Modeling and Mitigating Side Channels in Optical and Embedded Sensing Systems

by

Yan Long

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in the University of Michigan
2024

Doctoral Committee:

Professor Kevin Fu, Northeastern University, Co-Chair
Professor Mingyan Liu, Co-Chair
Assistant Professor Jean-Baptiste Jeannin
Associate Professor Alanson Sample
Associate Professor Pei Zhang

Yan Long
yanlong@umich.edu
ORCID iD: 0000-0002-3429-7127

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

ABSTRACT

This thesis investigates how to model and mitigate the security and privacy impact of undefined information channels in the analog-digital interfaces of sensing systems. Sensors bridge the physical and digital worlds and have become a fundamental building block of modern computer systems. The physical and analog nature of sensor hardware, however, creates an inherent gap in the abstraction of sensors when sensor hardware is interfaced with computer software. The lack of comprehensive and proper abstraction causes side channels in physical-digital information transformation, resulting in information leakage and manipulation problems that could compromise the data security and user privacy of emerging cyber-physical technologies.

By analyzing the sensing model and several representative examples of camera sensing and other embedded sensors, my thesis first investigates how three key factors of modern sensor design contribute to sensor data leakage problems. These factors include (1) the increasing resolution and sensitivity of sensors, (2) the increasing structural complexity of sensors, and (3) the more standardized but unprotected sensor data distributions. The first factor is demonstrated by a case study of webcams leaking user screen contents during video conferencing when the screen contents are reflected by users' eyeglasses, and generalized to another threat model of smartphone zero-permission accelerometers and gyroscopes leaking touchtone audio. The second factor is demonstrated by exploiting the rolling shutter and movable lens structures of smartphone cameras to extract ambient audio from a stream of photos. The third factor is demonstrated by eavesdropping on the electromagnetic leakage of camera data transmission interfaces to reconstruct confidential camera videos in real time. Besides information leakage, this thesis explains how such side channels in sensors also allow the injection of false information into sensing systems. Specifically, it shows how intentional electromagnetic interference can interact with analog sensing circuits to manipulate the temperature readings of vaccine temperature monitors and induce phantom keystroke inputs on various types of keyboards. Finally, the thesis generalizes the concept of sensor side channels and proposes that when properly controlled, such side channels could be utilized by system defenders to strengthen existing systems, such as synthesizing virtual sensors from existing sensor hardware to perform multimodal sensing and authentication.

CHAPTER 1

Introduction

My thesis investigates how to protect sensors from information eavesdropping and output manipulation by adversaries exploiting physics-based side channels of sensor semiconductors. Sensors are increasingly omnipresent in public and private spaces. Cyber-physical Systems (CPS) depend on sensors to make life-critical decisions ranging from steering an autonomous vehicle to defibrillating a patient's heart. It's extremely important to ensure confidential and trustworthy data from sensors to avoid leaking privacy and even life-critical information to unauthorized parties or making misinformed tragic decisions. However, a substantial gap persists between what system and application engineers expect from sensor semiconductors and what sensors actually provide in terms of trustworthiness for protecting the confidentiality and integrity of sensitive information. This gap leads to *sensor side channels* and an emerging branch of computer security vulnerabilities that threaten data security and user privacy.

The core research problem of this thesis is to *characterize and model the side channels in sensing systems to enhance the discovery, analysis, and mitigation of sensor side channel vulnerabilities*. My thesis uses an inductive approach to build the analytical framework for sensor side channels given that a thorough understanding of the representative instances of such problems is the key first step to achieving generalization and formalism. The thesis is heavily based on experiments, case studies, and quantitative measurements. It does not seek to directly investigate and address sensor side channel problems in every type of sensing device since there are hundreds of different types of sensors and even significantly more diverse implementations. Instead, my thesis focuses on embedded cameras, i.e., optical sensors widely embedded in consumer electronics including smartphones, laptops, and IoT devices, as an enlightening example. My thesis aims to generalize the analysis, measurement, and mitigation methodologies that can be applied to other types of sensors. Besides camera sensing, my thesis also provides case studies on inertial measurement units (IMUs, a.k.a. motion sensors), temperature sensors, and keyboard sensing to demonstrate how to generalize the problems and analysis.

Based on the background of existing sensor security problems, I first summarize a sensing model that demonstrates how today's sensing system designers mentally model the sensing process (Chapter 2). The model points out several key requirements that need to be satisfied to avoid security and privacy problems in this process. However, my thesis hypothesizes that these requirements are increasingly more challenging to fulfill in emerging sensing systems due to several key trends observed in sensor hardware, namely (1) the increasing resolution and sensitivity, (2) the increasing structural complexity, and (3) the increasing surface of unprotected data transmissions in sensing systems.

My thesis then analyzes how these three key trends create information leakage problems that provide experimental evidence for the thesis's hypotheses. In Chapter 3, I explain how the increasing resolution and sensitivity of webcams result in the leakage of user screen information in video conferences, and how smartphone motion sensors leak touch-tone audio due to similar problems. In Chapter 4, I explain how the increasing complexity of smartphone cameras' rolling shutters and movable lenses enable adversaries to extract room audio from camera photo streams. In Chapter 5, I explain how the unprotected data transmission interfaces in embedded cameras allow adversaries to eavesdrop on confidential camera videos in real-time by analyzing the electromagnetic leakage from the interfaces.

In the case studies of information leakage problems, we model sensor side channels as functions that map physical signals containing unintended secret information to digital sensor readings accessed by adversaries. Three key tasks of our experiments include (1) measuring how much secrete information adversaries can recover from the sensor readings they have access to, (2) identifying and characterizing the significant factors (variables) of these functions that affect the level of adversarial information recovery, and (3) proposing software and hardware defenses that improve the system design choices of the identified factors. We collect sensor data in both controlled lab environments and uncontrolled environments. The former is used to control, study, and develop hypotheses for the function factors while the latter is mainly used for testing our hypotheses and evaluation. In some cases, we also conduct user studies with human subjects to understand how human factors interact with these sensor side channels. To evaluate the capacity of sensor side channels, i.e., how much information can adversaries extract from them, we utilize both objective signal metrics such as signal-to-noise ratios and similarity scores, as well as machine learning-based information label classification tests.

While side channels are mostly known to cause information leakage problems, my thesis shows how the framework of Chapter 2 can also be used to analyze both sensor data integrity problems caused by physical signal injection attacks, and the potential opportunities for system designers to actively utilize such side channel information for good purposes such

as multimodal authentication. To substantiate the analysis of the first point, Chapter 6 provides two case studies on how intentional electromagnetic interference can inject false temperature readings into vaccine temperature monitors and inject phantom keystrokes into various types of keyboards by exploiting the side channels in the analog sensing circuits. For the second point, Chapter 7 provides a more dedicated analytical framework for sensor side channels in authenticating settings and a case study on synthesizing virtual motion sensors to capture smartphone users' hand tremor signals, which can then be used to strengthen existing face authentication systems against spoofing attacks.

Contribution & Published Results. To summarize, my thesis presents original research that advances the research of sensing security and privacy by providing the following major contributions.

- **New Vulnerability Characterization.** We discover and investigate a series of novel sensor side channel problems in cameras and other types of embedded sensors that pose threats of information leakage and false information injection. Particularly, the discovery of these camera-based channels reveals an orthogonal space of potential cybersecurity threats that can have real-world impact on a wide range of users who interact with camera-enabled electronic devices on a daily basis.

- **Theoretical Analysis and Modeling.** We provide a side channel-based analytical framework for modeling sensing security and privacy problems. The framework complements existing side channel literature by modeling the hardware-software interfaces of sensors. Such analysis and modeling enable more rapid and systematic discovery and evaluation of sensor side channels.

- **Mitigation and Utilization Methodology.** We provide a mitigation methodology that aims to address the modeled root causes of these sensor side channel problems. Such a methodology enables the identification of sensor design improvement that may be integrated into future sensing systems for preventive countermeasures. In addition, we explain how once mitigated and controlled, sensor side channels can be utilized by defenders to synthesize virtual sensors for multimodal authentication.

CHAPTER 2

Background & Problem Descriptions

2.1 Related Works

2.1.1 Sensor-based Eavesdropping Attacks

Eavesdropping attacks violate the confidentiality aspect of system security policies using side channel information. A good example of sensor side channel eavesdropping is the work of Gyrophone by Michalevsky et al [171]. The authors discover that adversaries can eavesdrop on sound using gyroscopes in smartphones. Since smartphone operating systems do not require user permission for applications to access gyroscope data, in contrast to microphone data access that requires permission, this technique allows adversaries to bypass the smartphone access control system. Later, several works also showed that the same principle applies to smartphone accelerometers and that advanced machine learning techniques can be used to achieve high accuracies of audio information inference [51, 123, 63, 45]. Another category of representative works investigated how to eavesdrop on smartphone PIN inputs using side channel information in sensors data, including using cameras and microphones [208], gyroscopes [66], light sensors [214], etc. These previous works closely align with my thesis, which investigates three new categories of camera-based sensor side channels. While most of these previous works study sensor side channels that allow software-space adversaries to eavesdrop on physical information, my thesis also investigates electromagnetic side channels of sensors that allow external, physical adversaries to infer confidential sensor data. Furthermore, my thesis builds a side-channel analysis framework for sensor side channels that aims to systematize the exploration and analysis of potential sensor side channels.

2.1.2 Transduction Attacks

Transduction attacks inject analog signals into sensors where victim sensor circuitry transduces an attacker's malicious physical signals to untrustworthy sensor measurements [102,

247]. Such malicious physical signals can often be in different physical modalities (e.g., acoustic vs. optical) or frequency ranges (e.g., audible vs. ultrasound) than what the sensors are designed to sense. For example, Light Commands [217] uses lasers to inject false speech signals into microphones. Works such as Walnut [209, 228] use acoustic injections to influence and control the output of MEMS gyroscopes and accelerometers. Ghost Talk [99] uses electromagnetic waves to inject audio signals into microphones. An SoK and a survey [247, 107] provide a comprehensive review of these attacks. Transduction attacks and side channel-based eavesdropping attacks are often two sides of the same coin. Although my thesis mainly employs the lens of side channel and eavesdropping attacks, the sensor side channels we discover and analyze are essentially low-level mechanisms that can also be exploited by transduction attacks, as further demonstrated in Chapter 6.

2.2 Sensing Security Problems

We first provide a high-level model of the sensing process to provide key definitions and show the potential security problems that can happen based on the model. The main goal of such modeling is to explain existing sensing security and privacy problems as well as predict future problems using the lens of side-channel analysis.

Sensor. A sensor is a device that can convert analog physical signals to digital software samples. The major components of a typical sensor include a transducer, a signal conditioning chain, an analog-to-digital converter (ADC), and data transmission interfaces. Sensors are mostly used as peripherals of computer systems for collecting physical information. This thesis calls a computer system with sensors to perform sensing functionalities a sensing system.

Transducer Unit. Mathematically, a transducer unit maps a set of random variables to a single random variable that the downstream ADC will process. The transducer of a sensor is the collection of all transducer units in this physical sensor.

Intended and Unintended Input. An input of a sensor/sensing system is a physical quantity modeled as a random variable. An intended input is a variable that the system designer intends to sample while unintentional inputs are the other random variables in the domain of the transducer units. Intended inputs provide a minimal set of information for the sensing system to achieve the designed functionality.

Sensing Process. A sensor has a set of N_{trans} transducer units, each denoted as a function $f^i_{trans}, i \in \{1, 2, ..., N_{trans}\}$. We denote a set of intended inputs that the system designer wants to measure with this sensor as s_{int}. Similarly, denote the set of all unintentional inputs

as s_{side}. The i-th transducer unit has:

$$d_i = f_{ADC}(f^i_{trans}(s_{int}, s_{side}) + n_i, f_s), \tag{2.1}$$

where f_s is the sampling rate of the sensor, n_i denotes the inherent noise of the transducer unit, and d_i denotes the digital data accessed in the software space. We further denote all digital data produced by this sensor as d_{sensor}

Secret. A secret is a piece of information, also modeled as a random variable, that once partially accessed by an untrusted party, results in a security problem. In this model, a secret is either an intended or an unintended input for a sensor. *A confidentiality problem* happens if a secret can be (at least partially) read by an untrusted party; an *integrity problem* happens if a secret can be (at least partially) written by an untrusted party. We denote a set of secrets as s_{sec}

Key Requirements. In the sensing model above, a trusted party must be able to access d_{sensor} in the software space to fulfill the system's functionality. More importantly, several key requirements need to be satisfied for the system to be secure:

- **KR1.** Confidentiality: When d_{sensor} are made accessible to untrusted parties in the software space, neither s_{int} nor s_{side} should contain secrets.
- **KR2.** Confidentiality: When s_{int} or s_{side} do contain secrets, untrusted parties should not have full or partial read access to d_{sensor} by all means.
- **KR3.** Integrity: When s_{int} contains secrets, untrusted parties should not have full or partial write access to s_{side} to change d_{sensor}.

2.2.1 Hypotheses

Analyzing the sensing model above, my thesis investigates several hypotheses on how observed key trends and factors make it increasingly more challenging to fulfill the key requirements in emerging sensing systems or provide new opportunities for system defenders to strengthen existing systems' security.

H1. Regarding **KR1**, we hypothesize that while the set of secrets s_{sec} could be constant, the sets s_{int} and s_{side} keep growing in size due to constant improvement of sensor hardware such as the increasing resolution and sensitivity (Chapter 3) as well as the increasing structural complexity of sensors (Chapter 4). This increases the coverage of $s_{sec} \cap (s_{int} \cup s_{side})$, creating side channels that communicate secrets to untrusted parties and compromise data confidentiality.

H2. Regarding **KR2**. We hypothesize that the challenge of data confidentiality increases as d_{sensor} are distributed and processed in multiple hardware components in modern computer systems in an unencrypted manner. The transmission of d_{sensor} between these components increases the attack surface, potentially generating side-channel leakage that allows untrusted parties to eavesdrop on d_{sensor} and further infer s_{sec} from outside of the system (Chapter 5).

