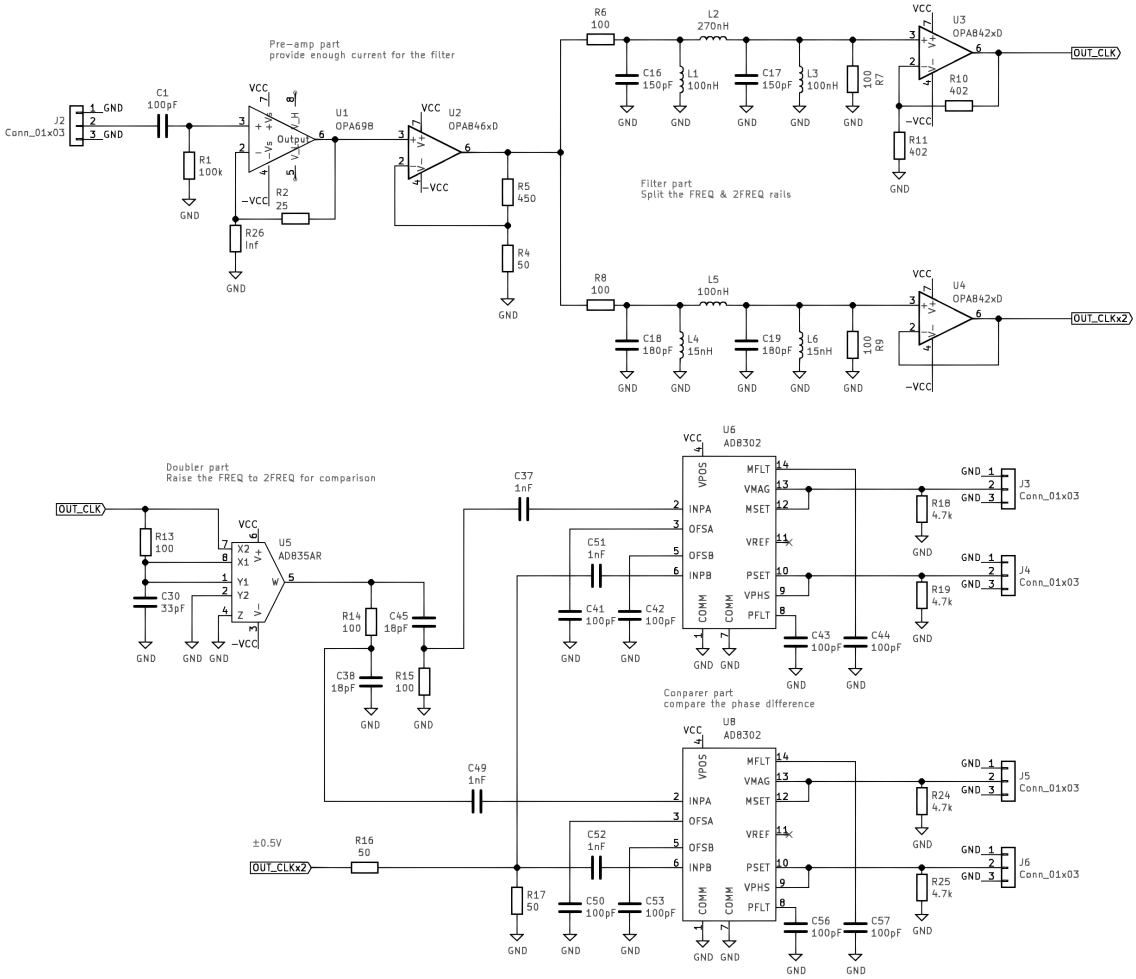


Fig. 4.2 The circuit diagram of the detector, first version

The pre-amplification stage acts as an amplifier and an impedance converter. This stage is made of a high-gain stage and a power-boost stage. Before the high-gain stage, a capacitor in series with the load termination resistor forms a high-pass filter cutting off at 1 MHz that filters out low-frequency noises, including the 50/60 Hz power line signal. Suppose we have an input signal of 10 mVpp. The high-gain stage is an OPA847 amplifier tuned to a gain of A = +20 V/V, which means a maximum 1.5 Vpp signal to drive the power-boost stage. The power-boost stage uses an OPA698 amplifier as a voltage follower to deliver enough signal power to the filtering stage.

The filtering stage splits the amplified signal into the 49 MHz and 98 MHz rails by two 3rd-order band-pass passive filters, where we expect to two rails to achieve over 20 dB separation. Fig.4.3 shows the predicted and measured filter Bode plot by a spectrum analyzer, proving that the filter design is successful. The aberration from the theoretical value may derive from the output resistance of the pre-amplification stage, or the impedance mismatch of the electronic components. The band-pass structure guarantees the frequency components out of this range, either higher or lower, to leave the following stage, in order not to disturb the detection result. Each filter follows an additional OPA820 amplifier as the output impedance buffer to drive the following stages.

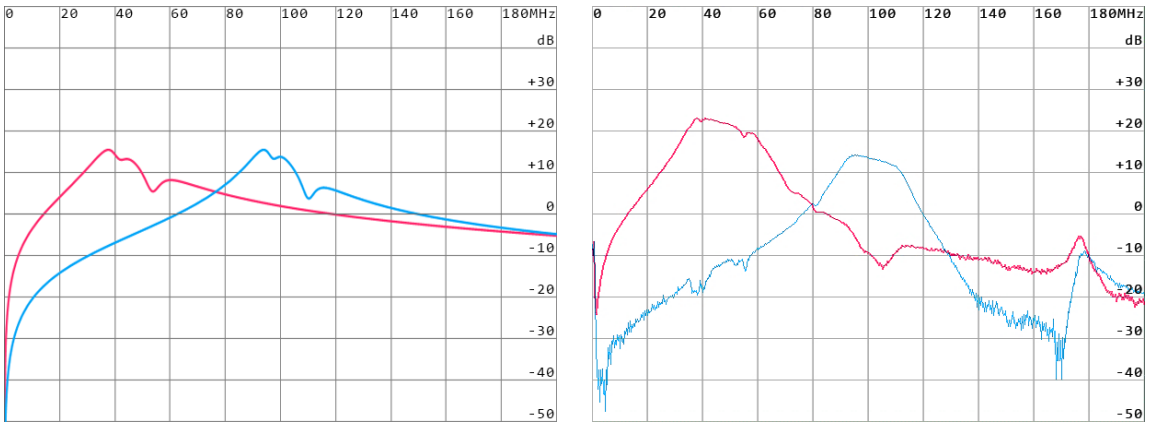


Fig. 4.3 The calculated (left) and measured (right) Bode plot of the filtering stage. Red: 49MHz channel, Blue: 98MHz channel

In order to get a second-order harmonic from the 49 MHz rail, a couple of ways could be used: passing the signal through a biased diode, introducing full-wave precise rectifier, or using a frequency multiplier, etc. As long as there is a doubled frequency component in the output signal, and the another rail (98 MHz rail) contains pure 98 MHz signal, recalling formula (4.1), we may ignore the other components and retrieve the phase difference information. Considering the frequency limit and signal amplitude restrictions, we choose the multiplier plan, which brings a certain draw-

back: the output magnitude difference isn't proportional to the input signal anymore (In fact it becomes quadratic), but this affects little to the final result.

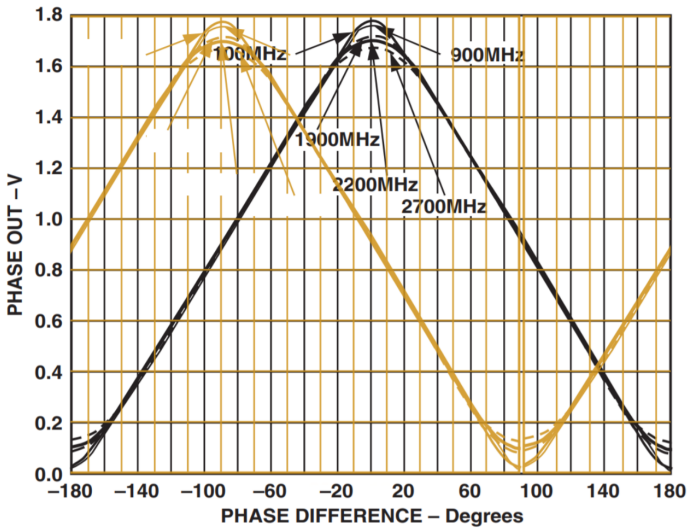


Fig. 4.4 The transfer properties of two phase-magnitude detectors (AD8302) in our circuit design. The black and yellow lines indicate the phase difference - output voltage relationships of two detectors placed at a 90 degree phase shift between them.

