EXTEND THE SENSING BOUNDARY OF MOBILE SYSTEMS: SECURITY AND NEW APPLICATIONS

by

WENQIANG JIN

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2021

To my family

ACKNOWLEDGEMENTS

First and foremost, I would like to given my sincerest gratitude to my Ph.D. supervisor, Prof. Ming Li. She is an outstanding mentor, a great friend, and the most intellectual researcher that I have ever known. Her immense knowledge and plentiful experience have encouraged me through all the time of my academic researches and daily life. It is my lifelong asset to learn from her.

I would also like to thank my committee members, Prof. Hao Che, Prof. Jia Rao, and Prof. Mohammad Atiqul Islam, for serving on my supervisory committee and for all the assistance in different stages of my Ph.D. study.

I could never achieve this without a group of great friends and colleagues. I would like to express my gratitude to all the fantastic MobiSec group members for providing me unconditional help, a family-like environment, and for their encouragements and collaboration in different research projects. I am so grateful to have Mingyan Xiao, Srinivasan Murali, Huadi Zhu, Chaowei Wang, and Tianhao Li. We had so many valuable discussions and all the good memories.

Lastly, but most importantly, I would like to thank my parents, beloved wife, and parents in law who have always been there when I needed them most. They encouraged me to study in the United States. Without their unconditional love and supports, I could never imagine what I have achieved.

April 27, 2021

ABSTRACT

EXTEND THE SENSING BOUNDARY OF MOBILE SYSTEMS: SECURITY AND NEW APPLICATIONS

Wenqiang Jin

The University of Texas at Arlington, 2021

Supervising Professor: Ming Li

The exploding growth of mobile devices like smartphones and wearables has envisioned various applications, which are developed to collect a wide spectrum of data using on-board device sensors and process them to serve peoples' life in all kinds of scenarios. Noticing the sensing capabilities of current mobile device are limited to its on-board sensors' default functionalities. We study the mechanism designs that extend the mobile device's sensing capabilities to perform new sensing tasks other than its defaults.

In this thesis, we investigate the mobile systems' security issues and develop new applications by exploring the device's sensing capabilities. Our contributions are mainly threefold. First, we address the challenges of device pairing between wearables by leveraging its on-board transceivers. Specifically, we use wearables' transceivers to harvest ambient radio frequency (RF) noise from open-air and turn them into the ingredients for the secret key establishment. Second, we introduce a novel side-channel attack to infer user's secret PINs typed on mobile device's touchscreens by eavesdropping and analyzing its electromagnetic emanations. In particular, we observe that the finger movements on the touchscreen leads to time-variant coupling between the human body and the touchscreen.

Consequently, it results variations of touchscreen's electromagnetic emanations, which can be captured by electrical potential sensors and leveraged to reconstruct the finger movement traces on the keypad during its typing process. Third, we develop an acoustic ranging application that assists pedestrians with vision impairment to across uncontrolled streets. The application leverages smartphone's microphones to sense motion status of oncoming vehicles. Based on the measurement results, it then detects the potential collisions and alert the pedestrians ahead.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Recently, we are witnessing a remarkable growth in the number of mobile devices. According to recent market reports [125], it is forecasted that the yearly shipment of mobile devices will reach 1.91 billion in the year 2023. The large scale deployment of mobile devices has driven the emergence of various mobile systems, such as body area networks (BAN), mobile crowdsensing (MCS), and Internet of Things (IoT). Applications based on these systems leverage mobile devices' on-board sensors (e.g., compass, accelerometer, gyroscope, camera, and GPS) to sense the environmental context of its surroundings. These sensors can be used anytime and anywhere because of the portable nature of mobile devices which introduces more flexibility and thus provides various functionalities that facilitate people's daily life.

However, the potentials of mobile devices' sensing capabilities are not well explored. Most of them only leverage the basis functionalities of their on-board sensors to perform simple tasks. For example, the speakers are usually used for playing music. Microphones are simply used to record sound tracks. WiFi transceiver boards are leveraged to meet the basic needs of data transmissions. However, recent researches [109, 89, 153, 29, 55, 119, 81, 97, 28] show that on-board sensors can be used to obtain a wide range of sensory results that are out of their default sensing spectrum. For example, Acousticcardiogram [109] leverages the smartphone's microphones and speakers to monitor heartbeats. Aim [89] uses the sensors to produce images of the covered objects. TouchPass [153] authenticates the mobile users by leveraging their finger's unique physical characters sensed by accelerometers. UbiBreathe [3] shows that WiFi transceivers can be leveraged to monitor users' breath patterns.

In this thesis, we intend to explore the mobile system's sensing boundaries to disclose and address the critical challenges of the system security and to develop new applications. The reset of the thesis is organized as follows.

In BAN, wearable devices capture user's rich information regarding their health conditions and daily activities. To secure communications among wearable devices, we propose a novel device pairing scheme in Chapter 2. Despite some prior efforts trying to fill this gap, they either rely on some sophisticated sensors, such as electromyogram (EMG)[160] or electrocardiogram (ECG)[117] pads that may not universally exist, or non-trivial design of communication transceivers that cannot be found easily on current commercial devices. We leverage wearables' RF transceivers to sense the RF noise, which is then exploited to distribute the secret pairing key. RF noise is ubiquitous presence, high random and unpredictable, thus serves as an ideal entropy source. We observe that the received RF noise power measured in the logarithmic scale at different parts of a human body surface experiences the same variation trend. In contrast, those from different human bodies or off the body are distinct. Under the touch-to-access policy, we present a protocol [63] that allows two legitimate devices to agree on a mutual secret key securely.

Besides the security of device paring, we show that mobile users' private inputs, such as passwords, PINs, and verification codes, also facing significant security challenges. In Chapter 3, we present a novel side-channel eavesdropping attack which leverages human-coupled electromagnetic (EM) emanations from touchscreens to infer victims' typing inputs at a remote distance. In particular, the capacitive touchscreens of mobile devices consists of a grid of transmitter (TX) and receiver (RX) electrodes, which are mutually coupled with a capacitance. TX electrodes are driven by an alternating current, which flows through the coupled capacitance to the RX electrodes. When a finger touches the screen, it extracts some electric charges from the RX electrode grid to the human body through their mutual coupling capacitance. The alternating driven currents generate time-variant EM fields and thus emit EM emanations to the open space. Meanwhile, we observe that the finger movement over the touchscreen leads to time-varying coupling between the finger and RX electrodes. Consequently, it impacts the screen's EM emanations that can be picked up by a remote sensory device. We intend to map between EM measurements and finger movements to recover the inputs. We build an analytic model that outputs finger movement trajectories based on given EM readings. As no training is needed, our approach does not require the collection of a user-specific dataset. We implement the system with simple electronic components and conduct a suite of experiments to validate this attack's impacts.

Chapter 4 further uses smartphones to help pedestrians with vision impairments across the uncontrolled streets, where no traffic-halting signal devices are present. Pedestrians with visual impairments must rely on their other senses to detect oncoming vehicles and estimate the correct crossing interval to avoid potentially fatal collisions. To overcome the limitations of human auditory performance, which can be particularly impacted by weather or background noise, we develop an assisting tool called Acoussist [64], which uses the smartphone's microphones to perform acoustic ranging and detect the oncoming vehicles. The vision-impaired people can use the tool to double-confirm surrounding traffic conditions before they proceed through a non-signaled crosswalk. To achieve this goal, it is essential to figure out movement status of each vehicle nearby, characterized by, for example, its velocity relative to the pedestrian, direction of arrival (DoA), and its distance to the pedestrian. Therefore, we propose to install external speakers on opt-in vehicles and have them emit acoustic chirps at a frequency range imperceptible by human ears, but detectable by smartphones operating the Acoussist app. By analyzing the received chirp signals, the app would then communicate to the user when it is safe to cross the roadway. We implement a proof-of-concept of Acoussist using commercial off-the-shelf (COTS) portable speakers and smartphones and prove the effectiveness of our designs.

Finally, we conclude the thesis in Chapter 5.

CHAPTER 2

HARNESSING THE AMBIENT RADIO FREQUENCY NOISE FOR WEARABLE DEVICE PAIRING [1]

2.1 Introduction

Recently, we are witnessing a remarkable growth in the number of smart wearable devices. According to recent market reports [56], it is forecasted that the yearly shipment of wearable devices will reach 279 million in the year 2023. Moreover, the wearable technology market is expected to reach a value of $58 billion by 2022, which is almost three times of that in 2015 ($19 billion) [114]. With the high penetration to people's daily life, smart wearable devices (e.g., wrist bands, earbuds, heartbeat meters, and step counter) have been gradually recognized as a compelling paradigm for e-healthcare and fitness applications. Sensitive information such as health conditions and physiological data are shared among wearables or synchronized from wearables to personal hubs [107]. Audio streaming is carried between earbuds and a smartwatch for better living and exercising experience [61, 8]. Securing their wireless communications is of critical importance to the wide deployment of wearable devices. In particular, newly deployed wearables must be able to securely associate and establish cryptographic key pairs with existing devices, also known as pairing, in a way that protects against man-in-the-middle (MitM) and protocol manipulation attacks. For devices on the Internet, pairing can be achieved by relying on certificate authorities to certify the device identities, providing a root of trust when establishing the identity of a communicating party. Unfortunately, many wearables lack direct Internet accesses. In-

[1] Used with permission of the publisher, 2021.

stead, they often use short-range radio technology (e.g., Bluetooth or Zigbee) as their first hop to connect to other existing wearables or personal hubs that are to pair. Conventional solutions typically rely on a human to certify the device validity when pairing, for example, by visually comparing short strings on a screen, or by typing a number displayed by one device into the other. These pairing protocols are cumbersome, error-prone, and do not scale well. More importantly, many wearables do not have a user interface, rendering password entry or management extremely challenging.

Efforts to reduce human involvement from the wearable device pairing process have brought the emergence of the idea *touch-to-access* [116, 157, 117, 160, 39, 152]. Two devices are allowed to be paired if and only if they are attached to the same human body at the same time. The rationale behind this idea is that if a wearable has direct physical contact with the human body, it is deemed as a legitimate device validated by the wearer. Under this policy, some existing schemes utilize dedicated sensors to extract physiological signals from the human body, such as ECG [117], EMG signals [160], and body movements [39, 152], and translate them to common randomness, forming the basis of a symmetric key agreement protocol. These approaches are based on the principle that the physiological signals captured from the same human body have similar features, whereas those collected from different human bodies are distinct. Apparently, their success relies on a common, properly calibrated sensing capability across all devices. In contrast, a wide diversity of sensing capabilities are present in commercial wearables; it is impractical to assume arbitrary two wearables equip with a common sensor.

In this work, we aim to develop an automatic pairing scheme for wearable devices without user involvement. Our design follows the touch-to-access policy. Rather than relying on any dedicated sensors, we propose to utilize the RF transceiver, one of the basic electronic components of wearables that are capable of data synchronization/transmission. For devices without this capacity, i.e., stand-alone wearables, there is no need

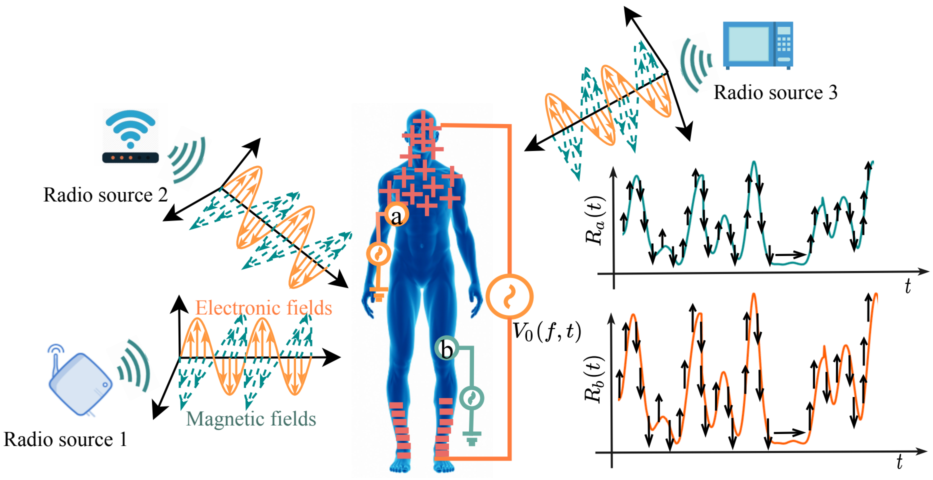


Figure 2.1: Physical basis of our design.

of pairing. Ambient radio frequency (RF) noise is captured from open air and turned into ingredients for secure pairing. RF noise is mainly a combination of natural electromagnetic atmospheric noise and manmade radio interference. Typical sources include lightning discharges, FM/AM radio stations, power systems, a wide variety of electronic equipment, and wireless transmissions. Due to the source diversity and the electromagnetic disturbance originating in a large number of discrete distances with unpredictable occurrences in time and amplitude, RF noise is highly random and unpredictable in temporal, frequential, and spatial domains. Thus, RF noise could be an ideal entropy source for key generation during pairing. Additionally, as RF noise is ubiquitously present, such a source is easily accessible almost everywhere.

Despite these promising features of RF noise, utilizing them for device pairing still faces a significant challenge. Essentially, under the touch-to-access policy, wearables on the same human body should be capable of extracting the same secret, even when they are separately mounted, say one on the wearer's wrist and the other on the chest. Such requirement, however, is primarily hindered by the noise dynamics exhibited in the spatial domain. As pointed out by Xi et al. [149], RF signals received at two locations are independent when they are more than half of the wavelength apart (6.25 cm@2.4 GHz). Thus, how to harvest correlated noise readings from open space at two remotely positioned wearables resides at the heart of our design.

This work is based on a key observation that RF noise, measured in logarithmic scale, experiences the same variation trend even at different parts of a human body surface (as shown in Figure 2.1). From the perspective of electrostatics, the human body can be viewed as a low-impedance conductor. Thus, placing an induced human body in electromagnetic fields will build up a charge distribution over the body surface to reach an electrostatic equilibrium [106]. As a result, an electrical potential is formed between an arbitrary part of a human body and the ground. Since radiation waves generate time-variant electromagnetic fields, the body surface electrostatic equilibrium distribution varies accordingly, so does the induced on-body potentials. Then, by creating physical contact between the wearable transceiver and the human body, the former can readily access RF noise, reflected as instant variant potentials, captured by the latter from open air. Besides, due to the above-mentioned phenomenon that variations of RF noise readings from the same body surface are highly synchronous, RF noise can be transformed into a common entropy source for key generation.

In practice, due to the uneven charge distribution over the skin surface, variation tendencies of RF noise measures at different parts of the body surface may not be perfectly the same. Directly applying them for device pairing would result in a high error rate. To address this issue, our scheme adopts the framework of *fuzzy commitment* [65, 95, 50]. It is able to correct at most t mismatched bits during paring, with t a tunable parameter that balances between security and usability.

To evaluate the proposed paring scheme, we build a prototype based on a CC2500 transceiver [59] and Arduino board [57]. Extensive experiments show that our scheme has an equal error rate (EER) as low as 1.4%. Its key generation rate reaches 138 bits/sec, which beats so-far existing pairing schemes. Tests also show that our scheme is robust against various attacks such as imitation attacks, MitM attacks, and synthesis attacks. Besides, our scheme can be efficiently executed within 0.97 s. Its incurred energy consumption is also negligible, as low as 0.27 J for the entire pairing procedure. We summarize the contributions of this paper as follows.

- While most of the previous studies focus on avoiding RF noise to improve the performance of wireless communication systems, we propose a novel and practical wearable device pairing scheme by harnessing ubiquitously present RF noise.
- We show analytically and experimentally that the log-scale RF noise measures at different parts of body surface experience the same variation tendency, which serves as the basis for our pairing scheme.
- We develop a prototype and perform extensive experiments to evaluate the security and usability of our scheme. It outperforms the state-of-art solutions in terms of key generation rate, time consumption, and energy cost.

2.2 Related work

2.2.1 Pairing for Wearable Devices

To secure wireless communications among wearable devices, quite a few novel methods have been proposed to facilitate automatic device pairing without any trusted authority. The idea of *touch-to-access* is *de facto* the pairing rule followed by existing works. Two wearables are allowed to be paired if and only if they are attached to the same human body.

Body movements have been explored as an entropy source for key extraction. Since human movements are unique across individuals, readings from the same human body are considered to share significant similarity, while that on different bodies are not. Thus, prior works [152, 39] turn readings from inertial sensors into common cryptographic keys. Along this line of research, some othere existing works utilize ECG or EMG signals [117, 160] for key establishment. These solutions require a common dedicated sensor across all devices (e.g., an accelerometer or ECG monitor), which largely hinders their wide adoption. Treating human body as a transmission medium, Roeschlin et al. [116] proposed to utilize this covert channel for key pairing. Nonetheless, it requires a non-trivial design of communication transceivers. A recent study TouchAuth [157] treats human body as an antenna that captures radiation from electrical cabling. It converts the corresponding potentials measured at two devices within close proximity into their common secrets. The scheme fails if two devices are reasonably distantly located even on the same body. Besides, cable radiations are mostly unavailable outdoors.

2.2.2 General Device Pairing

Pairing has also been studied for general IoT devices. Existing schemes mainly fall into two categories, channel reciprocity based and context-based paring.

Channel reciprocity based paring. Channel reciprocity states that a pair of wireless transceivers observe the same channel characteristics, which are then utilized by transceivers for key pairing. Zeng et al. [164] turned RSS readings into secret bits. Realizing that RSS-based pairing bears low bit generation rate, recent works use CSI [77, 148] and Channel impulse response (CIR) [93, 87, 139] as alternative entropy sources. Due to the involvement of phase information (in addition to signal amplitudes in RSS readings), CSI/CIR measurements can be converted to secret keys at a much higher speed. More importantly, the diverse information also renders spoofing attack more difficult to launch. Meanwhile, these works impose strict requirement over channel coherence–channel reciprocity exists only within channel coherent time, which is typically at a level of 10 ms for 2.4 GHz wireless signals. Existing schemes are implemented using the probing mechanism in IEEE 802.11. Alice broadcasts a Probe frame, upon receiving which Bob replies with an ACK. Only under the ideal case that ACK is replied immediately, i.e., the frame interval is deemed within coherence time. In practice, it is not rare that Probe or ACK is corrupted by other ongoing transmissions, especially in crowded 2.4 GHz ISM band. Consequently, the condition for channel reciprocity does not hold.

Context-based paring. This series of approaches are based on the assumption that co-present devices observe common contextual information that can be transformed into shared secrets. For example, Markus et al. [95] proposed to have devices compute a fingerprint of their ambient context using sensor modalities like ambient noise and luminosity. A similar idea is adopted in [121, 67]. Like many existing pairing schemes for wearables, [95, 121] assume the existence of commonly available sensors. To overcome this restriction, Han et al. [50] proposed a pairing scheme using heterogeneous sensor types. Their idea is that devices co-located within a physical boundary can observe more events in common over time, as opposed to devices outside. Nonetheless, sensors still need to capture certain contextual information, such as light, sound, movement, which are not accessible for many wearables. More importantly, these schemes are vulnerable to stealthy attackers that coexist with legitimate devices, say in the same room. As a result, the secret can be easily interpreted by attackers.

Another line of research also employs wireless channel characteristics for pairing. Rather than channel reciprocity, these approaches rely on the observation that received wireless signals are unique for co-presence devices. For example, Miettinen et al. [92] used the RSS to generate symmetric keys for two devices of close proximity. CSI measurements are also exploited [83, 149]. Nonetheless, these schemes only work when inter-device distance is within a certain threshold. Given a radio wave of frequency 2.4 GHz, two pairing devices should be located within 6.25 cm, as their received signals quickly de-correlate beyond the half-wavelength limit. Such a restriction is easily violated in a wearable system, for example, two devices worn on user's two wrists.

2.2.3 Human Body Sensing Capacity

The idea of leveraging conductive human body for sensing has been explored in various context. Pioneered by Zimmerman [170], IBC is investigated for information transmission among on-body devices by treating human body as a transmission medium. The research [52, 159, 73] also falls into this category, but with different focuses, such as propagation channel modeling and channel capacity quantification. Different from IBC, some existing works investigate the feasibility of using body electric potentials induced by power line radiation for gesture recognition [27], clock synchronization [156], object classification [158], and touch/motion sensing [26]. However, none of them is about device pairing.

2.3 System Overview

2.3.1 Problem Definition

We consider two wearables, Alice and Bob, who intend to establish a secure channel over a publicly accessible wireless media without any pre-shared secret. We focus on the problem of pairing between them. First of all, pairing must be established only between intended peers, i.e., wearables owned/admitted by the same user, which is referred to as *secure association*. Second, a secure symmetric key pair needs to be established between Alice and Bob that facilitates their secure communications in a later stage. Our discussion pertains to wearables attached to the body surface. The pairing for implantable devices is not covered in this work.

Our design follows the *de facto* paring policy for wearables, "touch-to-access" [116, 157, 117, 160, 39, 152]: A new wearable device is admitted to the system if and only if it has significant physical contact with the wearer's body. An important facet of touch-to-access is forward security. The authenticity to the system vanishes once the device loses physical contact with the wearer.

2.3.2 Threat Model

The goal of an adversary is to deceive the system as a legitimate device that is attached to the wearer's body. An adversary is assumed to be present during a pairing session. It is powerful in a sense to control the channel via, for example, eavesdropping and replaying signals through public wireless channels. It is also possible to forge MAC addresses to falsely claim as a valid device. In particular, we mainly consider the following three kinds of attacks that are severe to our scenarios.

Imitation attack. The adversary exploits physical co-presence with the wearer and tries to obtain similar RF measurements and thus extract the same secret keys as legitimate on-body devices. Depending on where the attacking device is placed, we further classify it into two types. It is called type-I attack if placed in free space, say on a desk, a floor, a stand, etc. It is called type-II attack, if attached to another wearer.

Synthesis attack. More advanced than simply generating random bit sequence, the adversary is able to observe and model the radio environment around the wearer in advance. Then it fabricates synthetic signals and tries to extract from them the same secret keys as legitimate devices.

MitM attack. The adversary tries to actively participate in pairing phases of two wearables. Its goal is to either get paired with a learned key or mess the pairing procedure by tampering the exchanged messages but still make one/both of them believe the pairing protocol has completed successfully. The adversary can relay and alter messages on the wireless channel.

We assume that the adversary cannot compromise the end devices; otherwise, it renders the secure pairing impossible. Besides, the scenarios that it seeks to jam the wireless channels or by any other means of destroying communications such as Denial-of-Service (DoS) attacks are out of the scope of this work.

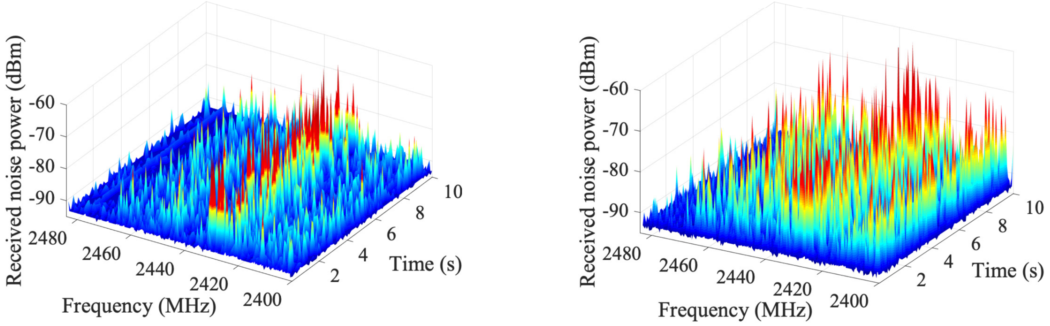
2.4 Background and Analytic Study

2.4.1 Ambient Radio Frequency Noise

Ambient RF noise is mainly a combination of natural electromagnetic atmospheric noise and manmade radio frequency interference. Typical sources include lightning discharges, FM/AM radio stations, automobile ignition systems, power systems, and a wide variety of electronic equipment such as computers, personal devices and microwave ovens. Due to the source diversity and the electromagnetic disturbance originating in a large number of discrete distances with unpredictable occurrences in time and amplitude, RF noise is highly random in temporal, frequnencial, and spatial domains, which is desirable as an entropy source. We did some prior study to show the distribution of RF noise in these three domains. Figure 2.2 plots the received noise by two CC2500 transceivers [59] over 2.4 GHz. The data are collected over the same time duration at two different locations separated by 10 cm. Since existing wearables, with their radio interfaces such as Bluetooth, ZigBee, or WiFi, operate over the 2.4 GHz ISM frequency band, we thus pertain our discussion to this band in this study. As a note, our idea can be easily extended to RF noise in other frequencies.

2.4.2 Body-as-A-Conductor

As suggested by [113, 157], a human body can be treated as a conductor with low impedance (a few $k\Omega$). Placing the induced human body in the electromagnetic fields will



(a) Location 1

(b) Location 2

Figure 2.2: RF radio noise heat map over the 2.4 GHz ISM band. RF noise is highly dynamic and randomly distributed over time, frequency, and space.

build up a charge distribution over the body surface to reach an electrostatic equilibrium [106]. From the perspective of electrostatics, a radio source creates electromagnetic waves at its propagation direction. Such electromagnetic waves from multiple radio sources overlap with each other that generate time-variant electromagnetic fields. In accordance, the electrostatic equilibrium distribution on the body-conductor surface will also vary. Then, by creating a physical contact (e.g., using an electrode or electrical conductor) between body skin and the wearable transceiver, the latter can access RF noise harvested by the wearer from the open air, as shown in Figure 2.3. Here, the transceiver is a common component of commercial off-the-shelf (COTS) wearables with data transmission capacity.

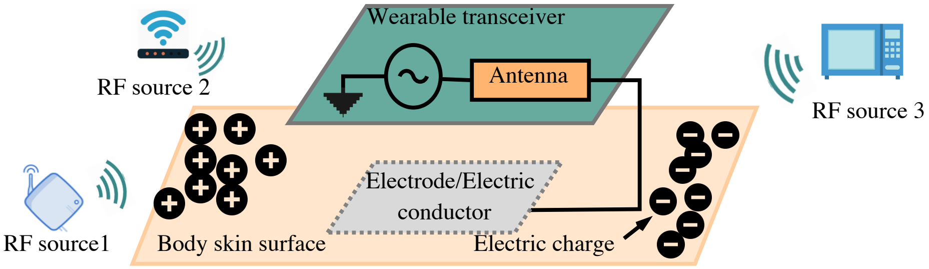


Figure 2.3: Illustration of how a wearable's transceiver accesses RF noise harvested by human body from open air.

Our design relies on a key observation that the received RF noise power measured in a logarithmic scale at different parts of a human body surface at the same frequency band and time instance experiences the same variation trend. We summarize the observation as the following theorem.

Denote by $R_a(f,t)$ and $R_b(f,t)$ the measured RF noise power (in dBm) at two arbitrary positions on a wearer's skin, we have $\frac{\partial R_a(f,t)}{\partial t} = \frac{\partial R_b(f,t)}{\partial t}$. We provide a proof sketch as follows. According to prior results on bio-electricity characterization [6, 44, 69], the human body can be deemed as a monopole cylinder conductor illuminated by waves from multiple radio sources. As mentioned above, the induced human body in electromagnetic fields builds up a charge distribution over the body surface. Assume that the axial current induced by the electric field is dominant. As shown in Figure 2.1, we denote $V_0(f,t)$ the voltage drop from the human head to the foot base of the body conductor. $V_0(f,t)$ is a complex value and variant with respect to f and t. Let $V(f,t)$ be the voltage at an arbitrary position of the human body. It can be expressed as $V(f,t) = \alpha V_0(f,t)$, where α is complex-valued coefficient with $|\alpha| \in [0,1]$. As indicated by [113, 68], the skin impedance Z at a given body position is positively correlated with its relative distance to the ground. Apparently, the closer the position to the foot base, the smaller $|\alpha|$ is. Moreover, Z is not only dependent on body height, but also the human biometric features, including weight, shape, body mass density, etc.

To obtain received RF noise power, most RF transceivers measure the average of the *apparent power* of RF signals in logarithmic scale [146, 80], which is expressed as

$$\begin{aligned} R(f,t) &= 10\lg(|\frac{V^2(f,t)}{2Z}|/1\,mW) \qquad (2.1) \\ =&10\lg|\frac{(|\alpha|^2|V_0(f,t)|^2 e^{2j(\theta_\alpha+\theta_0(f,t))})}{2|Z|e^{j\theta_Z}}| = 10\lg\frac{|\alpha|^2|V_0(f,t)|^2}{2|Z|} \end{aligned}$$

with its unit as dBm. In practice, the average of the apparent power is calculated based on the in-phase and quadrature (I/Q) components of received discrete samples. Here, we express the apparent power in its analog form for analysis simplicity. θ_α, $\theta_0(f,t)$, and θ_z stand for the corresponding phases of the α, $V_0(f,t)$, and Z, respectively. Let $R_a(f,t)$ and Z_a be the received RF noise and impedance at position a on body skin. Then, the partial derivative of $R_a(f,t)$ with respect to t is

$$\begin{aligned}\frac{\partial R_a(f,t)}{\partial t} &= \frac{\partial}{\partial t}(10\lg\frac{|\alpha_a|^2|V_0(f,t)|^2}{2|Z_a|}) \\ =&10\frac{\frac{|\alpha_a|^2}{2|Z_a|}\frac{\partial |V_0(f,t)|^2}{\partial t}}{\frac{|\alpha_a|^2|V_0(f,t)|^2}{2|Z_a|}\ln 10} = \frac{10}{\ln 10}\frac{1}{|V_0(f,t)|^2}\frac{\partial |V_0(f,t)|^2}{\partial t}\end{aligned} \tag{2.2}$$

where α_a is the fractional coefficient to $V_0(f,t)$. Similarly, we have

$$\frac{\partial R_b(f,t)}{\partial t} = \frac{10}{\ln 10}\frac{1}{|V_0(f,t)|^2}\frac{\partial |V_0(f,t)|^2}{\partial t} = \frac{\partial R_a(f,t)}{\partial t}$$

which ends the proof.

Theorem 2.4.2 states that the first order derivative of $R_a(f,t)$ and $R_b(f,t)$ over t are the same. Essentially, the first derivative reflects the direction the function is going, increasing or decreasing. It can be interpreted as an instantaneous rate of change. $R_a(f,t)$ and $R_b(f,t)$ are not necessarily exactly the same in their different amplitudes. Hence, rather than comparing the exact shape of received RF noise, we look into their variations. Besides, the equality in Theorem 2.4.2 exists only when noise power is measured in its logarithmic form, as otherwise $\frac{\partial R_a(f,t)}{\partial t} = \frac{|\alpha_a|^2|V_0(f,t)|}{|Z|}\frac{\partial |V_0(f,t)|^2}{\partial t} \neq \frac{\partial R_b(f,t)}{\partial t} = \frac{|\alpha_b|^2|V_0(f,t)|}{|Z|}\frac{\partial |V_0(f,t)|^2}{\partial t}$.