H3. Regarding **KR3**. We hypothesize that the challenge of sensor data integrity also keeps increasing. This is again mostly caused by the expansion of s_{side} because it gets more difficult to prevent untrusted parties from interacting with s_{side} to change d_{sensor}, essentially injecting false information into the system through sensor side channels (Chapter 6).

H4. Finally, we hypothesize that the side channels in sensing systems can be regarded as neutral information channels under certain conditions, and may be utilized by system designers and defenders to acquire useful information for strengthening the security of existing systems (Chapter 7).

It is worth pointing out that the key requirements **KR1-KR3** can themselves be treated as hypotheses of how the security and privacy and sensing system could be compromised. These hypotheses have been implicitly verified by prior related works (Section 2.1). The unique contributions of this thesis lie in the new hypotheses **H1-H4** that provide explicit and systematic characterization and modeling of how side channels could manifest in existing and future sensing systems due to sensor hardware evolution.

CHAPTER 3

Information Leakage Due to Increasing Sensor Resolution and Sensitivity

3.1 Overview

This section investigates hypothesis **H1** using two examples, namely the sensing process of cameras (Section 3.2–Section 3.8) and inertial measurement units (IMUs) (Section 3.9).

In cameras, N_{trans} takes the form of the number of pixels, i.e., camera resolution; n_i represents various imaging noises inherent to the camera circuit that determine the sensitivity of the camera. This case study [165] shows how the increasingly higher resolution and sensitivity of webcams allow adversaries to start eavesdropping on secret information displayed video conferencing users' computer screens through the eyeglass reflections of these users.

Besides camera sensing, we also provide another less detailed example of IMU sensing [64]. In an IMU, N_{trans} takes the form of the number of individual channels in its gyroscope (yaw, pitch, roll) and accelerometer (x, y, z) while n_i determines how sensitive these sensors are to motion signals. This case study shows how present-day IMUs in smartphones are sensitive enough to capture the tiny vibrations of touchtone audio generated by smartphone speakers and how strategically integrating information from multiple channels recovers more information of the original audio signals.

3.2 Threats of Webcams in Video Conferencing

Online video calls have become ubiquitous as a remote communication method, especially since the recent COVID-19 pandemic that caused almost universal work-from-home policies in major countries [90, 59, 49] and made video conference a norm for companies and schools to accommodate interpersonal communications even after the pandemic [19, 137, 194, 10].

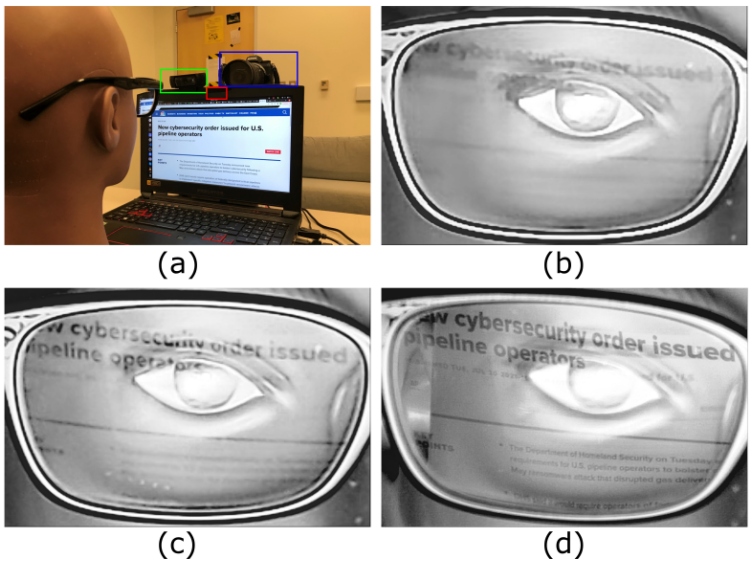


Figure 3.1: The optical emanations of the victim's screen are reflected by eyeglasses, captured by the victim's webcam, and streamed to the adversary, which can then be used to reconstruct the screen contents. The experimental setup (a) with a laptop built-in webcam (b) (red box, 720p), an external Logitech webcam (c) (green box, 1080p), and a Nikon DSLR (d) (blue box, 4K) helps us predict the future fidelity of the attacks as video conferencing technologies evolve.

While video conferencing provides people with the convenience and immersion of visual interactions, it unwittingly reveals sensitive textual information that could be exploited by a malicious party acting as a participant. Each video participant's screen could contain private information. The participant's own webcam could capture this information when it is reflected by the participant's eyeglasses and unwittingly provide the information to the adversary (Figure 3.1). We refer to this attack as a *webcam peeking attack*. Furthermore, adversary capabilities will only continue to increase with improvements to resolution, frame rate, and more. It is thus important to understand the consequences and limits of webcam peeking attacks in present-day and possible future settings.

Previous work shows that similar attacks exploiting optical reflection off nearby objects in controlled setups are feasible, such as observing teapots on a desk with high-end digital single-lens reflex (DSLR) cameras and telescopes at a distance [53, 52]. The challenge and characterization of peeking using the more ubiquitous webcams, however, are qualitatively different due to the lower-quality images of present-day webcams. The lower-quality webcam images are caused by unique types of distortions, namely the shot and ISO noise due to insufficient light reception, and call for new image-enhancing techniques. In addition, new mathematical models and analysis frameworks are needed to understand the threat model of webcam peeking attacks. Finally, this new threat model requires a dedicated evaluation

to clarify the potential threats and mitigations to the average video conference user.

There are many types of media that can leak over optical reflections, including text and graphics. We focus on textual leakage in this work as it's a natural starting point for measurable recognizability and modeling of the fundamental baseline of information leakage, but also provides insights into the leakage of non-textual information such as inferring displayed websites through recognizing graphical contents on the screen. We seek to answer the following three major questions: $\mathcal{Q}1$: What are the primary factors affecting the capability of the webcam peeking adversary? $\mathcal{Q}2$: What are the physical limits of the adversary's capability in the present day and the predictable future, and how can adversaries possibly extend the limits? $\mathcal{Q}3$: What are the corresponding threats of webcam peeking against cyberspace targets and the possible mitigations against the threats?

To answer $\mathcal{Q}1$, we propose a simplified yet reasonably accurate mathematical model for reflection pixel size. The model includes factors such as camera resolution and glass-screen distance and enables the prediction of webcam peeking limits as camera and video technology evolve. By using the complex-wavelet structural similarity index as an objective metric for reflection recognizability, we also provide semi-quantitative analysis for other physical factors including environmental light intensity that affect the signal-to-noise ratio of reflections.

To answer $\mathcal{Q}2$, we analyze the distortions in the webcam images and propose multi-frame super resolution reconstruction for effective image enhancement to extend the limits. We then gather eyeglass reflection data in optimized lab environments and evaluate the recognizability limits of the reflections through both crowdsourcing workers on Amazon Mechanical Turk and optical character recognition models. The evaluation shows over 75% accuracy on recognizing texts that have a physical height of 10 mm with a 720p webcam.

To answer $\mathcal{Q}3$, we focus on web textual targets to build a benchmark that enables meaningful comparisons between present-day and future webcam peeking threats. We first map the limits derived from the model and evaluations to web textual content by surveying previous reports on web text size and manually inspecting fonts in 117 big-font websites. Then, we conduct a user study with 20 participants and play a challenge-response game where one author acts as an adversary to infer HTML contents created by other authors. Results of the user study suggest that present-day 720p webcams can peek texts in the 117 big-font websites and future 4K webcams are predicted to pose threats to header texts from popular websites. We investigated the underlying factors enabling easier webcam peeking in the user study by analyzing the correlation between adversary recognition accuracy and multiple factors. We found, for example, user-specific parameters including browser zoom ratio play a more important role than the glass-screen distance. Besides texts, we also explored the feasibility of recognizing websites through graphical content with 10 participants and observed

accuracies as high as 94% on recognizing a closed-world dataset of Alexa top 100 websites.

Finally, we discuss possible near-term mitigations including adjusting environmental lighting and blurring the glass area in software. We also envision long-term solutions following an individual reflection assessment procedure and a principle of least privilege. In summary, the goal of this work is to provide a theoretical foundation and benchmark for the study of emerging webcam peeking threats with evolving webcam technologies and the development of more secure video conferencing infrastructures.

3.2.1 Related Work

The problem of screen reconstruction is a long-studied challenging problem. In this section, we analyze the past works that served as the foundations for our thinking in the context of video conferencing today and in the predicted future.

Screen Peeking Using Cameras. Screen-peeking with cameras through optical emanation reflections has been explored in previous works. In 2008, Backes et al. [53] showed that adversaries can use off-the-shelf telescopes and DSLR cameras to spy victims' LCD monitor screen contents from up to 30m away by utilizing the reflective objects that can be commonly found next to the monitor screen such as teapots placed on a desk. In 2009, the authors [52] took the attack to the next level by addressing the challenges of motion blur and out-of-focus blur by performing deconvolution on the photos with Point Spread Functions (PSF). Our work differs from these previous works by exploiting the victims' own webcams in video conferences for a remote attack. Such changes call for different imaging enhancing techniques due to the different types of image distortions. In addition, reflective objects on the desks and human eyes cannot be easily utilized due to very large curvatures. We thus exploit the glasses people wear to video conferences as a modern attack vector. [244] proposed a relevant idea of using adversary-controlled webcams to detect changes in webpage links' colors for inferring visited websites. It requires the adversary to take control over the victim's webcam with malicious web modules and exploits coarse-grain color variations, while our work studies more natural attack vectors in video conferencing and investigate the limits of textual reconstruction.

Screen Content Reconstruction With Other Emanations. Besides the direct optical emanations from the screen that we exploit in this work, previous works also explored other channels such as electromagnetic radiation [234, 144, 146] and acoustic emanations [106]. Reconstructing screen contents with such emanations usually requires using additional eavesdropping hardware that is placed close to the victims by the adversary. On the other hand, our work exploits the victim's own webcams, making the attack more accessible.

Remote Eavesdropping Via Audio/Video Calls. Similar to our work, such attacks assume the adversary and victim are both participants of an audio/video conference, and the adversary can eavesdrop on privacy-sensitive information by analyzing the audio/video channels. For example, Voice-over-IP attacks for keystroke inference eavesdrop on the victim's keyboard inputs by utilizing timing and/or spectrum information embedded in the keystroke acoustic emanations [68, 79, 207, 96]. Recently, Sabra et al. [194] proposed works solving the problem of inferring keystrokes by analyzing the dynamic body movements embedded in the videos during a video call. Hilgefort et al. [120] spies victims' nearby objects through virtual backgrounds in video calls by carrying out foreground-background analysis and accumulating background pixels. In contrast, our work explores the related problem of content reconstruction using only the optical reflections from participants' glasses embedded in the videos.

3.3 Threat Model & Background

3.3.1 Threat Model

In this work, we study the webcam peeking attack during online video conferences, where the adversary and the victim are both participants. We assume the device the victim uses to join the video conference consists of a display screen and either a built-in or an external webcam that is mounted on the top of the screen as in most cases, and the victims wear glasses with a reflectance larger than 0, i.e., at least a portion of the light emanated by the monitor screen can be reflected from the glasses to the webcams. We do not enforce constraints on the devices used by the adversary. When the adversary launches the attack, we assume the victim is facing the screen and webcam in the way that the screen emanated light has a single-reflection optical path into the webcam through the eyeglass lens's outer surface. We do not assume the adversary has any control or information on the victim's device.

We assume that the victim's up-link video stream is enabled during the attack, and the adversary can acquire the down-link video stream of the victim. The adversary can achieve that by either directly intercepting the down-link video stream data, or recording the victim's video with the video conferencing platform being used or even third-party screen recording services. Since the webcam peeking attack does not require active interaction between the victim and the adversary, the adversary does not need to attempt a real-time attack but can store the video recording and analyze the videos offline.

3.3.2 Glasses

The most common types of glasses that people wear in a video conferencing setting are prescription glasses [121] and blue-light blocking (BLB) glasses [186, 15]. BLB glasses can either have prescriptions with BLB coating or be non-prescription (flat). The reflectance and curvature of glass lenses are the two most important characteristics in the process of reflecting screen optical emanations.

Reflectance. Reflectance of a lens surface is the ratio between the light energy reflected and the total energy incident on a surface[9]. Reflectance is wavelength-dependent. The higher the reflectance, the more light can be reflected to and captured by a webcam.

Curvature. Curvature of a lens surface represents how much it deviates from a plane. The concepts of curvature, radius, and focal length of an eyeglass lens are used interchangeably on different occasions and are related by: $Curvature = 1/Radius = 2/FocalLength$. Smaller curvature leads to larger-size reflections. Both the outer and inner surfaces of a lens can reflect, but the outer surface often has smaller curvature and thus produce better quality reflections. This paper refers to the eyeglass lens curvature/radius/focal length as that of the outer surface if not specified otherwise.

Lens Power & Focal Length. The power/Diopter of a lens is defined as the reciprocal of the lens' nominal focal length. Different from the f_g used before, this nominal focal length corresponds to the optical effect produced by the combination of the outer and inner surfaces of the lens, and is related to the radius of the outer and inner surfaces by the Lens Maker's Formula [149]:

$$D = \frac{1}{f} = (n-1)(\frac{1}{R_o} - \frac{1}{R_i})$$

where R_o and R_i are the radius of the outer inner surfaces respectively, and n is the refractive index of lens material. When the lens power and materials are set, R_o and R_i can both be adjusted to produce the desired power. However, flatter outer surfaces, as known as base curves, are often used for higher power lenses [20]. This is why we observe a positive correlation between f_g and the lens power in Section 3.5.5.