In the frequency doubler design, the filter stage feeds the 49 MHz rail to an AD835 multiplier connected as a frequency doubler. From here, a doubled 98 MHz signal comes out, with a fixed phase shift relative to the input signal. This doubled signal then goes to the detector part along with the filtered-out 98 MHz signal. We used the AD8302 analog RF phase-gain detector IC as the detector stage, which outputs analog voltages proportional to the relative logarithmic magnitude gain and the phase difference. According to the datasheet, an AD8302 can only distinguish the phase difference within 0° to 180°, and thus, we used two AD8302s in parallel. To make the best of each AD8302 phase detection range, we devised a pair of RC networks to further split one of the 98MHz rails into two sub-rails whose phase is orthogonal to each other, and transmit them to the two detectors. The transfer properties is shown as Fig.4.4. This design enables us to retrieve the whole 360° phase information.

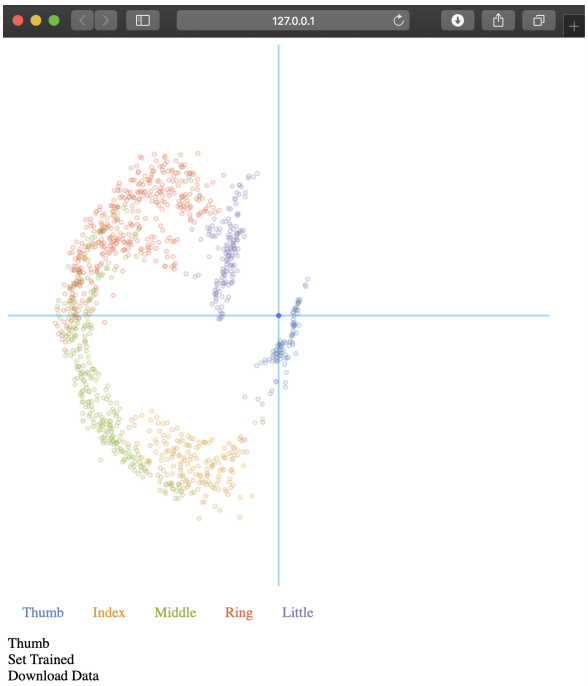


Fig. 4.5 The webpage for data collection and visualization in our system. Each colored dot is a data point of one single touch. The distance to the origin is the magnitude difference between two frequency components. The angle with x-axis is the relative phase difference. The color of the point represents the ground truth of the finger identity which we recorded by the buttons below the graph.

Finally, several additional rail-to-rail amplifiers expand the detector outputs into a full $0 \sim 5\,\mathrm{V}$ range, suitable for precise ADC operation. We use an Arduino Pro Mini module for the ADC application in order to read the values on a PC.

4.2 Software Configuration

We connect the Arduino serial port to a Python server to collect and filter data, and pass it on to the analysis and visualization part. The server reads and buffers data from the USB port, and distribute a sequence of data on a local network.

The data analysis and visualization part is a webpage (Fig.4.5) based on JavaScript. On the webpage lays a polar coordinate graph indicating the relative magnitude and

phase difference. There are also several buttons representing a finger identity class for data logging. Inside the webpage script, we implemented a k-nearest neighbors (k-NN) classifier taking the phase and magnitude differences as input, and the finger identity (1 to 5) as the output. There are also buttons controlling the training and examination process of the classifier.

Chapter 5

User Study 1

5.1 Experiment Process

The experiment consists of two stages: training and validation, both operated on the data analysis and visualization UI. For the touch target object, we chose a metal pad connected to the detector with a 50Ω transmission cable. We tightened the wristband to the wrist of the participant before the training stage. During the experiment, we asked the participant to keep the wrist firm so that the hand shape remains relatively solid, and move the palm by the arm.

During the training stage, the participant placed each of the five fingers on the metal pad by turns, while we click to save the finger identity number. We collected 50 data points in total, while the classifier trained an 8-NN model for identification.

During the validation stage, the participant placed random fingers on the metal pad, meanwhile recording the ground truth data on the UI. We collected a test dataset of 26 data point per finger, which is 130 data point in total.

After collecting the test dataset in the web UI, we downloaded the dataset containing the classification results and ground truth values of each data point. We printed

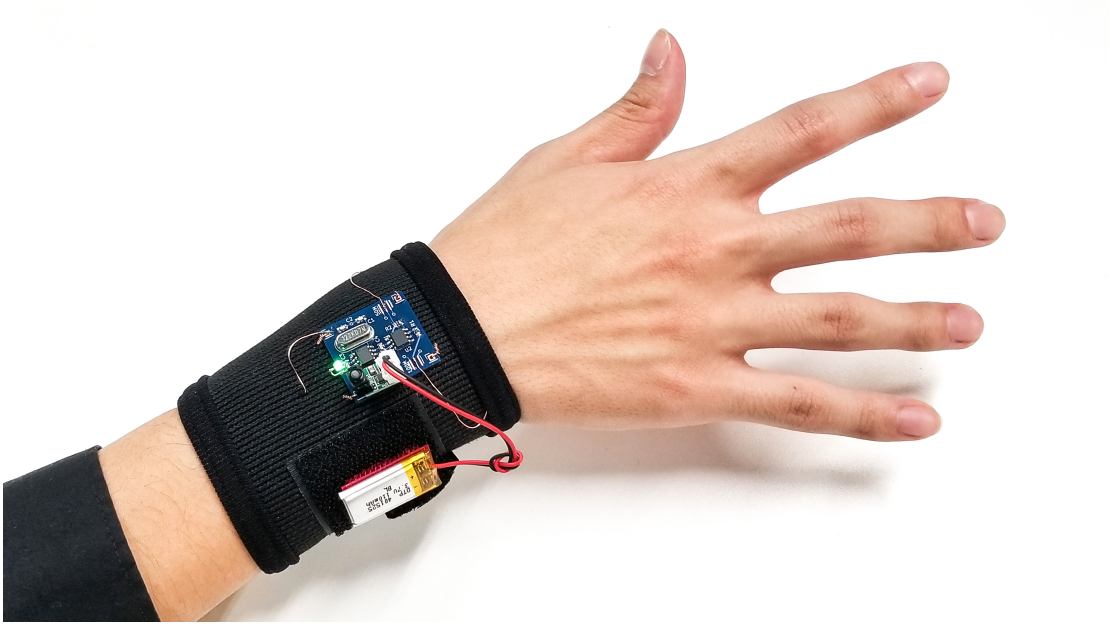


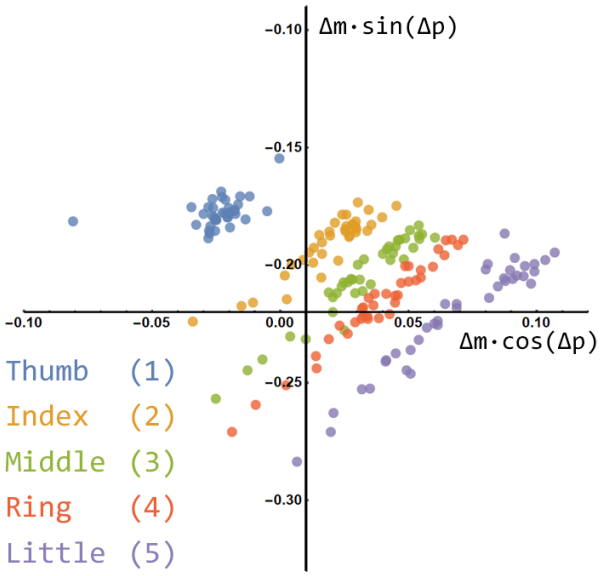
Fig. 5.1 The user is wearing our wristband (first version) during the experiment

out the visualization from the UI and the confusion matrix of the test dataset as the evaluation of our system.

5.2 Result

The visualization of the collected magnitude and phase difference data in 2D space and the classification results in confusion matrix format are as follows. We cropped out the valid part from the polar coordinate in the UI:

From the visualization 5.2 we could see that the data of the same finger gathered in clusters, which indicates that the data followed certain distributions. According to the test data, we achieved a 92.2% accuracy and 91.9% F1 score among five fingers. Noticing that the error mainly comes from the middle and ring finger, if we only focus on the first three fingers (thumb, index, middle), the accuracy would rise to 97.6%.