It is worth mentioning some prior works on establishing intra-body communication (IBC) between wearable devices [134, 52, 159]. The body serves as a transmission medium to host peer-to-peer communications. As these works focus on how to achieve high-speed low-energy data transmission, their goal and basis are quite different from ours. Given that

IBC can provide covert inter-body channels, Roeschlin et al. [116] proposed to utilize these channels for key pairing between wearables. However, there are at least two restrictions. First, they need to design their own transceivers for IBC. Second, covert channels only exist for low-frequent electromagnetic signals. Signals above 100 MHz can be easily radiated to the environment, rendering the whole procedure vulnerable to eavesdropping attacks. Unfortunately, most wearables operate over 2.4 GHz nowadays.

2.5 Feasibility Study

The objective of this section is to investigate the feasibility of leveraging ambient RF noise for wearable device paring via extensive measurement studies. Essentially, we need to validate two substrates. First, on-body and thus legitimate devices should be capable of extracting from RF noise common features. Second, it is infeasible for an adversary to do so.

2.5.1 Measurement Setup

Our experiments are conducted using Arduino nano boards and CC2500 radio chips [59]. Several watch-like wearables have been built as shown in Figure 2.4. CC2500 is an RF transceiver that operates over the 2.4 GHz ISM band and designed for very low-power wireless applications. Its role here is to collect and sample RF noise captured by the human body. A conductive wire is wound back of CC2500 to create physical contact between the CC2500's antenna and the wearer's skin. In this way, RF noise, which is first captured by the human body from the open air, is then conducted to the wearable transceiver through the wire, as demonstrated in Figure 2.3. Each CC2500 is associated with an Arduino nano that plays as a micro-controller to store and process the received noise. Arduino nano is powered by a lithiumion polymer battery. The system samples the noise at a rate of 7000

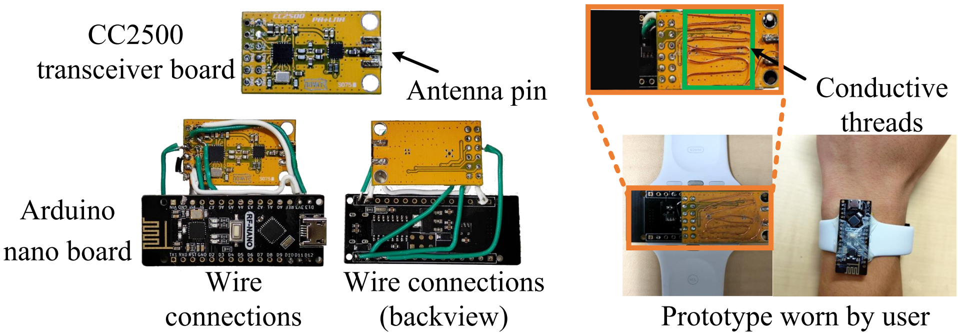


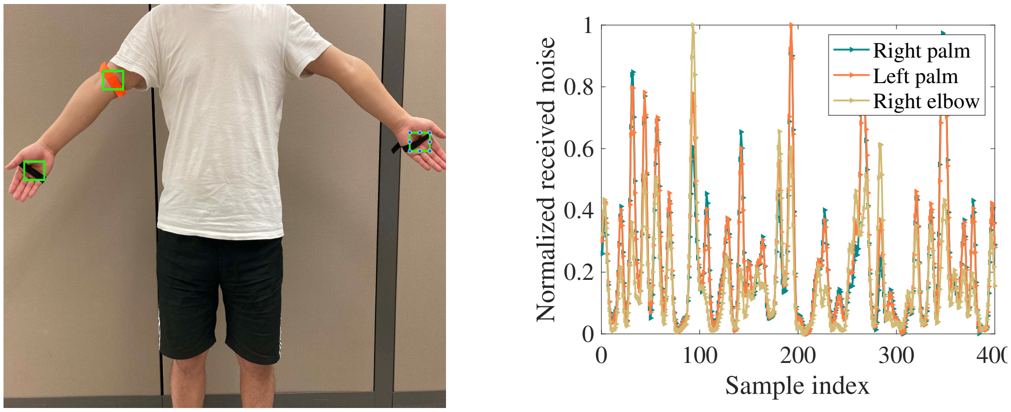
Figure 2.4: System setup.

samples/sec. Samples are timestamped using the system clock. A classical flooding time synchronization protocol (FTSP) [90] is implemented for clock synchronization between devices. We build a prototype for the measurement study instead of using COTS wearables. As they seal their transceiver chips inside of the out-shell case, it is challenging to create direct contact between a wearer's skin and transceiver chips, more specifically, antennas.

Although our prototype does not involve any dedicated sensors, such as accelerometer [152, 39], ECG, and EMG sensors [117, 160], we do need a minor modification on hardware. As mentioned above, a conductive wire or an electrode is needed to connect between the body skin and the wearable antenna.

2.5.2 Measurement Results

Devices on the same body. This part verifies the theoretical result of Theorem 2.4.2 that wearables attached to different parts of the body surface sense the same RF noise variation tendency. Three prototypes are used; two of them are held in the wearer's right and left hand palm, respectively, while the third one is attached to the right elbow (shown in Figure 2.5(a)). To examine noise variation tendencies, the raw signal is first fed into a Gaussian filter to remove low-frequency noise caused by, for example, loose skin contact or imperfect measurements. As our design does not rely on signal amplitudes, normaliza-



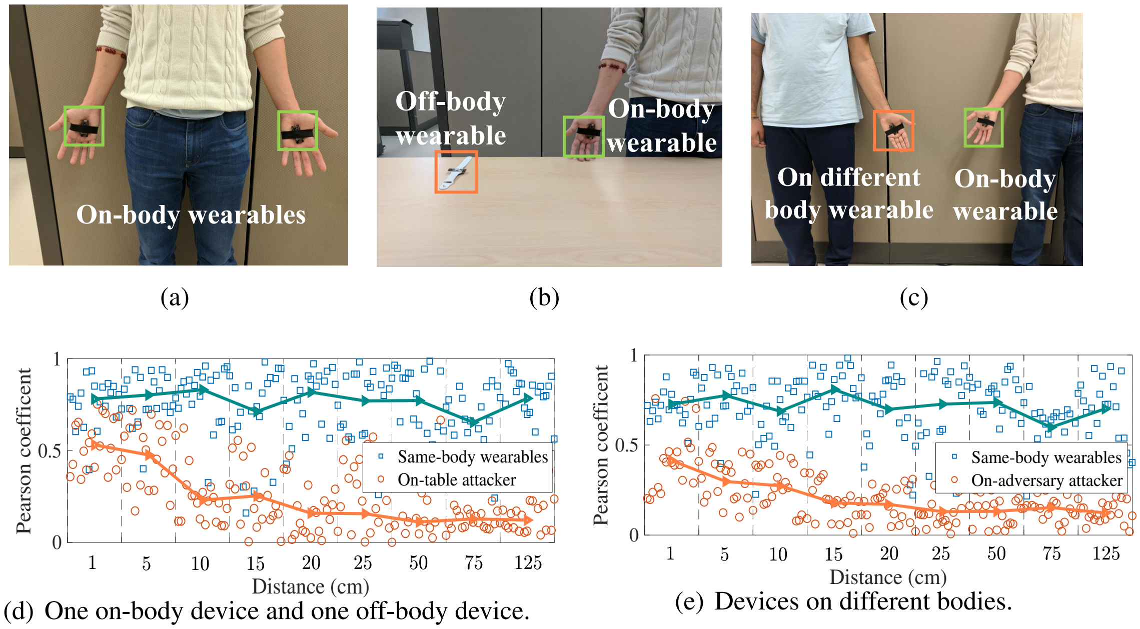
(a) Experiment setup

(b) Received RF noise after filtering and normalization

Figure 2.5: RF noise measures at different parts on body surface.

tion is further applied. Figure 2.5(b) depicts the received RF noise over a frequency band of 2400.0–2400.4 MHz. We observe that the three signals are synchronous across most samples, even when the right palm is about 125 cm away from the left palm. Such a phenomenon validates the basis of our pairing scheme: Wearables can observe almost identical RF noise variations when attached to the same body surface, regardless of their inter-device distance.

One on-body device and one off-body device. This part further shows that measures at an adversary, a device has no physical contact with the wearer, are distinct from the ones obtained at legitimate on-body devices. In the experiment, one device is placed on a table, while the other is held in the wearer's right palm (shown in Figure 2.6(b)). To evaluate the correlation between two measures, we examine their *Pearson correlation coefficients* [99]. Figure 2.6(d) plots the Pearson correlation coefficients by tuning the distance between two devices. As a comparison, the value is around 0.75 when both devices are on the same body (shown in Figure 2.6(a)). Besides, the correlation drops quickly as the distance grows. Therefore, the off-body adversary cannot extract meaningful information for key forgery even within close proximity to the wearer.



(a)

(b)

(c)

(d) One on-body device and one off-body device.

(e) Devices on different bodies.

Figure 2.6: Pearson correlation coefficients of RF noise received at wearables under various placement settings.

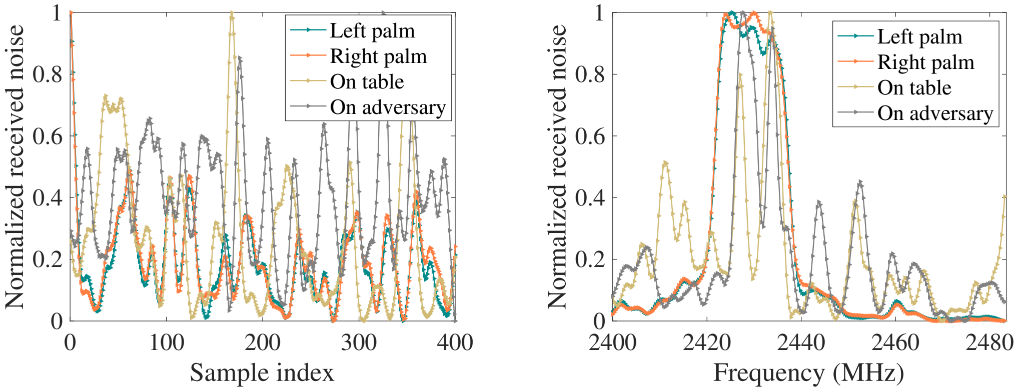
Devices on different bodies. We now demonstrate that an adversary, a device attached to an outlier, is incapable of extracting the same secret as a legitimate device. The setting is shown in Figure 2.6(c). Like above, we first examine the Pearson correlation coefficient of RF noise received at the two devices. Figure 2.6(e) shows that their correlation is relatively low throughout all inter-device distances. Specifically, when two wearers stand as close as 1 cm, the corresponding coefficient is merely 0.47, even lower than the value when the adversary is placed on the table. This is because the unique biometrics of different human bodies introduce another dimension of diversity to RF noise measures.

Figure 2.6 indicates that the adversary cannot generate the same pairing key as the legitimate device due to a lack of common entropy source. In contrast, some prior pairing schemes [149, 87, 77, 83, 139, 93, 164], which rely on propagation characteristics of wireless signals in free space, only work if the adversary is at least half the wavelength

(6.25 cm@2.4 GHz) away from the legitimate device, as otherwise wireless signals highly correlate. Apparently, our approach is free from such a restriction.

RF noise properties in time and frequency domains. Figure 2.7 shows the normalized RF noise in both time and frequency domains. Four devices are used, two on wearer's left and right palms, one on a table, and the last one on a second wearer's left palm. We observe in Figure 2.7(a) that measures from devices on wearer's left and right palms match well in the time domain. Besides, they are well separated from measures from the other two devices. We further show in Figure 2.7(b) RF noise measures across different spectrum bands around 2.4 GHz. A similar relation between these four measures is obtained. Interestingly, a surge is observed from all four measures at the frequency band between 2416 MHz and 2440 MHz, which is exactly the operating frequency of IEEE 802.11 WiFi router in the test room.

The properties of RF noise discussed above are essential for our pairing scheme. Since devices on the same wearer share RF noise measures of high similarity, common secret keys can be extracted. The details of key generation and device pairing will be discussed in Section 2.6. Besides, readings from an adversary, either placed off-body (denoted as type-I imitation attack) or worn by another wearer (denoted as type-II imitation attack), are distinct from the ones measured at the legitimate device even they are in close vicinity. Thus, it is extremely challenging for an adversary to extract a valid pairing key. Moreover, the above properties are consistent across time and frequency domains. It provides an ample choice of time slots and frequency bands from which pairing keys can be extracted.



(a) RF noise in temporal domain

(b) RF noise in frequential domain

Figure 2.7: RF noise properties in time and frequency domains.

2.6 Device Pairing

2.6.1 Overview

As a key component, Alice and Bob seek to produce common secret keys from their RF noise readings. A naive approach is *reciprocal quantization* [148, 87]. Setting two adaptive thresholds q_+ and q_-, sample readings above q_+ are mapped to 1's and those smaller than q_- are mapped to 0's. However, this approach imposes stringent requirement over device synchronization. If two devices are misaligned even by a single bit, the mismatch will be accumulated and result in high error rate eventually. Instead, we propose to utilize the noise variation tendency for key generation. By dividing samples into blocks, we examine the variation of samples in each block so as to tolerate the misalignment over a couple of individual samples. Besides, our scheme adopts the framework of *fuzzy commitment* [65, 95, 50] for key establishment. It is able to transform a secret value s into a commitment/opening value pair (σ, λ), such that σ does not reveal any information about the secret s. Another pair (σ, λ') will reveal s if the Hamming distance $\texttt{Ham}(\lambda, \lambda') \leq t$. It is computationally infeasible to find λ' with $\texttt{Ham}(\lambda, \lambda') > t$ that decommits σ. Due to the employment of Reed-Solomon (RS) codes [145], two devices are able to agree on a common key, if the opening values generated from their received RF noise differ in t bits at most. The value of t depends on the parameterization of the RS code and is tunable to strike a balance between security and usability. Its discussion is provided in Section 2.7.3.

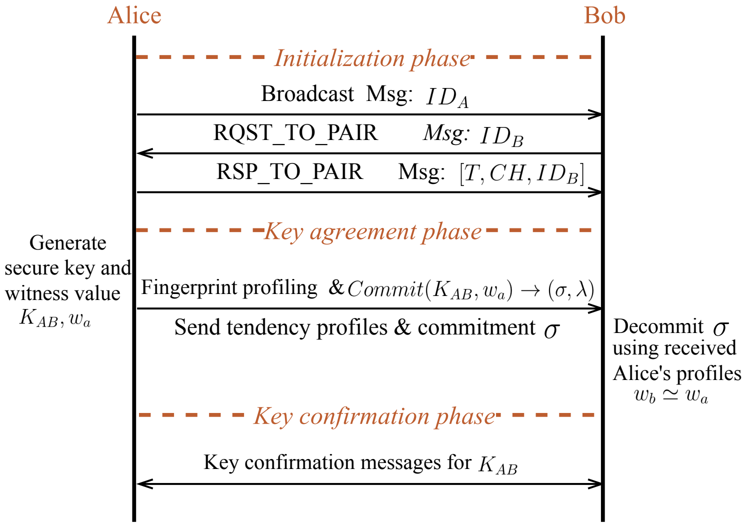

Alice
Bob
Initialization phase
Broadcast Msg: ID_A
RQST_TO_PAIR Msg: ID_B
RSP_TO_PAIR Msg: $[T, CH, ID_B]$
Key agreement phase
Generate secure key and witness value K_{AB}, w_a
Fingerprint profiling & $Commit(K_{AB}, w_a) \rightarrow (\sigma, \lambda)$
Send tendency profiles & commitment σ
Decommit σ using received Alice's profiles $w_b \simeq w_a$
Key confirmation phase
Key confirmation messages for K_{AB}

Figure 2.8: Pairing protocol.

As shown in Figure 2.8, the pairing protocol consists of three phases, initialization, key agreement, and key confirmation.

Initialization phase. Alice broadcasts her identifier ID_A to its vicinity. If a new wearable Bob wishes to pair with Alice, he sends a “RQST_TO_PAIR” message with his identifier ID_B. Alice confirms this request by replying with “RSP_TO_PAIR” and the specification of noise measurement duration T and frequency channel CH.

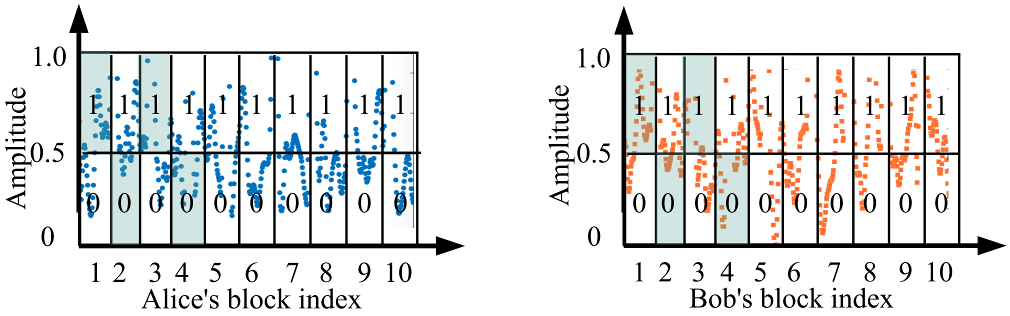
Key agreement phase. Denote by R_a and R_b the noise measurements collected by Alice and Bob, respectively. Alice generates a secret key $K_{AB} \in \{0,1\}^k$ and a *witness* $w_a \in \{0,1\}^k$ using her pseudo-random number generator (PNG). She then turns them into a commitment/opening pair $(\sigma, \lambda) \leftarrow \texttt{comit}(K_{AB}, w_a)$. The opening value λ is calculated as the codeword for K_{AB} using Reed-Solomon (RS) encoding, $\lambda = \texttt{RS}(K_{AB})$. The commitment σ is then calculated as the difference of w_a and λ, $\sigma = w_a \ominus \lambda$, where $\ominus$ denotes a subtraction in a finite field (analogous to an XOR operation). Then, Alice applies the proposed fingerprint profiling mechanism (discussed in Section 2.6.2) to derive her selected fingerprint profile, denoted by P. Alice releases σ and P to Bob. Neither K_{AB} nor w_a can be revealed from the transmitted message by any adversary. Unlike some prior works on device pairings [157, 117], which assume the existence of some secure (but unauthenticated) channel (e.g., TLS) to exchange messages, our protocol does not need such an assumption and is thus more practical for implementation.

Upon receiving the message, Bob produces another *witness* w_b based on P and his received RF noise R_b via the fingerprint profiling mechanism. If Bob is a legitimate device, then $w_b \simeq w_a$. Bob's opening value is calculated as $\lambda' = w_b \ominus \sigma$. If $\texttt{Ham}(\lambda, \lambda') \leq t$, K_{AB} is retrieved by decoding λ' as $K_{AB} = \texttt{RSD}(\lambda')$ using the RS decoding function $\texttt{RSD}()$. It means w_a and w_b can differ in t bits at most, the maximum number of mismatch bits the RS coding can correct.

Key confirmation phase. The aim of this phase is for peers to determine if their established keys are identical. We employ the classic challenging-and-replying protocol [51, 50]. Two devices use their established keys to encode a nonce and send both the nonce and its ciphertext to each other. The key is confirmed if its decoded ciphertext is the same with the nonce.

2.6.2 Fingerprint Profiling Module

This module enables Alice to select her fingerprint profile P that hides the information of *witness* w_a and Bob to recover *witness* w_b from P. A success design ensures $w_b \simeq w_a$ if Bob is a legitimate device. This module consists of components at both Alice and Bob. Our idea is inspired by [149] but different in several ways. For example, the scheme [149] employs instant channel state information (CSI) amplitude as device's fingerprint features. Singular value decomposition is applied to avoid RF noise interference



(a) Fingerprinting with $w_a = 1010$ (b) Recover *witness* as $w_b = 1010$

Figure 2.9: Fingerprint profiling.

for feature extraction. Instead, RF noise is treated as ingredients for key generation in this work. Besides, the noise variation, rather than its amplitude, is considered. As a result, the corresponding feature extraction algorithm will be different.

For Alice, she first segments RF noise samples R_a into a sequence of blocks. Denote by n the size of each block; it stands for the number of successive samples contained in a block. Then, the samples in a block is further divided into two groups G_0 and G_1 by a threshold which is set to 0.5 in this work. Particularly, samples with normalized amplitude above 0.5 belong to G_1; the rest belong to G_0. Figure 2.9(a) illustrates the formulation of blocks and groups. Alice applies the PNG to produce a pseudo-random number $w_a \in \{0, 1\}^k$, called *witness* under the framework of fuzzy commitment. Then, the corresponding group is selected in each block sequentially. In the example shown in Figure 2.9(a), given $w_a = 1010$, it indicates that G_1 is selected at the first and third blocks, while G_0 is selected at the second and fourth blocks, all marked in green.

As discussed in Section 2.4 and 3.5, the variation tendency of RF noise at two legitimate devices shares high similarity. Therefore, by applying the regression analysis over the samples from the same group given the same underlying model, Alice and Bob are able to derive the same set of parameters that characterize the model. For computation efficiency, in this work we apply the *linear regression* model, which is fitted using the least squares approach [20]. Once Alice derives a linear function to fit samples in each of her selected groups, she then gathers their slopes and sample statistics as her selected fingerprint profile, denoted by P.

The operations executed at Bob are similar to Alice. Blocks and groups are constructed based on his R_b. The same linear regression model is applied to derive linear functions and thus their slopes for both G_0 and G_1 in each block. Upon receiving P from Alice, for each element Bob decides which group, G_0 or G_1, in a block produces the closest slope by applying the t-test similarity comparisons [54]. If it is G_0, then a bit 0 is recorded, and 1 otherwise. This step can be viewed as the reverse of operations executed at Alice. Eventually, *witness* w_b is obtained at Bob. In the given example, w_b is equal to 1010 if Bob is a legitimate device.

If Alice intends to deliver a k-bit key K_{AB}, she constructs at least k blocks. The block size n influences the performance of key generation. A small n results in high bit error rate, while a large n reduces the key generation rate.

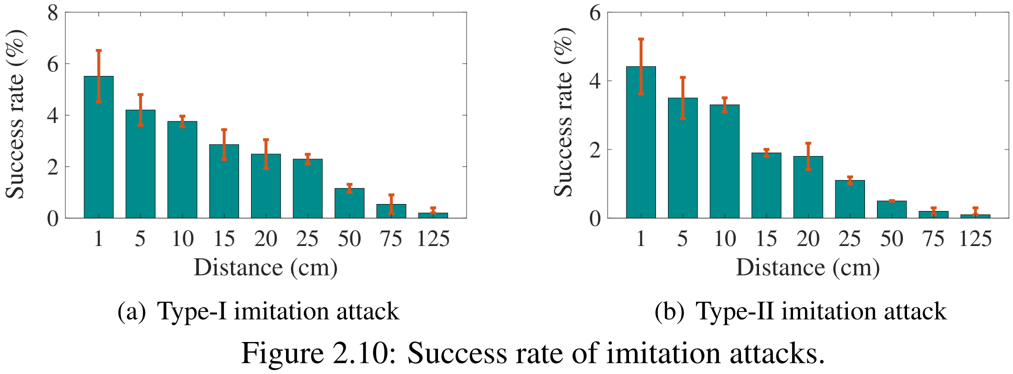
2.7 Evaluation

The experiments are conducted using our prototype devices described in Section 3.5. As human subjects are involved, the entire research has been approved by IRB. Since the prototype is built on a commercial transceiver that follows the FCC regulations, it poses a minimal risk to human health. During the data collection, the transceiver only passively measures the RF noise without injecting any current flow through the body. Subjects are fully informed before voluntarily participating in the experiments. Their discomfort anticipated in the research is not greater than those ordinarily encountered in daily life. All collected data are de-identified and properly stored locally from potential leakage.

The goal is to evaluate both the security and usability of the proposed mechanism. A wide spectrum of impact factors are thoroughly examined, such as system parameters, wireless environments, wearer motion status, and device types. Comprehensive performance comparison is also made with existing works. A total of 6 volunteers, 4 males and 2 females between 25 to 28 years old, are recruited for data collection. Before each experiment, detailed instructions regarding experimental procedures are provided.

2.7.1 Robustness Against Attacks

Imitation attack. In type-I imitation attacks, the adversary aims to derive a valid pairing key from its RF noise measurement for being at the vicinity of the wearer. Since RF noise is highly diverse in the spatial domain, its electromagnetic features are distinct even at two geographic locations with close proximity, as illustrated in our feasibility study. Thus, it is impossible for the adversary to extract an accurate *witness* to decode the session key K_{AB}. In the experiment, we test the adversary's success rate by varying its distance to the wearer. Figure 2.10(a) shows that the maximum success rate of type-I imitation attack is upper-bounded by 5.8% which is achieved at the closest distance to the wearer of 1 cm. The success rate further drops to 0.2% when the distance becomes 125 cm. This phenomenon is consistent with the result of Figure 2.6(d).



(a) Type-I imitation attack

(b) Type-II imitation attack

Figure 2.10: Success rate of imitation attacks.

In type-II imitation attacks, the adversary is attached to another wearer's body. The experiment is carried following the same setting in Section 2.5.2 for our feasibility study.

As shown in Figure 2.10(b), type-II attacker's success rate, 4.3% the maximum, is even lower than type-I attacker. This is because body biometric characteristics introduce an additional layer of uniqueness to the RF noise measures. Therefore, the adversary, attached to an outlier, is also unlikely to extract the same *witness* as a legitimate device to derive the valid session key.

Some existing schemes [92, 83, 149] that rely on wireless signals for device pairing typically assume the adversary no less than half of the signal wavelength away from the legitimate device; otherwise, their received signals will be highly correlated and thus fail the scheme. To be specific, the adversary gains a success rate of 93% at a distance of 5 cm [149], while this value is only 4.2% of our scheme.

Synthesis attack. In this attack, the adversary first models the statistic distribution of RF noise by collecting samples from the radio environment surrounding the wearer for a period of time. Then it fabricates synthetic measurements from the distribution to launch pairing attempts.

In the experiment, Gaussian process is adopted. For each noise sample $R(f, t), t \in T, f \in CH$, the synthetic value is randomly chosen following $\mathcal{N}(\mu, \delta)$. It estimates from collected RF noise measurements to get μ and δ. We generate 10,000 forgery noise samples to attack 120 pairing processes. As shown in Table 2.1, the attacker's success rate is relatively low, 3.2% the maximum. Besides, it gains no advantage for a longer observation period. Since RF noise is highly dynamic, demonstrating memoryless property in the time domain, historical statistics are of little help to predict its future values. Thus, it is unlikely for this type of adversary to extract useful information from historical data analysis. It is noteworthy that some existing approaches use cable radiations [157] and human movements [39, 152] as common entropy for pairing. As these signals exhibit certain patterns, they are vulnerable to synthesis attacks. For example, cable radiations follow a sinuous waveform at the frequency of 50Hz/60Hz. Such a pattern is easily predictable.

Table 2.1: Success rate of synthesis attacks.

Observation duration	10 min	15 min	20 min	30 min
Success rate (%)	2.9	1.8	3.2	2.1

MitM attack. In order to place between Alice and Bob, the adversary must either run the protocol with each of them or interfere, for example, replace or modify at least one of the key agreement messages, in an ongoing pairing session between Alice and Bob. As discussed above, the adversary cannot successfully complete the protocol alone with either Alice or Bob. Besides, any modification of the key messages will also cause Alice and Bob to disagree on the key. For example, if the adversary replaces Alice's commitment σ with its own commitment σ', then Bob derives a different opening value $\lambda' = w_b \ominus \sigma' \neq w_b \ominus \sigma$. The decoded symmetric key $K'_{AB} = \texttt{RSD}(\lambda')$ is different from K_{AB} generated by Alice. It results in a failure during the key confirmation phase. Now the only remaining option for the adversary is to initiate two sessions simultaneously with both Alice and Bob, and then rely on them for key establishment. To succeed, the adversary must correctly guess K_{AB} generated by Alice. The probability is $1/2^{128}$ given $k = 128$, which is negligible.

2.7.2 Key Generation Performances

Source entropy. Entropy characterizes the uncertainty associated with an information source. It is more difficult for an adversary to predict and deduce secrets from high-entropy sources. We examine the *spectral entropy* of RF noise in the experiment. It is calculated by applying the Shannon entropy concept over the power distribution of a given signal. This metric has been widely employed to quantify signal randomness and irregularities in the domain of speech recognition.

Figure 2.11 shows the spectral entropy of RF noise at different time instances. We observe its value is above 0.9 mostly with its average around 0.94. Hence, RF noise demonstrates satisfactory randomness to defend against brute force attacks and guessing attacks

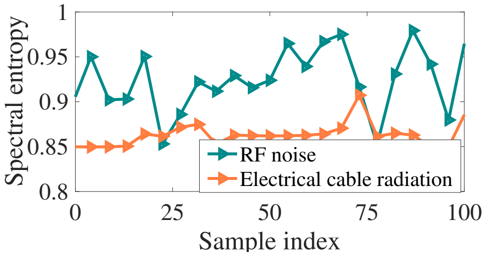


Figure 2.11: Spectral entropy.

on device pairing. For comparison, Figure 2.11 further shows the spectral entropy of electrical cable radiation. Its randomness has been recently explored for paring and wearable device on-body/off-body detection (e.g., [75]). We find that the electrical cable radiation exhibits a much lower spectral entropy around 0.85. As mentioned above, this is because cable radiations mainly follow a sinuous waveform at the frequency of 50Hz/60Hz. Thus, it bears much less uncertainty than RF noise.