3.3.3 Digital Camera Imaging System

Digital cameras have sensing units uniformly distributed on the sensor plane, each of which is a Charge-coupled Device (CCD) or Complementary Metal-oxide-semiconductor (CMOS) circuit unit that converts the energy of the photons it receives within a certain period of time, i.e., the exposure time, to an amplitude-modulated electric signal. Each sensing unit then corresponds to a "pixel" in the digital domain. The quality of a digital image to human

perception is mainly determined by its pixel resolution, color representation, the amount of received light that is of our interest, and various imaging noise. The two key imaging parameters that are closely related to webcam peeking attacks are described below.

Exposure Time. Theoretically, the longer the exposure time, the more photons will hit the imaging sensors, and thus there can be potentially more light of interest captured. The images with a longer exposure time will generally be brighter. The downside of having a longer exposure time is the aggravated motion blur when imaging a moving object.

ISO Value. The ISO value represents the amplification factor of the photon-induced electrical signals. In darker conditions, the user can often make the images brighter by increasing the ISO value. The downside of having a higher ISO is the simultaneous amplification of various imaging noises.

Webcam Parameter Estimation. Manufacturers of the laptop built-in webcams often do not share information about the webcam focal length f and imaging sensor physical size W. In this case, further estimation needs to be made. The term $\frac{f}{W}$ is a function of the vertical field-of-view (FoV) of the webcams. Specifically, the FoV angle α can be written as

$$\alpha = 2\tan^{-1}\frac{W}{2f}$$

Considering that typical webcams have a diagonal FoV of in the range $70 - 90°$, we can convert it to a typical vertical FoV of about $40 - 50°$ for a 720p webcam and thus get $\frac{f}{W}$ approximately in the range of $1.1 - 1.4$ [3, 2, 4].

3.3.4 Text Size Representations

It is important to select proper representations of text size in both digital and physical domains since the size of the smallest recognizable texts is the key metric for webcam peeking limits. When texts are digital, i.e., in the victim's software such as browsers and in the webcam image acquired by the adversary, we use point size and pixel size to represent the text size respectively. In the physical domain, i.e., when the texts are displayed on users' screens as physical objects, we use the cap height of the fonts and the physical unit mm to represent the size as it is invariant across different computer displays and enable quantitative analysis of the threats. Cap height is the uniform height of capitalized letters when font style and size are specified and is thus usually used as a convenient representation of physical text size and the base for other font parameters [47, 48].

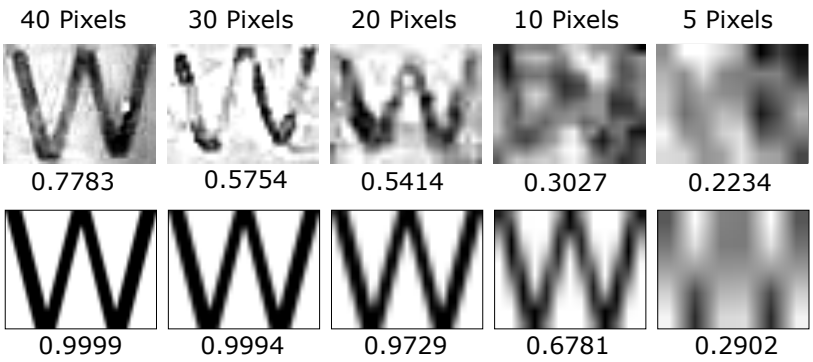


Figure 3.2: (Upper) The captured images of the reflections. Compared with the ideal reflections, additional distortions exist that undermine image recognizability. (Lower) The estimated ideal reflections in the feasibility test corresponding to letters with a height of 80, 60, 40, 20, 10 mm respectively. The images are subjected to aliasing when enlarged.

3.4 Modeling Webcam Peeking Through Glasses

In this section, we start with a feasibility test that reveals the 3 key building blocks of the webcam peeking threat model, namely (1) reflection pixel size, (2) viewing angle, and (3) light signal-to-noise ratio (SNR). For the first two building blocks, we develop a mathematical model that quantifies the related impact factors. For light SNR, we analyze one major factor it encompasses, i.e., image distortions caused by shot noise, and investigate using multi-frame super resolution (MFSR) to enhance reflection images. We will analyze other physical factors that affect light SNR in Section 3.5.4. Experiments are conducted with a Acer laptop with its built-in 720p webcam, the pair of BLB glasses, and a pair of prescription glasses.

Lab Setting Experiment Equipment. The Acer laptop [11] has a screen width of 38 cm and height of 190 mm and a 720p built-in webcam. The OS is Ubuntu 20.04. The OS and browser zoom ratios are default (100%). All the photos and videos are collected with the Cheese [17] webcam application. The photos are in PNG format and the videos are in WEBM format. The Samsung laptop used as the attacker device has OS Windows 10 Pro. The recordings are collected with OBS Studio in MP4 format.

The pair of BLB glasses [16] has lenses with a horizontal and vertical chord length of 5 cm and 4 cm respectively, and a focal length (f_g) of 8 cm. The pair of prescription glasses [16] has lenses with a horizontal and vertical chord length of 6 cm and 5 cm respectively, and a focal length of 50 cm.

Nikon Z7: The photos are in JPEG format (highest quality) and the videos are in MP4 format. We compared these formats with the compression-less (raw) photo and video formats provided by Nikon Z7 but didn't find an obvious difference in the image quality.

3.4.1 Feasibility Test

We conduct a feasibility test of recognizing single alphabet letters with a similar setup as in Figure 3.1. A mannequin wears the BLB glasses with a glass-screen distance of 30 cm. Capital letters with different cap heights (80, 60, 40, 20, 10 mm) are displayed and captured by the webcam. Figure 3.2 (upper) shows the captured reflections. We find that the 5 different cap heights resulted in letters with heights of 40, 30, 20, 10, and 5 pixels in the captured images. As expected, texts represented by fewer pixels are harder to recognize. The reflection pixel size acquired by adversaries is thus one key building block of the characteristics of webcam peeking attack that we need to model. In addition, Figure 3.2 (lower) shows the ideal reflections with these pixel sizes by resampling the template image. Comparing the two, we notice small-size texts are subjected to additional distortions besides the issue of small pixel resolution and noise caused by the face background, resulting in a bad signal-to-noise ratio (SNR) of the textual signals.

To quantify the differences using objective metrics, we embody the notion of reflection quality in the similarity between the reflected texts and the original templates. We compared multiple widely-used image structural and textural similarity indexes including structural similarity Index (SSIM) [242], complex-wavelet SSIM (CWSSIM) [199], feature similarity (FSIM) [257], deep image structure and texture similarity (DISTS) [92] as well as self-built indexes based on scale-invariant feature transform (SIFT) features [155]. Overall, we found CWSSIM which spans the interval $[0, 1]$ with larger numbers representing higher reflection quality produces the best match with human perception results. Figure 3.2 shows the CWSSIM scores under each image.

The differences show that the SNR of reflected light corresponding to the textual targets is another key building block we need to characterize. Finally, we notice that when we rotate the mannequin with an angle exceeding a certain threshold, the webcam images do not contain the displayed letters on the screen anymore. It suggests that the viewing angle is another critical building block of the webcam peeking threat model which acts as an on/off function for successful recognition of screen contents. In the following sections, we seek to characterize these three building blocks.

3.4.2 Reflection Pixel Size

In the attack, the embodiment of textual targets undergoes a two-stage conversion process: digital (victim software) $\rightarrow$ physical (victim screen) $\rightarrow$ digital (adversary camera). In the first stage, texts specified usually in point size in software by the user or web designers are rendered on the victim screen with corresponding physical cap heights. In the second

Table 3.1: Parameters for modeling reflection pixel size

Notation	Parameter
h_o	Physical size (cap height) of the object on the screen
h_s	Physical size of the object's projection on the sensor
s_p	Pixel size of the imaged object
h_i	Physical size of the object's virtual image
P	Physical size of a single imaging sensor pixel
N	Number of pixels the camera has in the dimension
W	Physical size of the imaging sensor in the dimension
f	Camera focal length
d_o	Distance between screen and glasses
d_i	Distance between glasses and virtual image
f_g	Focal length of the glasses convex outer surface

stage, the on-screen texts get reflected by the glass, captured by the camera, digitized, and transferred to the adversary's software as an image with certain pixel sizes. Generally, more usable pixels representing the texts enable adversaries to recognize texts more easily. The key is thus to understand the mechanism of point size → cap height → pixel size conversion.

Point Size → Cap Height. Mapping between digital point size and physical cap height is not unique but dependent on user-specific factors and software. The conversion formula for most web browsers can be summarized as follows:

$$h_o = \frac{4}{3} p_t \cdot \frac{H_{scr}}{N_{os}} \cdot s_{os} \cdot s_b \cdot r_{cap} \tag{3.1}$$

where h_o is the physical cap height of the text, $\frac{4}{3}p_t$ is the number of display hardware pixels most web browsers use to render the text given a point size p_t, H_{scr} is the physical height of the screen, N_{os} is the screen resolution on the height dimension set in the OS which can be equal to or smaller than the maximum supported resolution, s_{os} and s_b are the OS and browser zoom/scaling ratios respectively, and r_{cap} is the ratio between the cap height and the physical point size which is on average $\frac{2}{3}$ [47, 48].

Cap Height → Pixel Size. We would like to remind the readers that we only use pixel size to represent the size of texts living in the images acquired by the adversary[1]. Figure 3.3 shows the model for this conversion process. To simplify the model, we assume the glasses lens, screen contents, and webcam are aligned on the same line with the same angle. The result of this approximation is the loss of projective transformation information, which only causes small inaccuracies for reflection pixel size estimation in most webcam

[1]Since web/software designers sometimes also directly specify text size in pixel size ($\frac{4}{3}P_t$ in Equation 4.5), the two pixel sizes can be easily confused without explanation.

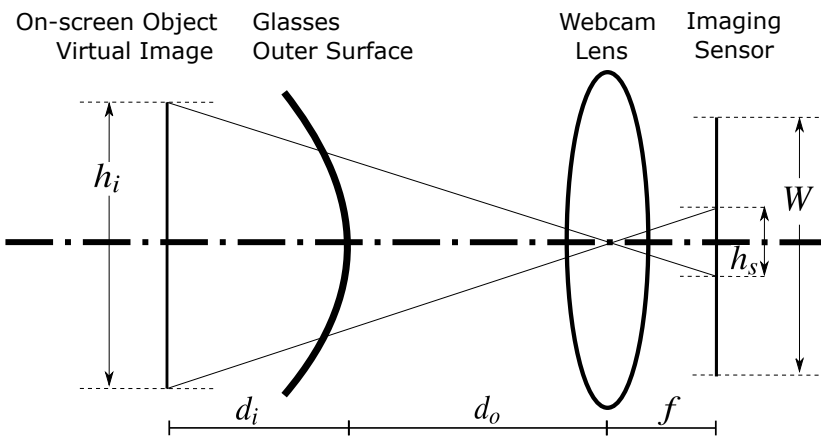


Figure 3.3: The model of reflection pixel size. To better depict the objects, the sizes are not drawn up to scale. The screen overlaps with the webcam lens and is omitted in the figure.

peeking scenarios. Figure 3.3 only depicts one dimension out of the horizontal and vertical dimensions of the optical system but can be used for both dimensions. In this work we focus on the vertical dimension for analysis, i.e., the reflection pixel size we discuss is the height of the captured reflections in pixels. We summarize the parameters of this optical imaging system model in Table 3.1. Through trigonometry, we know

$$\begin{cases} \frac{h_s}{f} = \frac{h_i}{d_o+d_i} \\ h_s = s_p P \\ P = \frac{W}{N} \end{cases} \quad \Rightarrow \quad s_p = \frac{h_i}{d_o+d_i} \cdot \frac{f}{W} \cdot N \tag{3.2}$$

As pointed out in Section 3.3.2, the reflective outer surface of glasses is mostly convex mirrors which shrink the size of the imaginary object h_i and also decrease d_i compared to an ideal flat mirror. To calculate the reflection pixel size s_p in this case, we can use the convex mirror equations [119]

$$\begin{cases} \frac{1}{(-f_g)} = \frac{1}{d_o} + \frac{1}{(-d_i)} \\ \frac{h_i}{h_o} = \frac{d_i}{d_o} \end{cases}$$

where f_g is the focal length of the convex mirror which is half of the radius of the glasses lens and is defined to be positive. Plugging the above equations into Equation 3.2 we can then get

$$s_p = \frac{h_o f_g}{d_o^2 + 2 d_o f_g} \cdot \frac{f}{W} \cdot N, \tag{3.3}$$

The term $\frac{f}{W}$ of typical laptop webcams can be estimated to be in the range $1.1 - 1.4$ (Section 3.3.3). With the Acer laptop and BLB glasses and a glass-screen distance of $d_o = 30$

Table 3.2: The predicted feasible attack ranges for the viewing angle.

Type	**Theoretical**	**Measurement**
Pres: All Page + Horizontal	±15°	±17°
Pres: Center + Horizontal	±5°	±8°
Pres: All Page + Vertical	±9°	±13°
Pres: Center + Vertical	±3°	±5°
BLB: All Page + Horizontal	±20°	±25°
BLB: Center + Horizontal	±10°	±13°
BLB: All Page + Vertical	±14°	±19°
BLB: Center + Vertical	±8°	±10°

cm, the estimated vertical pixel size of a 20 mm-tall object displayed on the screen is in the range of $9.2 - 11.7$ pixels, which matches with the 10 pixels found in the feasibility test and verifies the accuracy of the model despite the approximation we made.

3.4.3 Viewing Angle

To model the effect of viewing angle and the range of angles that enables webcam peeking attack, we model the lens as spherical with a radius $2f_g$.