Pred> / GT	1	2	3	4	5
1	26				
2		24	2		
3			25	1	
4			5	20	1
5				2	24

Fig. 5.2 Left: The visualization of the collected data points from each finger (test set). Right: The confusion matrix of the test result.

5.3 Summary

Based on a small test dataset, our capacitive sensing method has the ability to distinguish between different fingers. We will move to a newer design and a more decent analysis in the following chapter.

Chapter 6

System Design Version 2

From the previous studies, we know that the magnitude and phase difference do contribute to finger identification, and there is potential for a system efficiency improvement. Based on the experience we acquired from the experiment on system design version one, we proposed system design version 2, which features the following properties:

- Using a higher basic frequency of $f = 72\,\text{MHz}$, which is still below the signal emission frequency threshold of the arm so that the arm does not act as an antenna, but achieves better separation between the two frequency rails. This frequency makes the most capability of the electronic components we have.
- Using surface acoustic wave (SAW) filters as the frequency splitter.
- Adding an absolute signal magnitude detector to the design.

6.1 Emitter Configuration

We designed the second version of emitter as a battery-powered block that generates two rails of 72 MHz and 144 MHz signals. To maintain a stable voltage of 3.3 V from the battery, whose voltage will decrease as it discharges and can be either lower or

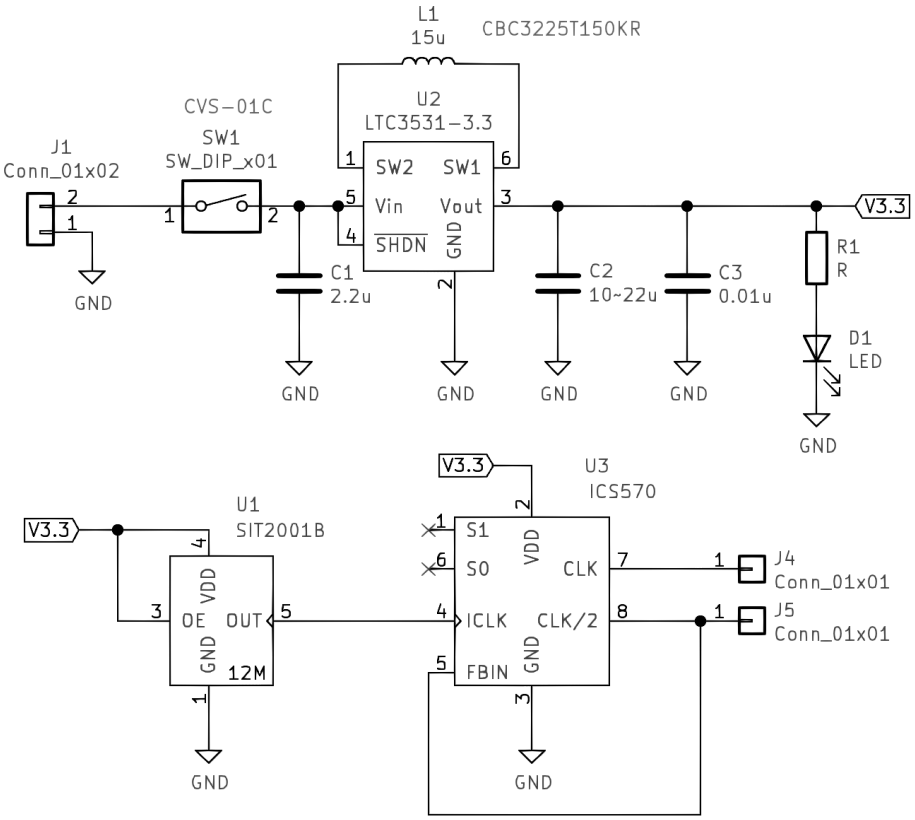


Fig. 6.1 The circuit diagram for the second version of the wristband. J1 is the battery power input.

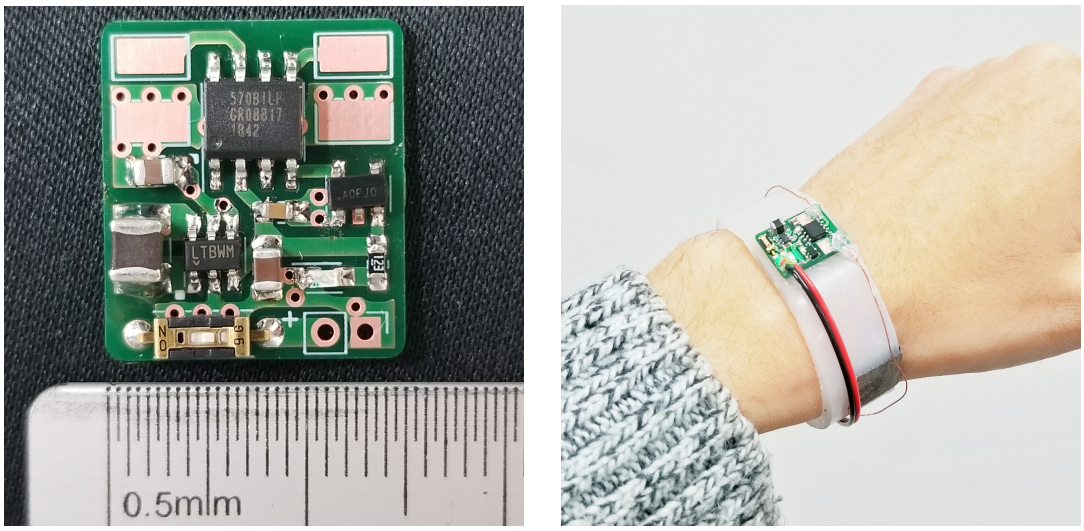


Fig. 6.2 Left: The complete PCB board of the emitter. Right: The assembled wristband wearing on the wrist. The battery is on the back side of the wrist. (not shown)

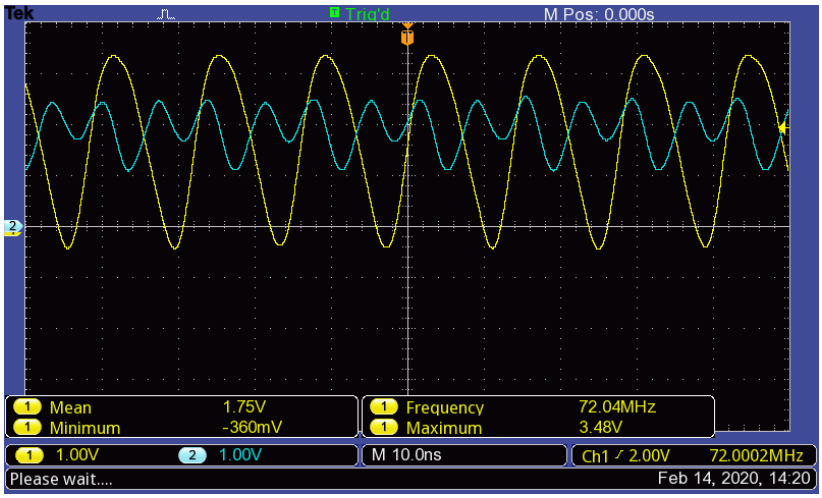


Fig. 6.3 The waveform observed from the signal emitter output. Yellow: 72 MHz rail. Cyan: 144 MHz rail.

higher than the desired voltage at any time, we use an LTC3531 IC as the DC-DC power manager. The working frequency of LTC3531 is only several kilohertz - much lower than our target signal frequency, which causes no spectrum conflict.

The signal generation part is extremely straight forward: we use an SIT2001B oscillator to feed a 12 MHz square wave signal to an ICS570 IC, configured as a 6x/12x phase-locked-loop frequency multiplier. We omitted the termination resistor recommended in the official datasheet similar to the first hardware version, as the termination resistor will degrade the frequency component separation when applying signals to the skin. The raw output waveform is like Fig.6.3 with no output load.

Moreover, we cast the model of the new wristband out of silicon gel: it enlarged the friction of the band to the skin greatly so that it won't slip during the experiment even if the user twists the wrist. The two electrodes are placed on the either side of the wrist and are connected to the PCB board with elastic metal wires as strain reliefs, which enable stretching the wristband to fit the circumferences of different wrists.