Table 2.2: Bit generation rate (bits/sec)

TDS	KEEP	ASBG	Telepathy	Proposed
96	28	13	12	138

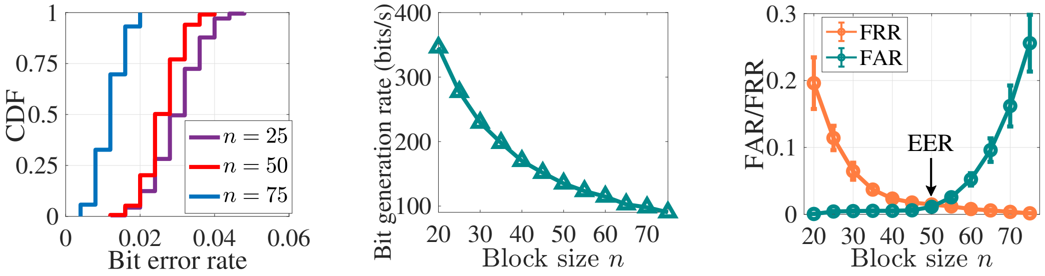
Bit generation rate. Bit generation rate is defined as the number of bits of the key over the time for the entire pairing process. Table 2.2 compares this metric between our scheme and other four schemes, including TDS [149], KEEP [148], ASBG [62], and Telepathy [93]. Specifically, TDS and KEEP leverage the common CSI measurements at two closely located devices for key pairing. ASBG and Telepathy rely on the reciprocity of radio wave propagation for transmission peers to extract common secret keys. For the sake of fairness, we copy the performance of these four schemes from their own papers. We notice that our scheme has the highest rate of 138 bits/sec. KEEP, ASBG and Telepathy have relatively low rates among the five. This is because they employ the *reciprocal quantization* mechanism when converting radio samples into bits. As the mechanism drops a large amount of samples to resolve the quantization ambiguity, it slows down the bit generation process. Although the rate has been improved significantly by TDS, it suffers from a time-consuming information reconciliation process, which is used to reconcile bit mismatch caused by imperfect clock synchronization. Since our scheme utilizes RF noise variation tendency for common secret extraction, imperfect clock synchronization imposes a rather limited impact on key agreement. Take Figure 2.9 as an illustration. Even if clocks at pairing devices are asynchronous by one-sample time duration, it is unlikely to change the slope and the corresponding bit derived from the entire samples in one block. Thus, the tedious information reconciliation process can be avoided.

2.7.3 Impact of Settings

Impact of key parameters. We first examine the impact of block size n on the performances of bit error rate, bit generation rate, and pairing accuracy. Specifically, bit error rate is defined as the number of mismatched bits over the number of all bits generated. False rejection rate (FRR) and false acceptance rate (FAR) are employed to characterize the pairing accuracy. FRR is the probability that a legitimate device is treated as an adversary. It is the ratio between the number of times that a legitimate device is wrongly classified and the total attempts. FAR is the probability that an adversary is treated as a legitimate device. In the experiment, two legitimate wearables are attached to a wearer while an adversarial device locates 20 cm away from the wearer.

Figure 2.12(a) shows the cumulative distribution function (CDF) of the bit error rate for generating 128-bit pairing keys. We observe that a larger n associates with a lower error rate. This is because a larger group size can better tolerate tendency mismatches caused by measurement errors. On the other hand, as shown in Figure 2.12(b), a larger n brings down the bit generation rate. This is because it takes a longer time for collecting enough samples for key generation. We further observe in Figure 2.12(c) that a larger n leads to a lower

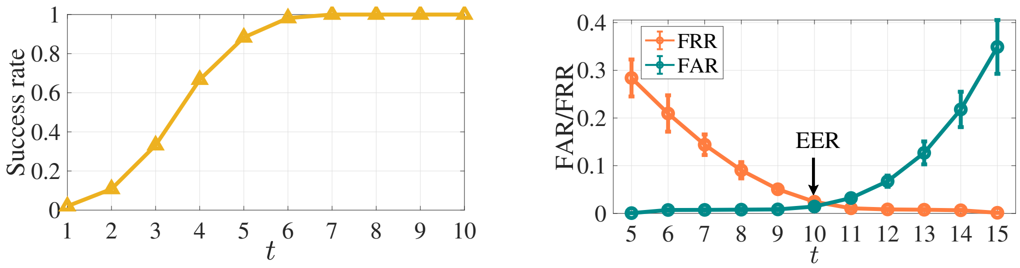
FRR but a higher FAR. This is because a smaller-size block contains fewer samples. The derived fingerprint profile P becomes sensitive to measurement errors. Thus, legitimate devices become easier to be wrongly rejected. Meanwhile, it renders adversaries even less likely to pair. On the contrary, a larger block size better tolerates measurement errors with a lower FRR. Since it indicates a loose detection rule, FAR increases accordingly. We observe that the EER, the point at which FRR and FAR are equal, is 1.4% when $n = 50$. Figure 2.12 provides insights for selecting proper n that strikes a balance among the metrics of bit error rate, bit generation rate, and pairing accuracy.



(a) CDF of bit error rate (b) Bit generation rate (c) Pairing accuracy

Figure 2.12: Impact of block size n.

We then investigate the selection of parameter t for the fuzzy commitment. Recall that t is the maximum Hamming distance between two opening values λ and λ' that recover the same key K_{AB}. Figure 2.13(a) depicts the pairing success rate with respect to t. It is observed that the success rate increases as t grows. It meets our expectation as a larger t tolerates a larger amount of bit mismatches in opening values. It also accounts for why FAR increases as t grows shown in Figure 2.13(b); an adversary becomes easier to get paired. On the other hand, a larger t effectively brings down FRR. EER is equal to 1.6%, when $t = 10$. Combining the results of Figure 2.13, we find $t = 10$ as a suitable setting for implementation, as it produces 96.8% pairing success rate and a low EER of 1.6%.

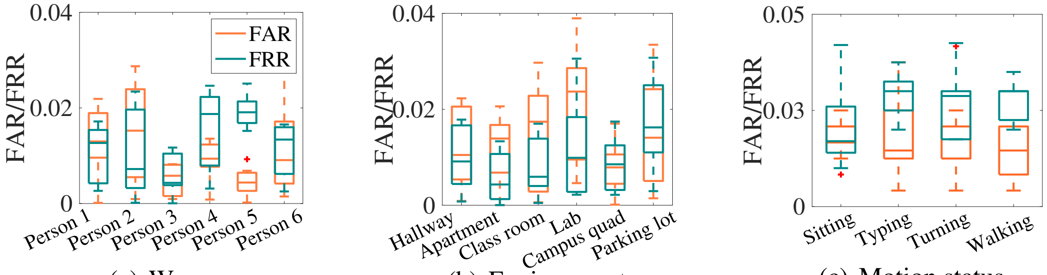


(a) Pairing success rate

(b) Pairing accuracy

Figure 2.13: Impact of fuzzy commitment parameter t.

Impact of wearers. Figure 2.14(a) shows the pairing accuracy across six different wearers. While each individual exhibits slightly different FAR/FRR, the overall performance is relatively consistent, with the average FAR and FRR both under 2.0%.



(a) Wearers

(b) Environments

(c) Motion status

Figure 2.14: Impact of usage settings.

Impact of environments. We further test the scheme at six different types of locations, including hallway, apartment, lab, classroom, campus quad, and parking lot. Figure 2.14(b) shows that the pairing accuracy performance is promising both indoors and outdoors. Since RF noise is ubiquitously accessible, our scheme does not impose any restrictions on its usage environment. In contrast, [157] relies on electrical cable radiation and thus is inapplicable in outdoors due to the unavailability of its signal sources.

Impact of motion status. Since a human body performs various types of motions due to daily activities, it is important to show that the proposed scheme is motion-insensitive. In the experiments, volunteers are asked to perform four types of motions, including sitting, typing, turning directions, and walking. The corresponding pairing accuracy is depicted in Figure 2.14(c). We find that the best performance is achieved at the sitting status with averaged FAR=2.8% and FRR=3.1%, while moving actions slightly bring up the error rate. Still, the pairing accuracy is practically acceptable. Some prior works on wearable device pairing extract common secrets from body movements [39, 152]. The source entropy comes from the randomness of body movements. Their schemes do not work when users are in a relatively static status, say sleeping and sitting.

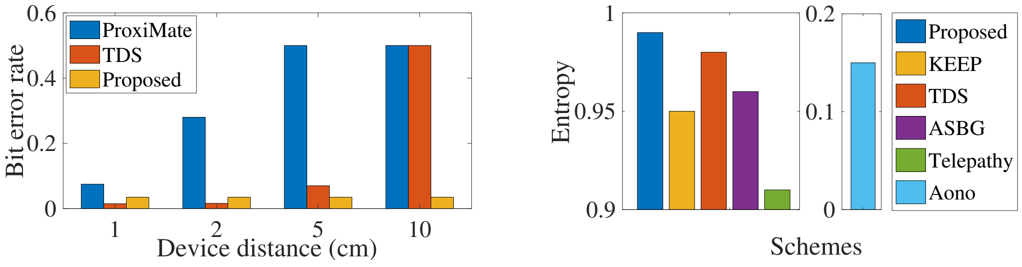
Impact of device placements. In this set of experiments, we evaluate the impact of device placements on the body surface. Several locations are examined, including user's palm, elbow, front head, wrist, and keen. Table 2.3 shows that the FRR is relatively stable for all locations, with the maximum value equal to 2.6%. It meets our expectations. The RF noise measures at different parts of a wearer's skin experience the same variation tendency. Therefore, the pairing performance, reflected by FRR here, is consistent. This is a desirable property. In practice, various wearables are attached to various parts of body skin to collect diverse biosignals. For instance, ECG monitors are sometimes attached to the chest. Then the above-mentioned prior designs [39, 152] that relay on body movements for key extraction do not work in this case as no significant movement is observable in the chest area.

Table 2.3: Impact of device placements.

Placement	R. elbow	F. head.	L. wrist	L. keen
FRR	2.1%	1.6%	1.8%	2.6%

2.7.4 Comparison with Other Pairing Schemes

In addition to the bit generation rate in Section 2.7.2, we present the performance comparison with prior works on bit error rate and key entropy. For the sake of fairness, we directly utilize the experimental results from these works.



(a) Bit error rate

(b) Key entropy

Figure 2.15: Comparison with other pairing schemes.

Figure 2.15(a) compares the bit error rate with two other schemes, ProxiMate [92] and TDS [149]. As mentioned previously, TDS leverages common CSI measurements at devices in close proximity to establish their symmetric keys, whereas ProxiMate utilizes FM radio and TV signals. The bit error rate is examined by tuning the inter-device distance. ProxiMate and TDS experience a surge in error after the distance surpasses certain thresholds. This is because the common secrets can only be extracted at two antennas within half wavelength. On the other hand, our pairing scheme is independent of the inter-device distance as long as they have physical contact to the same wearer.

Figure 2.15(b) compares the entropy of generated keys. Entropy reflects the randomness of keys from the perspective of uncertainty. Recall that Figure 2.11 shows the entropy of raw signals. We find that Aono [7] has the lowest entropy among the six, as it directly turns raw measurements into secret bits and raw signals are correlated in the temporal domain. To address this issue, KEEP, ASBG and Telepathy employ the reciprocal quantization mechanism; a certain amount of correlated signals are discarded during quantization. As shown, it effectively improves the entropy. The proposed scheme and TDS have the highest entropy, approximate to 1, since their keys are produced by PRN generators. As a note, our scheme extracts from RF noise the *witness* values instead of the key itself.

CHAPTER 3

PERISCOPE: A TRAINING-FREE KEYSTROKE INFERENCE ATTACK USING HUMAN COUPLED ELECTROMAGNETIC EMANATIONS [1]

3.1 Introduction

Mobile devices, such as smartphones and tablets, have penetrated into everyday life. They are commonly used to enter sensitive inputs with virtual keyboards, including bank card number, security code, and digit PIN. Prior research has shown that these secrets entered at keyboards can be inferred from onboard motion sensor readings [17, 105, 96, 155, 88, 82], acoustic signals at microphones [123, 167, 78, 86, 14, 41], video recordings [144, 143, 22, 111, 12, 11, 122, 154], and radio signals captured by surrounding wireless infrastructures [74, 4, 165, 37, 76]. To access these side channels, most existing works have to impose strong assumptions over attacker's capabilities or attacking scenarios. For example, motion sensor based attacks require the pre-installation of certain malware to victim's device to access sensor readings. Video based attacks rely on the line-of-sight (LoS) view of the typing process or object of interest that reflects typing motions. Radio signal based attacks analyze reflected signals to characterize environment disturbance caused by finger movements to learn which key is pressed. It cannot tolerate any background context changes, as otherwise, the subtle signal fluctuations introduced by finger movements

[1] Wenqiang Jin, Srinivasan Murali, Huadi Zhu, and Ming Li, "Periscope: A Training-Free Keystroke Inference Attack Using Human Coupled Electromagnetic Emanations." Submitted to ACM SIGSAC Conference on Computer and Communications Security (CCS), 2021.

are easily buried under large-scale signal variations caused by environment dynamics. All the above restrictions render many existing keystroke inference attacks impractical in real-world scenarios.

In this work, we present an attack that leverages electromagnetic (EM) emanations leaked from device's touchscreens to snoop keystrokes. While the EM emanations have been explored for keystroke inference attacks [136, 138, 33], previous efforts have been focused on physical keyboards. When a key is pressed, the keyboard sends a packet of information known as a *scan code* to the computer. The scan code is bound to a physical button on the keyboard. The information leakage threat exists because part of the internal circuit acts as an antenna and radiates unintentional encoded information in EM waves. The attacker can easily reproduce each keystroke by relating it to its unique EM wave pattern. For virtual keyboards on mobile devices, their working principle is quite different. The way to recognize a keystroke does not rely on the scan code, but rather the current changes in the electrode grid. (Details will be covered in Section 3.4.1.) Thus, the fingerprinting EM leakage from a specific physical button no longer exists.

For the first time, our attack analyzes touchscreen's EM emanations under the *human coupling effect*. As suggested by [113, 157], a human body can be treated as a conductor with low impedance (a few $k\Omega$). When a user's finger approaches the screen, it generates a radiative coupling with the touchscreen's circuit. A portion of electric charges are extracted from the electrode grid to the finger through the coupling capacitance. As the finger moves over a screen to enter inputs, it changes the coupling capacitance. Consequently, it influences the touchscreen's EM emanations, which can be detected by a remotely located eavesdropper. Our attack is built on this phenomenon to map EM emanation fluctuations with finger movements for performing keystrokes. Compared with state-of-the-art inference attacks, our scheme is more practical to execute from the following aspects. First, it eavesdrops keystrokes in a non-invasive way. Hence, it avoids the requirement to infect the victim device in advance of the attack. Second, as EM emanations can easily penetrate through obstacles, no LoS view is needed to launch the attack. Third, since our attack relies on direct EM radiations from touchscreens rather than reflected signals, it is robust against environment dynamics. We name our proposed attack as Periscope as it can observe and disclose victim's keystrokes covertly without a LoS view.

Despite these promising features, harnessing EM emanations for keystroke inference still faces a significant challenge, that is, to establish a relationship between observed EM emanations and a specific key press. A straightforward solution is to build a learning model that maps between these two. Under this framework, the attacker first needs collect labeled dataset of a reasonable size and train the model properly. During the attack phase, unknown EM emanations are fed into the trained model as inputs, with the output as which key was most likely pressed. In fact, this approach is adopted in most existing acoustic and radio signal based inference attacks [74, 4, 165, 37, 76, 123, 167, 78, 86, 14, 41]. However, training significantly hinders the deployment of these attacks. As users' typing behaviors are distinct, user-dependent inference models are preferred to capture this uniqueness. It requires either access to the victim's device for some time or possession of her labeled dataset.

To avoid the training hurdle, we aim to develop an analytic model that characterizes the relation between EM emanations and keystrokes. To facilitate the analysis, we divide the continuous EM readings of entering the entire PIN into several segments, each associated with one key pair. By looking into the equivalent circuits of the touchscreen with finger coupling, we first derive the closed-from expression between realtime EM readings and instant finger-screen distances. Nonetheless, the latter may not directly reflect specific keystrokes. To fill the gap, we further estimate the finger movement speed and direction of entering one key pair. With these parameters, time-dependent finger-screen distances are equivalently transformed to a 3D finger movement trajectory, which are further cast to two 2D planes. The projected trajectories reveal finger movement lengths for entering one key pair in both horizontal and vertical directions on the screen. After such projection and transformation, we establish an explicit relation between EM readings and finger movements. This knowledge requires no prior labeled dataset from a specific user and is completely training free. Meanwhile, we notice that different key pairs may share an identical finger movement trace. To alleviate the inference ambiguity, we propose to explore the inter-dependency between consecutive key pairs to narrow down possible keystrokes. We model the entire PIN entering process as a Hidden Markov Model (HMM), with the recovered finger movement traces as observations, whereas the exact key pairs as hidden states. Finally, HMM outputs a list of PINs ranked based on their probability of being the target PIN.

To evaluate the proposed Periscope, we build a prototype with an Arduino board [57] and a conductive wire, with the total cost around $10. Extensive experiments show that our Periscope achieves a recovery rate over 6-digit PINs of 56.2% at a distance of 90 cm. Tests also show that Periscope is robust against environment dynamics and transparent to attacker displacement. Besides, it stays effective for a diverse set of devices and environment context. We summarize the contributions of this paper as follows.

- We investigate a novel side-channel attack to eavesdrop user's digit inputs on mobile devices by analyzing human-coupled EM emanations from touchscreens. While EM emanation based inference attacks have been studied on physical keyboards before, they are inapplicable to virtual keyboards due to their distinctive working principles.
- By analyzing touchscreen circuits under the human coupling effect, a closed-form expression is derived to characterize the relation between EM readings and finger movements. With the analytic model, keystrokes can be easily recovered from EM readings without training hurdles.

- We develop a prototype and demonstrate the severity of the threat. It outperforms state-of-the-art inference attacks in terms of setup practicability with much fewer deployment restrictions. More importantly, the total cost of the prototype is as low as $10.

3.2 Related work

Existing keystroke inference attacks that explore side-channel information can be broadly classified into the following categories.

Motion sensor based attacks. Efforts have been made on inferring user's keystrokes from data generated by on-board motion sensors. Early works [17] and [105] utilize mobile device's accelerometer readings to infer victim's passwords. By further involving gyroscope, [96] and [155] are able to increase the attack success rate. In this line of research, some recent works [88, 82] show that the similar idea can be applied to wearables to snoop victim's inputs. However, these attacks cannot succeed unless the victim device is pre-installed with certain malware to acquire motion sensor data, limiting their applicability.

Acoustic signal based attacks. Some keypads such as ATM inputs and door keypads provide an audio feedback to the user for each button pressed. Such audio feedback is observable from a fair distance. Prior works [14, 41] quantify the delays between feedback pulses to reconstruct the keystrokes. This type of attack is susceptible to acoustic background noise. Besides, not all keypads emit audio feedback. Another line of research infers user inputs by employing acoustic ranging techniques. They utilize microphones to locate finger taps and thus the corresponding buttons on a screen [123, 167, 78]. It is not easy to derive an analytic model that characterizes finger movement trajectory with respect to audio sound. Researchers [168, 9, 137] have to resort to machine learning techniques and train classifiers to reconstruct the keystrokes so far. Tedious data sample collection and offline training process are unavoidable.

Video based attacks. Empowered by advanced computer vision techniques, video based attacks have been investigated for a while. Its idea is to use cameras to record the typing process or an object that reflects typing motion and then identify inputs by analyzing the recorded video. Prior works have demonstrated the feasibility of launching inference attacks by recording hand movement [122, 154], eye movement [144, 143, 22], tablet backside motion [126], reflections from nearby objects (e.g., glasses and plastic bottle) [111, 12, 11]. In these attacks, cameras should have a LoS view for object of interest; otherwise, keystroke activities cannot be detected. Besides, this type of attack does not work under poor lighting conditions.

Radio signal based attacks. Emerging research efforts have been made on eavesdropping keystrokes from radio signals due to the wide deployment of wireless infrastructures (e.g., WiFi and cellular towers). In particular, prior works [74, 4, 165, 37] reveal victim's keystrokes via the WiFi channel state information (CSI). Ling et al. [76] recovered the typed PIN on an ATM by analyzing the reflected cellular signals. As they rely on wireless infrastructures to launch the attack, the signal strength is relatively strong. Hence, the attacking distance is up to several meters. On the other hand, as radio signals are highly susceptible to environmental dynamics, these attacks cannot tolerate any changes in the environment other than the victim's hand or finger movement. Periscope utilizes EM radiations from device's touchscreen to recover keystrokes. Hence, no extra wireless infrastructure is needed. Because the effective sensing distance of touchscreen's EM radiations is around 1 meter, its attacking distance is constrained within this range too. Like acoustic signal based attacks, due to the complexity of formulating the relationship between observed wireless disturbances and specific key presses, radio signal based attacks also require a training phase to be effective. Recently, Fang et al. [37] proposed a training-free keystroke inference attack by leveraging structures of dictionary words. They built a prototype with USRP, with a total cost around several thousand dollars.

EM emanation based attacks. EM radiations unintentionally leak from electronic devices. It has been investigated as a side channel to infer victim's keystrokes on physical keyboards [136, 138, 33]. Notably, each key is associated with a unique scan code. Once it is pressed, the PC recognizes the key by reading the imported information through the data cable. The attack is based on the observation that the encoded keystroke information is radiated to the open air the form of EM emanations as it is transmitted over the cable. The working principle of soft keyboards is different. A keystroke is recognized by locating the touched position on a screen surface from current changes. Therefore, the existing eavesdropping method toward physical keyboards is inapplicable here. For the first time, Periscope examines the EM radiation changes caused by human coupling effects when a finger performs keystrokes. We then build a mapping relation between EM emanations and finger movement trajectory which serves as the foundation of our attack.

3.3 Adversary Model

Attack scenario. The attack scenario is considered as that an adversary seeks to infer a victim's secret PIN by eavesdropping her keystrokes on a mobile device. The victim places her device on a table and types on a soft numeric keyboard on the screen, as shown in Figure 3.1. Such scenarios are prevalent in daily life, such as in a library or a cafe where users unlock their smartphones by entering digit PINs. The attacker is in physical proximity to the victim. It is well concealed, e.g., placed underneath a table or in a bush nearby [91]. We focus on soft numeric keyboards with a classic layout, though the attack can target other layouts just as easily.

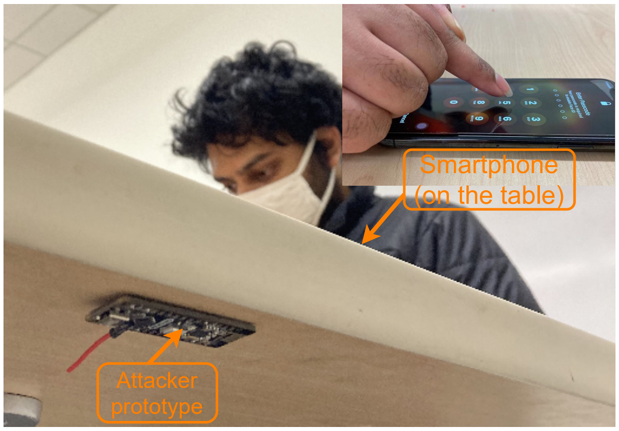


Figure 3.1: Eavesdrop EM emanations using an attack device.

What an attacker cannot do. Unlike many prior keystroke inference attacks, the attacker does not necessarily have a LoS view of victim's keyboard or any other object of interest, such as hand movement, eye movement, and tablet backside motion. We do not assume the existence of any covert channel that reveals victim's onboard sensor readings to the attacker either. Also, there is no ideal environment, static or quiet, to launch attacks. The victim can make free body movements during the typing process; other people may walk by or talk in the background. Besides, it is unlikely for an attacker to collect a large amount of data samples from a specific victim to train an individual keystroke inference model properly before the attack. The above settings render most of the existing keystroke inference attacks infeasible.

What an attacker can do. The attack is able to figure out which mobile device a victim is using and thus its numeric keyboard layout. In practice, the attacker can investigate the MAC address of the victim's WiFi traffics to obtain the device manufacturer information by looking up prefixes of MAC addresses [31]. Besides, the victim's DNS responses contain its device name [102, 53, 169]. Most mobile devices can be fingerprinted with the information above. Prior work [91] provides technique details on setting up a free WiFi access point to access the victim's MAC address and DNS responses for device fingerprinting unnoticeably.

3.4 Preliminaries

3.4.1 How Do Touchscreens Work?

The majority of current mobile devices, such as smartphones and tablets, are equipped with touchscreens. While there are various sensing touch technologies, mutual capacitive sensing has been the most prominent due to its high sensitivity, energy efficiency, and low manufacturing cost [108]. We thus pertain our discussion to this type of touch-sensing devices in this paper.

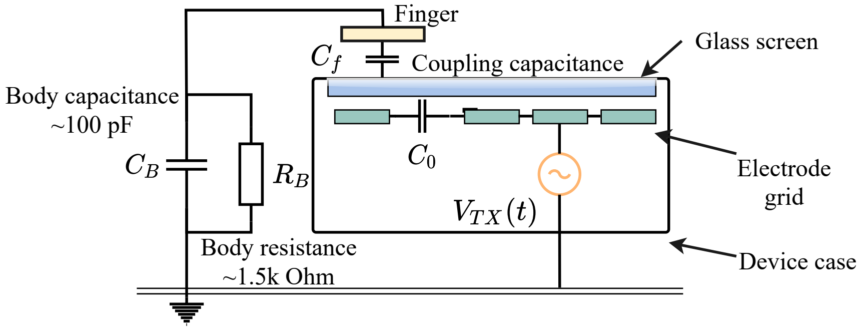


Figure 3.2: The composition of a mutual capacitive touchscreen.

As shown in Figure 3.2, a capacitive touchscreen consists of a grid of transmitter (TX) and receiver (RX) electrodes, which are mutually coupled with a capacitance of C_0. TX electrodes are driven by an alternating voltage signal $V_{TX}(t)$, which creates an alternating current flow from TX to RX electrodes. When a finger touches the screen, it extracts some electric charges from the electrode grid to the human body through a coupling capacitance C_f. The touchscreen controller monitors the changes in the current that flows into RX electrodes and reports the change as a touch event to the system OS. Meanwhile, it locates the current change in the electrode grid as the touched position on the screen. The input is then recognized accordingly.

3.4.2 Touchscreen EM Emanations and Measurements

The alternating currents between touchscreen's TX and RX electrodes generate time-variant EM fields that continuously emit EM radiations to the open space. Periscope intends to map the radiation to user's typing inputs.

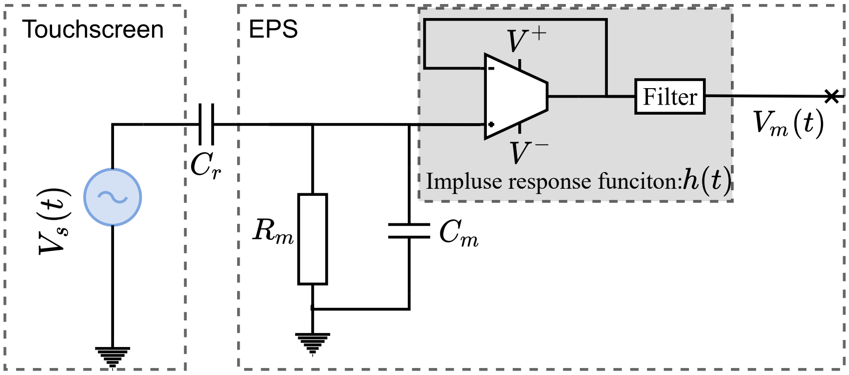


Figure 3.3: The circuit for touchscreen's EM emanation measurement.

Figure 3.3 depicts an equivalent circuit of using an electric potential sensor (EPS) to measure touchscreen's EM emanations. An EPS typically consists of a capacitor C_m, a resistance R_m, a voltage amplifier, and a low-pass filter. By placing the eavesdropper, i.e., EPS, within the EM field of victim's touchscreen, these two will be remotely coupled via a small capacitance C_r. Denote by $V_s(t)$ the time-variant voltage that drives EM emanations from the touchscreen. The captured EM emanation at EPS, measured in electric potential changes $V_m(t)$, is expressed as

$$V_m(t) = V_s(t) \cdot \frac{1/(\frac{1}{R_m} + j2\pi f C_m)}{\frac{1}{j2\pi f C_r} + 1/(\frac{1}{R_m} + j2\pi f C_m)} \cdot h(t). \tag{3.1}$$

Here $h(t)$ denotes the joint impulse response of the amplifier and the low-pass filter. f stands for the frequency of the driving voltage $V_{TX}(t)$. Among the parameters in (3.1), C_m, R_m, and $h(t)$ are fixed values. C_r depends on the attacker-victim distance. It can be treated as a fixed value too under a specific eavesdropping event. Now $V_m(t)$ is determined by $V_s(t)$. As demonstrated next, $V_s(t)$ is impacted by finger movement. Hence, we establish a connection between EM readings and finger movement. To validate this claim, we show in Figure 3.4 the spectrogram of EM readings $V_m(t)$ when a user enters a 6-digit PIN. There are 6 bars with intense magnitude, each representing the tap of one key. We also notice that the majority frequency components are scattered at the lower end of the spectrum band, below 60 Hz. It indicates that EM emanations can be easily captured by cheap EPS with a fair sampling rate.

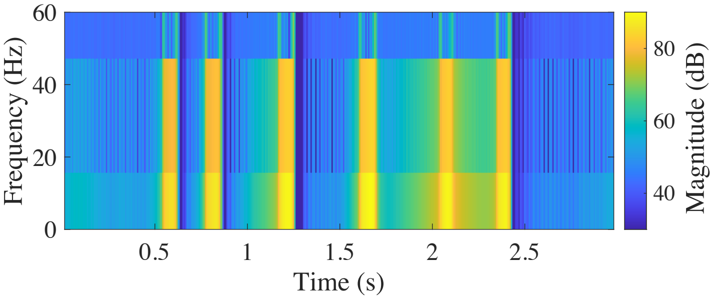
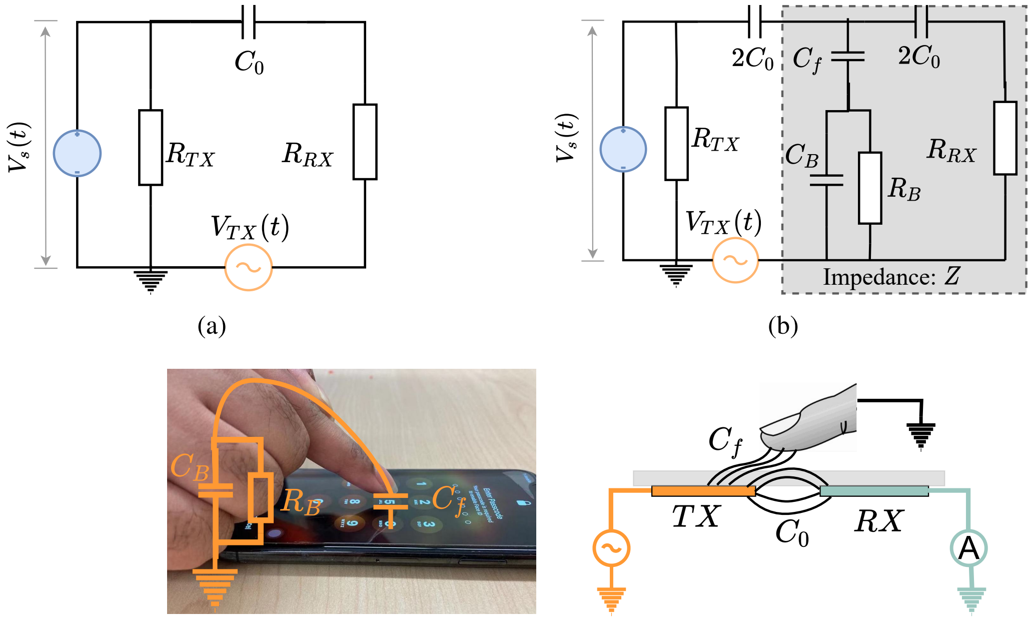


Figure 3.4: Spectrogram of EM emanation measurement $V_m(t)$.