Similar to the pixel size model, we only use 2D modeling (Figure 3.4) for simplicity which can represent either horizontal or vertical rotations, and we only consider one glass lens since the two lenses are symmetric. The lenses are further modeled as spherical with a radius $2f_g$. We set the origin O to the center of the head which is also treated as the rotation center, and assume the initial orientation without rotation is such that the center of the glass lens arc P_1 aligns with the rotation center and the laptop webcam P_4 on the X-axis. The distance between the glass lens center and the rotation center is s. To calculate the maximum feasible angles, we only need to consider the reflections from either one of the two boundary points of the glass lens since they are symmetric. We label the bottom boundary point as P_2. After a rotation of angle θ, P_1, P_2 are rotated to P_1', P_2' respectively, and the vector $\overrightarrow{P_1'P_2'}$ yields the normal $\vec{n}$ at the reflection point P_2'. P_3 denotes the point source on the screen whose light gets reflected to the camera with an incident angle β. With L_s being the length of the screen on the dimension, the camera should be able to peek reflections from the glass lens if P_3 falls in the range of the screen. C denotes the length of the glass lens chord.

In order to find a mapping from the rotation angle θ to the light-emission point P_3 on the screen, the key is to find the slope of the line $P_2'P_3$ which intersects with the screen. Since $P_1'P_2'$ bisects $P_2'P_4$ and $P_2'P_3$, we denote the slope of these three lines as b_1, b_2, b_3 respectively, and have

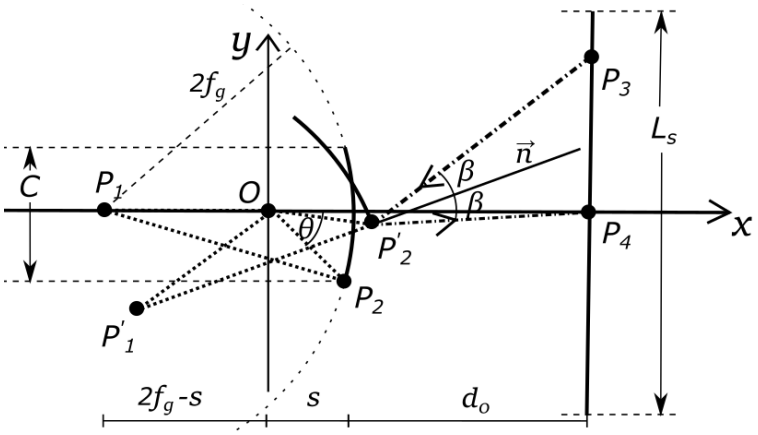


Figure 3.4: The model of viewing angle.

$$b_3 = \frac{b_2 - 2b_1 - b_1^2 b_2}{b_1^2 - 2b_1 b_2 - 1}$$

To calculate b_1 and b_2, the coordinate of P_1' and P_2', P_4 can be denoted as,

$$\begin{cases} P_1' : ((s-2f_g)cos\theta, (s-2f_g)sin\theta) \triangleq (C, D) \\ P_2' : (x_0 cos\theta - y_0 sin\theta, x_0 sin\theta + y_0 cos\theta) \triangleq (A, B) \\ P_2' : (s+d, 0) \triangleq (E, 0) \end{cases}$$

and thus

$$b_1 = \frac{B-D}{A-C}, \quad b_2 = \frac{B}{A-E}$$

The last missing piece is the coordinate of P_2, which is denoted as $P_2 : (x_0, y_0) = (r \times cos\alpha, r \times sin\alpha)$, where

$$\begin{cases} r = \sqrt{(\frac{C}{2})^2 + (\sqrt{R^2 - (\frac{C}{2})^2} - (R-s))^2} \\ \alpha = -arcsin(\frac{C}{2r}) \end{cases}$$

We note that the measured ranges in Table 3.2 are uniformly larger than the theoretical values, which could be caused by a coarse estimation of the distance s since the actual distance between the lens and the rotation center is hard to determine, and the fact that the model approximates the camera as a point instead of a surface.

We consider two cases of successful peeking with a rotation of the glass lens. The first case All Page claims success as long as there exists a point on the screen whose emitted light ray can reach the camera. The second case Center claims success only if the contents at the center of the screen have emitted lights that can be reflected to camera. Table 3.2 summarizes

the calculated theoretical angle ranges and the measured values. Both the theoretical model and measurements show that the webcam peeking attack is relatively robust to human positioning with different head viewing angles, which is validated later by the user study results (Section 3.6.2).

3.4.4 Image Distortion Characterization

Generally, the possible distortions are composed of imaging systems' inherent distortions and other external distortions. Inherent distortions mainly include out-of-focus blur and various imaging noises introduced by non-ideal camera circuits. Such inherent distortions exist in camera outputs even when no user interacts with the camera. External distortions, on the other hand, mainly include factors like motion blur caused by the movement of active webcam users.

User Movement-caused Motion Blur. When users move in front of their webcams, reflections from their glasses move accordingly which can cause blurs in the camera images. User motions can be decomposed into two components, namely involuntary periodic small-amplitude tremors that are always present [94], and intentional non-periodic large-amplitude movements that are occasionally caused by random events such as a user moving its head to look aside. By approximating user motions as displacements of h_o and utilizing Equation 3.3, the number of blurred pixels δ_p can be estimated by[2]:

$$\delta_p = \frac{\delta^T f_g}{d_o^2 + 2d_o f_g} \cdot \frac{f}{W} \cdot N$$

where δ^T is the motion displacement amplitude within the exposure time of a frame.

For tremor-based motion, existing research suggests the mean displacement amplitude of dystonia patients' head tremors is under 4 mm with a maximum frequency of about 6 Hz [95]. Since dystonia patients have stronger tremors than healthy people, this provides an estimation of the tremor amplitude upper bound. With the example glass in Section 3.4.2 and a 30 fps camera, the estimated pixel blur is under 1 pixel. Such a motion blur is likely to affect the recognition of extremely small reflections. Intentional motion is not a focus of this work due to its random, occasional, and individual-specific characteristics. We will experimentally involve the impacts of intentional user motions in the user study by letting users behave normally.

Distortion Analysis. To observe and analyze the dominant types of distortions, we recorded videos with the laptop webcam and a Nikon Z7 DSLR [21] representing a higher-

[2]We mainly consider motions that are parallel to the screen because generally, they cause larger blurs than other types of motions

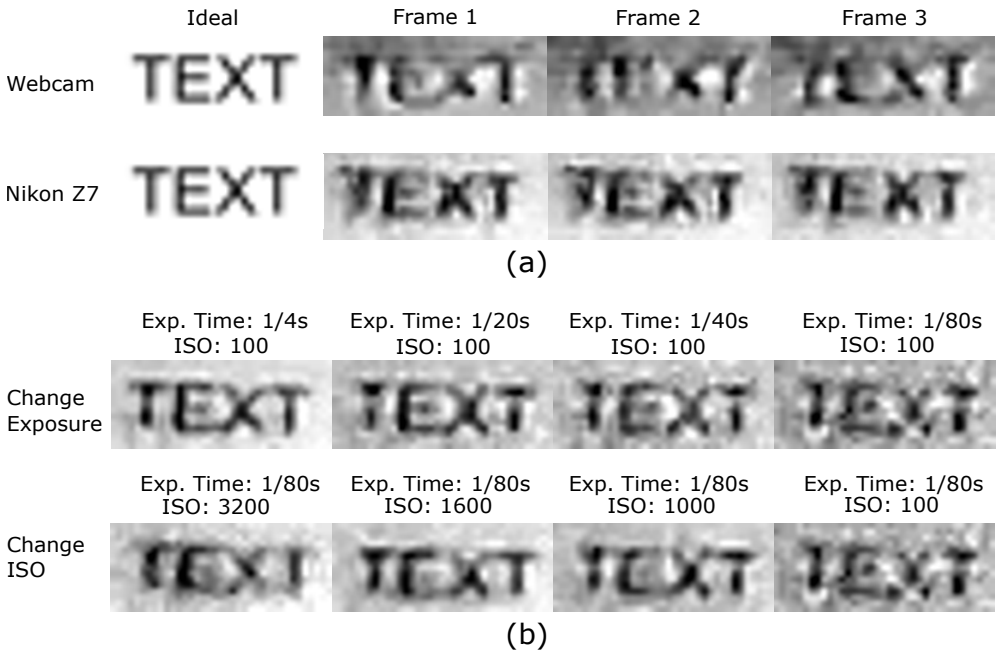


Figure 3.5: (a) The ideal capture versus the actual captures in three consecutive frames by webcam (1st row) and Nikon Z7 (2nd row). The distortions feature occlusions with inter-frame and intra-frame variance. The webcam yields larger variances. (b) Photos captured by Nikon Z7 under different exposure times and ISO settings. Longer exposure time and medium ISO yield smaller distortions and increase SNR.

quality imaging system. The setup is the same as the feasibility test except that we tested with both the still mannequin and a human to analyze the effects of human tremor. Figure 3.5 (a) shows the comparison between the ideal reflection capture and the actual captures in three consecutive video frames of the webcam (1st row) and Nikon Z7 (2nd row) when the human wears the glasses. Empirically, we observed the following three key features of the video frames in this setup with both the mannequin and human:

- Out-of-focus blur and tremor-caused motion blur are generally negligible when the reflected texts are recognizable.
- Inter-frame variance: The distortions at the same position of each frame are different, generating different noise patterns for each frame.
- Intra-frame variance: Even in a single frame, the distortion patterns are spatially non-uniform.

One key observation is that the captured texts are subjected to occlusions (the missing or faded parts) caused by shot noise [23] when there is an insufficient number of photons hitting the sensors. This can be easily reasoned in light of the short exposure time and small text

pixel size causing reduced photons emitted and received. In addition, other common imaging noise such as Gaussian noise gets visually amplified by relatively higher ISO values due to the bad light sensitivity of the webcam sensors. We call such noise ISO noise. Both two types of distortions have the potential to cause intra-frame and inter-frame variance. The shot and ISO noise in the webcam peeking attack plays on a see-saw with an equilibrium point posed by the quality of the camera imaging sensors. It suggests that the threat level will further increase (see the comparison between the webcam and Nikon Z7's images in Figure 3.5) as future webcams get equipped with better-quality sensors at lower costs.

3.4.5 Image Enhancing with MFSR.

The analysis of distortions calls for an image reconstruction scheme that can reduce multiple types of distortions and tolerate inter-frame and intra-frame variance. One possible method is to reconstruct a better-quality image from multiple low-quality frames. Such reconstruction problem is usually defined as multi-frame super resolution (MFSR) [249]. The basic idea is to combine non-redundant information in multiple frames to generate a better-quality frame.

We tested 3 common light-weight MFSR approaches that do not require a training phase, including cubic spline interpolation [249], fast and robust MFSR [98], and adaptive kernel regression (AKR) based MFSR [127]. Test results on the reflection images show that the AKR-based approach generally yields better results than the other two approaches in our specific application and setup. All three approaches outperform a simple averaging plus upsampling of the frames after frame registration, which may be viewed as a degraded form of MFSR. An example of the comparison between the different methods and the original 8 frames used for MFSR is shown in Figure 3.6 (a). We thus use the AKR-based approach for the following discussions.

One parameter to decide for the use of webcam peeking is the number of frames used to reconstruct the high-quality image. Figure 3.6 (b) shows the CWSSIM score improvement of the reconstructed image over the original frames with different numbers of frames used for MFSR when a human wears the glasses to generate the reflections. Note that increasing the number of frames do not monotonically increase the image quality since live users' occasional intentional movements can degrade image registration effectiveness in the MFSR process and thus undermine the reconstruction quality. Based on the results, we empirically choose to use 8 frames for the following evaluations. In addition, the improvement in CWSSIM scores also validates that MFSR-resulted images have better quality than most of the original frames. We thus only consider evaluation using the MFSR images in the following sections.

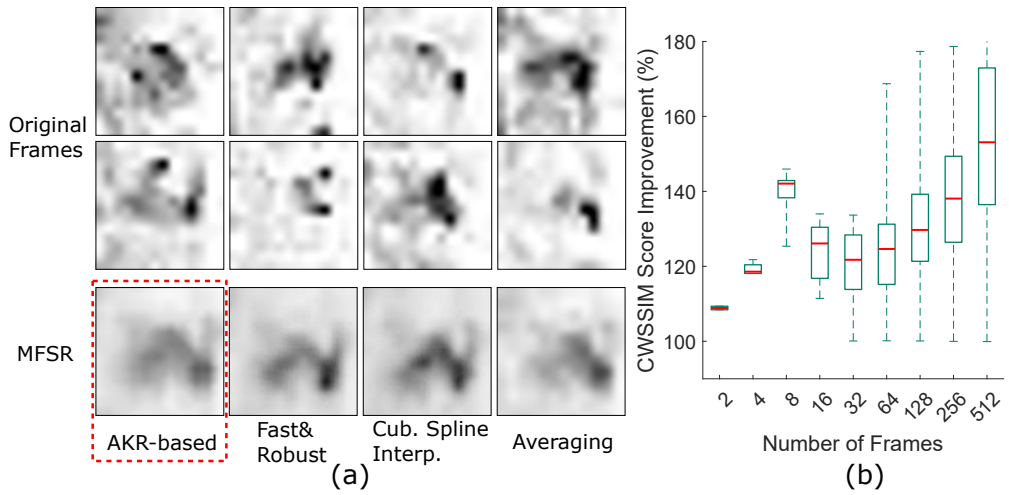


Figure 3.6: (a) Comparison between single frames and the MFSR-reconstructed images with 4 different MFSR approaches. The MFSR images are reconstructed with the 8 frames shown at the top. The AKR-based approach generally produces the best reconstruction results in our task of reflection image reconstruction. (b) The improvement of reflection reconstruction quality as the number of frames used for MFSR increases.

3.5 Reflection Recognizability & Factors

In this section, we evaluate the recognizability limits of reflected texts enhanced by the MFSR method given a specific set of webcams, glasses, and advantageous environmental conditions. We then investigate the impact of the most significant factors. The evaluations in this section are performed in a controlled lab environment and serve as the foundation for the analysis in Section 3.6.

3.5.1 Experimental Setup

Equipment. We collected all data with the aforementioned Acer laptop as the victim device, and another Samsung laptop [22] as the adversary's device. The two laptops were in a lab environment with WiFi network connection. The victim laptop was measured to have an internet download speed of 246 Mbps and upload speed of 137 Mbps while those for the adversary laptop were 144 Mbps and 133 Mbps respectively. We used two pairs of glasses, i.e., the pair of BLB glasses and prescription glasses.