6.2 Detector Configuration

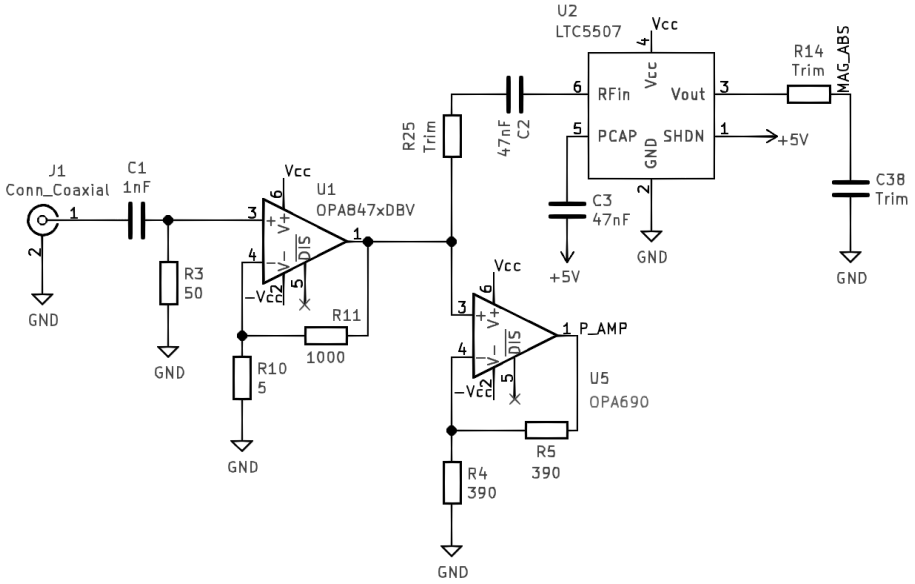


Fig. 6.4 The pre-amplification stage of the detector's second version

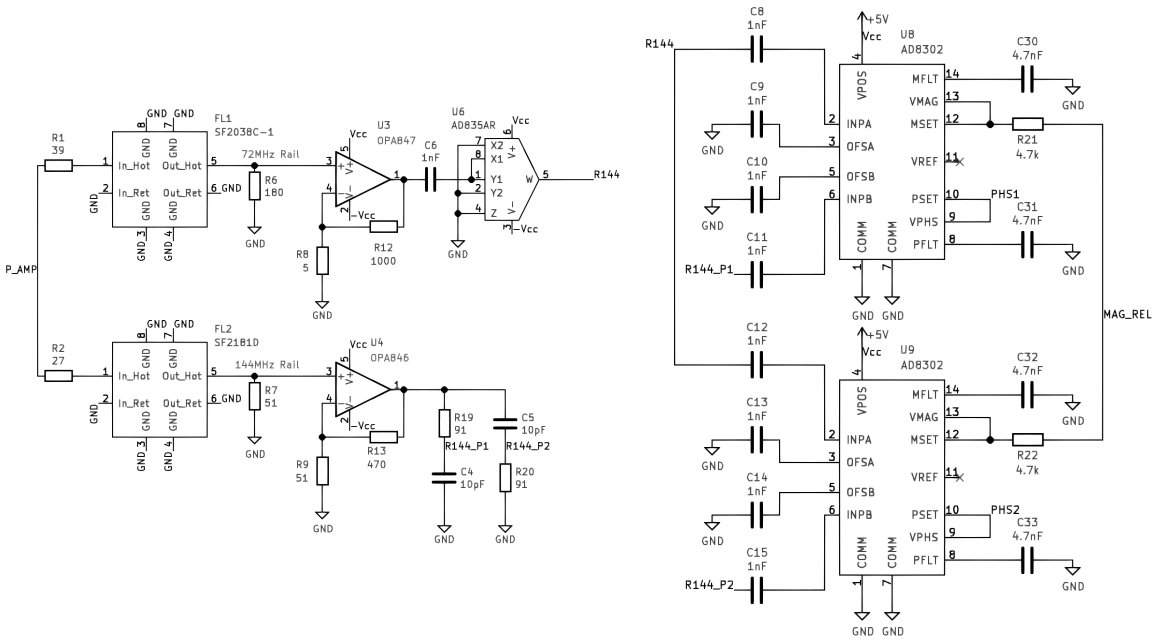


Fig. 6.5 The frequency-splitter and doubler stage (left) and the phase-magnitude detection stage (right) of the detector's second version

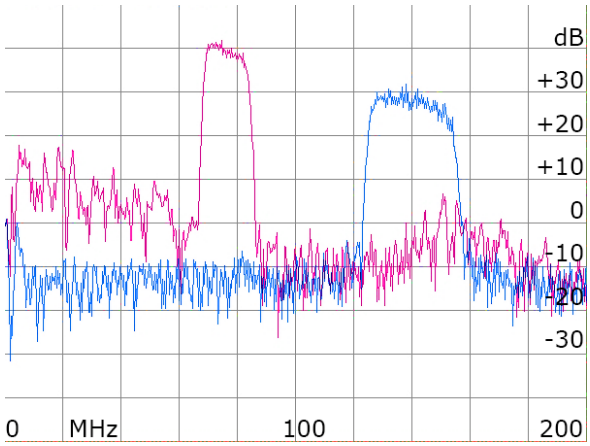


Fig. 6.6 The Bode plot of the SAW component filtering stage, after output buffering. Red: 72MHz channel, Blue: 144MHz channel.

The second version of detector also consists of the four stages as the first version does. We mainly introduce the differences.

At the pre-amplifier stage, we inserted an LTC5507 RF power detector to measure the absolute input signal amplitude. The LTC5507 is sensitive to frequencies over 100 kHz to 3 GHz. This gives us a useful condition to detect whether the user is touching the detector touchpad or not, and provide a convenient channel to measure and fine tune the gain of each stage.

At the frequency splitter stage, we selected two SAW filters that allow either 72 MHz or 144 MHz signal to pass through a narrow band efficiently. A SAW filter conducts RF signal through a layer of piezoelectric material as mechanical energy, and may filter out certain frequencies at extremely high selectivity. Each filter is terminated double sides using resistors, as a compromise of energy transfer efficiency and circuit complexity. The measured transmission properties of the two channels are as Fig.6.6, including the amplification factors of the output buffers.

At the frequency doubler stage, we directly send the filtered 72 MHz signal into the AD835 multiplier with AC coupling to minimize the loss during multiplication. There will be a DC component in the output signal, but as the output signal is AC coupled

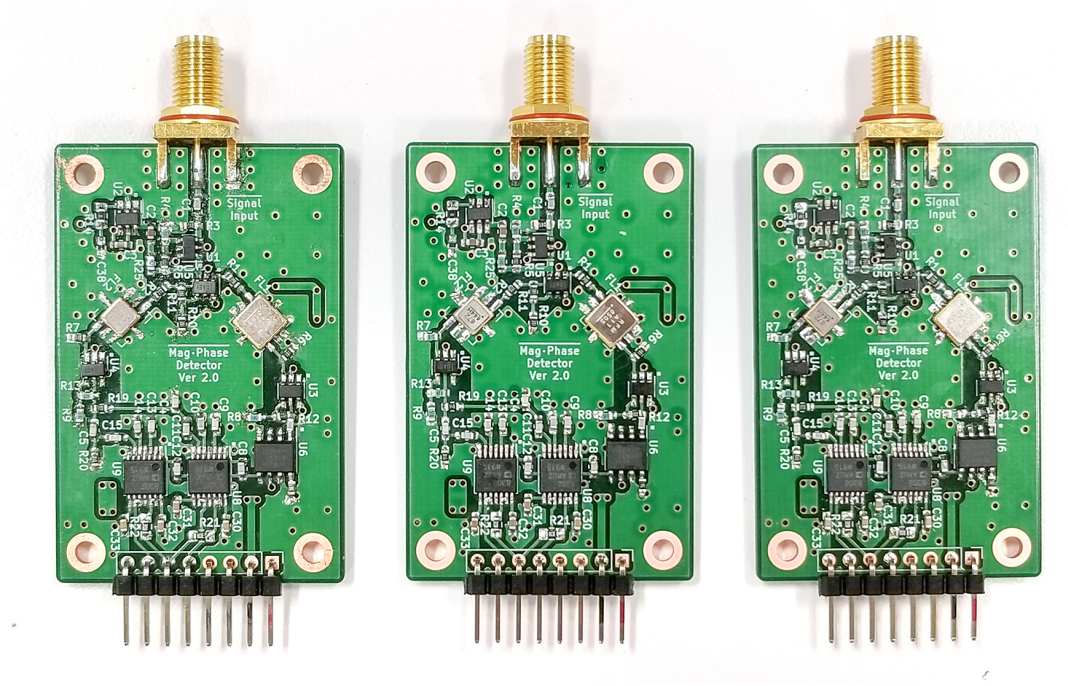


Fig. 6.7 The fully assembled PCB of the detector part. We prepared three pieces for the evaluation. The square shaped metal pieces are the SAW filters used in our design.

to the phase-magnitude detector, such a component is negligible. Also, moving the phase splitter to the original 144 MHz rail also helps enhance the signal-noise ratio by fewer conditioning stages.