3.4.3 Impact of Finger Coupling

The driving voltage of touchscreen EM emanation $V_s(t)$ is influenced by finger coupling that is modeled next.

Figure 3.5(a) is an equivalent circuit for a mutual capacitive touchscreen when no touching. R_{TX} (R_{RX}) represents the resistor at the TX (RX) electrode. Recall that C_0 is the TX-RX coupling capacitance. The equivalent circuit is transformed to Figure 3.5(b)



(a)

(b)

(c)

Figure 3.5: Illustration of finger coupling effect. (a) Equivalent circuit without finger touches. (b) Equivalent circuit with finger touches. (c) Screen-finger coupling.

when touching. As a finger moves close to the screen, they become remotely coupled via capacitance C_f. As shown in Figure 3.5(c), the finger extracts some electric charges through coupling to the human body (characterized in C_B and R_B). We call the above phenomenon as *finger/human coupling effect*. When a finger is coupled to the screen, $V_s(t)$ is expressed as

$$V_s(t) = V_{TX}(t) \cdot \frac{R_{TX}}{R_{TX} + 1/j4\pi f C_0 + Z(t)} \tag{3.2}$$

where $Z(t)$ denotes the equivalent time-variant impedance of the right-half circuit of Figure 3.5(b)

$$Z(t) = 1/(\frac{1}{1/j2\pi f C_f(t) + 1/j2\pi f C_B + 1/R_B} + \frac{1}{1/j4\pi f C_0 + R_{RX}}). \tag{3.3}$$

Let $z(t)$ be the instant finger-screen distance. According to [19], $C_f(t)$ can be expressed as

$$C_f(t) = \frac{\epsilon_0 \epsilon_r A}{z(t)}, \tag{3.4}$$

where ϵ_0 and ϵ_r are dielectric permeability coefficients. A is the overlap area between the fingertip and the screen. As ϵ_0, ϵ_r and A are fixed values in one keystroke, $C_f(t)$ is negatively correlated with $z(t)$. Together with (3.2) and (3.3), we have the following relation $z \downarrow, C_f \uparrow, Z \downarrow, V_s \uparrow$. In short, the touchscreen emits stronger EM emanations when the finger moves closer to it and vice versa. According to (3.1), V_m has a positive correlation with V_s. We thus have $z \downarrow, C_f \uparrow, Z \downarrow, V_s \uparrow, V_m \uparrow$. This relationship chain indicates that the finger coupling effect reveals a side channel to monitor finger movements: remote EM emanation measurements $V_m(t)$ reflect finger's realtime distance to the screen $z(t)$ when performing keystrokes.

We summarize all symbols and their definitions involved in this paper in Table 3.1.

3.4.4 How to Calculate $z(t)$ from $V_m(t)$?

While the above analysis exhibits a negative correlation between $z(t)$ and $V_m(t)$, we seek to further quantify this relationship, i.e., how to calculate $z(t)$ from $V_m(t)$ exactly? Essentially, our goal is to derive a closed-form expression of $z(t)$ as a function of $V_m(t)$ via (3.1)-(3.4). Nonetheless, this task is nontrivial. Some parameters in (3.1)-(3.4), such as R_m, C_m, C_r, and $h(t)$, are not readily available. For example, C_r is determined by the placement of the victim device and EPS. To resolve this issue, our trick here is to utilize multiple measurements that can cancel out the unknown parameters during the calculation.

Table 3.1: Summary of symbols and definitions.

Symbol	Definition
C_0	TX-RX coupling capacitance
C_f	Finger-screen coupling capacitance
C_r	Screen-EPS coupling capacitance
C_m	Capacitance of EPS
C_B	Body capacitance
R_m	Resistance of EPS
R_B	Body resistance
R_{TX}	Resistance of TX electrodes
R_{RX}	Resistance of RX electrodes
$V_{TX}(t)$	Driving voltage of TX electrodes
$V_s(t)$	Driving voltage of EM emanations
$V_m(t)$	EM measurement
$Z(t)$	Equivalent impedance in Figure 3.5(b)
$z(t)$	Finger-screen distance.
z_{min}	Minimal finger-screen distance
ϵ_0, ϵ_r	Dielectric permeability coefficients
$h(t)$	Impulse response function of EPS

As a note, EPS measures $V_m(t)$ in its amplitude, denoted as $|V_m(t)|$. Let $|V_m(t)|^*$ be the maximum value of $|V_m(t)|$. It is obtained the moment that a finger touches the screen. $|V_s(t)|$ and $|V_s(t)|^*$ are defined similarly. We have

$$\frac{|V_m(t)|}{|V_m(t)|^*} \overset{①}{=} \frac{|V_s(t)|}{|V_s(t)|^*} \overset{②}{=} \frac{|R_{TX} + 1/j4\pi f C_0 + Z(t)|^*}{|R_{TX} + 1/j4\pi f C_0 + Z(t)|} \tag{3.5}$$

where ① and ② are due to (3.1) and (3.2), respectively. As suggested by [43, 72], C_f and C_0 are generally very small, around 2 pF (2×10^{-12} F). Thus, their equivalent impedance is much larger than the body resistance R_B (around $1.5\ k\Omega$ [135]), the body capacitance C_B

(around 100 pF), as well as the resistance of electrodes R_{TX} and R_{RX} (around 160Ω [72]). Then, (3.5) is rewritten as[2]

$$\frac{|V_m(t)|}{|V_m(t)|^*} \simeq \frac{|1/j4\pi f C_0 + Z(t)|^*}{|1/j4\pi f C_0 + Z(t)|}, \tag{3.6}$$

Similarly, $Z(t)$ is approximated as

$$Z(t) \simeq 1/(\frac{1}{1/j2\pi f C_f(t)} + \frac{1}{1/j4\pi f C_0}) = 1/(j2\pi f C_f(t) + j4\pi f C_0). \tag{3.7}$$

Let the maximum finger coupling capacitance be C_f^*. In practice, manufacturers tend to set the TX-RX coupling capacitance C_0 approximate to C_f^*, called *impedance matching*, so that the touchscreen circuit tend to generate large current changes in the electrode grid. It helps to improve accuracy of touch event detection [43, 58, 72]. We thus have $C_f^* \simeq C_0$, by which C_f^* is achieved under the minimum finger-screen distance $z_{\min}$. From (3.4), we have $\frac{C_f(t)}{C_f^*} = \frac{z_{\min}}{z(t)}$ which leads to

$$C_f(t) = C_f^* \frac{z_{\min}}{z(t)} \simeq C_0 \frac{z_{\min}}{z(t)}. \tag{3.8}$$

Combining (3.6) - (3.8), we have

$$\frac{|V_m(t)|}{|V_m(t)|^*} \simeq \frac{|1/j4\pi f C_0 + Z(t)|^*}{|1/j4\pi f C_0 + Z(t)|} = \frac{\frac{z_{\min}}{z(t)} + 2}{\frac{z_{\min}}{z(t)} + 4} \cdot \frac{5}{3}. \tag{3.9}$$

and thus

$$z(t) = 1/(\frac{2}{1 - \frac{|V_m(t)|}{|V_m(t)|^*}\frac{3}{5}} - 4) \times z_{\min}. \tag{3.10}$$

[2] Given two complex values $a + jb$ and $c + jd$, if $b >> a$ then $|a + jb + c + jd| \simeq |jb + c + jd|$ since $\sqrt{(a+c)^2 + (b+d)^2} \simeq \sqrt{(c)^2 + (b+d)^2}$.

z_{min} is essentially the thickness of touchscreen's covering glass. It can be determined once the device manufacture information is figured out. For example, z_{min} is equal to 0.55 mm for iPhone 11 [70, 66]. $|V_m(t)|^*$ is the maximum measurement the EPS captured. Hence, it can be treated as a known value.

So far, we are able to express $z(t)$ into a function of $V_m(t)$. Given an instant EM emanation measurement, the corresponding finger-screen distance can be obtained following (3.10). More importantly, no training phase is needed. Unlike many wireless signal based inference attacks, our analytic model is transparent from underlying signal propagation channel conditions, as they have been incorporated into $|V_m(t)|$ and $|V_m(t)|^*$. Their impact is canceled with each other during the calculation. Still, the attacker cannot infer victim's typing inputs from $z(t)$ directly, unless it has the full knowledge of the finger movement trajectory. We present how to derive the latter from $z(t)$ in Section 3.6.

3.5 Measurement Study

The objective of this section is to validate the analytic result of Section 3.4 and investigate the feasibility of leveraging human-coupled EM emanations to launch keystroke inference attacks.

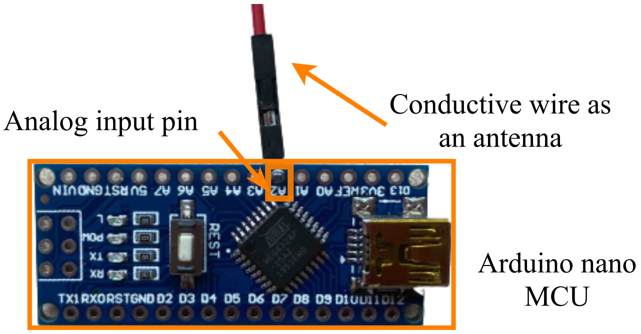


Figure 3.6: Prototype of Periscope.

We build our prototype using an Arduino nano board [57] as a microcontroller unit (MCU) and a conductive wire as an antenna. These two are connected via Arduino's analog input pin shown in Figure 3.6. The antenna senses the electric potential changes caused by touchscreen EM emanations. The system samples received signals with an analog-to-digital (A/D) converter at a rate of 4000 samples/sec. Recall that the frequency of touchscreen EM emanations is bounded within 60 Hz according to the spectrogram analysis in Section 3.4.2. Therefore, the prototype's sampling rate is more than enough to capture signal variances in EM emanations. The entire prototype costs less than $10, which renders the attack easily accessible and widely deployable.

To validate the analytic model for $z(t)$ derived in Section 3.4.4, Figure 3.7 compares it with the ground truth measurement. It is observed that the former generally complies with the latter. Meanwhile, the approximation operations involved in the derivation process do introduce some marginal discrepancies between these two. We plan to investigate its impact on the attack performance in experimental evaluations.

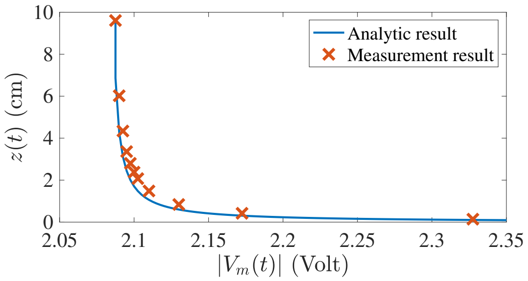


Figure 3.7: Estimation of $z(t)$ from $|V_m(t)|$

Figure 3.8 shows EM measurements when entering a 6-digit PIN. EM emanation variations reflect finger interactions with the screen. It also provides a zoom-in look of EM amplitude associated with the 5-th key tapping. We find that the signal experiences a sharp increase when the finger moves towards the screen. It then decays quickly the moment a physical contact takes place. This is because the finger draws some electric charges from the screen. With reduced electric charges, the EM radiation from the screen drops accordingly. Later on, as the finger leaves the screen for the next key, the EM amplitude keeps decreasing until the finger is de-coupled from the screen. This observation coincides with the analytic result derived previously.

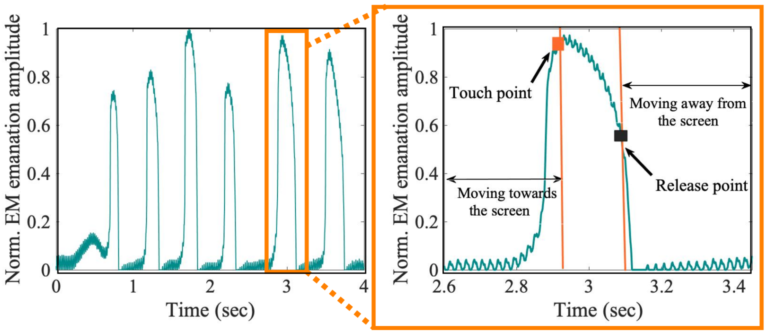


Figure 3.8: EM emanation measurements for entering a 6-digit PIN.

Figure 3.9 shows EM measurements by entering three different key pairs “42”, “46”, and “43”. It is observed that their EM readings are distinct to each other. For example, “42” is associated with the shortest time duration between two consecutive EM amplitude peaks, as a finger moves in the shortest path to enter this key pair among the three. We further evaluate the similarity of EM emanations among ten key pairs originated from “4” in Figure 3.10. Normalized DTW distance is employed. A small value represents a high similarity between two pairs, while a larger one means they are barely correlated. We find that except for the diagonal, i.e., a key pair and itself, DTW distances between EM readings from any two different pairs are relatively large.

Based on the above observation, we propose to recognize individual key pairs from their EM measurements first and then the whole PIN.

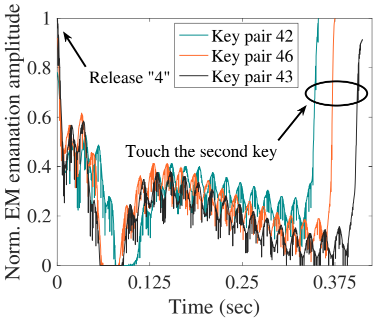


Figure 3.9: EM measurements of for entering different key pairs.

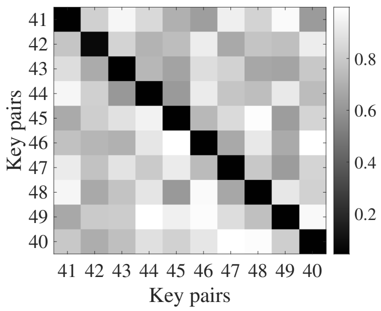
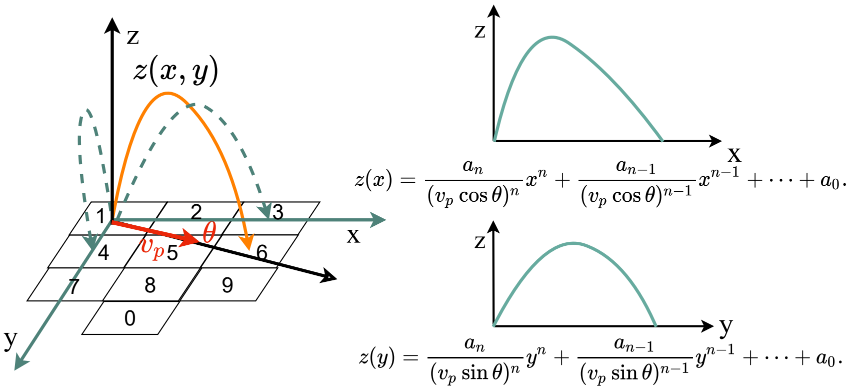


Figure 3.10: Similarity matrix for different key pairs using normalized DTW distance.



$$z(x) = \frac{a_n}{(v_p \cos\theta)^n} x^n + \frac{a_{n-1}}{(v_p \cos\theta)^{n-1}} x^{n-1} + \cdots + a_0.$$

$$z(y) = \frac{a_n}{(v_p \sin\theta)^n} y^n + \frac{a_{n-1}}{(v_p \sin\theta)^{n-1}} y^{n-1} + \cdots + a_0.$$

Figure 3.11: 3D finger movement trace and decomposition.

3.6 Design Rationale

Our design first separates the received continuous EM emanations into multiple segments, each representing signals from one key pair. Recall that we are able to map an instant EM reading to the associated finger-screen distance. Then we apply some transformations to convert time-dependent finger-screen distance to finger movement traces, which finally recover key pairs and thus the PIN.

Decomposition of 3D finger movement trace. As shown in Figure 3.11, the finger movement for entering one key pair can be characterized by a 3D trace. By treating the first keystroke as the origin point, we set up a 3D coordinate system, where the x-y plane is where the screen resides and z-axis is vertical to the screen. For any 3D finger movement trace, denoted as $z(x, y)$, let its projection on the x-z plane and y-z plane be $z(x)$ and $z(y)$, respectively. If we know the intersection between $z(x)$ ($z(y)$) and x-direction (y-direction) of the keyboard, the key pair is recovered. For this purpose, we further divide the x-direction, denoted as L_x, of the keyboard into 3 units; each represents one key. Similarly, the y-direction, denoted as L_y, is divided into 4 units. Under this setting, key pair "16", for example, can be represented as $L_x = 2$ units, $L_y = 1$ unit. Our task now becomes how to determine L_x and L_y of a specific key pair from its EM emanation readings.

Relation between L_x (L_y) and EM readings. Denote by θ the angle between 3D trace $z(x, y)$ and $z(x)$, its projection on the $x - z$ plane. Let $x(t)$ ($y(t)$) be the finger's instant position at time t cast on the x-axis (y-axis). Then we have

$$x(t) = v_p t \cos\theta,\ y(t) = v_p t \sin\theta, \tag{3.11}$$

where v_p is the finger movement speed[3]. Besides, the time-series finger-screen distance $z(t)$ of one key pair can be approximated with a high dimensional polynomial

$$z(t) = a_n t^n + a_{n-1} t^{n-1} + \cdots + a_i t^i + \cdots + a_0. \tag{3.12}$$

The $n + 1$ coefficients $a_0, \cdots, a_n$ can be determined by solving a linear equation system with $n + 1$ samples: $(t_1, z(t_1)), \cdots, (t_{n+1}, z(t_{n+1}))$, where $z(t)$ can be calculated from $V_m(t)$ following (3.10). Combining (3.11) and (3.12), $z(x)$ is expressed as

$$z(x) = \frac{a_n}{(v_p \cos\theta)^n} x^n + \frac{a_{n-1}}{(v_p \cos\theta)^{n-1}} x^{n-1} + \cdots + a_0. \tag{3.13}$$

[3] Soft keyboards are generally small in size. For most users, the entry of a PIN can be performed smoothly within 4 seconds. Thus, it is practical to assume a constant finger movement speed for each user. Speeds from different users are not necessarily the same though.

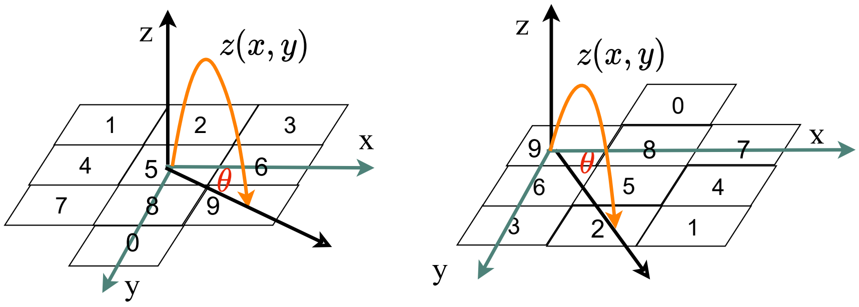


Figure 3.12: Coordinate systems for finger movement traces from arbitrary key pairs.

Similarly, $z(y)$ can be expressed as

$$z(y) = \frac{a_n}{(v_p \sin\theta)^n} y^n + \frac{a_{n-1}}{(v_p \sin\theta)^{n-1}} y^{n-1} + \cdots + a_0. \tag{3.14}$$

With $z(x)$ ($z(y)$), by examining its intersection with the x-axis (y-axis), we can easily obtain L_x (L_y). To be specific, solve x by setting $z(x) = 0$. L_x is the unit that x falls into. L_y is obtained similarly. The above calculation relies on the knowledge of v_p and θ. We will discuss in Section 3.7.2 how to derive these two critical parameters.

To sum up, for each key pair, the attacker collects at least $n+1$ samples of EM readings. Their corresponding finger-screen distances are calculated following (3.10). Then a polynomial of n-degree that characterizes time-series finger-screen distance $z(t)$ is constructed. With the knowledge of victim's finger movement speed v_p and direction θ, $z(t)$ is converted to $z(x)$ and $z(y)$. Their intersections with x-axis and y-axis are L_x and L_y, respectively.

Discussions. Note that a given pair of L_x and L_y may not uniquely identify a specific key pair, but a set of key pair candidates. For example when $L_x = 2$ units and $L_y = 1$ units, satisfying key pairs include "61", "16", "34", "43", "67", "76", "49", and "94". To alleviate the inference ambiguity, we propose to model transitions between key pairs into a HMM to eliminate impossible combinations of key pairs. Details will be elaborated in Section 3.7.3.

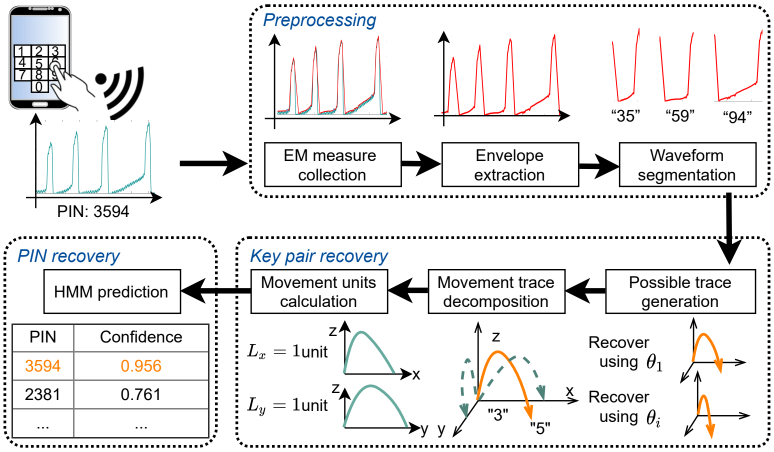


Figure 3.13: The system design of Periscope.

In the above analysis, we use the key pair "16" to illustrate how to model a finger movement trace shown in Figure 3.11. A coordinate system, with "1" as the origin point, is set up. In fact, our method is applicable to arbitrary key pairs. Figure 3.12 demonstrates the cases of two other key pairs "59" and "92". Their origin points become "5" and "9", separately. In either case, we ensure the trajectory exists in the first quadrant of the coordinate system and thus $\theta \in [0, \pi/2]$ to facilitate our analysis.

3.7 Design Details of Periscope

The system overview of Periscope is given in Figure 3.13. It consists of three main components: *preprocessing*, *key pair recovery*, and *PIN recovery*.

3.7.1 Preprocessing

The goal is to extract clean signal segments for individual key pairs from continuous raw EM emanation readings.

Envelope extraction. As shown in Figure 3.14, raw EM emanation readings are mixed with oscillating signals, which add small-scale variations to the envelope. Essen-

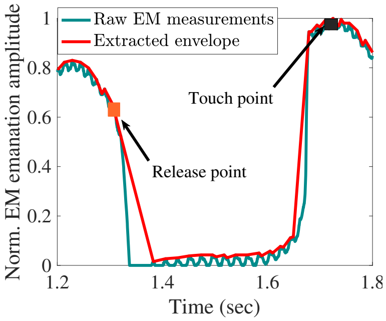


Figure 3.14: Envelope extraction and waveform segmentation.

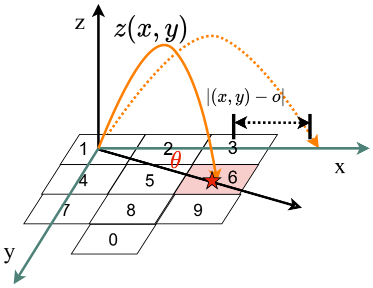


Figure 3.15: Estimation of θ.

tially, the envelop signal is caused by finger coupling effect and thus contains useful information regarding finger movements. The oscillating signals, on the other hand, are produced by touchscreen's alternating driving voltage and useless for the attack. To extract the envelope, the extrema sampling based algorithm is employed [161, 1]. Specially, a sliding time window Δt is applied over the raw reading. The local maximal value within this window, $\max V_m(t')$ ($t' \in [t, t + \Delta t]$), is deemed as the filtered output for Δt.

Waveform segmentation. The purpose of this step is to segment the signal for each key pair out of a continuous waveform. We first identify critical time instances associated with finger release/touch events. For finger touch, it appears at EM reading peaks. We thus apply the classic peak detection algorithm [16] over the envelope signal to identify such events. Once the finger leaves the screen, the discharging coupling capacitance causes a sudden drop in EM readings as shown in Figure 3.14. Hence, the finger release event is identified by locating the maximum derivative along the EM signal envelope between two consecutive peaks. Upon identifying the above critical events, the EM signal of one key pair is the waveform segment between the finger release (of the first key) and the next finger touch (of the second key).

3.7.2 Key Pair Recovery

Section 3.6 presents how to recover a key pair, recognized via L_x, L_y, from EM readings. As discussed, the attacker should be aware of the victim's finger movement speed v_p and direction θ. In the following, we focus on the estimation of these two parameters.

Estimation of θ. Let Θ be the set of possible directions of finger movement for entering a key pair. To estimate θ, our idea is compare among all the possible candidates in Θ and figure out the one that produces the highest estimation confidence level.

As discussed in Section 3.6, we take the first key of a key pair as the origin and set up a 3D coordinate system. Following the steps, the coordinate of the second key (x, y) is derived by solving $z(x) = 0$ and $z(y) = 0$. L_x and L_y are obtained accordingly. Let o be the geometry center of the key identified by L_x and L_y. A user typically taps the center of a key to enter an input. If θ is the correct direction, the derived (x, y) should be close to a key's center. Otherwise, (x, y) is tend to deviate from the center, as illustrated in Figure 3.15. We then define the confidence level under θ as

$$l = 1 - \frac{|(x, y) - o|}{\sum_{\theta \in \Theta} |(x, y) - o|}. \tag{3.15}$$

l is a value between $[0, 1]$. It tends to be 1 if (x, y) is close to a key center. Finally, finger movement direction is deemed as the one that produces the maximum confidence level among all the candidates, $\theta = \arg\max_{\theta \in \Theta} l$.

Estimation of v_p. As examined in prior works [38, 5], finger movement speed for typing is deemed consistent for each individual. We propose to estimate it by eliciting victims to enter some digits in their devices and estimating the speed from the collected samples. Specifically, the attacker can set up free WiFi. Once a victim is connected, the access point requires user approval by displaying a dialog box and asking the victim to enter designated numbers as a confirmation message [4]. An alternative approach is to set up a Text Captchas that asks the victim to input the chosen numbers [74]. Following the same key pair segmentation approach, we first separate victim entered number sequence into a series of key pairs. Then the time duration for entering one key pair is known. Since the exact key pair is known, so is the inter-key distance. The finger movement speed is estimated by dividing the distance by the time duration. We set v_p as the median value of measured speeds of all key pairs in one number sequence. To improve the estimation accuracy, the attacker can have the victim enter more than one sequence. According to our experiment result, three such digit sequences are sufficient to deliver satisfactory estimation. As a note, the limited number of samples should not be deemed as a user-specific training dataset.

3.7.3 PIN Recovery

So far, the attacker is able to infer L_x and L_y of a given EM waveform segment. As discussed, a pair of L_x and L_y can be mapped to multiple key pairs. We propose to leverage the interdependence of consecutive key pairs to resolve the inference ambiguity. For example, given $L_x = 2$ units and $L_y = 1$ units for the first waveform segment, satisfying key pairs include "61", "16", "34", "43", "67", "76", "49", and "94". Given $L_x = 2$ units and $L_y = 2$ units for the second waveform segment, satisfying key pairs include "19", "91", "37", and "73". Considering interdependence, the existing candidates for the first key pair "16", "34", "76", and "94" can be eliminated immediately, as none of them ends with "1", "9", "3", or "7", the first digit of the second key pair. Hence, viable candidates for the first key pair are narrowed down to "61", "43", "67", and "49", by 50%. As more key pairs are considered, this side information can be propagated back-and-forth to further reduce the ambiguity. We propose to model such interdependence between consecutive key pairs for PIN recovery using HMM.

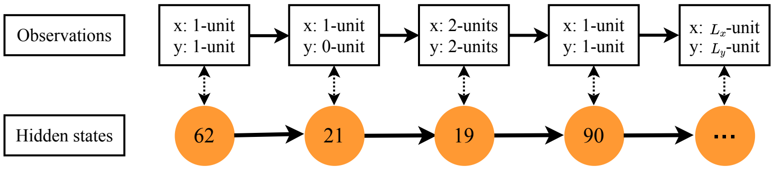


Figure 3.16: The state and transitions of HMM.

We model the keystroke process as HMM characterized by $\lambda = (N, M, A, B, \pi)$. In the HMM, N is the number of hidden states. We treat key pairs as hidden states. As there are 100 possible key pairs, i.e., from "00" to "99", we have $N = 100$. The parameter M represents the number of possible observations for hidden states, i.e., L_x and L_y. As there are three and four possible values of L_x and L_y, respectively, we have $M = 3 \times 4 - 1 = 11$. A, with the size of $N \times N$, stands for the transition probability matrix, with each element denoting the transition probability from one hidden state to another. The observation probability matrix B, of the size $N \times M$, gives the possibility that a given observation can be observed in a hidden state. The initial state distribution vector π represents the belief about which state the HMM is in when our scheme is called for the first time.

To build the HMM, we need to determine parameters A, B, and π. The transition probability matrix A can be predefined by the natural continuity of the typing process. For example, if we assume equal probability of typing any keys, the hidden state "61" has a chance of 0.1 to transfer to each hidden state "1x", while the chance to other states is 0. B is obtained by evaluating the probability of a given key pair that generates certain observations. It actually reflects the accuracy of our proposed key recovery scheme. Due to random errors occurred in EM measurements, our scheme may generate false observations other than the ground truth at a certain probability. We propose to run our scheme offline ahead of the attack to derive B. In our design, we employ a uniform distribution for the initial state distribution π.