Data Collection. We asked a person to wear the glasses and sit in front of the victim's laptop. The glass-screen distance was chosen to be 40 cm which was also found to be close to the average distance in the user study (see Figure 3.11 (b)). The screen brightness was 100%. To estimate the limits of recognition, we used an environmental light intensity of

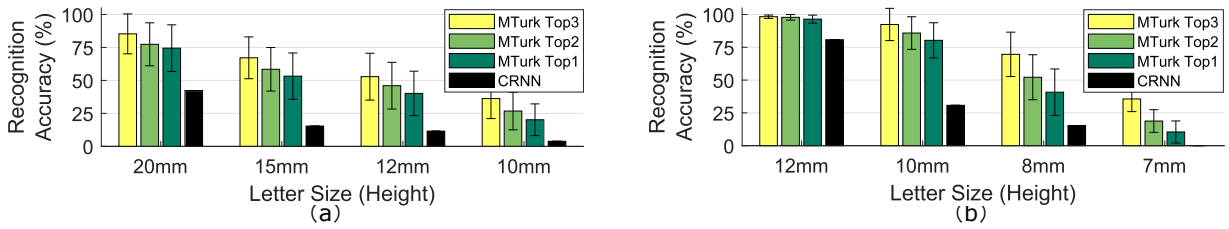


Figure 3.7: The recognition accuracy of letters in different sizes with (a) the BLB glasses and (b) the prescription glasses. Although the pair of BLB glasses have higher reflectance than the prescription glasses, the prescription glasses enable reading smaller on-screen texts because of their smaller curvature leading to larger reflection pixel size. Note that the conclusion is device-specific and cannot be applied to general BLB-prescription glass comparison. Humans are found more capable of recognizing the reflected texts than SOTA OCR models.

100 lux to generate the best reflections. We then displayed single capital letters (26 letters) on the victim screen with different heights ranging from 20 mm to 7 mm. The victim and adversary laptops had a Zoom [25] session with a video resolution of 1280×720. For each display of the letters, we recorded a 3s video of the victim's images on the adversary's laptop. We then used 8 consecutive frames starting from 1s for MFSR reconstruction and generated one corresponding image for each video. We generated 208 images in total for the 2 glasses each with 4 different sizes.

Recognizability Evaluation. In order to evaluate the recognizability of the reconstructed single-letter images and avoid potential bias introduced by the authors' prior knowledge of the reflections, we acquired recognition accuracy by (1) using multiple SOTA pretrained deep-learning OCR models including Google Tesseract and Keras CRNN, and (2) conducting a survey on Amazon Mechanical Turk (AMT) [13]. For the AMT study, we collected answers from 25 crowdsourcing workers for each reconstructed image and thus collected 5200 answers in total. We showed to the workers all reconstructed images in a randomized manner without providing them with any information on the original letters on the screen. We asked the workers to provide 3 best guesses of the single letter in each reconstructed image. They were allowed to input the same answer for multiple guesses if they feel confident in a guess, or if they have no clue about making subsequent guesses. The recognizability of the texts in the reconstructed images is then represented by the recognition accuracy, i.e., correctly recognized number of letters over the total number of letters in each case.

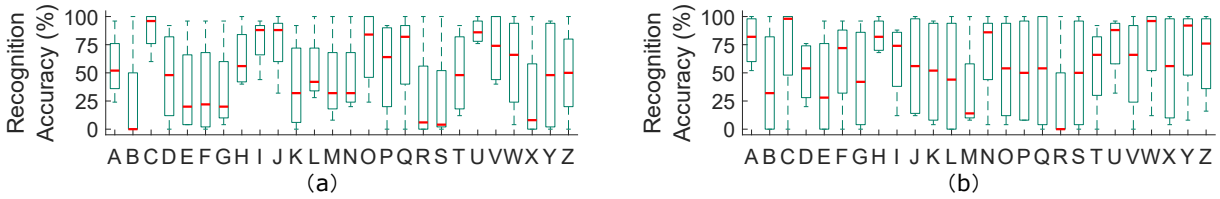


Figure 3.8: The human recognition accuracy of different letters with (a) the BLB glasses and (b) the prescription glasses. Letters such as "R" have been found the most difficult to read in the reflections while letters such as "C" and "U" have high recognizability. The difference is mostly due to the simplicity and symmetry in the letters' structures which lead to smaller degradation of recognizability when the reflections are subject to distortions.

3.5.2 Recognizability vs. Size & Letter

Figure 3.7 shows the recognition accuracy with the BLB and prescription glasses respectively with different letter sizes. The AMT accuracy for each letter size is calculated by including all 25 answers for all 26 letters, i.e., with a denominator of $25 \times 26 = 650$. We picked 4 representative letter sizes for each pair of glasses respectively, and show the top 1, 2, and 3 recognition accuracy. we also use error bars to show the standard deviations. The SOTA OCR models performed considerably worse than AMT workers. We believe the main reason is that data distribution in the models' training sets is very different from the actual data in webcam peeking. After testing different image data on the models, we found the two main causes for their bad performance are (1) significantly lower contrast, (2) occlusions caused by insufficient photons. Surprisingly, we also found the models sensitive to how we crop the images, which suggests the convolutional layer features and potential data augmentation schemes employed by these models are not dealing well with our data's distribution. We think future researchers can potentially utilize these pretrained models and collect their own webcam peeking dataset to fine-tune the model weights to better adapt machine learning recognition models to this scenario.

The prescription glasses generally yield better results for the webcam peeking attack, showing that 10 mm texts can be recognized in the reconstructed images with over 75% accuracy. Although not as good as the prescription glasses, the recognition accuracy with the BLB glasses is also high enough to support efficient peeking attacks against texts of 10-20 mm. Despite the better reflective characteristics of the BLB glasses, the prescription glasses still generate better results due to their smaller curvature, highlighting the risks of the peeking attack even without highly reflective glasses.

Intuitively, different letters in the alphabet would be recognized with different levels of hardships due to their structural characteristics (see Figure 3.8). For instance, the letters "R" and "B" have been found the hardest to recognize in both cases of the two pairs of

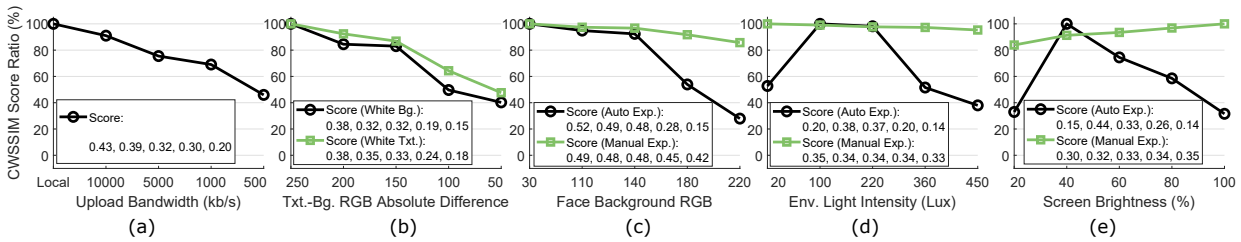


Figure 3.9: Effects of impact factors evaluated by CWSSIM scores. The original score numbers are displayed along with the legend at the bottom, and we plot the ratio between each score and the highest score in each case as a percentage. Visualizations of the effects can be found in the appendix.

glasses. On the other hand, letters such as "C", "U", "I", and "O" have generally the highest recognizability in all the sizes, which we suspect is due to their simple or highly symmetric structures that prevent the recognizability of such letters from dropping too seriously when the texts are down-sampled and occluded. Furthermore, we found letters having similar structures are confused with each other more easily in the recognition. For instance, "J" and "L" are mostly recognized as "I" when the letter size gets small because the distortions to the bottom part of "J" and "L" makes them just like "I" in the reflection images.

3.5.3 Network Influence

Video conferencing platforms like Zoom cause different levels of distortions in the images through video encoding and decoding under various network bandwidths. To analyze the impact, we compared the quality of the reconstructed images under different network bandwidths to that when the video is recorded by the victim's local device without going through Zoom. A visual demonstration of the effect is quantified with CWSSIM scores and shown in Figure 3.9 (a). We found that when the upload bandwidth is larger than 10 Mbps, the quality of the reconstructed images generally remains the same, and is close to the locally-captured and reconstructed images with a minor degree of added distortions. An upload bandwidth smaller than 10 Mbps starts to undermine the reconstructed image quality over Zoom. When the bandwidth is smaller than 1000 kpbs, the letters get hard to recognize. It's almost unrecognizable with a bandwidth smaller than 500 kbps. When the bandwidth was larger than 1500 kbps, Zoom was generally able to maintain a 720p video resolution with a frame rate close to 30 fps.

3.5.4 Physical Factors

The recognizability of the reflections is a highly complex multi-variate function over many physical factors. We categorize the factors into 2 groups, namely those mainly affecting the reflection pixel size (Section 3.4.2) and those affecting the light SNR. Comprehensive quantitative modeling of light SNR is very challenging due to the need for accurate imaging sensor models. Nevertheless, we provide qualitative analysis and quantify representative cases by calculating changes in CWSSIM scores (Figure 3.9).

In light SNR, the signal portion comes from the light emanating from the screen, reflected by the glasses, and then captured by the imaging sensors corresponding to the area of the screen. Other light captured by sensors in this area can be treated as noise. Counter-intuitively, more reflected light does not always lead to higher reflection recognizability as we will discuss next. Figure 3.9 (b-e) show the factors that can change light SNR most significantly. (c-e) also inspect how auto exposure and manual (fixed) exposure can affect the light SNR-recognizability relationships in surprisingly different ways by using the laptop built-in webcam and the configurable Nikon Z7 respectively.

Text Color Contrast. Different colors of texts can affect the reflection recognizability because the texts and screen background colors produce a certain contrast. We found that chroma has smaller effects than luma and show how luma affects reflection quality in Figure 3.9 (b) by using the absolute difference in RGB values of gray-scale text and background colors to represent the contrast. As expected, lower contrast (smaller RGB difference) undermines the reflection recognizability.

Face Background Reflectance. Face background reflectance is decided by sub-factors such as skin color. We tested different background reflectance by pasting the inner side of the glasses with papers of different gray-scale colors that have the same values for RGB. When the background has a higher reflectance (larger RGB values), more light from the environment as well as the screen will be reflected by it, increasing the noise portion of the light SNR and thus undermining the recognizability of the reflections as shown in Figure 3.9 (c).

Environment Light Intensity. A decrease in the environmental light intensity causes a smaller degree of noise and thus increases the light SNR. This increase, however, does not necessarily lead to better recognizability in the case of webcams which often have auto-exposure control to adjust the overall brightness of the videos they take. When the overall environment is too dark, the webcam's firmware automatically increases the exposure time trying to compensate for the dark environment. This increase in the exposure time can cause an over-exposure for the reflected contents on the glasses which could have much higher light intensity than the environment, leading to smaller contrast and thus harder-to-

read images. Such over-exposure is found in multiple participants' videos in the user study (Section 3.6.2). On the other hand, the recognizability monotonically increases in the case of manual-exposure cameras such as the Nikon Z7 in manual mode. Figure 3.9 (d) shows the different behaviors of auto and manual exposure.

Screen Brightness. Screen brightness is the opposite of environmental light intensity in terms of its impact on the reflection recognizability. When the screen is brighter, the signal portion in the light SNR increases and can lead to more readable reflections for manual-exposure cameras. However, auto-exposure of most webcams can again negatively affect recognizability. Specifically, if the screen gets too bright compared to the environmental lighting condition, the webcams will often adjust their exposure time and ISO based on the dominant environmental light condition, and thus cause over-exposure to the screen reflections. Figure 3.9 (e) shows the effects.

Summary. The results show that variations in physical conditions can change the actual limits of the attack dramatically. The fact that reflection recognizability does not change monotonically with some factors like environmental light intensity and screen brightness further challenges the attack by making it more difficult to predict the possible outcomes in uncontrolled settings.

3.5.5 Eyeglass Lens

The difference in recognition accuracies between the pair of BLB and prescription glasses (Figure 3.7) suggests parameters of different eyeglass lenses will influence the performance of webcam peeking. To examine the impact, we analyzed 16 pairs of eyeglasses by inspecting the correlation between their reflection quality quantified by CWSSIM scores and several lens factors. The CWSSIM scores are acquired with the 16 glasses when all other factors are kept the same.

The results suggest lens focal length, which determines the pixel size of reflections (Equation 3.3), has the strongest influence on the reflections with a correlation score of 0.56. The minimum, mean, and maximum focal length of the 16 pairs of glasses are 10, 268, and 110 cm respectively. With a correlation score of 0.42, the second strongest factor is found to be prescription strength (lens power) as lens power usually has a positive correlation with focal length following design conventions. Lens reflectance and surface coating conditions that mainly affect reflection light SNR produce correlation scores of 0.32 and 0.31 respectively. We empirically defined and added the factor of lens coating condition that gauges how much the lens coatings have worn off with higher values representing more intact coating. The motivation is our observation that damage in lens coating reduces the recognizability of re-

flections. We also estimated lens reflection spectrum by calculating the ratio between RGB values of the reflections in the image but only found correlation scores lower than 0.15. This suggests the glass type (e.g., BLB or non-BLB) does not have a strong influence on reflection quality. Finally, we expect the parameters analyzed above have certain relationships with lens and coating materials, which require specialized optical equipment to measure and determine.

3.6 Cyberspace Textual Target Susceptibility

The evaluations so far are based on the text's physical size and carried out in controlled environments to better characterize user-independent components of the reflection model as well as the range of theoretical limits for webcam peeking. In this section, we start by mapping the limits to common cyberspace objects in order to understand the potential susceptible targets. We then conduct a 20-participant user study with both local and Zoom recordings to investigate the feasibility and challenges of peeking these targets and various factors' impact.