At the phase-magnitude detector stage, we connect the two magnitude measuring output directly with resistors. We may now acquire 4 measurings from the detector: 2 orthogonal phase difference value, 1 absolute magnitude (from the RF power detector), and 1 relative magnitude difference value (in logarithmic scale). These signals are as well amplified, fed to the Arduino and transferred into digital values.

6.3 Software Configuration

For the new software part, we modified the classifier, data visualization UI, and training data collection process.

6.3.1 Classifier

We implemented a 2D Gaussian classifier to replace the k-NN classifier used in the previous version. After we notice the cluster's shape of data from each finger identity, we concluded that a bivariate Gaussian model might be more suitable for this task. Our Gaussian classifier takes the raw phase and magnitude difference values as 2 input dimensions, and try to find the $\mu_x, \mu_y, \sigma_x, \sigma_y, \rho$ parameters that fits the model

$$p = p(x, y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(x-\mu_x)^2}{\sigma_x^2} + \frac{(y-\mu_y)^2}{\sigma_y^2} - \frac{2\rho(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y}\right]\right)$$

for each cluster of a finger identity. Practically, we could calculate $\log 2\pi p$ for the reason that only the relative order matters during prediction.

According to the maximum likelihood estimation (MLE) for a Gaussian model, we have:

$$\begin{aligned}
\mu_x &= \overline{x} \\
\mu_y &= \overline{y} \\
\sigma_x^2 &= \overline{x^2} - \mu_x^2 \\
\sigma_y^2 &= \overline{y^2} - \mu_y^2 \\
\rho &= \frac{\overline{xy} - \mu_x\mu_y}{\sqrt{\sigma_x\sigma_y}}
\end{aligned}$$

where the covariance matrix part is:

$$\mathbf{\Sigma} = \begin{bmatrix} \overline{x^2} - \mu_x^2 & \overline{xy} - \mu_x\mu_y \\ \overline{xy} - \mu_x\mu_y & \overline{y^2} - \mu_y^2 \end{bmatrix}$$

One thing to notice is that the phase difference value is a self-wrapping scale of $0 \sim 2\pi$, where we normalize it into $0 \sim 1$ in our algorithm. The value space shapes

like a cylinder, and a continuous change through the preceding 0 or 1 border along the x-axis will cause a jump in a Cartesian coordinate. We use the following modified algorithm to deal with this discontinuity: calculate the horizontal distance of two points by the smallest one among three values - the horizontal distance between one point and another, one point and the position of another point in the previous cycle (minus one), and one point and the position of another point in the following cycle (plus one). When calculating the μ_x value of a class, the new value might exceed the $0 \sim 1$ range due do the addition or subtraction, thus we always take the fractal part of the result as the new μ_x value to make sure that at any time, either any point or the center of a class is within the 0 to 1 range.

The complete updating algorithm is:

Algorithm 1: Updating the μ_x value of a class during training

Data: a new x value, previous center μ'_x, total count N and sum Σ_x of previous x values

Result: The new μ_x value

$x\text{List} \leftarrow [x-1, x, x+1]$;

$x_0 \leftarrow \underset{x' \in x\text{List}}{\operatorname{argmin}} d(x', \mu'_x)$;

$\mu_x \leftarrow \frac{\Sigma_x + x_0}{N+1}$;

$\mu_x \leftarrow \mu_x \bmod 1$;

6.3.2 Data Visualization

The new visualization treats the phase and magnitude values in a Cartesian coordinate - x-axis for phase and y for magnitude , and plots the error ellipse of each finger identity cluster.

From the previous sample UI we could also see that, in some cases, data points of a same cluster distribute on the two edges of the graph. To perform Gaussian MLE under such a condition, we have to update the μ_x and μ_y position according to the nearest point. The distance across the phase axis shall be:

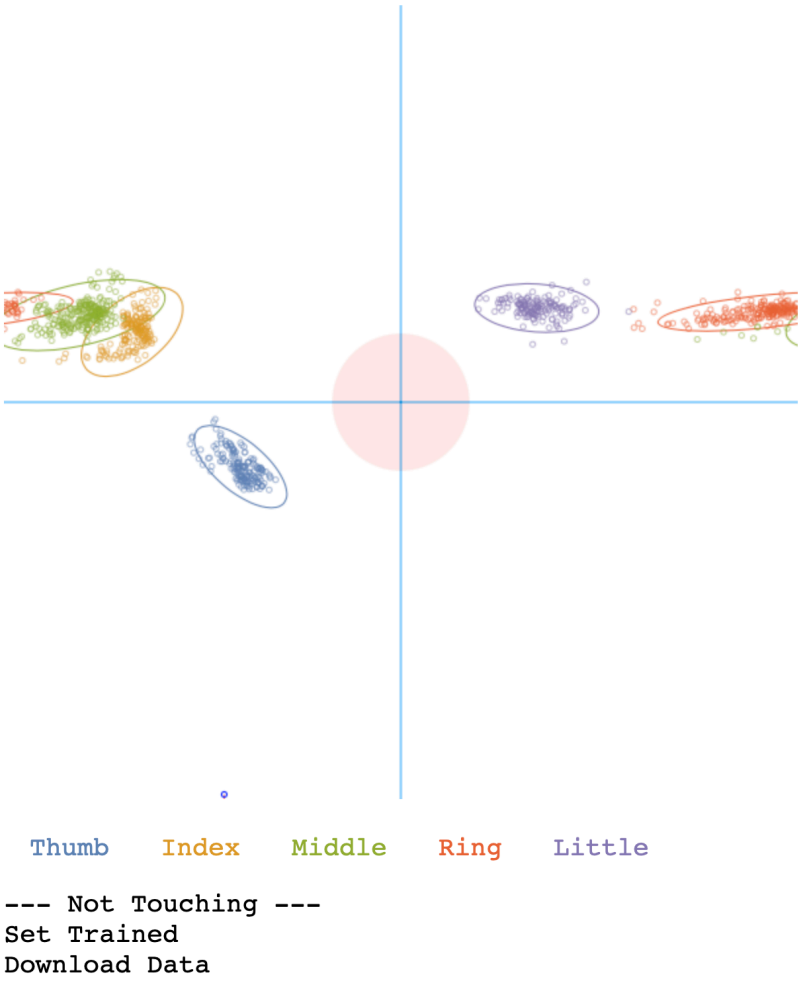


Fig. 6.8 The second version of the web UI

$$d(u, v) = |(u - v + 0.5) \bmod 1 - 0.5|$$

to make sure that the x-distance falls into the range of $[0, 0.5]$. Every time when a new data point comes, we calculate the distance from the data point to the center via the previous formula, and update the position of the center so that it always drops in the range of $[0, 1)$. The same distance formula applies to prediction process.

The next modification to the interface is that we added an error ellipse to demonstrate the confidence intervals of each class. The parameters of the ellipse should follow:

$$
\begin{aligned}
O_x &= \mu_x \\
O_y &= \mu_y \\
r_1 &= \sqrt{K\lambda_1} \\
r_2 &= \sqrt{K\lambda_2} \\
a_{r_1,x} &= \arctan \frac{\Sigma_{xy}}{\lambda_1 - \Sigma_{yy}}
\end{aligned}
$$

where the λ_1 and λ_2 are the greater and lesser eigenvalues of the covariance matrix $\mathbf{\Sigma}$. $a_{r_1,x}$ is the angle between the long main axis of the ellipse and the x-axis, which is the angle between the eigenvector of greater eigenvalue and the x-axis. K is a parameter related to χ^2 distribution controlling the confidence of the ellipse area. Assuming $K = 2$ gives a 63% confidence inside the ellipse.