Given the observation sequence of key pairs $O = O_1 O_2 \cdots O_S$, the PIN recovery problem is to find optimal hidden sequence $Q = Q_1 Q_2 \cdots Q_S]$ to maximize $P(Q|O, \lambda)$. This problem can be solved by the Viterbi algorithm [110], a commonly adopted approach for HMM. In addition to finding the most likely PIN, we also calculate the probability of all possible PINs generated by the HMM. The attacker can thus sort them according to their probabilities and form a list of candidates to infer the target PIN with multiple trials.

3.8 Experimental Evaluations

The experiments are conducted using our prototype described in Section 3.5. It is built on a commercialized Arduino board that follows the FCC regulations and passively collects EM emanations. Hence, no risk is posed to human health. The collected data are anonymized and properly stored locally from potential leakage. The entire research has been approved by IRB.

The goal is to evaluate the performance of our proposed attack Periscope under different settings. A wide spectrum of impact factors are examined, such as system parameters, attack distances, environmental contexts, etc. A comprehensive comparison is also made with existing schemes. A total of 20 volunteers, 12 males and 8 females between 22 to 28 years old, are recruited for the experiments. Before each experiment, detailed instructions regarding experimental procedures are provided. We design an App that mimics the UI that allows users to unlock the screen via digit PINs. During the experiment, each volunteer is asked to enter 60 randomly generated PINs into smartphones.

3.8.1 Key Pair Recovery Accuracy

As the basis of our attack, we first examine the accuracy of key pair recovery. Figure 3.17 shows the success rate over all the 100 possible key pairs from “00” to “99”. Each

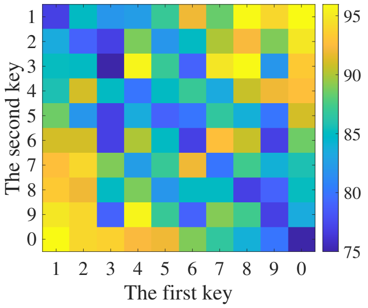


Figure 3.17: Key pair recovery accuracy.

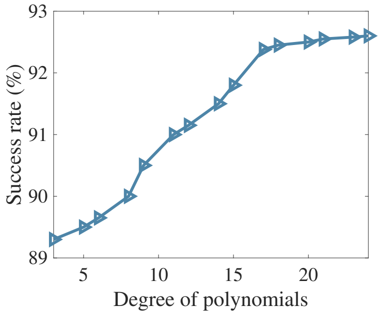
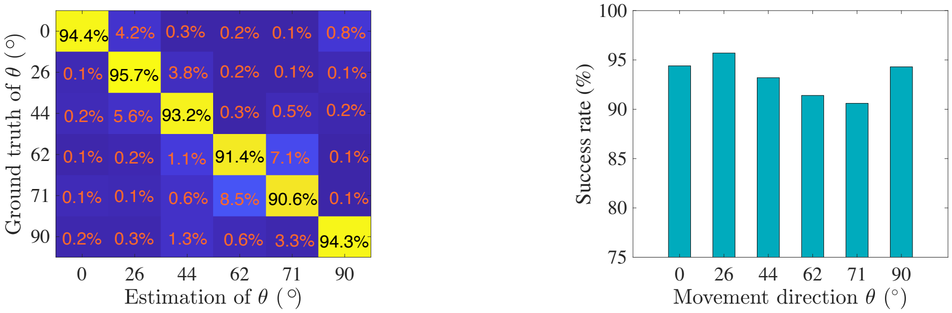


Figure 3.18: Impact of the order of polynomials.

row represents the first key, whereas each column represents the second key. It is observed that key pairs with longer inter-key distances tend to have better recovery accuracy. For example, the recovery rate of “01” is 94%; it becomes 89.5% for “08”. Besides, we find that the success rate is not perfectly symmetric with respect to key pairs. In other words, the success rates of “ab” and “ba” are not exactly the same. This is because users may exhibit different typing behaviors when entering the same pair of keys but with reverse orders.

Impact of degree of polynomials. To establish the relation between EM readings and L_x (L_y), we employ an n-degree polynomial to characterize the time-dependent finger-screen distance $z(t)$. Figure 3.18 shows key pair recovery accuracy with respect to the degree n. The success rate experiences a slight increase by adopting a higher degree polynomial. For example, the success rate is 89.6% when $n = 6$ and then raised to 91.2% when $n = 12$. It indicates that a polynomial with a higher degree can nicely tract the finger movement trace. Once n surpasses 17, such benefit becomes negligible. At the same time, a polynomial of higher degree incurs larger computation overhead in solving $z(x) = 0$ and $z(y) = 0$. To strike a balance between accuracy and efficiency, we set $n = 17$ by default.

Estimation of θ. The estimation of finger movement direction θ is critical to key pair recovery. Figure 3.19(a) shows the confusion matrix of θ estimation. A Google Pixel phone is adopted in the experiment. The rows represent all possible finger movement directions



(a)

(b)

Figure 3.19: Recovery accuracy of θ. (a) Confusion matrix. (b) Recovery success rate of key pairs with different θ's.

as the ground truth, whereas columns represent estimated results. As discussed in Section 3.6, $\theta \in [0, \pi/2]$. The figure easily tells whether our scheme causes any confusion between classes. The average recognition accuracy is 93.3%. We observe that most of the errors come from adjacent directions. For example, when the ground truth is $71°$, the chance it recognized as $62°$ is 8.5%, which is the highest among all the cases. We also notice in Figure 3.19(b) that $62°$ and $71°$ are associated with relatively lower success rate, at 91.4% and 90.6% respectively, compared with other directions. This is because they are separated by a small margin of $9°$.

No. of Seq.	V 1	V 2	V 3
1	5.12	4.43	3.24
2	2.72	3.0	2.16
3	1.96	2.46	1.74
4	1.49	2.27	1.25
5	1.39	1.97	0.82

Table 3.2: Estimation error of v_p (cm/s).

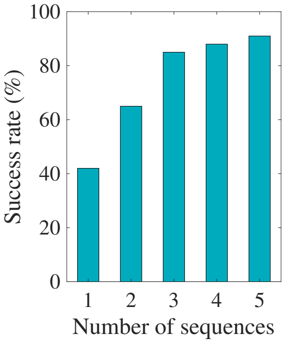


Figure 3.20: Key pair recovery performance.

Estimation of v_p. It is difficult to measure user's finger movement speed directly. To approximate the ground truth, we divide the distance between two touch points for entering a key pair by its time duration. The distance can be readily computed from coordinates of the two touches, accessible from smartphone API. Table 3.2 exhibits the estimation error over v_p from three randomly selected volunteers. We find that the error decreases as the user is asked to enter more digit sequences in advance of the attack. Take volunteer 1 as an example, the estimation error is 5.12 cm/s with one digit sequence and drops to 1.39 cm/s under five digit sequences. It meets our expectation; the estimation becomes more robust to variations introduced by an individual sample. We further evaluate in Figure 3.20 the impact of number of digit sequences to key pair recovery. The recovery success rate quickly increases to 85% under three digit sequences. Beyond that, the growth becomes incremental. To trade between practicality and accuracy, we suggest having victims enter three digit sequences in advance of the attack.

3.8.2 PIN Recovery Accuracy

We now examine the recovery performance over an entire PIN that consists of multiple key pairs.

Table 3.3: PIN recovery success rate with top-10 candidates.

PIN length	3-digit	4-digit	6-digit	8-digit
Success rate	71.7%	61.7%	43.3%	35%

Impact of PIN length. In this experiment, volunteers are asked to input PINs with lengths varying from 3 to 8 digits. Table 3.3 shows the recovery success rate with top-10 candidates. As a note, our scheme can produce a list of candidate PINs. If the list of K candidate PINs contain the target PIN entered by the victim, then the correct PIN is deemed among the top-K candidates. This metric reflects the recovery accuracy and has been widely adopted in prior works. We find that the highest success rate 71.7% is achieved for 3-digit PINs. It decreases as PIN length grows, since successive correct inferences of all key pairs are needed to recover the entire PIN. We find the success rate is 43.3% with 6-digit PINs, the mostly commonly PIN length adopted by mobile devices nowadays. Our attack does pose a real threat to these devices.

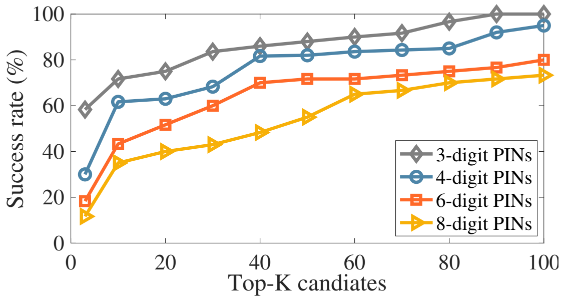
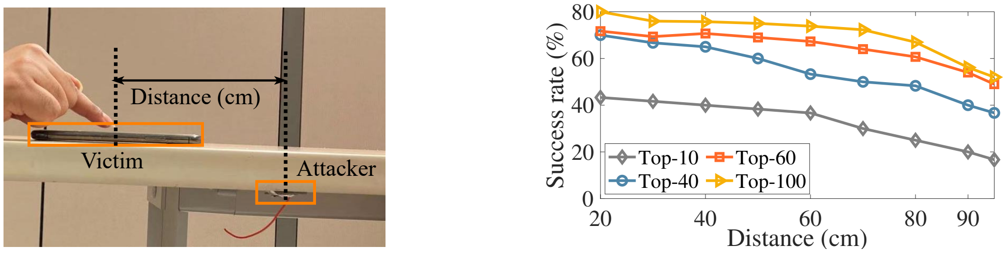


Figure 3.21: PIN recovery success rate with top-K candidates.

Impact of the number of candidates. We further study how many candidates are needed to succeed in inferring the target PIN. In the experiments, we sort candidates generated by the HMM model according to their probability of being the target PIN in a descending order and select the top-K candidates to evaluate the recovery accuracy. In Figure 3.21, we give the PIN inference success rate under top-K candidates, where K ranges from 1 to 100. The result is encouraging. It is shown that, given top-1 candidate, the recovery accuracy is 18.3% for 6-digit PINs. That is, our attack can correctly hit a victim's 6-digit PIN at a probability of 18.3% in one shot. The rate can be significantly improved if given top-10 candidates or top-20 candidates, which corresponds to 43.3% and 51.7%, respectively. As shown in the figure, if given top-40 candidates, the success rate reaches almost 70% for 6-digit PINs. As a comparison, WindTalker [74], a well-cited radio signal based inference attack, delivers a similar performance with more than 60 candidates, not to mention that WindTalker needs a training stage while Periscope is completely training free.

3.8.3 Performance Under Different Settings

Impact of victim-attacker distances. In practice, a victim device may be placed at different distances away from the attacker. It is thus necessary to examine the impact of this factor to the attack accuracy. In the experiments, we set the distance from 20 cm to 100 cm. All are carried out with 6-digit PINs. As shown in Figure 3.22, the recovery success rate exhibits negative correlation with the distance. This is because a longer distance leads to weaker EM emanation receptions at the attacker. As a result, it becomes challenging to precisely recover finger movement traces from the EM readings. Still, the attacker can successfully disclose a target PIN at a probability of 20% even 90 cm away from the victim with top-10 candidates. It is worth mentioning that the victim and the attacker reside at two sides of a wood table, a non-LoS scenario shown in Figure 3.22(a). We anticipate an even higher success rate under a LoS scenario. Besides, we only deploy one prototype in the experiment. In practice, many of them can be used. Such settings can potentially further enhance the performance.

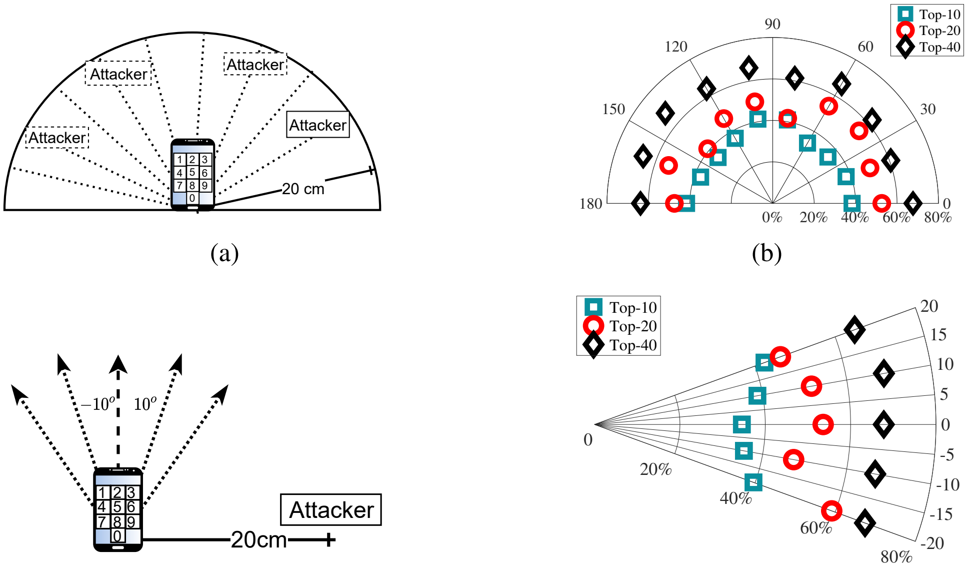


(a)

(b)

Figure 3.22: Impact of victim-attacker distance. (a) Positions of the victim and the attacker. (b) PIN recovery success rate.

Impact of victim-attacker relative direction. We also examine if the relative direction between the victim and the attacker impacts the recovery accuracy. In the first set of experiments, we fix their distance at 20 cm and place the attacker at different directions to the victim as shown in Figure 3.23(a). Figure 3.23(b) shows the PIN recovery success rate at these positions. We find that the accuracy is almost the same for all the cases. In the second set of experiments, the attacker's and victim's positions are fixed while varying the smartphone orientation. Again, no apparent difference in recovery accuracy is observed. Hence, the performance of Periscope is independent of victim-attacker relative direction. This is not the case for radio signal based attacks. Essentially, the attacker needs a LoS view over the victim; otherwise, the signal variance caused by multi-path propagation renders the signal hard to tract. In addition, the video-based attack also imposes stringent requirement over the recording angle. For example, Eyetell [22] experiences about 70% accuracy degradation when the two parties have a 10° displacement angle.



(a)

(b)

(c)

(d)

Figure 3.23: Impact of victim-attacker relative direction. (a) Test scenarios for the first set of experiments. (b) PIN recovery success rate. (c) Test scenarios for the second set of experiments. (d) PIN recovery success rate.

Impact of target diversity. People may have distinct typing behaviors during PIN inputs. Hence, it is critical to find out if Periscope is susceptible to this factor. Table 3.4

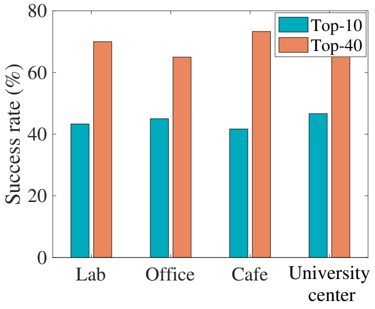


Figure 3.24: Impact of environmental context.

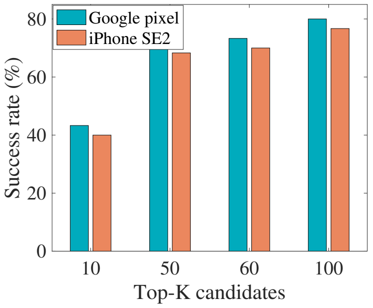


Figure 3.25: Impact of device diversity.

shows the PIN recovery accuracy across seven volunteers. While each individual exhibits a slightly different success rate, the overall performance is relatively consistent, with the average success rate all above 40% with top-10 candidates. It means our analytic model is capable of handling target diversity. As discussed, most acoustic and radio signal based attacks need to train user-specific models to accommodate diverse typing behaviors and is thus less practical for broad deployment.

Table 3.4: PIN recovery success rate over different victims.

Index	1	2	3	4	5	6	7
Top-10	52.7%	46.7%	42.5%	50.1%	40.2%	40.6%	45.6%
Top-40	68.9%	72.1%	69.5%	74.1%	63.4%	67.8%	62.2%

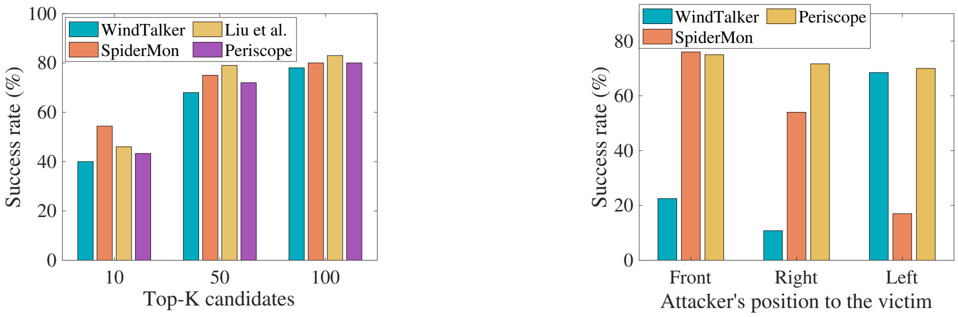
Impact of environmental context. We further evaluate the attack performance in four different environments, including lab, office, coffee shop, and university center. Figure 3.24 exhibits a promising performance in all the environments, no matter whether it is quiet or noisy, static or dynamic. In contrast, most acoustic based attacks can only succeed in quiet places, whereas radio signal based attacks do not work with dynamic backgrounds. In fact, public places tend to be noisy and dynamic.

Impact of different devices. To demonstrate the usability of Periscope, we also experimented on two smartphones, an iPhone SE2 with a 4.7-inch touchscreen and a Google Pixel phone with a 6-inch touchscreen. Figure 3.25 compares their PIN recovery accuracy. The performance is similar for both devices. It means our attack works for a diverse set of devices as long as they are equipped with a multi-capacitance touchscreen. We also notice that the success rate on Google pixel is slightly higher than that on iPhone SE2. This is attributed to the larger screen size of the former. Finger movement traces are more distinct on a larger screen.

3.8.4 Comparison with Other Schemes

In this part, we present the performance comparison with prior keystroke inference attacks on digit PINs. For the sake of fairness, we directly utilize the experimental results from these works. Three schemes WindTalker [74], SpiderMon [76], and the attack proposed by Liu et al. [82] are considered. Specifically, WindTalker measures the fluctuations of WiFi channels caused by victim's typing motions. SpiderMon utilizes variations of multi-path LTE signals to infer victim's inputs. Liu et al. [82] analyzed the motion status of smartwatches to launch the attack.

Figure 3.26(a) compares their success rates of recovering 6-digit PINs. It is observed that the performance of Periscope is similar to WindTalker and Liu et al. with top-10 candidates, while SpiderMon is the best. All their success rates reach 80% with top-100 candidates. Note that all other three schemes need a training phase. A large amount of training samples should be collected from the target victim in advance of the attack. To do this, Liu et al. even require the pre-installation of malware on the victim's smartwatch for sample collection. These restrictions make them hardly practical in real-world scenarios. In contrast, Periscope is a non-intrusive and training-free attack.



(a)

(b)

Figure 3.26: Performance comparison with other schemes. (a) Recovery rate of 6-digit PINs. (b) Impact of attacker's position.

Figure 3.26(b) compares their performance under the impact of victim-attacker relative direction. We find that both SpiderMon and WindTalker experience significant performance variance by placing the attacker in different directions with respect to the victim. In contrast, the success rate of Periscope is relatively stable regardless of the displacement. This is because the former two extract wireless channel disturbances to monitor finger movements. They heavily rely on LoS propagation channels as they provide the most tractable signals. These channels can be easily blocked by the victim's body if positioning the attacker to the left/right of the victim. On the other hand, as discussed in Section 3.4.4, the analytic model of Periscope is irrelevant of surrounding environmental conditions. Its performance is thus independent of victim-attacker relative direction.

3.9 Discussions

3.9.1 Limitations and Future Work

Extending attack distance. Results in Section 3.8.3 show that Periscope's PIN recovery rate with top-10 candidates drops to 20% when the attack distance is beyond 90 cm. The threat can be more severe if the attack can be successfully performed remotely. The primary reason of the confined distance here is the weak signal strength of EM leakage.

Besides, the EM field decays quickly over distance. Note that our prototype is built with simple electronic pieces, including an Arduino nano board and a conductive wire. Neither advanced transceiver module nor sophisticated signal processing unit is utilized. As our future work, we plan to build a more powerful prototype with dedicated components that can pick up useful signals from noisy and weak EM measurements so as to extend attack distance.

Recognizing letters. Our discussion has been focused on soft numeric keyboards. We plan to extend Periscope to recognize letter inputs. The challenge is to distinguish subtle EM emanations from more diverse combinations of key pairs, as the number of keys will almost be tripled. We propose to employ multiple eavesdroppers and explore their collaboration to launch attacks. *Sensor fusion* techniques [88, 34] will be applied. It combines EM readings from disparate sources such that the resulting information has less ambiguity than would be possible when these sources were used individually. It is expected that the aggregated EM measurement will provide more fine-granular recognition of finger movements.

3.9.2 Defense Solutions

Periscope explores human-coupled EM emanations to recover victim's inputs on soft keyboards. An intuitive defense solution is thus to adopt shuffled keyboards. This idea has been proposed before [120, 112]; the system adopts a new randomly generated keyboard layout each time a user intends to enter a credential. Although attackers can still derive finger movement traces, they can be hardly mapped to specific keystrokes without the knowledge of keyboard layout. While leaving the key inference almost impossible, as pointed by [163], this idea sacrifices the authentication usability. Extra effort is incurred to the user in searching for keys on a shuffled keyboard. More input errors might also be introduced thereby.

In a more practical way, users may intentionally disrupt their typing behaviors, for instance, adding random pauses between keystrokes and/or adopting variant typing speeds when entering different key pairs. For both cases, attackers will tend to make mistakes in transforming time-dependent finger-screen distances to 3D finger movement traces. For the former, the trace length will appear much longer than the ground truth. For the latter, as a user adopts a dynamic speed, it is impossible for an attacker to generate a meaningful finger movement trace with v_p, a constant finger movement speed that is estimated in advance the attack. As a result, the derived L_x and L_y become error-prone in both cases. The attacker is less likely to accurately recover individual key pairs, let alone the whole PIN.

It is also possible to apply electromagnetic interference (EMI) shielding on touchscreens. This technique has been widely employed on many electronic devices; it refers to the shielding of radio waves so that radiations cannot penetrate the shield. In our case, it can serve as a barrier that prevents EM emanation leakage, or at least reduces the radiation strength. Nonetheless, this approach may be expensive and require hardware modifications, including the introduction of new EMI materials and touchscreen circuit redesign. Another alternative is to intentionally obfuscate the EM emanations emitted by the touchscreen, so that the trajectories of EM readings are not recognizable. A straightforward approach is to add well-calibrated noise to the touchscreen driving signal $V_{TX}(t)$. Then attacker's EM measurements $V_m(t)$ become polluted. Since the attacker is unaware of the injected noise pattern, it is hard to tell if observed EM variations are incurred by finger movements or intentionally injected noise.

3.10 Conclusion

In this paper, we present Periscope, a new eavesdropping attack that leverages human-coupled EM emanations from touchscreens to infer victims' typing inputs at a remote distance. We implemented the proposed attack with a prototype that costs less than $10. Its effectiveness is evaluated from various aspects. Periscope exhibits promising recovery accuracy over a distance up to 90 cm. It can well adapt to diverse device models and setting contexts. Compared with prior works, our approach is built on an analytic model that characterizes the relationship between EM measurements and finger movement traces. Therefore, it is completely training-free, without the need to collect user-specific datasets in advance of the attack. In summary, we believe that Periscope outperforms state-of-the-art keystroke inference attacks, especially in terms of practicality. Meanwhile, it can be further extended with longer attack distance and inference over letter inputs, which are deemed as our future work.

CHAPTER 4

ACOUSSIST: AN ACOUSTIC ASSISTING TOOL FOR PEOPLE WITH VISUAL IMPAIRMENTS TO CROSS UNCONTROLLED STREETS[1]

4.1 Introduction

Motivation: According to the records provided by the World Health Organization in 2018, the global-wide visually impaired people are estimated at 2.2 billion [104]. How to navigate them to cross streets is a long-lasting topic. The state-of-art solution is to install the Accessible Pedestrian Signals (APS) at intersections or crossing sections to assist the visually impaired in determining when it is safe to cross. However, there are even more uncontrolled crosswalks where no traffic control (i.e., traffic signals or APS) is present. These common crossing types occur at non-intersection or midblock locations where they may be marked or not. They typically exist in residential communities, local streets, suburban areas, etc. where sophisticated traffic infrastructures are too expensive to fully deploy around. In these road sections, the visually impaired have to mostly depend on themselves to judge the surrounding traffic condition and decide whether it is safe to proceed to the crosswalks. In practice, the pedestrian leverage hearing to discriminate between traffic sounds that are too far away to pose a hazard to crossing and those that are within close proximity. Nonetheless, hearing based judgment is not always reliable. Hearing capability varies from person to person; young people generally have more sensitive hearing than se-

[1] Used with permission of the publisher, 2021.

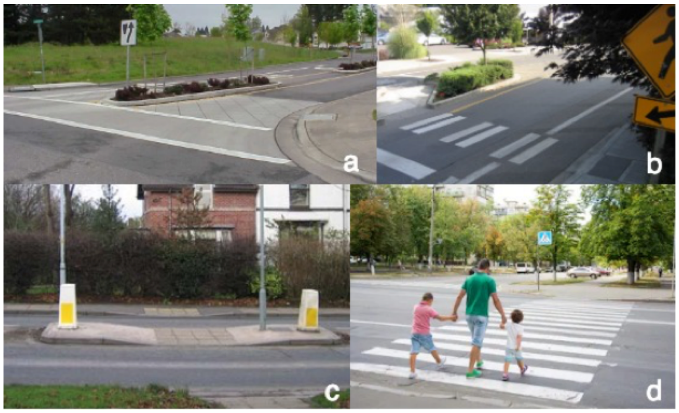


Figure 4.1: Some examples of uncontrolled crosswalks.

niors. Besides, environmental conditions may affect traffic sounds. For example, rain and wind may enhance or distort sounds; snow can muffle sounds; background construction sounds or talking from people nearby may even overwhelm the traffic sounds.

This paper aims to develop a portable tool that assists pedestrians with vision impairments to cross uncontrolled streets. The tool alerts pedestrians with the presence of oncoming vehicles that may cause hazard. To achieve this goal, it is essential to figure out movement status of each vehicle nearby, characterized by, for example, its velocity relative to the pedestrian, direction of arrival (DoA), and its distance to the pedestrian. To measure these parameters, one possible solution is to use radar [85]. Due to its stringent requirements over the received signal quality [147, 151, 124], the existing radar applications mostly operate over licensed spectrum bands. For instance, Federal Communications Commission (FCC) designates the X band frequencies between 10.500-10.550 GHz and the K band frequencies between 24.050-24.250 GHz for police radar gun. In our case, applicable radio frequencies the already over-crowded ISM bands. Besides, given that our problem requires the ranging distance up to 200 ft, the perceived signal-to-noise ratio (SNR) would be too low to achieve meaningful detection. Moreover, we need build our own transmitter or/and receiver by using radar techniques. In contrast, the proposed design can be implemented on commercial off-the-shelf (COTS) devices.

LiDAR [40, 10, 118], which uses predominantly infrared light from lasers rather than radio waves, is another potential alternative. It has been widely used in autonomous vehicles to monitor surrounding environments to avoid traffic incidents. However, LiDAR is a technology that requires a powerful computation and storage capacity to handle the huge collected dataset. Its cost is another concern.

Given the above analysis, our idea leverages acoustic ranging. It is competent for our implementation in the following aspects. First, acoustic signals can propagate around obstructions through diffraction on their edges or reflection from their surfaces. This capability supports measuring vehicles behind obstructions. Second, its performance does not depend on lighting conditions and is effective even in darkness. Third, we can easily customize transmission signals on commercial COTS speakers and process received signals in smartphones. We propose to utilize ultrasound signals ranging from 17 KHz to 19 KHz. To our knowledge, there is no application with wide deployment operating on this band. Thus, it is less likely to experience cross- application interference.

Motivated by the above observations, we develop Acoussist, an **acou**stic based a**ssist**ing tool for the visually impaired to cross uncontrolled streets. Acoussists consists of speakers that are mounted on the front of vehicles to emit ultrasonic chirps and an app running on the pedestrian's smartphone for signal analysis. Whenever a pedestrian senses a clear street and tends to proceed to the crosswalk, she turns on the app to double-confirm her judgment. The app analyzes the received chirps to detect if any oncoming vehicle would cause potential collisions. If yes, an alert is generated. Then, the pedestrian should take precaution and wait at the curb until the street is clear. For vehicles operating in all-electric

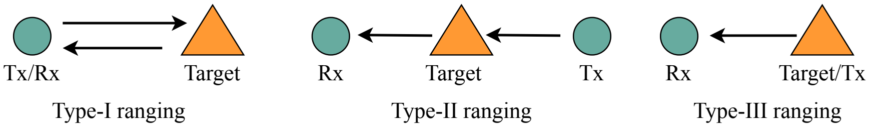


Figure 4.2: Illustration of three types of ranging.

or hybrid mode, the chirps can be played by their *warning sounds system*[2]. For traditional combustion engine vehicles, chirps are proposed to emit from COTS portable speakers. While pedestrians are "stakeholders" in our scenario, drivers do not necessarily lack the motivation to install the speakers. Department of Public Safety (DPS) [32] regulates that a driver who fails to yield to the pedestrians with vision impairments (regardless of any reasons) is fully/partially liable for any injury caused to the pedestrian. As demonstrated in this work, a participatory vehicle can effectively alert a pedestrian regarding its presence.

Challenges: Although the idea of acoustic ranging is not new, turning it into a tool for collision detection is faced with several unique challenges. 1) *Vehicle velocity measurement with mutual interference:* One of the key ingredients to decide if a vehicle causes potential hazard is to figure out its velocity relative to the pedestrian. A straightforward solution is to analyze the Doppler frequency shift of the received chirps at the receiver. This task is easy if only one vehicle is nearby. In our case, oftentimes several vehicles are present. Their emitted chirps overlap, rendering distinguishing among them an extremely challenging task, let alone analyzing the frequency shift for velocity measurement. 2) *Dynamic multi-source localization:* It is also indispensable to figure out the DoA of each vehicle. Acoustic multi-source localization has been studied in the domain of speaker tracking in video conferences [79, 36, 84], indoor localization [35], and noise identification [30]. Existing approaches either assume the number of sources are fixed and *a prior* known, or signals from different sources are of distinct frequency patterns. Thus, none of

[2] Many countries have approved legislation to enforce "quiet" vehicles install the *warning sounds system*, an array of external speakers that emit artificial engine sounds for pedestrians to be aware of their presence. For example, EU requires all new models of electric and hybrid vehicles developed and sold in EU to equip the system by July 2019 [49].

them is applicable to our problem. It is also worth-mentioning a set of novel ranging-based applications, such as breathing pattern detection [21, 141, 100, 162, 150] and hand and finger gesture detection [119, 142, 127, 60, 101]. These applications can be classified as either *type-I ranging*[3] or *type-II ranging*, while our system belongs to *type-III ranging*. For the former two types, the analysis is carried over target-reflected signals. For the latter type, the analysis is over target-emitted signals. Thus, their design rationale and applied techniques are quite different.