3.6.1 Mapping Theoretical Limits to Targets

We use web texts as an enlightening example of cyberspace textual targets considering their wide use and the relatively mature conventions of HTML and CSS. The discussion is based upon (1) a previous report [152] scraping the most popular 1000 websites on Alex web ranking [12], and (2) a manual inspection of 117 big-font websites archived on SiteInspire [14]. We further divide the inspected web texts into the three groups $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ below, in order to discuss separately how the webcam peeking attack with current and future cameras could have effects on them.

Text Sizes. We summarize the text sizes investigated in Table 3.3 where The cap height values are measured with the Acer laptop and default OS and browser settings.

$\mathcal{G}_1$ and $\mathcal{G}_2$: The first group represents the median HTML P, H1, H2, H3 texts of the 1000 websites. [152] reports that the median size of the P elements is about 12 pt and H1, H2, H3 sizes are close to the 2, 1.5, 1.17 em ratios recommended [18]. We thus use these point sizes for $\mathcal{G}_1$ and specify the corresponding cap heights in Table 3.3. The second group represents the largest HTML P, H1, H2, H3 texts of the 1000 websites in [152] with the same recommended em ratios for the headers. [152] finds that about 4% of the 1000 websites use a P size as large as 21 pt. This results in H1, H2, H3 sizes of 25, 32, and 45 pt respectively.

$\mathcal{G}_3$: The third group represents the 117 big-font websites' texts. We manually inspected

Table 3.3: Text sizes of web contents

Target	**Point Size**	**Cap Height (mm)**
$\mathcal{G}_1$ P	12	2.1
$\mathcal{G}_1$ H3	14	2.5
$\mathcal{G}_1$ H2	18	3.2
$\mathcal{G}_1$ H1	24	4.3
$\mathcal{G}_2$ P	21	3.7
$\mathcal{G}_2$ H3	25	4.3
$\mathcal{G}_2$ H2	32	5.6
$\mathcal{G}_2$ H1 (S1)	42	7.4
$\mathcal{G}_3$ 0% (S2)	56	10
$\mathcal{G}_3$ 20% (S3)	80	14
$\mathcal{G}_3$ 40% (S4)	102	18
$\mathcal{G}_3$ 60%	136	24
$\mathcal{G}_3$ 80% (S5)	253	35
$\mathcal{G}_3$ 95% (S6)	340	60

all the 427 websites archived on SiteInspire[14]. The reason for manual analysis rather than scraping is that many large-font texts on the websites are embedded in the form of images instead of HTML text elements in order to create more flexible font styles. We then selected 117 of them based on the following criteria: (1) The webpage is still active. (2) The largest static texts that enable an adversary to identify the website through google search have a cap height of at least 10 mm when displayed on the Acer laptop. We show the different quantiles of the largest physical cap heights on the 117 websites and the converted point sizes in Table 3.3. We find that most websites in $\mathcal{G}_3$ are related to art, design, and cinema industry which like to present their stylish design skills but unfortunately make the web peeking attack easier. About 1/3 of the websites are designers' or studios' websites that computer science/security researchers may overlook. Furthermore, 72 out of the 117 websites are ranked on Alexa from 38 to 8,851,402 with 5 websites among the top 10,000.

As pointed out in Section 3.4.2, the conversion between digital point size and physical cap height is dependent on specific user settings such as browser zoom ratio. The cap height values in Table 3.3 are thus measured with the Acer laptop with default OS and browser settings as a case study.

Based on the results in Figure 3.7, we hypothesize that the smallest cap heights adversaries can peek using mainstream 720p cameras is 7-10 mm. We then calculate the corresponding limits with 1080p and 4K cameras with Equation 3.3 and show them in the Theoretical column of Table 3.3. Considering participants are most likely to use 720p cameras, we then choose point sizes S1-S6 in Table 3.3 for evaluations.

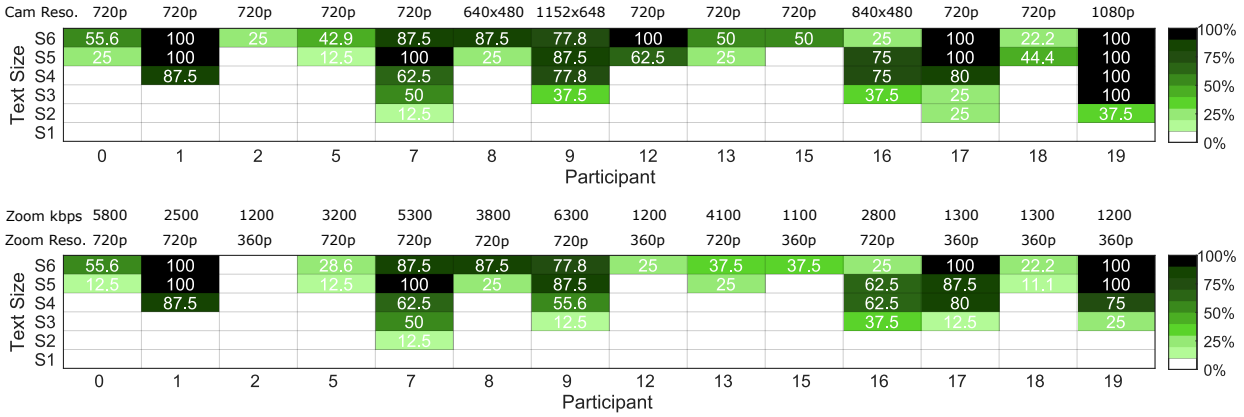


Figure 3.10: The recognition results of textual reflections collected with local and Zoom-based remote video recordings from 20 user study participants. Participants 4, 14, and 3, 6, 10, 11 did not generate glass reflections that allow successful recognition due to problems of out-of-range viewing angles and very low light SNR respectively and are thus omitted from the figure.

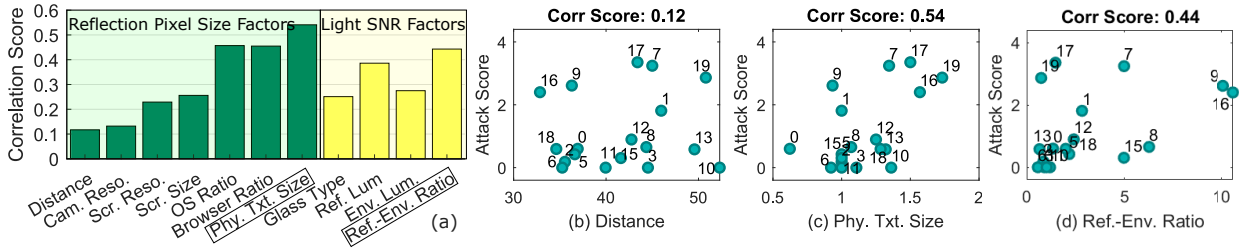


Figure 3.11: (a) The degree of influence of different factors on the reflection recognition performance evaluated by the correlation scores. Factors highlighted with boxes are computed with other raw factors according to our model. (b-d) The joint distribution of three factors and the recognition results.

3.6.2 User Study

The user study is designed in the following challenge-response way: An author generates HTML files each with one randomly selected headline sentence containing 7-9 words [3] from the widely-used "A Million News Headlines" dataset [148]. Only each word's first letter is capitalized. The participants display the HTML page in their browsers when they are recorded, and another author acting as the adversary tries to recognize the words from the videos containing the 20 participants' reflections without knowing the HTML contents by using the same techniques as in Section 4.6. We then calculate the percentage of correctly recognized words.

[3]Uniform lengths (e.g., all 8 words) are avoided to prevent the adversary from guessing the words by knowing how long the sentences are.

Data Collection. Each participant was given 6 HTML files of increasing point sizes from S1 to S6 as shown in Table 3.3. Note that the 6 sizes are specified in point size in HTML so that user-dependent factors such as screen size and browser zoom ratio can be studied (Equation 4.5). The participants display each HTML file on their own computer display in their accustomed rooms and behave normally as in video conferences. We allow participants to choose their preferred environmental lighting condition except asking them to avoid other close light sources besides the screen in front of their face. The reason is that we found a close frontal light source can seriously decrease light SNR, which can potentially be used as a physical mitigation against this attack but prevents us from examining the impact of all the other factors. We did not tell the participants to stay stationary and let them behave normally as in browsing screen contents. Their webcams record their image for 30 seconds for each HTML.

Network bandwidth and resulted video quality are artifacts of video conferencing platforms that improve in a rapid way [7] compared to other user-dependent physical factors. To study the present-day and possible future impact of video conferencing platforms, we record the 20 participants' videos both locally and remotely through Zoom. Our experiments focused on Zoom since it is the most used platform and also provides the most detailed video and network statistics.

We asked the participants to report their user-dependent parameters including screen resolution (N_{os}), screen physical size (H_{sr}), OS and browser zoom ratio (s_{os}, s_b) webcam resolution in Equation 4.5, webcam resolution (N) in Equation 3.3, and the type of their glasses. Some other physical factors including environmental light intensity, screen brightness, glass-screen distance, and the physical size of displayed texts are difficult to be measured by the participants themselves and are not reported. We thus estimated the values of these factors by utilizing their videos.

General Adversary Recognition Results. The recognition results achieved by the adversary with local and remote recordings are shown in Figure 3.10 (upper and lower respectively). Two participants (4 and 14) did not generate glass reflections of their screens in the video recordings due to the problem of out-of-range vertical viewing angles as predicted in Section 3.4.2. Four participants (3, 6, 10, 11) yield 0% textual recognition accuracy due to a very low light SNR.

With local video recordings, the percentage out of the 20 participants that are subjected to non-zero recognition accuracy against S6-S1 are 70%, 60%, 30%, 25%, 15%, and 0% respectively. Videos of participants 7 and 17 using 720p cameras allowed the adversary to achieve 12.5% and 25% accuracies on recognizing S2. Videos of participant 16 using a 480p camera allowed the adversary to achieve an 37.5% accuracy on recognizing S3. These results

translate to the predicted susceptible targets with cameras of different resolutions as listed in the User column of Table 3.3, where 720p webcams pose threats to large-font webs ($\mathcal{G}_3$) and future 4K cameras pose threats to various header texts on popular websites ($\mathcal{G}_1$ and $\mathcal{G}_2$). As expected, this result is worse than the theoretical limits in the table that are derived with prescription glass data in the controlled lab setting (Section 4.6). Our observations suggest the main reasons include: (1) The environmental lighting conditions of the users are more diverse and less advantageous to screen peeking than the lab setup, generating reflections with worse light SNR. (2) Texts in the user study are mostly lower-case and have thus smaller physical sizes than the upper-case letters used in Section 4.6. (3) The prescription glasses used in Section 4.6 have a larger focal length than the average user's glasses. (4) More intentional movements exist in the user study leading to more motion blur.

With Zoom-based remote recordings, the percentage of participants with non-zero recognition accuracy against S6-S1 degraded to 65%, 55%, 30%, 25%, 5%, and 0% respectively. We logged the video network bandwidth and resolution reported by Zoom as shown in Figure 3.10. The correlation between Zoom bandwidth, resolution, and their impact on video quality agrees with the observations in Section 3.5.3. Generally, bandwidths smaller than 1500 kbps led to 360p resolutions for most of the time and decreased the recognizable text size by 1 level. Zoom's 720p videos also caused degradation in recognition accuracy but mostly kept the recognizable text size to the same level as the local recordings, suggesting the same predictions of susceptible text sizes and corresponding cyberspace targets.

Besides the mostly used platform Zoom, we also acquired remote recordings of participant 19 with Skype and Google Meet. The adversary achieved better results with Skype than Zoom by recognizing S3 and S2 with 89% and 25% accuracies respectively, which is likely due to Skype's capability of maintaining better-quality video streams with a 1200 kbps bandwidth. The web-based Google Meet platform provided the lowest quality videos and only allowed the adversary to achieve 22% accuracy on recognizing S4.

Underlying Reasons. To find out the dominant reasons enabling easier webcam peeking by analyzing the correlation between the recognition results and different factors, we turn each participant's results (6 sizes) into a single *attack score* that is a rectified weighted sum of the recognition accuracy of the six text sizes tested. Figure 3.11 (a) shows correlation scores with 11 factors that affect reflection pixel size (left) and light SNR (right) respectively when $w = 1.5$. The glass type includes prescription (15/20) and prescription with BLB coatings (5/20). The physical text size and reflection-environment light ratio highlighted in the boxes are two composite factors. In short, the physical text size represents the ratio between the actual physical size of texts displayed on each participant's screen and the case study values in Table 3.3 and is calculated with Equation 4.5 with other raw factors such as browser zoom

ratios. The reflection-environment light ratio represents how strong the screen brightness is compared to the environmental light intensity and is calculated by dividing glass luminance by environmental luminance. Basically, these two composite factors represent our model's prediction of reflection pixel size and light SNR and are found to generate higher correlation scores than the other raw factors, which validates the effectiveness of our models. Figure 3.11 (b-d) further show the joint distribution of the attack score and three representative factors. It can be seen from (b) that the 40 mm screen-glass distance used in the evaluation of Section 4.6 is about the average of the participants' values, and distances of these participants actually only have a very weak correlation with the easiness of webcam peeking attack. Figure 3.11 (d) suggests that when the screen brightness-environmental light intensity ratio gets lower than a certain threshold, the likelihood of preventing adversaries from peeking is very high, which may be considered as a temporary mitigation.

3.7 Website Recognition

The results so far suggest it may still be challenging for present-day webcam peeking adversaries with mainstream 720p cameras to eavesdrop on common textual contents displayed on user's screens. During our experimentation, we observed that recognizing graphical contents such as shapes and layouts on the screen is generally easier than reading texts. Although shapes and layouts contain more coarse-grained information compared to texts, a webcam peeking adversary may still pose non-trivial threats by correlating such graphical information with privacy-sensitive contexts. This work further explored to which degree can a webcam peeking adversary recognize on-screen websites by utilizing non-textual graphical information.