Also, because the same group of data could split into several sections along the x-axis in some cases, we draw each same ellipse two times additionally with the O_x value shifts ± 1, as we can see on the left and right edges in Fig.6.8.

6.3.3 Automatic Data Sampler

The next modification to the user interface is that during the training process, we do not collect each data point manually. Instead, during the participant's pressing on the touchpad, we collect a series of data points automatically. The script we use in the UI detects if the absolute signal strength (measured with LTC5507 detector) is over a threshold that indicates a press on the touchpad. If the signal is identified as pressing-down, the script sends all the data received during this period to the classifier

as training data. Such a process is triggered only once every time the user presses the corresponding tag button on the UI, and stops when the user moves the finger away.

6.3.4 Calibration

We design an additional calibration step to compensate for the profile drifting: when the wristband stays on the user's wrist, minor physical changes such as sweat, humidity, or small slipping over the skin may cause the phase-magnitude difference profiles to change over time. The calibration stage uses a single finger profile collected in the same way as the training stage to adjust the finger profile pattern for higher detection accuracy. During the calibration stage, the user place only one finger on the touchpad, and we record data points for once by the automatic data sampler.

During the training stage, we have the center for each finger profile $\mu_x^{(i)}$ and $\mu_y^{(i)}$, where $i = 1, 2, 3, 4, 5$ indicates the finger identity. For a certain finger j, we also have a its profile center $\mu'^{(j)}_x$ and $\mu'^{(j)}_y$. For each old finger profile calculated during the training stage using algorithm 1, we change the new profile center into:

$$\mu'^{(i)}_x = \mu_x^{(i)} + (\mu_x^{(j)} - \mu'^{(j)}_x) \bmod 1$$
$$\mu'^{(i)}_y = \mu_y^{(i)} + (\mu_y^{(j)} - \mu'^{(j)}_y)$$

which aligns the center of the old profile j to the center of the newly collected profile.

Chapter 7

User Study 2

We conducted an informal user study to evaluate the performance of this system. For this user study, we first collected a dataset from the participants. All participants are from the IIS-Lab. There are 3 participants engaged in this user study.

7.1 Finger Profile Data Collection

We collected four groups of finger profiles from each participant: the training data, the test data immediately after training, the test data after wearing the wristband for 20 minutes, and the test data after remounting the wristband. During the whole process, the user is blind to the training or test data. There are also three stages during the data collection: ready stage, training stage, and test stage. The whole experiment process for one participant is as follows.

7.1.1 Ready Stage

1. Disinfect the wristband, touchpad, and the desk.

2. Record the age, gender, and dominant hand of each user.

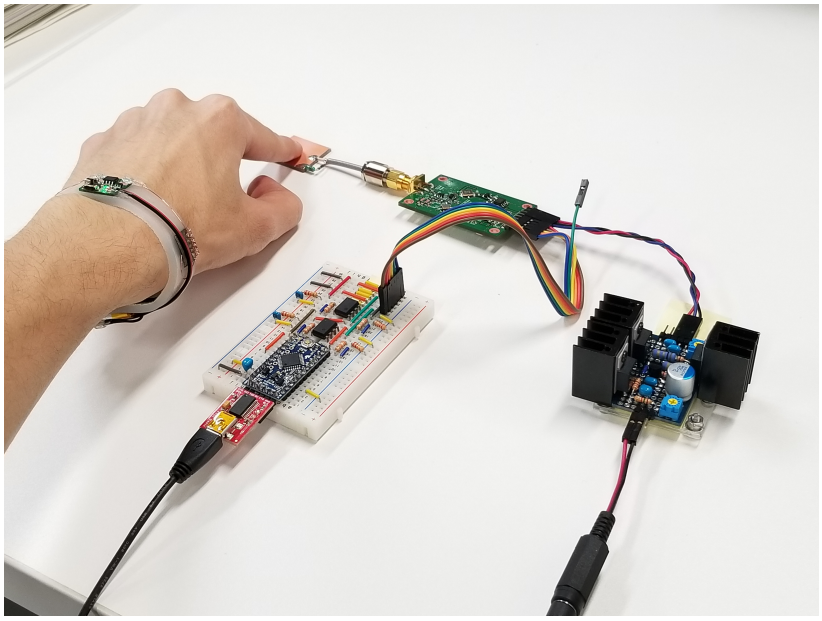


Fig. 7.1 The user is touching the sensor pad of the detector, wearing the emitter wristband. There are power and data connections to the hardware part.

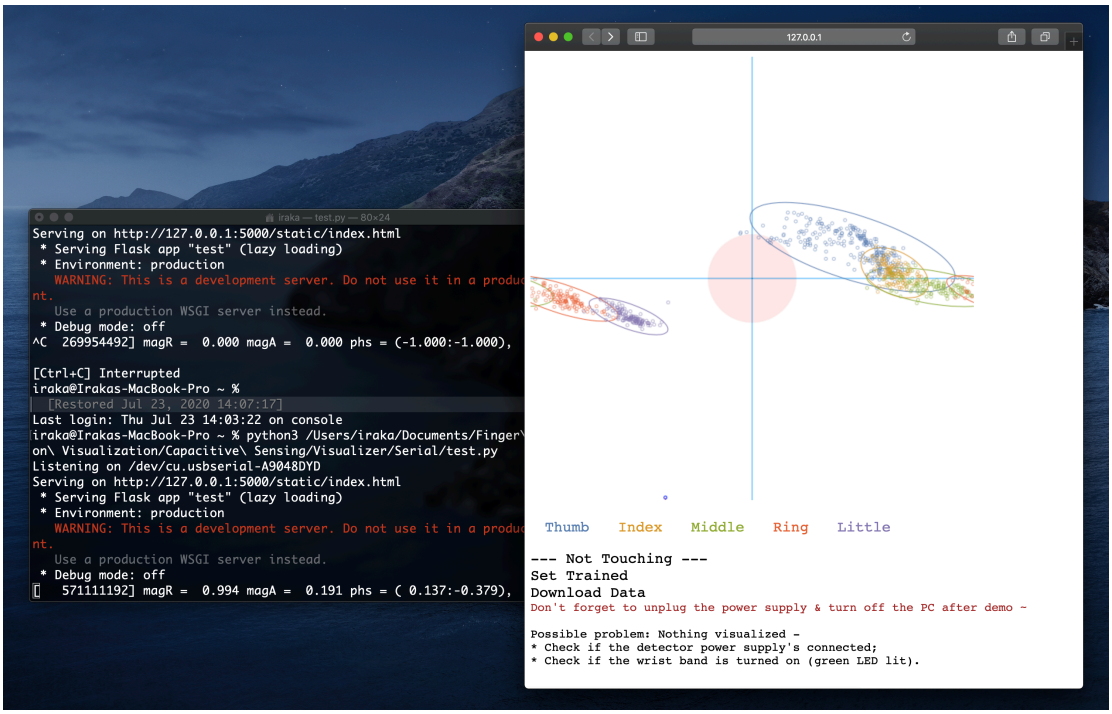


Fig. 7.2 The interface for this user study.

Table 7.1 The information of the participants in user study 2

	Age	Gender	Dominant Hand
P1	24	Female	Right
P2	23	Male	Right
P3	29	Male	Right

3. Set up the system, make sure it is working in the training mode. Place the touchpad on a platform.

4. Switch on the wristband, make sure it is charged.

5. The user stretches out the dominant hand (palm downwards), and attaches the wristband over the carpel bones, with the chip on the wristband facing upwards.

6. The user sits towards the touchpad, with the fingers open. (Do not grab or stick two fingers together)

7.1.2 Training Stage

For each of the 5 fingers on the user's dominant hand:

1. (The experimenter) clicks on the corresponding finger identity tag.

2. The user puts the fingertip on the touchpad, rubs the pad in several ways (knuckle angle, wrist movement, etc.), and then leave the finger from the touchpad. The system will automatically start and stop recording training data.

Repeat the process by turns until every finger is recorded 3 times in total.

7.1.3 Test Stage 1: Immediate

The experimenter manually records the result and ground truth during this process.

1. (The experimenter) Switch the system into test mode.

2. For each of the 5 fingers (from thumb to little finger) on the user's dominant hand, the user puts the fingertip on the touchpad in a natural way, wait for the recording of the identification result, and then leave the finger from the touchpad.

Repeat the process by turns for 10 rounds (50 touches) in total.