Our Approach: We find that the received samples associated with each acoustic source generate a series of pulses in the time-frequency (t-f) domain. Lining up these pulses produces a "t-f sweep line" that can identify the corresponding acoustic source due to the unique combination of its signal offset time and the moving speed. Our formal analysis reveals that the slope of each t-f sweep line is a function of the vehicle's relative velocity. We thus address the first challenge by exploiting this relationship. *To our best knowledge, no existing literature has provided any closed-form formula of a moving source's t-f sweep line in the expression of its relative velocity, let alone leveraging the relationship for velocity estimation.* We address the second challenge by developing a modified generalized cross correlation method, called *MGCC*. MGCC consists of three major components: 1) extracting the LoS transmission component from the received signal for each acoustic source, 2) applying the generalized cross correlation function over the extracted signals received by two microphones on the smartphone to obtain the time difference of arrival (TDoA), and 3) calculating the DoA for each vehicle based on its TDoA. Comparing with conventional

[3] By ranging, we mean localizing/detecting objects based on their reflected or emitted electromagnetic (EM) or acoustic signals. According to the relation among the target, transmitter, and receiver in a ranging system, we define the following three types of ranging. For *type-I ranging*, an EM/acoustic wave is emitted from a transmitter and reflects off the target the wave encounters. The signal is reflected back to the receiver that picks up the echoed signal. The transmitter and the receiver are collocated. For *type-II ranging*, the signal is also emitted from the transmitter, reflected by the target, and captured by the receiver. The only difference is that the transmitter and the receiver are located differently. They are either cooperative or not. For *type-III ranging*, the target is localized through the analysis over its own emitted signals. Thus, the target is also the transmitter.

GCC, which is incapable of dealing with association ambiguity caused by coexistence of multiple sources, the proposed MGCC avoids this issue by analyzing t-f profiles of each source extracted from the previous step.

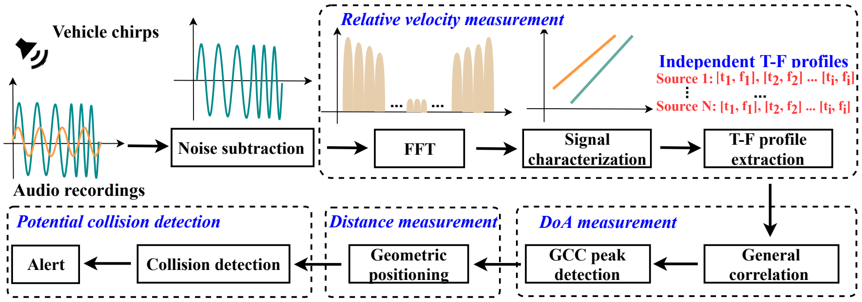


Figure 4.3: Acoussist system architecture.

The processing flow of Acoussist is summarized in Figure 4.3. It consists of vehicle-side mounted external speakers that emit acoustic chirps ranging from 17 KHz to 19 KHz and a pedestrian-side detection app. To facilitate the multi-source localization, two microphones at the pedestrian's smartphone are utilized. Upon receiving acoustic samples, Acoussist applies a high-pass filter to remove low-frequency components in the recorded samples. It then performs the short-time Fourier transform (STFT) over the de-noised samples. A t-f sweep line is identified for each acoustic source. It further estimates the relative velocity for each vehicle by analyzing the slope for each t-f sweep line. Following that, it measures the DoA of acoustic sources via the proposed MGCC method; its inputs are the t-f profiles from each acoustic source obtained from the previous step. By employing the geometric relations between the acoustic source and the pedestrian, it then calculates each vehicle's moving velocity and its distance to the pedestrian. Finally, the app decides if a pedestrian is safe to proceed to the crosswalk by analyzing the relations among all derived movement parameters, and produces an alert if needed.

As mentioned, Acoussist adopts the framework of *type-III ranging* that analyzes target's self-emitted signal for detection. If we adopt the other two types of ranging that utilizes target's reflected signal, the ranging distance would be significantly reduced due to the high decay coefficient experienced by acoustic signals. As shown in our experiments, the signal is still detectable by a smartphone when the v-p distance is as long as 240 ft under type-III ranging, but it drops to 48 ft under type-I/-II ranging which is unsuitable for moving object collision detection. Besides, Acoussist works for smartphones with more than one microphone. Luckily, most of current smartphones meet this requirement. As a note, Apple equips their devices with even four embedded microphones since iPhone 7 released in 2016.

The key contribution of this paper is summarized as follows:

- We develop an acoustic based collision detection system that assists pedestrians with vision impairments to perceive surrounding traffic conditions before crossing uncontrolled streets. It supplements the conventional hearing based solution. While collision avoidance systems for automobiles have been investigated for more than a decade, human-centered collision detection has rarely been studied.
- We address unique challenges when applying acoustic ranging to collision detection. Two salient technical contributions have been made. First, we propose a novel t-f sweep line based analysis that derives vehicle's relative velocity. With this basis, MGCC is developed to calculate vehicle's DoA according to its TDoA with respect to the smartphone's two mics. Both detection methods are capable of differentiating among multiple vehicles.
- From a generalized point of view, we study a *type-III homogeneous multi-source ranging* problem that has rarely been investigated in prior *ranging* literature.
- We implement Acoussist on COTS speakers and mobiles without involving any central server. We demonstrate the feasibility of Acoussist via extensive in-field testing.

Clarifications: *First*, Acoussist does not intend to overwrite a pedestrian's judgement; instead, it should be treated as an assisting tool that provides an added layer of protection for the visually impaired. That being said, if a conflict occurs between a pedestrian's judgement and the detection result, the pedestrian still holds the responsibility of decision making. A suggestive choice is to wait by the curb when either source indicates a potential collision. *Second*, Acoussist is designed to use at uncontrolled crosswalks existing in residential communities, local streets, and suburban areas, where there are common needs from the visually impaired for daily activities and commute. These venues generally impose relatively conservative vehicle speed limits. In an interview with five visually impaired students in our university, all of them claim that they would never consider crossing any less regulated road sections that allow 45 mph speed limit or higher on their own. *Third*, we stress that the functionality of Acoussist does not require all vehicles to participate. It can perform collision detection only to the participatory ones. In a worst case that no vehicle in the pedestrian's vicinity opt-in, it degrades to the hearing-based judgment scenario. Therefore, Acoussist will not perform worse than the current solution.

4.2 Overview and Background

4.2.1 Design Rationale

The White Cane Laws give visually impaired pedestrians the right-of-way in crosswalks, whether or not they are marked [103]. The laws require drivers to stop and yield to the blind who is crossing the street. On the other hand, the visually impaired rely on themselves to judge if the street is clear and when to proceed to the crosswalk by mainly referring to hearing. Since hearing is not always reliable, for example, mislead by the background noise, the blind may wrongly judge the traffic condition and enter streets even there are oncoming vehicles within close proximity. To avoid this hazard situation, Acoussist senses surrounding traffic conditions and estimates whether there is sufficient time for

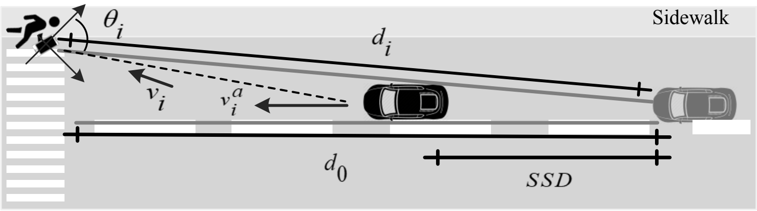
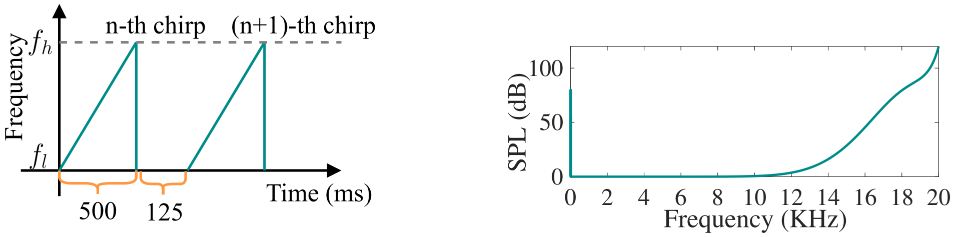


Figure 4.4: Design rationale of Acoussist.

nearby drivers to spot the pedestrian and then take reaction to stop cars. If not, an alert is generated at the pedestrian's smartphone, keeping her from proceeding to the crosswalk.

The design of Acoussist relies on a basic assumption that drivers obey the laws; for the collisions that are caused by careless driving are out of the scope of our discussion. Besides, there are some scenarios that people are using headphones/earphones listening to music or talking on the phone when crossing streets without care. They are not the focus of this work too. As shown in Figure 4.4, denote by d_0 the distance between a vehicle and the crosswalk, when the pedestrian is first spotted entering the crosswalk. Then the driver takes reaction and stops the vehicle. The distance that the vehicle travels before complete stop is called *stopping sight distance* (SSD) [71]. It is a near worst-case distance a driver needs to be able to see in order to have room to stop before colliding with something in the roadway. SSD is also one of several types of sight distance commonly used in road design. If $d_0 >$ SSD, i.e., the driver can stop the car before hitting into the pedestrian, the pedestrian can safely cross. This inequality can be used as a condition for pedestrian safety. Nonetheless, it is challenging for the pedestrian to measure d_0. Alternatively, we consider d_i, the v-p distance. Generally, $d_i \approx d_0$ when the vehicle is faraway. For example, given $d_0 = 150$ ft and the street width 30 ft, then $d_i \leq \sqrt{150^2 + 15^2} = 150.7$ ft. Thus, the pedestrian safety condition can be rewritten as $d_i >$ SSD.

As discussed later, d_i is a function of DoA of the vehicle to the pedestrian, denoted by θ_i, and SSD is a function of the vehicle's moving velocity, denoted by v_i^a. Besides, v_i^a is dependent of θ_i and the vehicle's velocity relative to the pedestrian, denoted by v_i.



(a) (b)

Figure 4.5: (a) Time-frequency domain representation of the chirp signal. (b) Human hearing threshold.

Eventually, our problem to determine if $d_i >$ SSD is satisfied is converted to estimate the values of v_i, θ_i, v_i^a, and d_i.

4.2.2 Acoussist Signal Design

Acoussist uses external speakers to emit acoustic chirps periodically. As shown in Figure 4.5(a), a chirp's frequency linearly sweeps from the minimum f_l to the maximum f_h over time. Chirp signals are widely used in radar applications for its capability of resolving multi-path propagation. In the time domain, the expression for one chirp is

$$s_c(t) = A\cos(\pi \frac{B}{T} t^2 + 2\pi f_l t) \tag{4.1}$$

where A is the amplitude, $B = f_h - f_l$, $t \in (0, T]$, and T is the chirp duration. We choose a high frequency chirp ranging from $f_l = 17$ KHz to $f_h = 19$ KHz. Such a range has been adopted by quite a few novel applications, such as biometric sensing [109] and acoustic imaging [89].

Although the frequency range of human hearing is generally considered from 20 Hz to 20 KHz, high frequency sounds must be much louder to be noticeable (including children and young adults) [115]. This is characterized by the absolute threshold of hearing (ATH), which refers to the minimum sound pressure that can be perceived in a quiet environment.

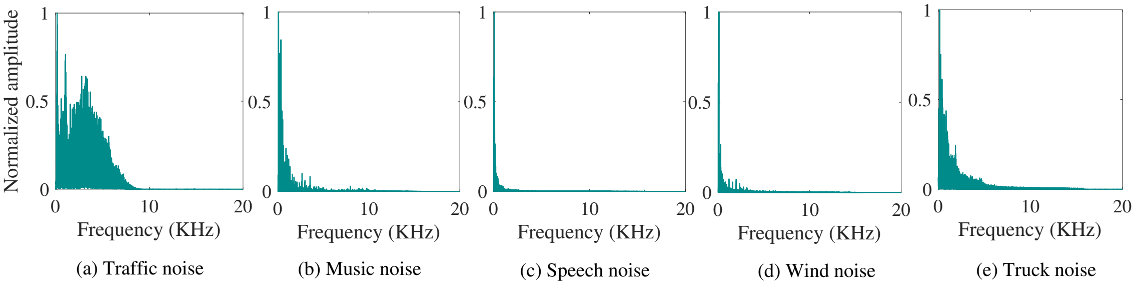


Figure 4.6: Frequency spreading of different background noises.

According to [128], we depict in Figure 4.5(b) the ATH with respect to sound frequency. ATH increases sharply for frequencies over 10 KHz. In particular, human ears can detect sounds of 1 KHz at 0 dB sound pressure level (SPL), but above 75 dB SPL for sound beyond 17 KHz, which has about 10,000 fold amplitude increase. In our implementation, the chirp signal is played at 69.3 dB and thus hardly perceptible by human hearing.

Another concern of applying acoustic ranging is that the signal may be polluted by background noise in outdoor environments. We extract some recordings for commonly seen outdoor noise from the well acknowledged dataset provided by the Google Audioset [47] and analyze their spectrum distribution. We find in Figure 4.6 that the background noise mainly concentrates on the lower-end of the frequency, mostly lower than 10 KHz. As our chirp signals are above 17 KHz, there is a clear gap between these two. By applying a high-pass filter to the received signal can easily filter out background noise.

The chirp duration and the separation between two consecutive chirps impact the overall performance of Acoussist. Too short a chirp will cause blurs in t-f profiles of received signals. Also, the chirp separation should be large enough to ensure that the main reflected signals of the current chirp are received before the next chirp is transmitted. However, too long a chirp duration or the separation will add delay to the system. As examined in 4.5.2, we found empirically that a duration of 500 ms and a separation of 125 ms represent a good tradeoff.

4.3 Multi-Vehicle Signal Characterization

In this section, we model the received signal with the presence of multiple vehicles[4]. The insight that we develop from this model guides the design of different modules of Acoussist.

As surrounding objects, such as buildings and trees, reflect acoustic signals, a chirp emitted from vehicle s_i arrives at a microphone r_m from multiple paths. Denote by $d_{i,m}^k$ the length of the k-th path associated with s_i and r_m. v_a is the speed of acoustic signals, which is considered as 340 m/s in our system. Let $\phi_{i,k}$ be the angle between the DoA of the k-th path and the line-of-sight (LoS) path respect to mic r_m. Following [129], the corresponding time-dependent signal propagation delay is calculated by

$$\tau_{i,m}^k(t) = \frac{d_{i,m}^k - v_i \cos(\phi_{i,k})t}{v_a}.$$

Let $a_{i,m}^k(t)$ be the attenuation experienced by the acoustic signal transmitted via the k-th path. Then, the aggregated time-domain channel response between s_i and r_m is expressed by

$$h_{i,m}(\tau, t) = \sum_{k=1}^{K} a_{i,m}^k(t)\delta(\tau - \tau_{i,m}^k(t))$$

i.e., the accumulated pulses arriving at different propagation delays. Given a particular time instance t, the source signal (4.1) can be rewritten as $s_c(t) = A\cos(2\pi f_t t)$, in which

[4] In this paper, we use “vehicle” and “source” interchangeably without causing confusion.

$f_t = \frac{\mathrm{d}(\pi\frac{B}{T}t^2+2\pi f_l t)}{2\pi \mathrm{d}t} = \frac{B}{T}t + f_l$. Then, the time-domain expression for mic r_m received signal coming from vehicle s_i is

$$y_{i,m}(\tau,t) = \sum_{k=1}^{K} Aa_{i,m}^{k}(t)\cos(2\pi f_t(t-\tau_{i,m}^{k}(t)))$$
$$= \sum_{k=1}^{K} Aa_{i,m}^{k}(t)\cos(2\pi f_t(1+\frac{v_i\cos(\phi_{i,k})}{v_a})t - \frac{2\pi f_t d_{i,m}^{k}}{v_a}).$$

By applying the continuous Fourier transformation over $y_{i,m}(\tau,t)$, its t-f representation is

$$Y_{i,m}(f,t) = \frac{A}{2}\sum_{k=1}^{K}\frac{a_{i,m}^{k}(t)v_a}{v_a+v_i\cos\phi_{i,k}}e^{-j2\pi f\frac{d_{i,m}^{k}}{v_a+v_i\cos\phi_{i,k}}}$$
$$\times[\delta(f-f_t\frac{v_a+v_i\cos\phi_{i,k}}{v_a})+\delta(f+f_t\frac{v_a+v_i\cos\phi_{i,k}}{v_a})].$$

Then, the t-f representation of the aggregated received signal from all N vehicles at mic r_m is written as

$$\begin{aligned}&Y_m(f,t) \\ &= \frac{A}{2}\sum_{i=1}^{N}\sum_{k=1}^{K}\frac{a_{i,m}^{k}(t)v_a}{v_a+v_i\cos\phi_{i,k}}e^{-j2\pi f\frac{d_{i,m}^{k}}{v_a+v_i\cos\phi_{i,k}}}\times[\delta(f-(\frac{B}{T}t+f_l)\frac{v_a+v_i\cos\phi_{i,k}}{v_a}) \\ &+\delta(f+(\frac{B}{T}t+f_l)\frac{v_a+v_i\cos\phi_{i,k}}{v_a})].\end{aligned} \tag{4.2}$$

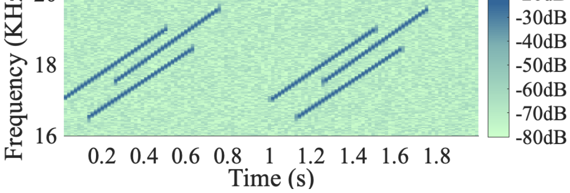


Figure 4.7: T-F profile of the received signal with the presence of three vehicles.

Figure 4.7 depicts the t-f profile of the received signal $Y_m(f,t)$ at a microphone when three vehicles are present, with their relative velocities v_i's at 0 mph, 20 mph, and -20 mph, respectively. The darker part in this heat map indicates the components of large power.

Insight: We can tell from (4.2) that the t-f profile of received samples consist of a series of impulses that exist when the corresponding f and t satisfy a set of linear equations $f = (\frac{B}{T}t + f_l)\frac{v_a+v_i\cos\phi_{i,k}}{v_a}$ $(k \in \{1,\cdots,K\})$. Besides, according to [18], $a^k_{i,m}(t) \propto e^{-\gamma d^k_{i,m}(t)}$ where γ is the acoustic amplitude decay coefficient. In the air propagation environment, γ is at least 40.3 dB/100m when the acoustic signal's frequencies are between [17 KHz, 19 KHz] [133]. Thus, the signals coming from the LoS transmission path (with index $k = 1$) out-weight other components from the rest paths. This is also another benefit of employing acoustic signals rather than radio signals. It is more convenient to extract LoS component from received signals that are mixed with multi-path transmissions. Each "line" in Figure 4.7 can be specified by

$$f = (\frac{B}{T}t + f_l)\frac{v_a + v_i\cos\phi_{i,1}}{v_a} = (\frac{B}{T}t + f_l)\frac{v_a + v_i}{v_a} \tag{4.3}$$

as $\phi_{i,1} = 0$. We call such a line as a *t-f sweep line*. As shown in Figure 4.7, each vehicle corresponds to a unique t-f sweep line, due to its unique combination of v_i and the chirp signal offset time. Therefore, the number of different t-f sweep lines that a mic detects implies the number vehicles within its proximity. More importantly, we also notice that the slope of a line contains the information of v_i. Therefore, we are able to infer the number of nearby vehicles and their relative velocities by analyzing t-f profile of received signals.

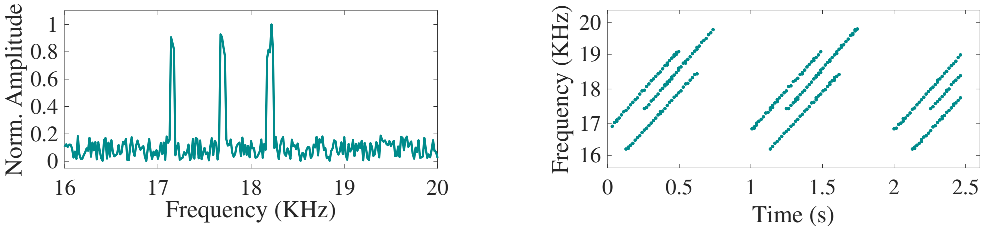
While the t-f signal analysis is a common approach for target tracking in a radar system, no existing work has tried to establish an explicit relation between the received signal's t-f sweep line and the source relative velocity. Such a relation enables velocity estimation when multiple sources with homogeneous signals are present. As discussed later, the analysis also lays the basis for the DoA measurement module. One reason that this idea has not been explored previously is because the generalized problem, *type-III homogeneous multi-source ranging*, is hardly observed in any other real-world ranging-based applications. For example, speaker localization [79, 36, 84] and noise identification [30] can be classified as *type-III heterogeneous multi-source ranging*. Radio-based indoor localization [23] can be treated as *type-III single-source ranging*. Radio-based breathing pattern detection [3], sleep monitoring [81], gesture detection [131], and radar guns all belong to *type-I ranging* or *type-II ranging*.

4.4 Design Details of Acoussist

Next, we discuss the design details of Acoussist's modules and describe how they interact to perform multi-vehicle detection.

4.4.1 Measurement of Relative Velocity

The objective of this module is to estimate each vehicle's velocity relative to the pedestrian, v_i. A straightforward solution is to analyze the Doppler frequency shift of the received chirps at the receiver. This task is easy if only one source is nearby or source signals are heterogeneous, e.g., different combinations of frequency components. In our case, oftentimes several vehicles are present. Additionally, they emit homogeneous chirps. These chirps overlap at the receiver, rendering distinguishing among them an extremely challenging task, let alone analyzing the frequency shift for velocity measurement. To address this issue, in the previous section, we formulate the received signal into a generalized expression that quantifies the effect of source movements by jointly characterizing the received signal's time and frequency properties. More importantly, we model the t-f profile of a source as a closed-form expression of its relative velocity. Specifically, the slope of a t-f sweep line, denoted as κ, is unique and dependent of v_i, i.e., $\kappa = \frac{B}{T}\frac{v_a+v_i}{v_a}$. Hence, v_i is calculated as $v_i = v_a(\kappa T/B - 1)$. As v_a, T and B are known values, the remaining task is to find out κ of each t-f sweep line.



(a)

(b)

Figure 4.8: (a) Normalized amplitude of all frequency components in a window. (b) All detected peaks in the received signal's t-f profile.

The app takes the denoised-stream at runtime as input and continuously slides a window of short time-width over it to get t-f profile by applying short-time Fourier transform (STFT) at each window. Consider a sliding window indexed by l; the window size is Δt. Figure 4.8(a) depicts all frequency components contained in this window. As a note, Figure 4.8(a) is actually a slice of Figure 4.7 at window l. Each peak exists at the frequency $f_i = (\frac{B}{T}t + f_l)\frac{v_a+v_i}{v_a}$ with $t = l \cdot \Delta t$. The rest components are the signals from multi-path propagation or ambient noise that may consist of sound of sudden wind or machinery in a construction site or their harmonics in the higher frequency range. We then identify all the peaks in window l by applying the *peak detection algorithm* [15]. Denote by (l, f_i) an index-frequency pair of window l. We are able to identify three such pairs in Figure 4.8(a), indicating three detectable vehicles.

Figure 4.8(b) plots all the index-frequency pairs obtained at all windows of the received signal's t-f profile; each dot is associated with one pair. To extract the t-f sweep lines from a set of discrete dots, we employ the *Hough transformation* technique [13], which is a classic scheme in detecting edges, including lines, circles and ellipses, from a digital image. In our scenario, by treating the collected index-frequency pairs as the entire dataset, the t-f sweep lines are then detected as the edges by applying Hough transmissions over the dataset. Since there are substantial prior discussions on Hough transformation, we omit its implementation details here. So far, we are able to extract the t-f sweep lines and thus the slope κ for each of them. Then the number of lines is exactly the number of detectable vehicles. Each relative velocity is computed by $v_i = v_a(\kappa T/B - 1)$.

One critical issue in STFT is to decide the frequency resolution (Δf) and time resolution, i.e., sliding window size (Δt). These two parameters determine whether frequency components close together can be separated and the time at which frequencies change. A properly selected set of Δf and Δt produces concentrated, rather than blurred, t-f sweep lines, which are essential to measure relative velocity accurately. Suppose the microphone's sampling rate is 64 KHz. Here, 64KHz is a conservative value. Many smartphones, such as Razer phone 2 and Pixel XL, even support a sampling rate of 192KHz. Δt and Δf are computed as $\Delta t = N/64$ KHz and $\Delta f = 64$ KHz$/N$, respectively, where N is the number of samples taken from a window. To strike a balance between these two, N is set to 2048 in our system. Given the chirp duration as 500 ms, the time resolution $\Delta t = 2048/64$ KHz $= 32$ ms is precise enough to capture the variance of received chirps in the time domain caused by multi-path propagation. Consider that most of vehicles are not traveling with speed lower than 10 mph, i.e., 4.5 m/s. According to (4.3), its spectrum offset is $f_t \frac{v_i}{v_a}$, which ranges from 225 Hz ($f_t = 17$ KHz) to 251 Hz ($f_t = 19$ KHz). Thus, the frequency resolution $\Delta f = 64$ KHz$/2048 = 31$ Hz is satisfactory to measure the spectrum offset.

4.4.2 Measurement of DoA

This module estimates the vehicle's DoA with respect to the pedestrian, θ_i. As shown in Figure 4.9, it is the angle between the LoS transmission and the line connecting two mics on the phone. Its measurement is achieved by analyzing the time difference of arrival (TDoA), denoted by τ_i, at the two mics. Since the v-p distance is much larger than the inter-mic distance, denoted by D, LoS propagation paths to the two mics are deemed parallel with each other. Then, θ_i is calculated as $\theta_i = \arccos(\frac{\tau_i v_a}{D})$. As v_a and D are available values, the remaining task is to find out τ_i. The inter-distance of two microphones of a mobile phone is available in open source dataset like [48]. This value can be instantiated during the initiation of the app.

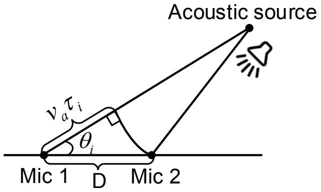


Figure 4.9: Illustration of vehicle DoA.

A naive approach: It examines the inter-channel phase difference (ICPD) to derive DoA [25, 166]. Specifically, the ICPD observed by two mics can be expressed as

$$\psi_i(f) = \angle\frac{Y_1(f,l)}{Y_2(f,l)} = 2\pi f\tau_i + 2\pi p_f \tag{4.4}$$

where $Y_m(f,l)$ ($m = 1$ or 2) is the T-F representation of the signal received at mic r_m under discrete time. The wrapping factor p_f is a frequency-dependent integer and $2\pi p_f$ represents possible phase wrapping. If $p_f = 0$, then τ_i is calculated by $\angle\frac{Y_1(f,l_1)}{Y_2(f,l_2)}/2\pi f$ and thus our problem is solved. In theory, $p_f = 0$ when D is smaller than half the wavelength [25]. In our system, the longest wavelength is about 2cm ($\frac{340\text{m/s}}{17\text{KHz}}$). Its halve, i.e., 1cm, is apparently smaller than the inter-mic distance for many smartphones, e.g., the ones with one mic located on bottom and the other on top. Therefore, the ICPD based TDOA measurement is unreliable in our case.

The proposed approach: We start from a conventional approach, called generalized cross correlation (GCC), to identify TDoA [24, 130]. Consider a function

$$R(\tau) = |\sum_{l}\sum_{f}\frac{Y_1^*(f,l)Y_2(f,l)}{|Y_1(f,l)Y_2(f,l)|}e^{-j2\pi f\tau}| = |\sum_{i}\sum_{l}\sum_{f}e^{-j\psi_i(f)}e^{-j2\pi f\tau}|$$

where $Y_m(f,l)$ ($m = 1$ or 2) follows the definition above. $Y_m^*(f,l)$ stands for the conjugate of $Y_m(f,l)$. $\psi_i(f,l) = 2\pi f\tau_i$ is the phase difference between Y_1 and Y_2. Ideally, $R(\tau)$ shows a peak at $\tau = \tau_i$. However, due to the presence of multiple vehicles, the strong interference from multiple acoustic sources generates multiple peaks, as shown in Figure 4.10(a).

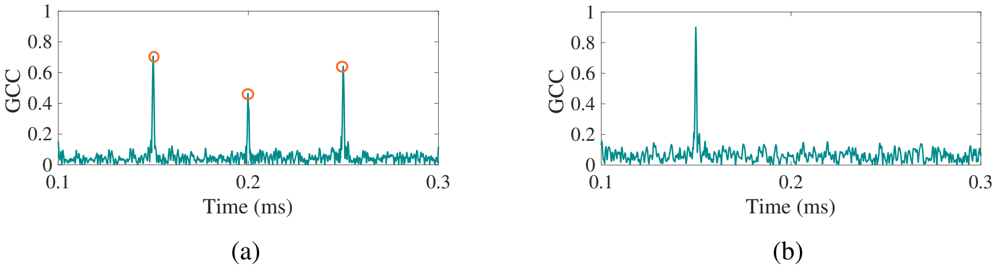


Figure 4.10: (a) Association ambiguity caused by multi-source and multi-path interference. (b) Eliminate association ambiguity by calculating GCC of LoS signals from a separated single source.

To avoid the association ambiguity in the conventional GCC approach, we adapt it to multi-source scenarios. Recall that in Section 4.4.1 we are able to extract the discrete index-frequency pairs for each source s_i, and thus the t-f profile of the received signal from s_i via the LoS path, denoted as $Y_{i,m,1}(f,l)$. Here, “1” means the index of the LoS path. Then, a modified GCC (*MGCC*) function that associated with a particular source s_i is expressed as

$$\begin{aligned} R_i(\tau) &= |\sum_l \sum_f \frac{Y^*_{i,1,1}(f,l) Y_{i,2,1}(f,l)}{|Y_{i,1,1}(f,l) Y_{i,2,1}(f,l)|} e^{-j2\pi f\tau}| \\ &= |\sum_l \sum_f e^{-j\psi_i(f)} e^{-j2\pi f\tau}|. \end{aligned} \tag{4.5}$$

Since $Y_{i,1,1}$ and $Y_{i,2,1}$ only contain the signals that are transmitted via the LoS paths from source s_i, there is only one peak at $\tau = \tau_i$, as shown in Figure 4.10(b). Thus, the TDoA τ_i of source r_i can be calculated by $\tau_i = \arg\max R_i(\tau)\ i \in [1, N]$. Then, θ_i is derived accordingly.