Data Collection. 10 out of the 20 participants in the user study participated in the website recognition evaluation. Following a similar methodology as in [136], we used the Alexa top 100 websites as a closed-world dataset. We only investigate the recognition of the home page of each website in this work. [136] shows that other pages of a website can also lead to the recognition of the website. We believe the easiness of recognizing a website using different pages is worth exploring in future works. The experiment followed a similar procedure as the textual recognition experiment in Section 3.6. For each participant, one author generates a unique random sequence of 25 websites for the participant to browse (10 seconds for each website) while another author acts as the adversary that analyzes the video recordings. Both local and Zoom-based remote recordings were obtained and recognized by the adversary. The adversary was given the whole recording and was asked to match each segment of the video to a specific website out of the 100 websites in the correct order. A

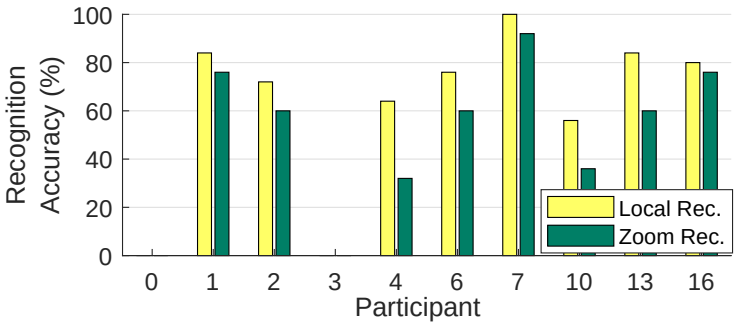


Figure 3.12: Accuracy of recognizing Alexa top 100 websites from eyeglass reflections. Each participant browsed 25 websites. Participant 0 and 4 did not yield recognizable reflections due to bad light SNR and viewing angles.

random guess naive adversary is supposed to have a success rate of about 1%. Note that some participants changed their environment and ambient lighting compared to the previous textual recognition experiment since the two experiments were conducted five months apart.

Recognition Results. Figure 3.12 shows the percentage of websites (out of 25) correctly recognized by the adversary. Participants 0 and 4 did not yield recognizable reflections due to bad light SNR and viewing angles respectively. This ratio of zero recognition (2 out of 10) agrees with that in the textual recognition test (6 out of 20), suggesting that webcam peeking may be impossible in 20-30% video conferencing occasions due to extreme user environment configurations.

As expected, participants with higher textual recognition accuracies such as participant 7 generally yield higher website recognition accuracies too. In addition, we observe that website recognition is more robust to various lighting conditions in the participants' ambient environment. For example, we found participant 10 who had 0% textual recognition accuracy due to bad light SNR produced 56% (local) and 36% (remote) accuracies in website recognition with the same environment and lighting. The reasons are two-fold. First, solid graphical contents such as color blocks commonly found on web pages occupy larger areas than the body of texts and are thus much easier to identify in low-quality videos. Second, compared to black texts on white backgrounds which only have two different colors, the overall web pages with multiple graphical contents have more colors and contrast, leading to better robustness against over- and under-exposure of the usable screen contents in the webcam videos.

Recognition Easiness and Web Characteristics. Compared to texts, websites feature more abundant and diverse characteristics. We conducted qualitative and quantitative analyses to identify the characteristics that make certain websites more susceptible to webcam peeking. To that end, we ranked the 100 websites by their easiness of recognition

Figure 3.13: A spectrum of Alexa top 100 websites that are found to be the easiest (upper) and hardest (lower) to recognize in our evaluation of website recognition under webcam peeking attacks. Screenshots of each website are rotated by 90 degrees and concatenated horizontally. Correlations scores between the rank of website recognition easiness and website pixel values' average and standard deviation are -0.33 and 0.45 respectively, suggesting darker websites with high-contrast graphical contents are easier to recognize.

utilizing recognition accuracies. Figure 3.13 shows rotated screenshots of the websites that rank the top and bottom 15 by their recognition easiness. Visual inspections suggest websites with higher contrast, larger color blocks, and more salient relative positions between different color blocks are easier to recognize. Websites that are mostly white with sparse textual and graphical components on them are the hardest to recognize. We calculated the correlation scores between the rank of each website and the average as well as the standard deviation of the websites' pixel values. Generally, a higher average means the website is closer to a pure white screen; a higher standard deviation means the website has more abundant high-contrast textures. The correlation scores obtained are -0.33 and 0.45.

3.8 Mitigation

3.8.1 Near-Term Mitigations

Given the threats, it is worthwhile exploring feasible mitigations that can be applied immediately. A straightforward approach involves users modifying the dominant physical factors identified in this work to reduce reflections' light SNR, e.g., by placing a lamp facing their face whose light increases the noise portion of light SNR. For software mitigations, we notice Zoom provides virtual filters of non-transparent cartoon glasses that can completely block the eye areas and thus eliminate reflections. Such features are not found in Skype or Google Meet. Other software-based approaches that support better usability involve fine-tuned blurring of the glass area. Although none of the platforms supports it now, we have implemented a real-time eyeglass blurring prototype that can inject a modified video stream

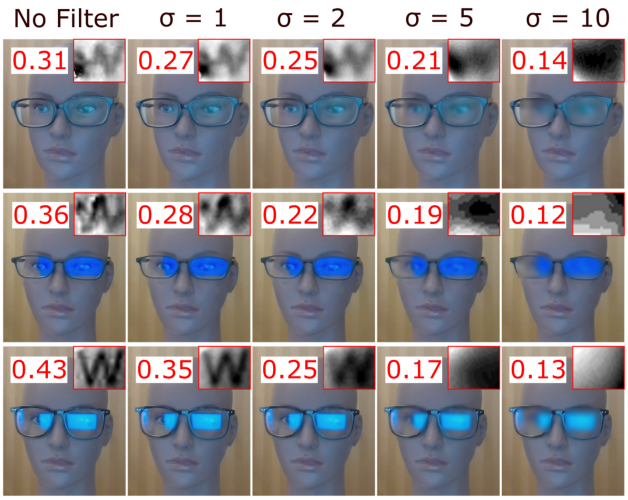


Figure 3.14: Different strengths of Gaussian filtering applied on three pairs of glasses. The reflected texts and their CWSSIM scores in each case are shown. Different glasses require different strengths of filters to reduce the reflection. We thus advocate an individual reflection testing procedure to determine protection scheme and settings.

into the video conferencing software. The prototype program[4] locates the eyeglass area and apply a Gaussian filter to blur the area. Figure 3.14 demonstrates the effect of using different strengths of Gaussian filtering by tuning the σ parameter. Stronger filtering (higher σ) reduces reflection quality more but also undermines usability and user experience to a larger degree as it makes the users' eye areas look more unnatural. We believe the usable strength also depends on the characteristics of specific glasses. For example, Figure 3.14 shows three pairs of glasses with increasing reflectance. Since glasses with higher reflectance (e.g., the 3rd row) may already have produced screen reflections that occupy and distort images of users' eye areas, applying stronger filtering may cause less degradation in user experience in this case. On the other hand, lower-reflectance (e.g., the 1st row) glasses may require weaker filtering to maintain the same degree of usability. In general, we believe it is a good idea for future platforms incorporating this protection mechanism to allow users to adjust filtering strength by themselves.

3.8.2 Improve Video-conferencing Infrastructure

Individual Reflection Assessment Procedure. Our analysis and evaluation reveal that different individuals face varying degrees of potential information leakage when subjected to webcam peeking. Specifically, various factors of software settings, hardware devices,

[4]Details and open-source code of this prototype implementation can be found at https://github.com/longyan97/EyeglassFilter.

and environmental conditions affect the quality of reflections. Even for the same user, the potential level of threats varies when the user joins video conferences from different places or at different times of the day. These factors make it infeasible to recommend or implement a single set of protection settings (e.g., what glasses/cameras/filter strength to use) before the actual user settings are known.

Providing usable security requires an understanding of how serious the problem is before trying to eliminate the problem. In light of this, we advocate an individual reflection assessment procedure that can potentially be provided by future video conferencing platforms. The testing procedure can be made optional to users after notifying them of the potential risk of webcam peeking. The procedure may follow a similar methodology as the one used in this work by (1) displaying test patterns such as texts and graphics, (2) collecting webcam videos for a certain period of time, (3) comparing reflection quality in the video with test patterns to estimate the level of threats of webcam peeking. With the estimated level of threats, the platform can then notify the user of the types of on-screen content that might be affected and offers options for protection such as filtering or entering the meeting with the PoLP principle that will be discussed below.

Principle of Least Pixels. Cameras are getting more capable than what average users can understand—unwittingly exposing information beyond what users intend to share. The fundamental privacy design challenge with webcam technology is "oversensing" [62] where overly-capable sensors can provide too much information to downstream processing—more data than is needed to complete a function, such as a meaningful face-to-face conversation. This oversensing leads to a violation of the sensor equivalent to the classic *Principle of Least Privilege (PoLP)* [197]. We believe long-term protection of users ought to follow a PoLP (perhaps a Principle of Least Pixels) as webcam hardware and computer vision algorithms continue to improve. Thus, we recommend that future infrastructure and privacy-enhancing modules follow the PoLP not just for software, but for the camera data streams themselves. In sensitive conversations, the infrastructure could provide only the minimal amount of information needed and allow users to incrementally grant higher access privileges to the other parties. For example, PoLP blurring techniques might blur all objects in the video meeting at the beginning and then intelligently unblur what is absolutely necessary to hold natural conversations.

3.8.3 User Opinion Survey

We collected opinions on our findings of webcam peeking risks and expectations of protections from 60 people including the 20 people who participated in the user study and 40 people

who did not. We did not find apparent differences between the two group's opinions. The overall opinions are reported below.

Textual Recognition. For the discovered risk of textual recognition, 40% of the interviewees found it a larger risk than what they expected; 48.3% thought it was almost the same as their expectation; 11.7% expected worse consequences than what we found. In addition, 76.7% of the interviewees think this problem needs to be addressed while 23.3% think they can tolerate this level of privacy leakage.

Website Recognition. 61.7% of the interviewees found it a larger risk than what they expected; 30% thought it was almost the same as their expectation; 8.3% expected worse consequences than what we found. In addition, 86.7% of the interviewees think this problem needs to be addressed while 13.3% think they can tolerate this level of privacy leakage.

Reflection Assessment. Regarding the proposed idea of reflection assessment procedures that may be provided by video conferencing platforms in the future, 95% of the interviewees said they would like to use it; 85%, 68.3%, 45%, and 20% of the 60 interviewees would like to use it when meeting with strangers, colleagues, classes, and family/friends respectively.

Glass-blur Filters. Regarding the possible protection of using filters to blur the glass area, 83.3% of the interviewees said they would like to use it; 78.3%, 51.7%, 43.3%, and 11.7% of the 60 interviewees would like to use it when meeting with strangers, colleagues, classes, and family/friends respectively.

3.9 Touchtone Eavesdropping with Zero-permission Inertail Measurement Units

Touchtones, the sounds produced by a smartphone when a numerical key is pressed, are an established communication standard widely used to encode user feedback in telephony channels [233]. In modern telephony systems, touchtones often encode important information such as credit card numbers (during call-based activation), bank pins, various account numbers, social security numbers, selections for various options in automated services, and possibly even votes in a phone-based federal election [235].

Recent side channel research has shown that sound produced by a smartphone's speaker may "leak" into the same phone's motion sensors, particularly speech audio during phone calls. This side-channel vulnerability results in a security breach in smartphone operating systems including the most prevalent Android system because third-party applications do not need any user permissions to use these motion sensors including gyroscopes and ac-

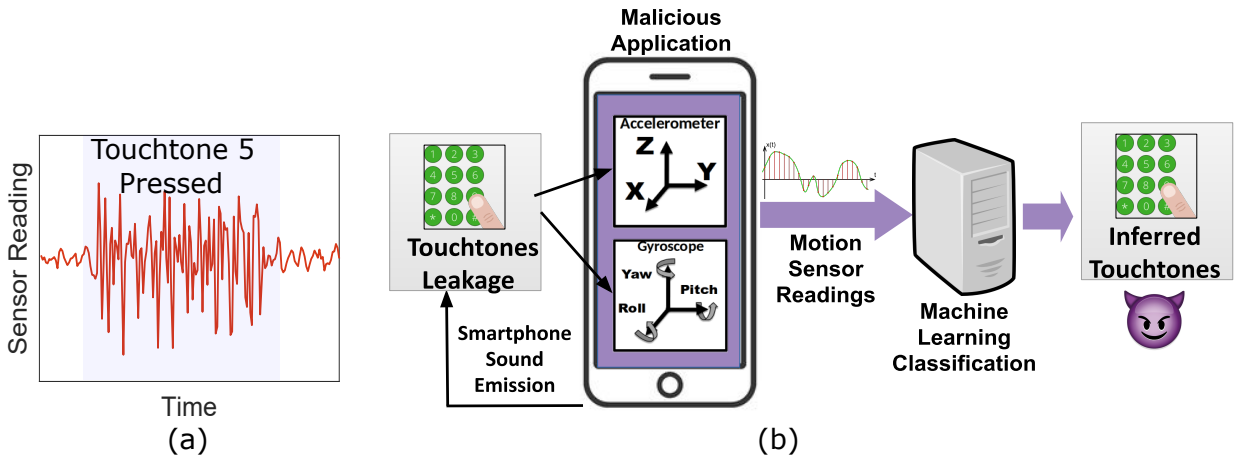


Figure 3.15: Touchtone leakage and eavesdropping. (a) A touchtone, indicating a "5" on a smartphone number pad, leaks into accelerometer data. (b) A malicious smartphone application can classify this leakage to discern that a "5" touchtone was emitted, inferring user input of a "5" for purposes such as dialing a phone number or inputting information into automated services.

celerometers. In contrast, the use of smartphone microphones, which are usually perceived as the sensor for receiving acoustic information, do require explicit user permissions. As a result, malicious applications may stealthily collect motion sensor data that can contain acoustic information without user notice.