7.1.4 Test Stage 2: Delayed

After test stage 1, the user wears the wristband for 20 more minutes, and repeat the process in test 1.

7.1.5 Test Stage 3: Remounting

After test stage 2, the user takes off the wristband for 20 minutes. Then, the user remount the wristband where he/she feels almost the same place, and repeat the process in test 1.

7.1.6 Collected Data

During each stage, we collect all the phase-magnitude difference profile, and save them for the following algorithm evaluation. We visualized the collected finger profile. Fig. 7.3 shows the visualization for the profile of one participant.

7.2 Experiment Result

We ran the algorithm (without calibration) with the three groups of data collected, and measured the accuracy over five fingers with F_1 score.

$$F_1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

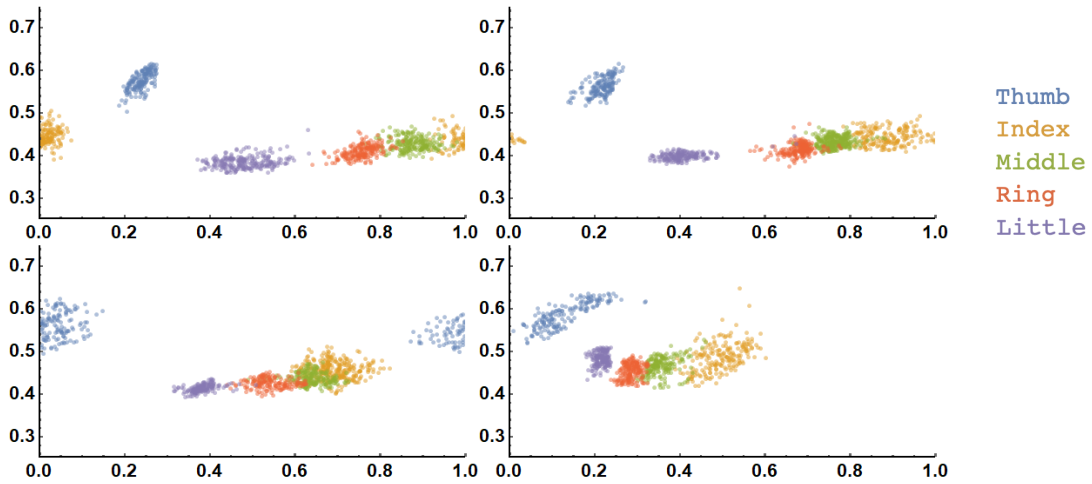


Fig. 7.3 The visualization of finger identity profile for one participant. Up-Left: training data. Up-Right: test data 1. Down-Left: test data 2. Down-Right: test data 3. The X and Y axes are relative phase and magnitude difference.

Table 7.2 Detection accuracy of the algorithm without calibration

F_1 Score	Test 1	Test 2	Test 3
P1	0.75	0.53	0.61
P2	0.67	0.24	0.55
P3	0.87	0.78	0.32
Total	0.76	0.52	0.49

Table 7.2 shows the accuracy of each participant and the overall accuracy in each test. The baseline for a random guess is $F_1 = 0.2$.

We can achieve a 76% accuracy in the immediate test after the training stage, but after a while, either the delayed test or the remounting test shows an accuracy decline to about 50%. If we visualize the distribution of the training and test data, like the left image of Fig. 7.4, we could see that a large number of data points, mainly among the index, middle, and ring finger, are at a bias from the distribution of the training data.

We then applied the calibration process on the middle finger, by randomly sample a 20% subset from the middle finger profile in each test dataset and calculate the bias. We may shift the training data pattern (the Gaussian model parameters) to a new

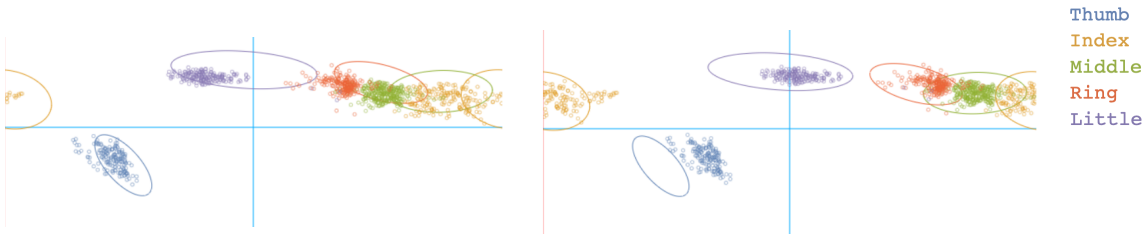


Fig. 7.4 The training data error ellipse and the test data points from one participant. Test data from test 2 (delayed test). Left: Without calibration. Right: With middle finger for calibration.

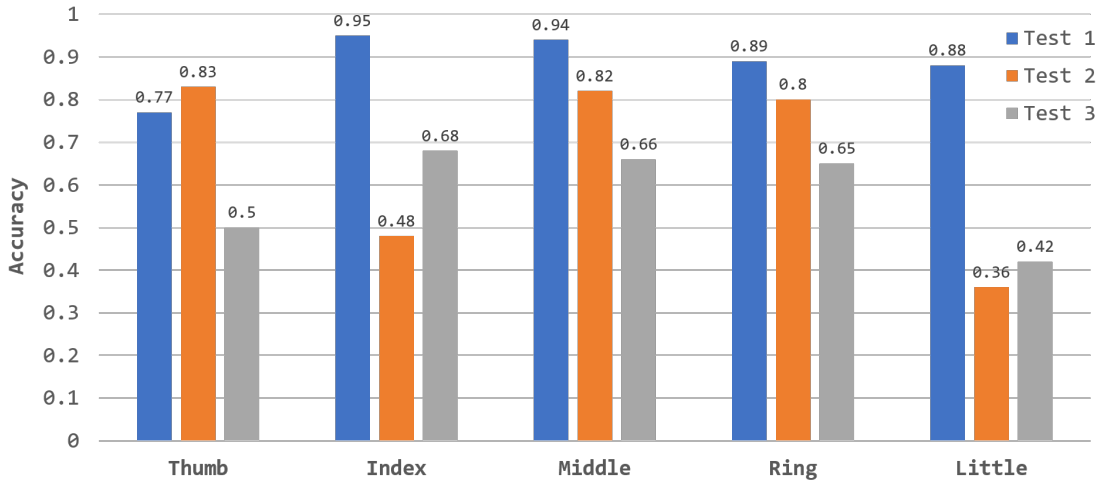


Fig. 7.5 The accuracy of each test using different fingers for calibration.

place. Testing the algorithm on the calibrated model gives the accuracy listed in table 7.3.

From the results, we may observe a great performance improvement on especially test 1 and test 2. The only case whose accuracy dropped is the case of test 3, participant 1. We deduce that such a result may result from a misalignment of other finger profiles when focusing on the middle finger.

Moreover, we tested using the profile of each finger as the origin of alignment during the calibration step. Fig 7.5 includes the overall accuracy of each test using different fingers. From the chart, we may conclude that using the middle finger for calibration gives the best performance among all.

Table 7.3 Detection accuracy of the algorithm with middle finger calibration

F_1 Score	Test 1	Test 2	Test 3
P1	0.94	0.82	0.49
P2	0.96	0.72	0.86
P3	0.92	0.93	0.62
Total	0.94	0.82	0.66

7.3 Summary

The second version of our finger identification system is capable of detecting the finger identity when touching a touchpad. Accompanied with a calibration process, the accuracy is 94% immediately after training, 82% after 20 minutes' wearing, and 66% after the users' remounting the wristband. Calibration gives the best performance when using the middle finger.

Chapter 8

Discussion and Future Works

8.1 Discussion

For the results we obtained from the user studies, we may answer the initial questions to the finger identification system.

8.1.1 Q1: What hardware and algorithm do we need for the finger identification?

Based on the performance of the system we design, we knew that a capacitive-sensing-based hardware is capable of extracting the features of the finger used to touch an object. Specifically, when emitting signals at two locations around the wrist, the phase and magnitude differences between two frequencies are reliable features for finger identification.

We implemented two versions of hardware. As for the second version, we found that a Gaussian model in the Cartesian space is appropriate for transforming the signal features into a detection result. With an additional calibration step, we could get an accuracy (in F_1 score) of 94% in an immediate test, 82% in a delayed test, and

still 66% after remounting the system. The automatic sampler in the second version of user interface also made data collection easier.