4.4.3 Measurement of Vehicle Velocity and V-P Distance

Vehicle velocity: The measurement of vehicle s_i's velocity v_i^a relies on the estimation results of its relative velocity v_i and DoA θ_i derived in Section 4.4.1 and 4.4.2, respectively.

Denote by $v_i^a(t_1)$ and $v_i^a(t_2)$ the vehicle's instance velocities at t_1 and t_2, respectively. Let $\delta_t = t_2 - t_1$ be the measurement interval. In the implementation, δ_t is set to 500 ms. Thus, $v_i^a(t_1)$ and $v_i^a(t_2)$ are deemed equal. As shown in Figure 4.11, denote by α the angle between the vehicle's moving direction AB and the line connecting two mics O_1O_2. Besides, let β be the angle

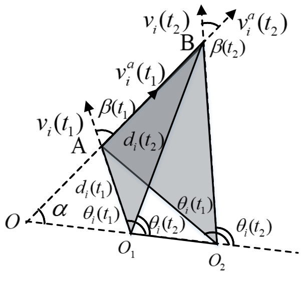


Figure 4.11: Geometric relation illustration.

between the vehicle's velocity v_i^a and its velocity relative to the pedestrian v_i. Then we have the following system of equations

$$\begin{cases} v_i^a(t_1)\cos\beta(t_1) = v_i(t_1),\ v_i^a(t_2)\cos\beta(t_2) = v_i(t_2) \\ \alpha + \beta(t_1) = \theta_i(t_1),\ \alpha + \beta(t_2) = \theta_i(t_2),\ v_i^a(t_1) = v_i^a(t_2) \end{cases}$$

The first two equations are from the relation between v_i^a and v_i. Regarding the third equation, $\theta(t_1) = \alpha + \angle OAO_1$ due to the application of exterior angle theorem in the triangle ΔAOO_1 in Figure 4.11. Similarly, the fourth equation holds. Since there are five variables (α, $\beta(t_1)$, $\beta(t_2)$, $v_i^a(t_1)$, $v_i^a(t_2)$) and five uncorrelated equations above, we can derive the closed form expression for $v_i^a(t)$ as

$$v_i^a(t) = v_i(t_2)\big(\cos(\arctan(\frac{\cos(\theta(t_1)-\theta(t_2)) - \frac{v_i(t_1)}{v_i(t_2)}}{\sin(\theta(t_1)-\theta(t_2))}))\big)^{-1}. \tag{4.6}$$

V-P distance: The v-p distance at t, i.e., $d_i(t)$, is calculated based on the knowledge of $v_i^a(t)$, $\theta_i(t)$, and $\tau_i(t)$ which are all known values by now. Consider two triangles ΔAO_1B and ΔAO_2B in Figure 4.11. Since v-p distance is significantly larger than the inter-mic distance, vehicle's DoAs with respect to the two mics are deemed the same, denoted by $\theta_i(t)$. Thus, $\angle AO_1B = \theta_i(t_1) - \theta_i(t_2)$ and $\angle AO_2B = \theta_i(t_1) - \theta_i(t_2)$. Due to the law of cosines in trigonometry, we have the following relation

$$\begin{cases} \cos(\theta_i(t_1) - \theta_i(t_2)) = \frac{d_i^2(t_1)+d_i^2(t_2)-(v_i^a\Delta t)^2}{2d_i(t_1)d_i(t_2)} \\ \cos(\theta_i(t_1) - \theta_i(t_2)) = \frac{(d_i(t_1)+v_a\tau_i(t_1))^2+(d_i(t_2)+v_a\tau_i(t_2))^2-(v_i^a\Delta t)^2}{2(d_i(t_1)+v_a\tau_i(t_1))(d_i(t_2)+v_a\tau_i(t_2))} \end{cases} \tag{4.7}$$

which is a system of two quadratic equations of two variables, $d_i(t_1)$ and $d_i(t_2)$. Thus, it is not difficult to solve it by some existing libraries. In the implementation, we use the GSL [45] that provides a library to compute the root of polynomials.

4.4.4 Piecing All Components Together

The design rationale of Acoussist is to estimate whether nearby drivers have sufficient time to spot the blind pedestrian and stop their vehicles when the pedestrian tends to enter the crosswalk.

As discussed, it is equivalent to have driver's SSD larger than the v-p distance d_i. In practice, we should further take into account the processing latency of the system, denoted by t_{dl}. We thus adopt the following conservative pedestrian safety condition

$$d_i > \text{SSD}_i + v_i^a \times t_{dl}. \tag{4.8}$$

Acoussist generates an alarm as long as any detectable vehicle s_i violates the above condition. In our implementation, t_{dl} is instantiated with a device-dependent value that is associated with 90% confidence level. To obtain this value, app runs on the smartphone dozens of times prior the usage. More details will be discussed in Section 4.5.2.

Since d_i and v_i^a are all known, the remaining task is to find out a particular vehicle i's SSD. As recommended by design standard of American Association of State Highway and Transportation Officials (AASHTO) [2], SSD is estimated by

$$\text{SSD} = 1.47 \times v_i^a t_{pr} + 1.075(v_i^a)^2/a \tag{4.9}$$

where v_i^a is the instant vehicle velocity that is derived in Section 4.4.3. t_{pr} and a stand for the driver's perception-reaction time and acceleration rate, respectively. AASHTO allows 2.5 seconds for t_{pr} and 11.2 ft/s^2 for a to accommodate approximately 90% of all drivers when confronted with simple to moderately complex road situations. SSD is the sum of two distances: 1) brake reaction distance (i.e., the distance traversed by the vehicle from the instant the driver sights an object necessitating a stop to the instant the brakes are applied);

and 2) braking distance (i.e., the distance needed to stop the vehicle from the instant brake application begins). According to AASHTO, in the above expression for SSD, conservative parameters are used, including a generous amount of time given for the perception-reaction process, and a fairly low rate of deceleration, such that it allows a below-average driver to stop in time to avoid a collision in most cases.

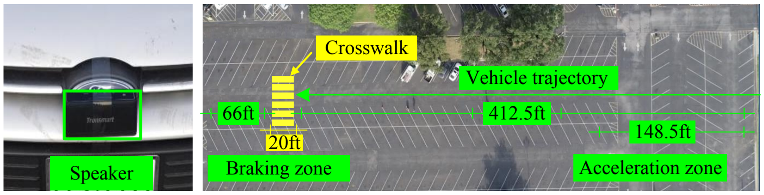
It is noteworthy that our system does not impose any requirement on the position/orientation of the phone during usage. Even though the calculation of DoA θ_i is dependent on the phone orientation, the pedestrian safety condition is related to v_i^a and d_i (4.8). As shown in (4.6) and (4.7), v_i^a and d_i are relevant to $\theta_i(t_1) - \theta_i(t_2)$ which is independent of the phone orientation.

4.5 Implementation and In-field Testing

4.5.1 Implementation Setup

Implementation: As a proof-of-concept implementation, we develop the prototype of Acoussist on four Tronsmart portable speakers, around $40 each, and three Android smartphones, Google Pixel XL, Galaxy S8 and Nexus 2. Four vehicles, Ford Focus 2014, Ford escape 2019, Toyota corolla 2017 and Honda Accord 2016, are used in the testing. A speaker is mounted in front of the vehicle, shown in Figure 4.12(a), playing the preloaded chirps that sweep from 17 KHz to 19 KHz at 69.3 dB [5]. We use the two built-in microphones at the smartphones to receive signals. An Android app is developed to process received signals and generate alarm when needed. We use NDK [46] to implement the STFT operations, and GNU Scientific Library (GSL) for other mathematical operations in our design.

[5] This value is measured at the speaker directly, which equals to 8.5×10^{-7} mW/cm^2 [132]. As a reference, the FDA regulation over preamendments diagnostic ultrasound equipment is $\leq$ 94 mW/cm^2 [42]. It restricts the maximal intense level of ultrasound exposed when people take medical evaluations such as peripheral vessel detection, cardiac diagnostic, and fetal imaging.



(a) Testing vehicle (b) A holistic view of testing site

Figure 4.12: In-field testing setup.

In-field testing setup: All testings are conducted at the campus parking lot as shown in Figure 4.12(b) during weekends when the space is relatively empty. A pedestrian stands at the end of the crosswalk and records the performance. In each testing round, a driver accelerates the vehicle to a target speed. Meanwhile, the pedestrian activates the app to sense the environment. If no alarm is generated, the pedestrian waves a flag, indicating the action of street crossing. Otherwise, she keeps the flag down, indicating waiting at the curb. Upon noticing a waving flag, the driver takes reaction and stops the car. The reason we use flag signals instead of having a pedestrian physically proceed to the crosswalk is for safety consideration. Besides, as the driver needs be signaled with the pedestrian's action of street crossing by waving a flag, the pedestrian cannot be simulated by a stand mounted with a smartphone. A test is viewed success, if a) the flag is not waved, since a potential collision is detected, or b) the flag is waved while the vehicle stops completely before reaching the crosswalk.

Acoussist requires users to hold their smartphones steady for about 1 second to have an accurate detection of oncoming vehicles. It is also the time duration between the time point that a user activates the app and the time point that he/she decides to wave the flag or not. Beyond the 1 second time limit, the user can choose to wave the flag at any time, as long as no alert is observed. The 1 second is attributed from two aspects, the duration of two consecutive measures to derive the v-p distance (δ_t =500 ms) and the processing delay (the 90-percentile value of t_{dl} =220.7 ms as shown in Section 5.2).

Evaluation metrics: The performance of our system is evaluated via the following metrics: ranging distance, warning distance, miss detection ratio (MDR), and false alarm ratio (FAR). Particularly, ranging distance is the v-p distance at which the app is able to measure this value for the first time. It implies the largest detectable range of our system. Warning distance is the v-p distance at which the pedestrian safety condition (4.8) is violated for the first time. MDR is the probability that a vehicle which has violated (4.8) but not detected. FAR is the probability that a vehicle satisfies (4.8) but wrongly reported. Traffic cones are placed along the vehicle trajectory. To measure the ranging distance, the pedestrian records the parking slot number, where the pedestrian stands denotes the first slot. With the assistance of traffic cones, the v-p distance becomes measurable. Then, the ranging distance is approximated by multiplying the slot number with the width of each slot, 10.7 ft in our case. Warning distance is obtained similarly.

4.5.2 Micro Benchmark

Impact of parameter settings: We first examine the impact of two most crucial parameters of our system, Δt and δ_t. Recall that Δt is the STFT window size and δ_t is the time interval between two consecutive measures. Figure 4.13(a) shows the accuracy performance of Acoussist with various Δt. The best performance 93.6% exists when $\Delta t = 2048$. As discussed in Section 4.4.1, the value of Δt strikes a trade-off balance between frequency resolution Δf and time resolution Δt of STFT. Figure 4.13(b) shows the accuracy performance with various δ_t. The best performance is achieved for $\delta_t = 500$ ms. On one hand, a large δ_t and thus an apparent difference between $\theta_i(t_1)$ and $\theta_i(t_2)$ is beneficial for deriving an accurate v_i^a. On the other hand, it inevitably leads to a long processing delay which impacts the detection accuracy. Δt and δ_t are set to 2048 and 500 ms, respectively, in the rest experiments.

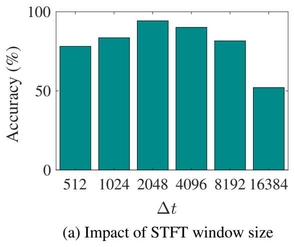


Figure 4.13: Impact of parameters.

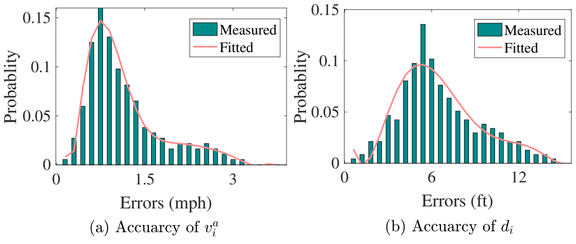


Figure 4.14: Accuracy of measurements.

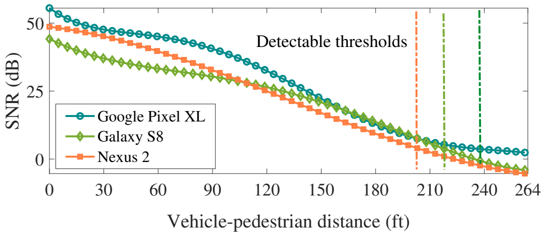


Figure 4.15: The received signal SNR with different v-p distance.

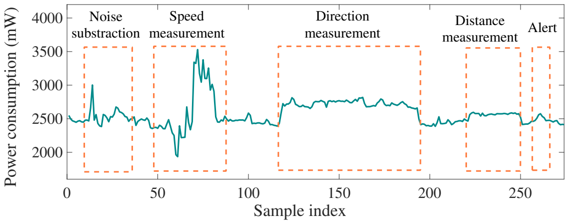
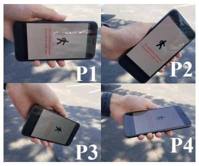


Figure 4.16: Instant power readings.

Table 4.1: Detection performance of different devices.

Device	Pixel XL	Galaxy S8	Nexus 2
MDR	3.4%	4.1%	3.6%
FAR	3.2%	3.7%	4.8%

Table 4.2: Impact of phone orientation.

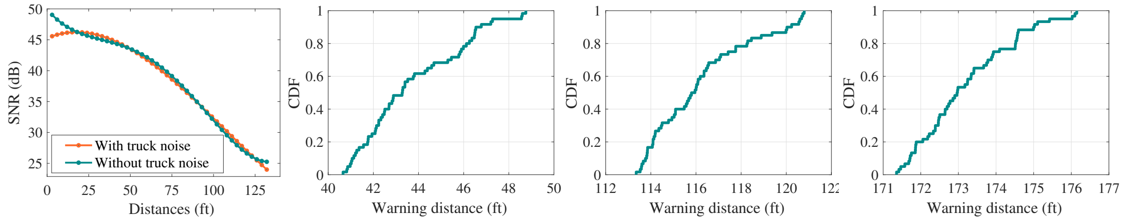
Orientation		**Ranging dist.**	**Warning dist.**
	P1	189.8 ± 10.1 ft	157.6 ± 10.4 ft
	P2	181.1 ± 13.2 ft	155.3 ± 15.3 ft
	P3	186.9 ± 11.8 ft	150.7 ± 10.5 ft
	P4	178.1 ± 14.4 ft	159.2 ± 10.5 ft

Measurement performance of motion parameters: Figure 4.14 depicts the distribution of measurement errors of velocity v_i^a and v-p distance d_i. The measurement error is defined as the difference between measures and the ground truth. Here, the ground truth of v_i^a and d_i is set to 25 mph and 120 ft, respectively. We observe that 90% of errors for these parameters are within 2.4 mph and 11.8 ft, respectively, which are acceptable for implementation.

Ranging distance and warning distance: Figure 4.19(a) evaluates the ranging distance of our system with respect to the vehicle speed. The ranging distance decreases from 208.1 ft to 165.5 ft on average when the speed changes from 5 mph to 45 mph. This is because the vehicle travels a longer distance at a higher speed given δ_t and thus perceives a shorter v-p distance when this value is first obtained. As shown in Figure 4.19(b), the warning distance increases almost linearly as the speed grows from 5 mph to 30mph. It reaches 175.2 ft when $v_i^a = 30$ mph. The result meets our expectation; when a vehicle moves faster, the driver needs longer distance to react and stops the vehicle which corresponds to a larger warning distance. However, as the speed continues to increase, the warning distance experiences slight decrease. This is because the warning distance is capped by the ranging distance. As the latter decreases, it also brings down the former.

Impact of different devices: Figure 4.15 shows the smartphone's received SNRs of chirps with frequency 17 KHz-19 KHz at different v-p distances. Three devices exhibit different sensitivity responding to high-frequency signals. Particularly, the detectable threshold, defined as the maximum distance within which the received signal are perceivable from the background noise, is 237.6 ft, 221.1 ft and 207.9 ft for Google Pixel, Galaxy S8, and Nexus 2, respectively. They bring about various detection performances as shown in Table 4.1. Combining the results of Figure 4.15 and Table 4.1, we observe a positive correlation between a device's detectable threshold and its detection accuracy, and Pixel XL has the best performance among the three.

Impact of phone orientation: We also examine if the phone orientation in usage impacts the detection performance. We test four different positions, the combinations of the screen facing above/aside and the head pointing up/down, as shown in Table 4.2. We find that the ranging distance and the warning distance are almost the same for all four positions. Thus, the performance of Acoussist is independent of how the pedestrian holds the phone. It meets our discussion in Section 4.4.4.



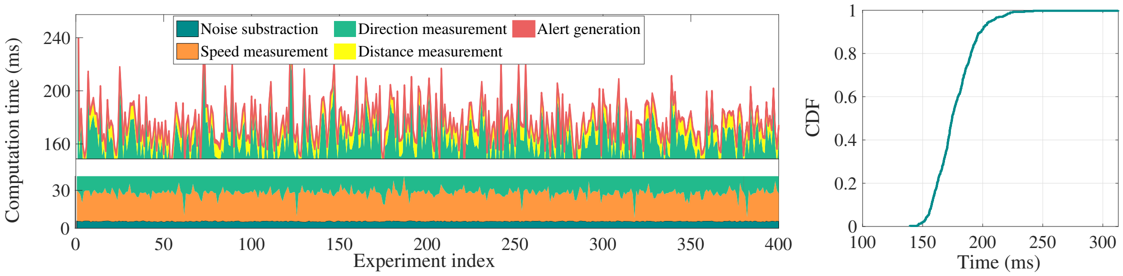
(a) Received signal SNR. (b) Vehicle speed v_i^a = 10 mph. (c) Vehicle speed v_i^a = 20 mph. (d) Vehicle speed v_i^a = 30 mph.

Figure 4.17: Impact of the background noise.

Impact of background noise: Among commonly observed background noise in streets, truck sound is typically the most powerful one. We thus evaluate its impact to the performance of Acoussist. First of all, as shown in Figure 4.6 (e), the signal frequency components are mainly concentrated on the lower-end of the frequency. Particularly, 88.7% of them reside lower than 10 KHz. Recall that the acoustic chirp signal used by Acoussist ranges between 17 KHz and 19 KHz. Thus, there is a clear gap between the truck sound and the acoustic chirp signal. In our design, a high-pass filter, with a cutting frequency of 10 KHz, is then applied to get rid of most background noise, including traffic noise, music noise, speech noise, construction noise, as well as truck sound.

On the other hand, we notice that truck noise does have frequency components above 10 KHz. To examine its impact to f-t analysis of our system, we first add truck sound as background noise to the chirp signals recorded at different distances by using Matlab audio toolbox [94]. Then SNR is measured after passing the received signal through all noise removal modules. The relation of SNR versus the v-p distance is plotted in Figure 4.17(a). As a comparison, we also show the SNR without truck noise. We observe that the two curves are quite similar to each other, except when the pedestrian is very close to the noise source, i.e., within 25 ft. We further depict in Figure 4.17(b)-(d) the CDF of warning distance under different vehicle speeds, from 10 mph to 30 mph. Recall that warning distance is the v-p distance at which the alert is triggered. Take Figure 4.17(b) as an illustration. When the vehicle speed is at 10 mph, all alerts are generated when the v-p distance is between 40.6 ft and 48.7 ft. Thus, vehicles are all detected even at distances much longer than 25 ft away from the pedestrian. Combining the observations above, we can infer that a truck will trigger the alert even it is larger than 25 ft away from the pedestrian. Besides, its detection performance should be similar to regular vehicles. If a truck is within 25 ft from the pedestrian, its presence can be easily picked up by human ears. In this scenario, the visually impaired pedestrians can simply rely on their hearings to detect the potential hazard.

Energy consumption: A dedicated hardware, Monsoon power monitor [98], is applied to measure the energy consumption of mobile phones for running our app. During the measurement, we keep other components, e.g., WiFi and Bluetooth, offline. Figure 4.16 shows the instant power reading via the power monitor when executing one detection. We clearly specify the part dedicated to each module. We can tell that the measurement of relative velocity and DoA consumes a larger amount of power among all modules, which is about 2967.2 mW and 2656.8 mW on average, separately. As a note, the average power consumption of some common smartphone tasks, such as video call, map service, and web browsing take 3351.6 mW, 2642.8 mW, and 1732.7 mW, respectively. Besides, our app is only activated when a pedestrian tends to cross streets and thus offline most of the time. Thus, the power consumption of our app is practically acceptable.



(a) Stacked computation time

(b) Total computation time

Figure 4.18: Processing latency.

Table 4.3: Detection performance when a vehicle is at different speeds.

Speed (mph)	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**
MDR	11.3%	9.9%	6.4%	5.4%	4.7%	3.4%	5.8%	6.3%	8.2%
FAR	10.0%	9.5%	7.2%	6.5%	5.3%	3.2%	2.8%	2.5%	2.1%

Processing latency: Figure 4.18(a) gives the stacked computation time of each system module. The module for DoA measurement incurs the largest delay, which is about 137.2 ms on average. This is because it involves an exhaustive search for the solution of the MGCC function. Figure 4.18(b) further illustrates the cumulative distribution function (CDF) of the total processing latency of the app. The average value is 186.3ms, with 90% of measurements lower than 220.7 ms. We thus instantiate t_{dl} with 220.7 ms for the implementation (4.8). We also believe that by leveraging the parallelization, we can further bring down the processing latency.

4.5.3 System Benchmark

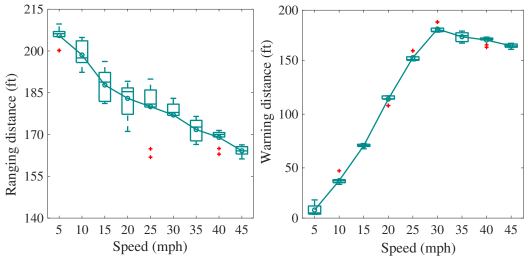
Impact of vehicle speeds: Table 4.3 gives the detection accuracy of Acoussist toward a vehicle in a wider range of speeds. Interestingly, both MDR and FAR experience significant decrease when the speed increases from 5 mph to 45 mph. This is because a higher speed generates a larger slope of the t-f sweep line as revealed in (4.3). As a result, the difference between $f(t + \Delta t)$ and $f(t)$ will be more apparent to tolerate errors caused by insufficient frequency resolution Δf. Therefore, when a target vehicle moves in a higher speed, its t-f profile tends to more accurate which leads to a better detection accuracy. MDR grows as the speed continues to increase from 30 mph to 45 mph. This is because the ranging distance becomes close or even shorter than the the warning distance when a vehicle is at a high speed. As a result, some hazard situations are missed in the detection.

Table 4.4: Detection performance with the presence of multiple vehicles.

Speed (mph)	**5**	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**
MDR	8.5%	6.8%	4.3%	3.7%	2.6%	1.9%	4.0%	5.5%	6.2%
FAR	12.0%	11.6%	10.1%	8.4%	7.3%	6.7%	5.0%	4.5%	3.2%

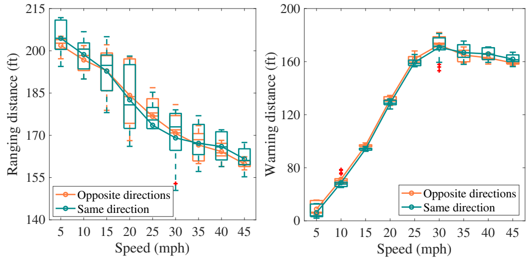
Impact of multiple vehicles: We examine the detection performance of Acoussist with the presence of four vehicles in Table 4.4. Compared with Table 4.3, we notice that FAR slightly increases with the presence of more cars. This is because a false alarm is generated by the system when any of the four vehicles is falsely reported to incur a potential collision. In contrast, MDR becomes smaller when there are more vehicles. This is because a collision is correctly forecast, when any one of the vehicles triggers the alarm. While FAR experiences a slight increase, it does not impact the performance of Acoussist much. A visually impaired pedestrian uses Acoussist to double-confirm the situation when sensing a clear street with hearing. Thus, MDR is more crucial than FAR in practical usage.

Figure 4.20 shows the ranging distance and warning distance when vehicles move in the same/opposite direction(s). While the average measures are closely the same, the variance associated with opposite directions is smaller than the same direction. This is because t-f sweep lines of the four vehicles are better separated and easier to extract in the former case.



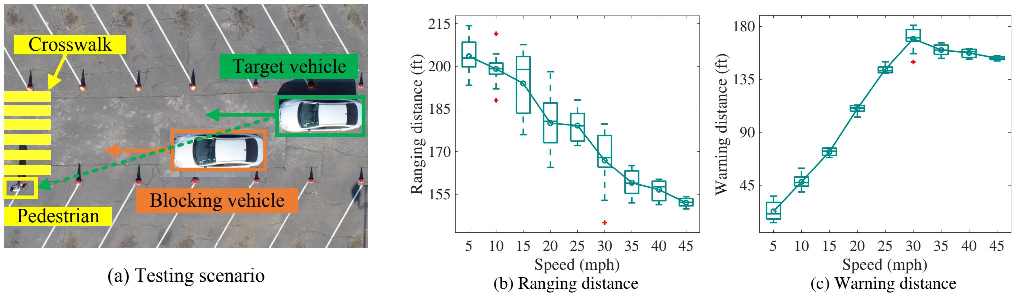
(a) Ranging distance (b) Warning distance

Figure 4.19: Impact of vehicle speeds.



(a) Ranging distance (b) Warning distance

Figure 4.20: Impact of multiple vehicles.



(a) Testing scenario (b) Ranging distance (c) Warning distance

Figure 4.21: Impact of nearby objects.

Impact of nearby objects: To evaluate the impact of nearby objects, we place a second vehicle to partially block the line of sight between the target vehicle and the pedestrian (as shown in Figure 4.21(a)). The ranging distance and the warning distance toward the target vehicle is shown in Figure 4.21(b) and 4.21(c), respectively. The trend of these two distances with respect to the vehicle speed is very similar to that in Figure 4.19(a) and 4.19(b), the performance without any blocking object. However, the two distances exhibit a larger variance with the existence of a blocking object. This is because the blocking object absorbs a portion of energy of chirps in the LoS path and thus slightly impacts the detection performance.

Impact of time/weather of usage: Table 4.5 shows the detection accuracy of Acoussist at different time of a day. The performance is relatively stable. Acoussist performs well at evenings when there are typically lack of visible light. Note that some existing systems for pedestrian safety, such as WalkSafe [140], rely on back camera of mobile phones to detect hazard vehicles. Thus, their performance is largely impacted by the time of usage. Table 4.6 further compares the detection performance under different weather conditions. We notice that both MDR and FAR experience slight increase in rainy days due to higher loss of acoustic signals when propagating in saturated air.

Table 4.5: Detection performance at different time of a day.

Usage time	Morning	Noon	Afternoon	Evening
MDR	3.8%	4.5%	4.7%	5.2%
FAR	6.2%	4.9%	5.3%	4.8%

Table 4.6: Detection performance under different weather conditions.

Usage weather	Sunny	Windy	Cloudy	Rainy
MDR	3.7%	3.5%	4.2%	4.4%
FAR	4.8%	5.3%	4.8%	5.2%

CHAPTER 5

CONCLUSIONS

In this thesis, we discuss three mechanism designs that extends the sensing boundary of mobile systems to secure the device pairings, disclose potential security threats, and develop novel assistive applications to facilitate blind pedestrians' daily travels.

First, we propose a novel pairing scheme for wearables devices, which builds upon the core idea of treating the human body as a conductor. We observe that the on-body wearables can receive identical RF noise variations, regardless of which skin positions they contact. RF noise serves as an ideal entropy source due to its ubiquitous presence and high randomness. Under the touch-to-access policy, we present a pairing protocol that turn the RF variation trends as the ingredients to securely distribute the pairing key between two legitimate devices. A prototype is developed to evaluate the effectiveness of the proposed scheme from the aspects of robustness against various types of attackers, key generation performances, and time and energy consumption.

Second, we present, Periscope, a new side-channel eavesdropping attack to infer users' PIN from the human-coupled EM emanations from touchscreens during the typing process. Periscope is motivated by the observation that finger movements over the touchscreen leads to time-varying coupling between these two. Consequently, it impacts the screen's EM emanations that can be picked up by a remote sensory device. We developed an comprehensive analytic model to formulate the relationship between screen-finger distances and EM measurements. Based on the analysis, the finger's movement traces are reconstructed to recover the typed PINs. Meanwhile, the proposed attack is completely training-free and do not need to collect large amount of user-specific datasets in advance of the attack. We build the prototype on a low-cost MCU device and conduct a suite of experiments to validate this attack's impact.

Last, we design a new application, Acoussist, that uses COTS smartphones' microphones to measure the motion status of oncoming vehicles and then provide assistive guidance to the pedestrians with impairments before they take actions across the uncontrolled streets. The key novelty of Acoussist is to leverage the t-f sweep line of received chirp signals to derive multiple vehicles' relative speeds and resolve the association ambiguity issue in their DoA measurements. We exploit geometric relations of vehicle's sequential motion status to calculate the important movement parameters, such as vehicle velocity and distance to the pedestrian. Based on these parameters, a carefully designed alert generation mechanism is proposed to warn the pedestrian if there is one vehicle cannot stop before the crosswalk and thus lead to potential collision hazards.

REFERENCES

[1] Envelope extraction. https://www.mathworks.com/help/signal/ug/envelope-extraction-using-the-analytic-signal.html.

[2] AASHTO. A policy on geometric design of highway and streets. `https://www.academia.edu/33524500/AASHTO_Green_Book_2011.PDF`, 2011.

[3] H. Abdelnasser, K. A. Harras, and M. Youssef. Ubibreathe: A ubiquitous non-invasive wifi-based breathing estimator. In *Proceedings of the International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)*, pages 277–286, 2015.

[4] K. Ali, A. X. Liu, W. Wang, and M. Shahzad. Keystroke recognition using wifi signals. In *Proceedings of the 21st annual international conference on mobile computing and networking*, pages 90–102, 2015.