However, it is hard to give an explicit judgement of whether the second version outperformed the fist one. Although the accuracy results indicate that the second version will have a higher accuracy after calibration, we lack sufficient data to ensure this result to be statistically significant.

Compared with other methods of finger identification mentioned in chapter 2, our method achieved all of the following features:

- Fast speed: Our method reacts to the touch on the touchpad instantly, and the delay comes from only the data transfer from the ADC to the classifier, which is less than 100ms in our case.
- Detecting among all fingers: Our method may distinguish the identity among all of the 5 fingers on one hand, and may potentially extend to several hands.
- Ease of use: The user only have to wear a wristband over the wrist. There is no additional hardware attached to extra places on the body, which means our method will not hinder the user from manipulating the body parts, especially to moving and feeling with fingers.

8.1.2 Q2: What features contribute the most accuracy to the identification?

From the visualization of the training and test data, we observed the phase and magnitude features both contributed to the classification. The most important factor is the data from the middle finger, which decides the final accuracy to a greater extent.

We are still unable to explain the further factors that caused the false detection results. Even with a calibration step, we cannot make the finger profile pattern strictly

overlap between training and test stages. The conductive properties of the hand tissues owns more complex behaviors than our model may describe.

8.1.3 Application Design Space

By comparing the advantages to the related works, we may find our system useful in the following cases.

When a touch user interface involves several tools for operation, or some menus for choosing a function with the touch (such as a right-click menu), we may assign a function to each finger to avoid an extra selecting operation. For example, in a sketchpad application, different fingers may represent paint brush, eraser, or transform tools. In another case, for a number in an instant messaging application, a click on the number with different fingers may activate a call, adding friend with this number ID, or copying the number as a validation code. These operations are suitable for the user interface on a tiny surface, such as a smartwatch, a pair of smart glasses, or a Bluetooth earphone.

For another aspect, there are applications where finger identities are bound to the value of a variable. In a virtual musical instrument case, finger identities could be related to an audio effect such as the pitch, or the tone of the instrument. Using the finger identity in a virtual instrument application also allows us to record the finger used to play every note, and allow us to replay the piece or comment the musical score with fingering information.

8.2 Future Work

In our system, there are only two frequency components of a single “doubled-frequency” relation. The frequency selection, 49 MHz and 72 MHz in our study, are mainly based on the convenience of circuit design. Our current calibration process only utilizes the

profile from a single finger, which we only used for validating the idea of calibration. This hardware detects only a single-finger touch, where we expect to be able to detect all finger identities of a multiple touch. Also, due to the current situation, we did not carry out a formal user study with ample data for a statistically solid conclusion.

Future work should consider the best frequency that provides the highest accuracy, and the possibility of adding more frequency components in the signal from the wristband - requiring more advanced circuit design both to the wristband and the detector.

Meanwhile, the future work should investigate the calibration algorithm of using more fingers to generate a better pattern from the training data, or a better feature space transform with a higher accuracy. All these evaluations should include a larger-scale user study to guarantee statistical significance.

Chapter 9

Conclusion

Despite the large amount of previous work exploring the methods for finger identification, a method that is swift, versatile, and easy to mount is still vacant. We created a system that achieves all of these advantages with capacitive sensing technology. The system only requires the user to wear a signal emitting wristband on the wrist, which generates signals of two frequency components. We designed a hardware piece to detect the relative phase and magnitude difference of the two frequency components, and an algorithm to classify these data into finger identities. Our evaluation study showed that such a system achieved fast and accurate finger identification among 5 fingers on one hand, with an extra calibration process.

Publications

Publication related to this thesis

国内研究会

- 陳明輝，チャコン・サラス・ディマス・アントニー，矢谷浩司. 「複数周波数帯での静電容量センシングによる指の識別手法」第 65 回ユビキタスコンピューティングシステム研究発表会.

References

[1] Au, O. K.-C. and Tai, C.-L. (2010). Multitouch finger registration and its applications. In *OZCHI '10*.

[2] Benko, H., Saponas, T. S., Morris, D., and Tan, D. S. (2009). Enhancing input on and above the interactive surface with muscle sensing. In *ITS '09*.

[3] Colley, A. and Häkkilä, J. (2014). Exploring finger specific touch screen interaction for mobile phone user interfaces. In *OZCHI*.

[4] Dietz, P. H. and Leigh, D. (2001). Diamondtouch: a multi-user touch technology. In *UIST '01*.

[5] Dove, I. (2014). Analysis of radio propagation inside the humanbody for in-body localization purposes.

[6] Fukumoto, M. and Suenaga, Y. (1994). "fingering" : a full-time wearable interface. In *CHI '94*.

[7] Gabriel, C., Gabriel, S., and Corthout, E. (1996). The dielectric properties of biological tissues: I. literature survey. *Physics in medicine and biology*, 41 11:2231–49.

[8] Ghomi, E., Huot, S., Bau, O., Beaudouin-Lafon, M., and Mackay, W. E. (2013). Arpège: learning multitouch chord gestures vocabularies. In *ITS '13*.

[9] Gil, H., Lee, D., Im, S., and Oakley, I. (2017). Tritap: Identifying finger touches on smartwatches. In *CHI '17*.

[10] Goguey, A., Vogel, D., Chevalier, F., Pietrzak, T., Roussel, N., and Casiez, G. (2017). Leveraging finger identification to integrate multi-touch command selection and parameter manipulation. *Int. J. Hum.-Comput. Stud.*, 99:21–36.

[11] Gupta, A. and Balakrishnan, R. (2016). Dualkey: Miniature screen text entry via finger identification. In *CHI '16*.

[12] Harrison, C., Schwarz, J., and Hudson, S. E. (2011). Tapsense: enhancing finger interaction on touch surfaces. In *UIST '11*.

[13] Holz, C. and Baudisch, P. (2013). Fiberio: a touchscreen that senses fingerprints. In *UIST '13*.

[14] Kim, I., Park, K.-W., Yoon, Y., and Lee, G. (2018). Touch180: Finger identification on mobile touchscreen using fisheye camera and convolutional neural network. In *UIST '18 Adjunct*.

[15] Le, H. V., Mayer, S., and Henze, N. (2018). Infinitouch: Finger-aware interaction on fully touch sensitive smartphones. In *UIST '18*.

[16] Le, H. V., Mayer, S., and Henze, N. (2019). Investigating the feasibility of finger identification on capacitive touchscreens using deep learning. In *IUI '19*.

[17] Masson, D., Goguey, A., Malacria, S., and Casiez, G. (2017). Whichfingers: Identifying fingers on touch surfaces and keyboards using vibration sensors. In *UIST '17*.

[18] Mayer, S., Le, H. V., and Henze, N. (2018). Designing finger orientation input for mobile touchscreens. In *MobileHCI '18*.

[19] Parizi, F. S., Whitmire, E., and Patel, S. N. (2019). Auraring: Precise electromagnetic finger tracking. In *IMWUT*.

[20] Park, K.-W. and Lee, G. (2019). Fingmag: Finger identification method for smartwatch. In *CHI EA '19*.

[21] Roy, Q., Guiard, Y., Bailly, G., Lecolinet, E., and Rioul, O. (2015). Glass+skin: An empirical evaluation of the added value of finger identification to basic single-touch interaction on touch screens. *ArXiv*, abs/1901.08325.

[22] Sato, M., Poupyrev, I., and Harrison, C. (2012). Touché: enhancing touch interaction on humans, screens, liquids, and everyday objects. In *CHI '12*.

[23] Schweigert, R., Leusmann, J., Hagenmayer, S., Weiss, M., Le, H. V., Mayer, S., and Bulling, A. (2019). Knuckletouch: Enabling knuckle gestures on capacitive touchscreens using deep learning. In *MuC'19*.

[24] Suzuki, Y., Sekimori, K., Shizuki, B., and Takahashi, S. (2019). Touch sensing on the forearm using the electrical impedance method. *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 255–260.

[25] Vega, K. and Fuks, H. (2014). Beauty tech nails: interactive technology at your fingertips. In *TEI '14*.

[26] Zhang, Y., Xiao, R., and Harrison, C. (2016a). Advancing hand gesture recognition with high resolution electrical impedance tomography. In *UIST '16*.

[27] Zhang, Y., Zhou, J., Laput, G., and Harrison, C. (2016b). Skintrack: Using the body as an electrical waveguide for continuous finger tracking on the skin. In *CHI '16*.