[5] S. AlShowarah. The effectiveness of dynamic features of finger based gestures on smartphones' touchscreens for user identification. *International Journal of Interactive Mobile Technologies (iJIM)*, 11(1):133–142, 2017.

[6] J. Andersen and P. Balling. Admittance and radiation efficiency of the human body in the resonance region. *Proceedings of the IEEE*, 60(7):900–901, 1972.

[7] T. Aono, K. Higuchi, T. Ohira, B. Komiyama, and H. Sasaoka. Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels. *IEEE Transactions on Antennas and Propagation*, 53(11):3776–3784, 2005.

[8] Apple. Use airpods and other bluetooth accessories with apple watch. https://support.apple.com/en-us/HT204218, assessed at January, 2020.

[9] D. Asonov and R. Agrawal. Keyboard acoustic emanations. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2004.

[10] A. Asvadi, C. Premebida, P. Peixoto, and U. Nunes. 3d lidar-based static and moving obstacle detection in driving environments: An approach based on voxels and multi-region ground planes. *Robotics and Autonomous Systems*, 83:299–311, 2016.

[11] M. Backes, T. Chen, M. Duermuth, H. P. Lensch, and M. Welk. Tempest in a teapot: Compromising reflections revisited. In *2009 30th IEEE Symposium on Security and Privacy*, pages 315–327. IEEE, 2009.

[12] M. Backes, M. Dürmuth, and D. Unruh. Compromising reflections-or-how to read lcd monitors around the corner. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 158–169. IEEE, 2008.

[13] D. H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern recognition*, 13(2):111–122, 1981.

[14] Y. Berger, A. Wool, and A. Yeredor. Dictionary attacks using keyboard acoustic emanations. In *Proceedings of the 13th ACM conference on Computer and communications security*, pages 245–254, 2006.

[15] E. Billauer. Peak detection. `http://billauer.co.il/peakdet.html`, 2019.

[16] E. Billauer. Peak detection algorithm. http://billauer.co.il/peakdet.html, 2020.

[17] L. Cai and H. Chen. Touchlogger: Inferring keystrokes on touch screen from smartphone motion. *HotSec*, 11(2011):9, 2011.

[18] N. R. Center. Attenuation of sound waves. `https://www.nde-ed.org/EducationResources/CommunityCollege/Ultrasonics/Physics/attenuation.htm`, 2020.

[19] J.-S. Chang, A. J. Kelly, and J. M. Crowley. *Handbook of electrostatic processes*. CRC Press, 1995.

[20] S. Chatterjee and D. McLeish. Fitting linear regression models to censored data by least squares and maximum likelihood methods. *Communications in Statistics-Theory and Methods*, 15(11):3227–3243, 1986.

[21] J. Chauhan, Y. Hu, S. Seneviratne, A. Misra, A. Seneviratne, and Y. Lee. Breathprint: Breathing acoustics-based user authentication. In *Proceedings of the International Conference on Mobile Systems, Applications, and Services*, pages 278–291, 2017.

[22] Y. Chen, T. Li, R. Zhang, Y. Zhang, and T. Hedgpeth. Eyetell: Video-assisted touchscreen keystroke inference from eye movements. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 144–160. IEEE, 2018.

[23] S. Chumkamon, P. Tuvaphanthaphiphat, and P. Keeratiwintakorn. A blind navigation system using rfid for indoor environments. In *Proceedings of the International Conference on Electrical Engineering/-Electronics, Computer, Telecommunications and Information Technology*, volume 2, pages 765–768, 2008.

[24] A. Clifford and J. Reiss. Calculating time delays of multiple active sources in live sound. In *Proceedings of the Audio Engineering Society Convention*, pages 1–8, 2010.

[25] M. Cobos, J. J. Lopez, and D. Martinez. Two-microphone multi-speaker localization based on a laplacian mixture model. *Digital Signal Processing*, 21(1):66–76, 2011.

[26] G. Cohn, S. Gupta, T.-J. Lee, D. Morris, J. R. Smith, M. S. Reynolds, D. S. Tan, and S. N. Patel. An ultra-low-power human body motion sensor using static electric field sensing. In *Proceedings of the ACM Conference on Ubiquitous Computing (UbiComp)*, 2012.

[27] G. Cohn, D. Morris, S. Patel, and D. Tan. Humantenna: using the body as an antenna for real-time whole-body interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, 2012.

[28] G. Cohn, D. Morris, S. N. Patel, and D. S. Tan. Your noise is my command: sensing gestures using the body as an antenna. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, 2011.

[29] H. Crawford and E. Ahmadzadeh. Authentication on the go: Assessing the effect of movement on mobile device keystroke dynamics. In *Thirteenth Symposium on Usable Privacy and Security ({SOUPS} 2017)*, pages 163–173, 2017.

[30] N. Dey and A. S. Ashour. Applied examples and applications of localization and tracking problem of multiple speech sources. In *Direction of arrival estimation and localization of multi-speech sources*, pages 35–48. Springer, 2018.

[31] Dnschecker.org. Mac address lookup. https://dnschecker.org/mac-lookup.php, 2020.

[32] DPS. Department of public safety guide book. `http://www.dps.texas.gov/internetforms/forms/dl-7.pdf`, 2019.

[33] Y.-L. Du, Y.-H. Lu, and J.-L. Zhang. Novel method to detect and recover the keystrokes of ps/2 keyboard. *Progress In Electromagnetics Research*, 41:151–161, 2013.

[34] W. Elmenreich. An introduction to sensor fusion. *Vienna University of Technology, Austria*, 502:1–28, 2002.

[35] V. Erdélyi, T.-K. Le, B. Bhattacharjee, P. Druschel, and N. Ono. Sonoloc: Scalable positioning of commodity mobile devices. In *Proceedings of the Annual International Conference on Mobile Systems, Applications, and Services*, pages 136–149, 2018.

[36] C. Evers, A. H. Moore, and P. A. Naylor. Acoustic simultaneous localization and mapping (a-slam) of a moving microphone array and its surrounding speakers. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6–10, 2016.

[37] S. Fang, I. Markwood, Y. Liu, S. Zhao, Z. Lu, and H. Zhu. No training hurdles: Fast training-agnostic attacks to infer your typing. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 1747–1760, 2018.

[38] T. Feng, Z. Liu, K.-A. Kwon, W. Shi, B. Carbunar, Y. Jiang, and N. Nguyen. Continuous mobile authentication using touchscreen gestures. In *Proceedings of the IEEE Conference on Technologies for Homeland Security*, 2012.

[39] R. D. Findling, M. Muaaz, D. Hintze, and R. Mayrhofer. Shakeunlock: Securely transfer authentication states between mobile devices. *IEEE Transactions on Mobile Computing*, 16(4):1163–1175, 2016.

[40] C. Flores, P. Merdrignac, R. de Charette, F. Navas, V. Milanés, and F. Nashashibi. A cooperative car-following/emergency braking system with prediction-based pedestrian avoidance capabilities. *IEEE Transactions on Intelligent Transportation Systems*, 20(99):1–10, 2018.

[41] D. Foo Kune and Y. Kim. Timing attacks on pin input devices. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 678–680, 2010.

[42] Food, D. Administration, et al. Guidance for industry and fda staff information for manufacturers seeking marketing clearance of diagnostic ultrasound systems and transducers. *Rockville, MD: FDA*, 2008.

[43] Fujitsu. Capacitive touch sensors: application fields, technology overview and implementation example. https://www.fujitsu.com/downloads/MICRO/fme/articles/fujitsu-whitepaper-capacitive-touch-sensors.pdf, 2021.

[44] M. Ghaddar, L. Talbi, T. A. Denidni, and A. Sebak. A conducting cylinder for modeling human body presence in indoor propagation channel. *IEEE Transactions on Antennas and Propagation*, 55(11):3099–3103, 2007.

[45] GNU. Gnu scientific library. `https://www.gnu.org/software/gsl/`, 2019.

[46] Google. Android ndk. `https://developer.android.com/ndk`, 2019.

[47] Google. Audioset. https://research.google.com/audioset/, 2020.

[48] Google. Phone device metrics. `https://material.io/resources/devices/`, 2020.

[49] J. Guy. Eu requires carmakers to add fake engine noises to electric cars. `https://www.cnn.com/2019/07/01/business/electric-vehicles-warning-noises-scli-intl-gbr/index.html`, 2019.

[50] J. Han, A. J. Chung, M. K. Sinha, M. Harishankar, S. Pan, H. Y. Noh, P. Zhang, and P. Tague. Do you feel what i hear? enabling autonomous iot device pairing using different sensor types. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2018.

[51] J. Han, M. Harishankar, X. Wang, A. J. Chung, and P. Tague. Convoy: Physical context verification for vehicle platoon admission. In *Proceedings of the International Workshop on Mobile Computing Systems and Applications (HotMobile)*, 2017.

[52] M. Hessar, V. Iyer, and S. Gollakota. Enabling on-body transmissions with commodity devices. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, 2016.

[53] C. Hoffman. How to see who's connected to your wi-fi network. hhttps://www.howtogeek.com/204057/how-to-see-who%E2%80%99s-connected-to-your-wi-fi-network/, 2020.

[54] D. C. Howell. *Statistical methods for psychology*. Cengage Learning, 2009.

[55] W. Huang, Y. Xiong, X.-Y. Li, H. Lin, X. Mao, P. Yang, and Y. Liu. Shake and walk: Acoustic direction finding and fine-grained indoor localization using smartphones. In *Proceedings of the IEEE Conference on Computer Communications*, pages 370–378, 2014.

[56] I. D. C. (IDC). Idc forecasts steady double-digit growth for wearables as new capabilities and use cases expand the market opportunities. https://www.idc.com/getdoc.jsp?containerId=prUS44930019, assessed at January, 2020.

[57] A. Inc. Arduino chips. https://www.arduino.cc, assessed at January, 2020.

[58] T. instruments. Capacitive sensing basics. http://software-dl.ti.com, 2020.

[59] T. instruments. Cc2500 transceiver. http://www.ti.com/product/CC2500, assessed at January, 2020.

[60] Y. Iravantchi, M. Goel, and C. Harrison. Beamband: Hand gesture sensing with ultrasonic beamforming. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, page 15, 2019.

[61] Jabra. Jabra elite 75t compatible with smartwatches. https://www.jabra.com/supportpages/jabra-elite-75t/100-99090000-02/faq/What-tablet-smartphone-and-smartwatch-operating-systems-are-compatible-with-Jabra-SoundPlus/, assessed at January, 2020.

[62] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. On the effectiveness of secret key extraction from wireless signal strength in real environments. In *Proceedings of the Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2009.

[63] W. Jin, M. Li, S. Murali, and L. Guo. Harnessing the ambient radio frequency noise for wearable device pairing. https://doi.org/10.1145/3372297.3417288, 2020.

[64] W. Jin, M. Xiao, H. Zhu, S. Deb, C. Kan, and M. Li. Acoussist: An acoustic assisting tool for people with visual impairments to cross uncontrolled streets. https://doi.org/10.1145/3432216, Dec. 2020.

[65] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 1999.

[66] L. Kahney. Your iphone could be 'unbreakable,' if it were just 1 mm thicker. https://www.cultofmac.com/624356/your-iphone-could-be-unbreakable-if-it-were-just-1mm-thicker/, 2019.

[67] N. Karapanos, C. Marforio, C. Soriente, and S. Capkun. Sound-proof: usable two-factor authentication based on ambient sound. In *USENIX Security Symposium (USENIX Security)*, 2015.

[68] B. Kibret, A. K. Teshome, and D. T. Lai. Human body as antenna and its effect on human body communications. *Progress In Electromagnetics Research*, 148:193–207, 2014.

[69] B. Kibret, A. K. Teshome, and D. T. Lai. Characterizing the human body as a monopole antenna. *IEEE Transactions on Antennas and Propagation*, 63(10):4384–4392, 2015.

[70] K. Kyoung and R. Hattori. Electromagnetic field analysis of capacitive touch panels. *Journal of Information Display*, 15(3):145–155, 2014.

[71] R. Layton and K. Dixon. Stopping sight distance. `https://cce.oregonstate.edu/sites/cce.oregonstate.edu/files/12-2-stopping-sight-distance.pdf`, 2012.

[72] C.-J. Lee, J. K. Park, C. Piao, H.-E. Seo, J. Choi, and J.-H. Chun. Mutual capacitive sensing touch screen controller for ultrathin display with extended signal passband using negative capacitance. *Sensors*, 18(11):3637, 2018.

[73] J. Li, Z. Nie, Y. Liu, L. Wang, and Y. Hao. Evaluation of propagation characteristics using the human body as an antenna. *Sensors*, 17(12):2878–2893, 2017.

[74] M. Li, Y. Meng, J. Liu, H. Zhu, X. Liang, Y. Liu, and N. Ruan. When csi meets public wifi: Inferring your mobile phone password via wifi signals. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1068–1079, 2016.

[75] X. Li, F. Yan, F. Zuo, Q. Zeng, and L. Luo. Touch well before use: Intuitive and secure authentication for iot devices. In *Proceedings of the Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2019.

[76] K. Ling, Y. Liu, K. Sun, W. Wang, L. Xie, and Q. Gu. Spidermon: Towards using cell towers as illuminating sources for keystroke monitoring.

[77] H. Liu, Y. Wang, J. Yang, and Y. Chen. Fast and practical secret key extraction by exploiting channel response. In *Proceedings IEEE International Conference on Computer Communications (INFOCOM)*, 2013.

[78] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser. Snooping keystrokes with mm-level audio ranging on a single phone. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 142–154, 2015.

[79] Q. Liu, W. Wang, T. de Campos, P. J. Jackson, and A. Hilton. Multiple speaker tracking in spatial audio via phd filtering and depth-audio fusion. *IEEE Transactions on Multimedia*, 20(7):1767–1780, 2017.

[80] W. Liu, M. Kulin, T. Kazaz, A. Shahid, I. Moerman, and E. De Poorter. Wireless technology recognition based on rssi distribution at sub-nyquist sampling rate for constrained devices. *Sensors*, 17(9):2081–2104, 2017.

[81] X. Liu, J. Cao, S. Tang, and J. Wen. Wi-sleep: Contactless sleep monitoring via wifi signals. In *Proceedings of the IEEE Real-Time Systems Symposium*, pages 346–355, 2014.

[82] X. Liu, Z. Zhou, W. Diao, Z. Li, and K. Zhang. When good becomes evil: Keystroke inference with smartwatch. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1273–1285, 2015.

[83] Y. Liu, S. C. Draper, and A. M. Sayeed. Exploiting channel diversity in secret key generation from multipath fading randomness. *IEEE Transactions on information forensics and security*, 7(5):1484–1497, 2012.

[84] H. W. Löllmann, C. Evers, A. Schmidt, H. Mellmann, H. Barfuss, P. A. Naylor, and W. Kellermann. The locata challenge data corpus for acoustic source localization and tracking. In *Proceedings of the Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 410–414, 2018.

[85] N. Long, K. Wang, R. Cheng, K. Yang, and J. Bai. Fusion of millimeter wave radar and rgb-depth sensors for assisted navigation of the visually impaired. In *Proceedings of the Millimetre Wave and Terahertz Sensors and Technology XI*, pages 1–8, 2018.

[86] L. Lu, J. Yu, Y. Chen, Y. Zhu, X. Xu, G. Xue, and M. Li. Keylisterber: Inferring keystrokes on qwerty keyboard of touch screen through acoustic signals. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 775–783. IEEE, 2019.

[87] Y. Lu, F. Wu, S. Tang, L. Kong, and G. Chen. Free: A fast and robust key extraction mechanism via inaudible acoustic signal. In *Proceedings of the International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2019.

[88] A. Maiti, O. Armbruster, M. Jadliwala, and J. He. Smartwatch-based keystroke inference attacks and context-aware protection mechanisms. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, pages 795–806, 2016.

[89] W. Mao, M. Wang, and L. Qiu. Aim: acoustic imaging on a mobile. In *Proceedings of the International Conference on Mobile Systems, Applications, and Services (Mobisys)*, pages 468–481, 2018.

[90] M. Maróti, B. Kusy, G. Simon, and Á. Lédeczi. The flooding time synchronization protocol. In *Proceedings of the International Conference on Embedded Network Sensor Systems (SenSys)*, 2004.

[91] S. Maruyama, S. Wakabayashi, and T. Mori. Tap'n ghost: A compilation of novel attack techniques against smartphone touchscreens. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 620–637. IEEE, 2019.

[92] S. Mathur, R. Miller, A. Varshavsky, W. Trappe, and N. Mandayam. Proximate: proximity-based secure pairing using ambient wireless signals. In *Proceedings of the International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2011.

[93] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2008.

[94] MathWorks. Matlab audio toolbox. https://www.mathworks.com/products/audio.html, 2020.

[95] M. Miettinen, N. Asokan, T. D. Nguyen, A.-R. Sadeghi, and M. Sobhani. Context-based zero-interaction pairing and key evolution for advanced personal devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2014.

[96] E. Miluzzo, A. Varshavsky, S. Balakrishnan, and R. R. Choudhury. Tapprints: your finger taps have fingerprints. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 323–336, 2012.

[97] F. Moiz, W. D. Leon-Salas, Y. Lee, and R. Derakhshani. A low bit rate wearable motion sensing platform for gesture classification. In *Proceedings of the IEEE International Symposium on Computer-Based Medical Systems (CBMS)*, 2017.

[98] Monsoon. Power monitor. `https://www.msoon.com/online-store`, 2019.

[99] R. Müller and P. Büttner. A critical discussion of intraclass correlation coefficients. *Statistics in medicine*, 13(23-24):2465–2476, 1994.

[100] R. Nandakumar, S. Gollakota, and N. Watson. Contactless sleep apnea detection on smartphones. In *Proceedings of the International Conference on Mobile Systems, Applications, and Services*, pages 45–57, 2015.

[101] R. Nandakumar, V. Iyer, D. Tan, and S. Gollakota. Fingerio: Using active sonar for fine-grained finger tracking. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1515–1525, 2016.

[102] Nirsoft. Who is connected sniffer, 2020.

[103] A. C. of the Blind. White cane law. `https://www.acb.org/whitecane`, 2019.

[104] W. H. Organization. Blindness and vision impairment report. `https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment`, 2018.

[105] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang. Accessory: password inference using accelerometers on smartphones. In *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*, pages 1–6, 2012.

[106] W. K. Panofsky and M. Phillips. *Classical electricity and magnetism.* Courier Corporation, 2005.

[107] Polar. Pair heart rate sensor with your watch, assessed at January, 2020.

[108] M. Pruvost, W. J. Smit, C. Monteux, P. Poulin, and A. Colin. Polymeric foams for flexible and highly sensitive low-pressure capacitive sensors. *npj Flexible Electronics*, 3(1):1–6, 2019.

[109] K. Qian, C. Wu, F. Xiao, Y. Zheng, Y. Zhang, Z. Yang, and Y. Liu. Acousticcardiogram: Monitoring heartbeats using acoustic signals on smart devices. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, pages 1574–1582, 2018.

[110] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[111] R. Raguram, A. M. White, D. Goswami, F. Monrose, and J.-M. Frahm. ispy: automatic reconstruction of typed input from compromising reflections. In *Proceedings of the 18th ACM conference on Computer and communications security*, pages 527–536, 2011.

[112] S. Rajarajan, K. Maheswari, R. Hemapriya, and S. Sriharilakshmi. Shoulder surfing resistant virtual keyboard for internet banking. *World Applied Sciences Journal*, 31(7):1297–1304, 2014.

[113] J. P. Reilly. *Applied bioelectricity: from electrical stimulation to electropathology*. Springer Science & Business Media, 2012.

[114] A. M. Research. Wearable technology market by device, by product type and geography - global opportunity analysis and industry forecast, 2014-2022. https://www.alliedmarketresearch.com/wearable-technology-market, assessed at January, 2020.

[115] A. Rodríguez Valiente, A. Trinidad, J. García Berrocal, C. Górriz, and R. Ramírez Camacho. Extended high-frequency (9–20 khz) audiometry reference thresholds in 645 healthy subjects. *International Journal of Audiology*, 53(8):531–545, 2014.

[116] M. Roeschlin, I. Martinovic, and K. B. Rasmussen. Device pairing at the touch of an electrode. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2018.

[117] M. Rostami, A. Juels, and F. Koushanfar. Heart-to-heart (h2h): authentication for implanted medical devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2013.

[118] Z. Rozsa and T. Sziranyi. Obstacle prediction for automated guided vehicles based on point clouds measured by a tilted lidar sensor. *IEEE Transactions on Intelligent Transportation Systems*, 19(8):2708–2720, 2018.

[119] W. Ruan, Q. Z. Sheng, L. Yang, T. Gu, P. Xu, and L. Shangguan. Audiogest: enabling fine-grained hand gesture detection by decoding echo signal. In *Proceedings of the International Joint Conference on Pervasive and Ubiquitous Computing*, pages 474–485, 2016.

[120] D. Schneider, A. Otte, T. Gesslein, P. Gagel, B. Kuth, M. S. Damlakhi, O. Dietz, E. Ofek, M. Pahud, P. O. Kristensson, et al. Reconviguration: Reconfiguring physical keyboards in virtual reality. *IEEE transactions on visualization and computer graphics*, 25(11):3190–3201, 2019.

[121] D. Schürmann and S. Sigg. Secure communication based on ambient audio. *IEEE Transactions on Mobile Computing*, 12(2):358–370, 2011.

[122] D. Shukla, R. Kumar, A. Serwadda, and V. V. Phoha. Beware, your hands reveal your secrets! In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 904–917, 2014.

[123] I. Shumailov, L. Simon, J. Yan, and R. Anderson. Hearing your touch: A new acoustic side channel on smartphones. *arXiv preprint arXiv:1903.11137*, 2019.

[124] B. Smith and D. Sandwell. Accuracy and resolution of shuttle radar topography mission data. *Geophysical Research Letters*, 30(9), 2003.

[125] Statista. Unit shipments of mobile phones worldwide from 2009 to 2023.

[126] J. Sun, X. Jin, Y. Chen, J. Zhang, Y. Zhang, and R. Zhang. Visible: Video-assisted keystroke inference from tablet backside motion. In *NDSS*, 2016.

[127] K. Sun, T. Zhao, W. Wang, and L. Xie. Vskin: Sensing touch gestures on surfaces of mobile devices using acoustic signals. In *Proceedings of the International Conference on Mobile Computing and Networking*, pages 591–605, 2018.

[128] Y. Suzuki and H. Takeshima. Equal-loudness-level contours for pure tones. *The Journal of the Acoustical Society of America*, 116(2):918–933, 2004.

[129] D. Tse and P. Viswanath. *Fundamentals of wireless communication*. Cambridge university press, 2005.

[130] J. Velasco, M. J. Taghizadeh, A. Asaei, H. Bourlard, C. J. Martín-Arguedas, J. Macias-Guarasa, and D. Pizarro. Novel gcc-phat model in diffuse sound field for microphone array pairwise distance based calibration. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2669–2673, 2015.

[131] R. H. Venkatnarayan, G. Page, and M. Shahzad. Multi-user gesture recognition using wifi. In *Proceedings of the ACM International Conference on Mobile Systems, Applications, and Services*, pages 401–413, 2018.

[132] D. Version. Conversion of sound units. `http://www.sengpielaudio.com/calculator-soundlevel.htm`, 2020.

[133] D. Version. Damping of air at high frequencies. `http://www.sengpielaudio.com/calculator-air.htm`, 2020.

[134] T. Vu, A. Baid, S. Gao, M. Gruteser, R. Howard, J. Lindqvist, P. Spasojevic, and J. Walling. Distinguishing users with capacitive touch communication. In *Proceedings of the international conference on Mobile computing and networking (Mobicom)*, 2012.

[135] T. Vu, A. Baid, S. Gao, M. Gruteser, R. Howard, J. Lindqvist, P. Spasojevic, and J. Walling. Capacitive touch communication: A technique to input data through devices' touch screen. *IEEE Transactions on Mobile Computing*, 13(1):4–19, 2013.

[136] M. Vuagnoux and S. Pasini. Compromising electromagnetic emanations of wired and wireless keyboards. In *USENIX security symposium*, pages 1–16, 2009.

[137] J. Wang, K. Zhao, X. Zhang, and C. Peng. 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*, pages 14–27, 2014.

[138] L. Wang and B. Yu. Analysis and measurement on the electromagnetic compromising emanations of computer keyboards. In *2011 Seventh International Conference on Computational Intelligence and Security*, pages 640–643. IEEE, 2011.

[139] Q. Wang, K. Xu, and K. Ren. Cooperative secret key generation from phase estimation in narrowband fading channels. *IEEE Journal on selected areas in communications*, 30(9):1666–1674, 2012.

[140] T. Wang, G. Cardone, A. Corradi, L. Torresani, and A. T. Campbell. Walksafe: a pedestrian safety app for mobile phone users who walk and talk while crossing roads. In *Proceedings of the ACM Workshop on Mobile Computing Systems & Applications*, pages 5–10, 2012.

[141] T. Wang, D. Zhang, Y. Zheng, T. Gu, X. Zhou, and B. Dorizzi. C-fmcw based contactless respiration detection using acoustic signal. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(4):170, 2018.

[142] W. Wang, A. X. Liu, and K. Sun. Device-free gesture tracking using acoustic signals. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking*, pages 82–94, 2016.

[143] Y. Wang, W. Cai, T. Gu, and W. Shao. Your eyes reveal your secrets: an eye movement based password inference on smartphone. *IEEE transactions on mobile computing*, 2019.

[144] Y. Wang, W. Cai, T. Gu, W. Shao, I. Khalil, and X. Xu. Gazerevealer: Inferring password using smartphone front camera. In *Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pages 254–263, 2018.

[145] S. B. Wicker and V. K. Bhargava. *Reed-Solomon codes and their applications*. John Wiley & Sons, 1999.

[146] A. Wolke. Calculating rf power from iq samples. https://www.tek.com/blog/calculating-rf-power-iq-samples, assessed at January, 2020.

[147] L. Xi, L. Guosui, and J. Ni. Autofocusing of isar images based on entropy minimization. *IEEE Transactions on Aerospace and Electronic Systems*, 35(4):1240–1252, 1999.

[148] W. Xi, X.-Y. Li, C. Qian, J. Han, S. Tang, J. Zhao, and K. Zhao. Keep: Fast secret key extraction protocol for d2d communication. In *Proceedings of the IEEE International Symposium of Quality of Service*, 2014.

[149] W. Xi, C. Qian, J. Han, K. Zhao, S. Zhong, X.-Y. Li, and J. Zhao. Instant and robust authentication and key agreement among mobile devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2016.

[150] Y. Xiao, J. Lin, O. Boric-Lubecke, and M. Lubecke. Frequency-tuning technique for remote detection of heartbeat and respiration using low-power double-sideband transmission in the ka-band. *IEEE Transactions on Microwave Theory and Techniques*, 54(5):2023–2032, 2006.

[151] J. Xu, X.-G. Xia, S.-B. Peng, J. Yu, Y.-N. Peng, and L.-C. Qian. Radar maneuvering target motion estimation based on generalized radon-fourier transform. *IEEE Transactions on Signal Processing*, 60(12):6190–6201, 2012.

[152] W. Xu, C. Javali, G. Revadigar, C. Luo, N. Bergmann, and W. Hu. Gait-key: A gait-based shared secret key generation protocol for wearable devices. *ACM Transactions on Sensor Networks*, 13(1):6–33, 2017.

[153] X. Xu, J. Yu, Y. Chen, Q. Hua, Y. Zhu, Y.-C. Chen, and M. Li. Touchpass: towards behavior-irrelevant on-touch user authentication on smartphones leveraging vibrations. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pages 1–13, 2020.

[154] Y. Xu, J. Heinly, A. M. White, F. Monrose, and J.-M. Frahm. Seeing double: Reconstructing obscured typed input from repeated compromising reflections. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 1063–1074, 2013.

[155] Z. Xu, K. Bai, and S. Zhu. Taplogger: Inferring user inputs on smartphone touchscreens using onboard motion sensors. In *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks*, pages 113–124, 2012.

[156] Z. Yan, Y. Li, R. Tan, and J. Huang. Application-layer clock synchronization for wearables using skin electric potentials induced by powerline radiation. In *Proceedings of the Conference on Embedded Network Sensor Systems (SenSys)*, 2017.

[157] Z. Yan, Q. Song, R. Tan, Y. Li, and A. W. K. Kong. Towards touch-to-access device authentication using induced body electric potentials. In *Proceedings of the Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2019.

[158] C. Yang and A. P. Sample. Em-id: Tag-less identification of electrical devices via electromagnetic emissions. In *Proceedings of the IEEE International Conference on RFID*, 2016.

[159] C. J. Yang and A. P. Sample. Em-comm: Touch-based communication via modulated electromagnetic emissions. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (Ubicomp)*, 2017.

[160] L. Yang, W. Wang, and Q. Zhang. Secret from muscle: Enabling secure pairing with electromyography. In *Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys)*, 2016.

[161] Y. Yang. A signal theoretic approach for envelope analysis of real-valued signals. *IEEE Access*, 5:5623–5630, 2017.

[162] Y. Yang, J. Cao, X. Liu, and X. Liu. Multi-breath: Separate respiration monitoring for multiple persons with uwb radar. In *Proceedings of the IEEE Computer Software and Applications Conference*, pages 840–849, 2019.

[163] Q. Yue, Z. Ling, X. Fu, B. Liu, K. Ren, and W. Zhao. Blind recognition of touched keys on mobile devices. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2014.

[164] K. Zeng, D. Wu, A. Chan, and P. Mohapatra. Exploiting multiple-antenna diversity for shared secret key generation in wireless networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, 2010.

[165] J. Zhang, X. Zheng, Z. Tang, T. Xing, X. Chen, D. Fang, R. Li, X. Gong, and F. Chen. Privacy leakage in mobile sensing: Your unlock passwords can be leaked through wireless hotspot functionality. *Mobile Information Systems*, 2016, 2016.

[166] W. Zhang and B. D. Rao. A two microphone-based approach for source localization of multiple speech sources. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(8):1913–1928, 2010.

[167] T. Zhu, Q. Ma, S. Zhang, and Y. Liu. Context-free attacks using keyboard acoustic emanations. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 453–464, 2014.

[168] L. Zhuang, F. Zhou, and J. D. Tygar. Keyboard acoustic emanations revisited. *Transactions on Information and System Security*, 13(1):1–26, 2009.

[169] A. Zhukova. How to see who is connected to my wifi. https://helpdeskgeek.com/how-to/determine-computers-connected-to-wireless-network/, 2020.

[170] T. G. Zimmerman. Personal area networks: near-field intrabody communication. *IBM systems Journal*, 35(3.4):609–617, 1996.