Our work investigates *touchtone leakage*, a new security threat that this side-channel vulnerability causes where touchtone's acoustic information leaks into motion sensor data. Touchtone leakage occurs with a signal-to-noise ratio sufficient to be observed even visibly (Figure 3.15a). This leakage enables malicious smartphone applications (e.g., a seemingly benign health monitoring app running in the background) to eavesdrop on numerical user input that produces touchtones as shown in Figure 3.15. This work seeks to characterize the root causes and limits of touchtone leakage and eavesdropping. Specifically, we investigate why acoustic information is hidden in motion sensor data and how signal processing and physical phenomenon, such as aliasing or varying frequency responses, aid adversarial recovery of the original user keypress. These phenomena cause artifacts of touchtone information to manifest in a multitude of ways such as harmonics and aliases of harmonics. An adversary only needs to be able to ascertain user input through one of those manifestations. More advanced techniques such as selective integration of multiple sensors and sensor axes via machine learning can instead utilize several of these manifestations simultaneously for a more proficient attack. We demonstrate these ideas by designing an eavesdropping classifier based on the XGBoost machine learning model. Our experiments with four Android smartphones suggest that 12 smartphone touchtones can be recovered by an adversary at over 99 % accuracies.

3.9.1 Touchtone Leakage through Motion Sensors

This section analyzes the information leakage threats posed by touchtone eavesdropping using smartphone motion sensors (Fig 3.15) and the multitude of reasons why it can be difficult to mitigate. We investigate how touchtones produced by a phone's speaker leak distinguishable, deterministic side-channel signals into the smartphone's accelerometer and gyroscope sensor readings.

3.9.1.1 Background

Touchtones. Touchtones, also known as dual-tone multi-frequency (DTMF) signals, are a standardized code of two-tone audible acoustic signals that play upon a numerical key press [233]. Touchtones are often used in telecommunications and various other applications with a numerical touchpad [75, 151]. The sound is produced by a phone when users press an individual key to dial a phone number, answer an automated telephony question (e.g. "press 1 to...."), register credit card numbers or bank pins over the phone, etc. There are 12 unique touchtones commonly used by smartphones (Figure 3.16), each consisting of two frequencies taken from two separate frequency sets, used for the numbers 0-9, the symbols * and #. As they are unique, hearing one touchtone is indicative of a certain number press. These dual-tone combinations have been chosen specifically to be easily understood in the presence of noise for reliable communication.

Aliasing. Aliasing can have several definitions depending on the context, but the most relevant definition in the context of this paper refers to distortions caused by the improper sampling of a signal [166, 229]. As defined by the Nyquist sampling theorem [204, 168] the highest frequency a sensor with sampling rate f_s can properly sample is the Nyquist frequency $f_N = f_s/2$. If a signal has frequencies greater than f_N the sensor output *will* contain aliases of the original signal. The formula for the frequency of the alias, f_a, given the Nyquist frequency f_N and the frequency of the original signal f is $f_a = |2mf_N - f|$.

3.9.1.2 Threat Model

This paper considers an adversary whose goal is to determine a user's numerical key presses on a smartphone using access to a smartphone's motion sensor data and the knowledge of touchtone leakage, an attack we term *touchtone eavesdropping*.

We assume the adversary can obtain and save motion sensor data through means such as a malicious application running in the background with motion sensor access, as has been assumed in all previous works of acoustic eavesdropping using smartphone motion sensors. Note that popular smartphone platforms such as Android do not require applications to

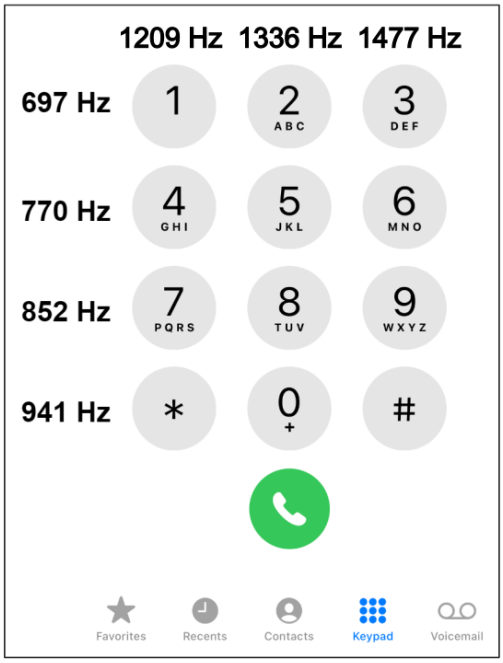


Figure 3.16: Touchtone frequencies. Touchtones are comprised of two single-frequency tones emitted simultaneously to convey numerical input.

ask for user permission to use motion sensors. As a result, any applications running in the background can potentially collect motion sensor data stealthily for malicious purposes.

We also assume the adversary has access to the same model as the victim's phone(s); a phone's model can be determined by an application using fingerprinting techniques [61, 188, 259]. The adversary can use their duplicate phone(s) to collect training data to build a classification system. Last, the adversary has unlimited time to classify victim data as the victim data can be saved and sensitive information (e.g. credit card numbers, bank pins, social security numbers) may not change often.

3.9.1.3 Acoustic Waves and Sensor Construction

Acoustic waves produced by the smartphone's speaker alter the output of microelectricalmechanical systems (MEMS) accelerometers and gyroscopes [58] due to how these sensors *approximate* motion. MEMS accelerometers and gyroscopes approximate the motion of a larger body (i.e. a smartphone) via the motion of a small sensing mass(es) attached to capacitive springs. When the mass(es) moves, the springs create a representative voltage which is then amplified, filtered, digitized, and sent to the processor. However, while the linear or angular acceleration of the sensing mass(es) are usually accurate representations of the body's acceleration, they are not exact. For example, small acoustic vibrations via the air or contacted

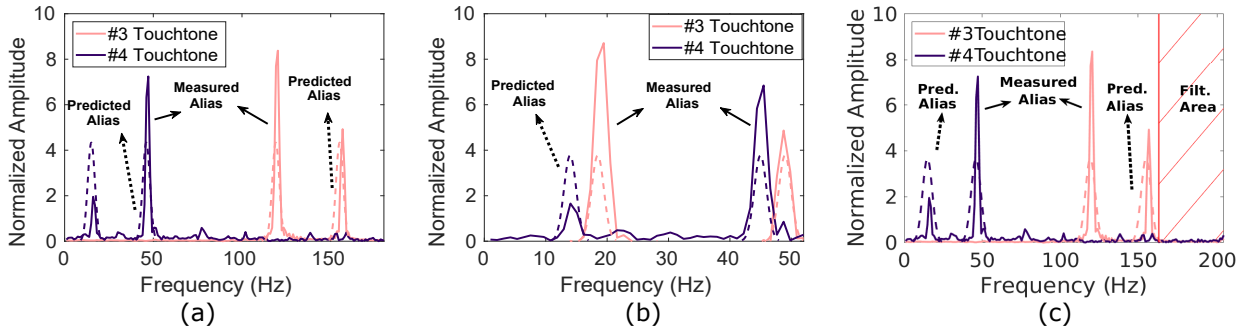


Figure 3.17: Predictable and discernible touchtone leakage. Touchtone leakage for #3 and #4 touchtones in a Google Pixel 2's accelerometer's x-axis. These signals remain discernable and predictable in the frequency domain with (a) a normal, unaltered signal, and also despite apparent mitigations suggested by previous research including (b) reduced sampling rates and (c) digital low-pass filtering.

surfaces can move the small sensing masses even if minimally affecting the connected body (i.e. smartphone) due to effects such as varying frequency responses [171, 228, 44]. In this case, MEMS accelerometers and gyroscopes may capture acoustic signals.

3.9.1.4 Touchtone Aliasing

Aliasing is a key factor in both making touchtone leakage occur and making it difficult to mitigate. Touchtones have frequencies higher than the Nyquist sampling rate for most smartphone motion sensors (lower than 250 Hz), and thus have aliases. However, the frequencies of these aliases can be predicted as the touchtone frequency and sampling rate are both known (Fig 3.17). An attacker can use these known aliases to indicate the presence of the missing original touchtone frequencies. Furthermore, the placement of these aliases — how all touchtone frequencies can lie somewhere in the sampled signal's frequency band — can make touchtone eavesdropping resistant to certain apparent mitigations. For example, reducing the sampling rate will not get rid of aliases, but only move them to other deterministic frequencies (Fig 3.17b). Furthermore, low-pass filters may remain ineffective unless the cutoff frequency is placed low, as touchtone aliases could be close to 0 Hz (Fig 3.17c).

3.9.1.5 Leakage Signal Manifestations

The two above factors enable touchtone leakage, but information leakage may manifest in a multitude of forms simultaneously (e.g., different axes of a sensor's readings having different signals caused by touchtones) due to a variety of physical and signal processing phenomena; these manifestations can provide complementary and distinct information for the purpose of

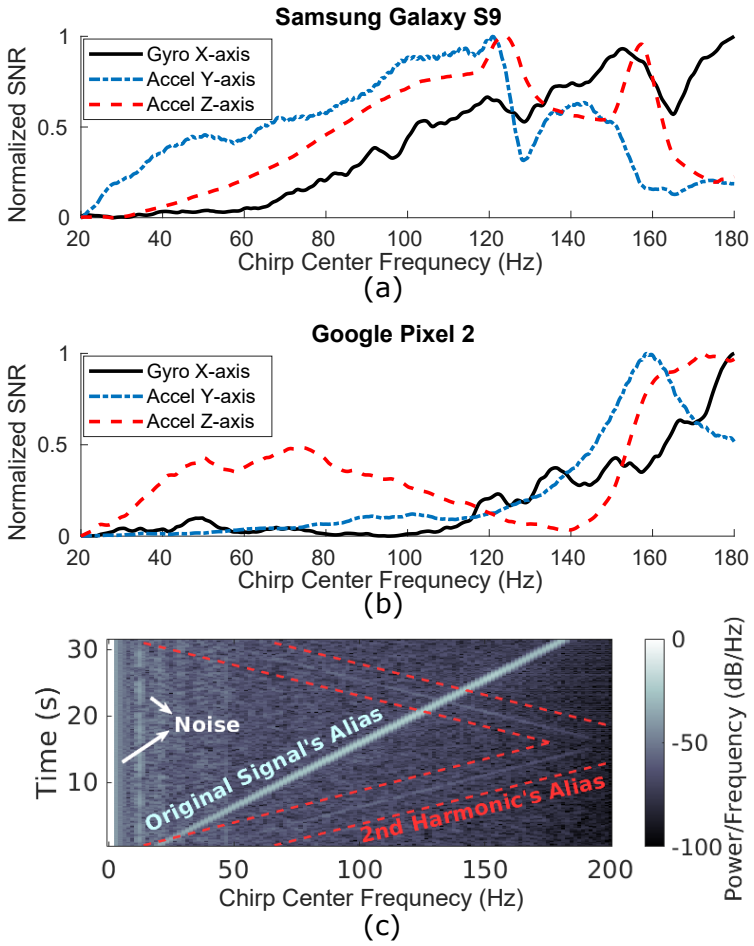


Figure 3.18: Touchtone information manifestations. Touchtone information can be manifested in a variety of forms or to varying extents in motion sensor data. In (a) and (b), two axes have distinct non-linear frequency responses to a 420 Hz to 580 Hz chirp from the speaker of smartphones. Different axes may thus be better predictors for certain tones. (c) shows how there may be many subtle artifacts in touchtone data. An attacker could use any of these artifacts to perform touchtone eavesdropping.

classifying touchtones (and thereby user input) and an attacker may need only one of these manifestations in some cases to determine the touchtones.

First, different types of sensors or different axes of a single sensor can contain complementary or different information about the same set of touchtones. One factor that can affect how information manifests is the varying frequency responses in phone mechanical construction, speakers, sensors, or even individual axes of sensors. Different frequency responses inherent to physical materials and sensors can lead to one sensor axis having a higher signal-to-noise ratio (SNR) for certain frequencies (i.e. touchtones) where a separate axis could have a higher SNR for other frequencies [65, 99, 209], as shown in Fig 3.18a. With access to

readings of all sensor axes, an adversary may be able to exploit this fact and combine useful information.

Additionally, even in the same sensor axis, information about the same touchtone can manifest in different manners. For example, an axis will have information on an alias of the touchtone frequency, but could also have information on the harmonics of the same touchtone as shown in Fig 3.18b. A touchtone eavesdropping attack would only need to recognize one of a touchtone's alias, harmonic, or even an alias of the harmonic to be successful.

3.9.2 Experiments

3.9.2.1 Setup

We have three different hardware setups for motion sensor data collection. The first two setups collect data from the four Android phones listed in Table 3.4 for baseline (no mitigation) and software-only mitigation evaluation; these two setups differ only in physical locations: a quieter conference room versus a noisy server room. The conference room was next to a busy atrium with the door closed to mimic a conference call setting, while the server room was chosen to mimic a noisy environment measured at an average of 67 dB SPL as measured by a General DSM403SD sound level meter[227]. Each setup used an Intel NUC running Ubuntu 18.04 [126] as a base station, smart-phones (Table 3.4), cables, and base station peripherals on a table (Figure 3.19). In this setup, the acoustic speaker was a phone's loudspeaker and the motion sensors (accelerometer and gyroscope) were the same phone's sensors. The base station used a python API for the Android Debug Bridge [91] to upload a custom Android data collection program to each phone and for other communications.

The third hardware setup collects data at faster sampling rates for software anti-aliasing filters and for testing onboard hardware anti-aliasing filtering. Phone hardware can collect at rates faster than what is made available to applications in smartphones to limit power consumption. Although current smartphone software API does not support it, we test it with external sensors to emulate possible future mitigation. To that end, our setup uses an LSM9DS1 breakout board, a very similar chip to the ones in three of the phones (Table 3.4), a Teensy 3.6 micro-controller, the same Intel NUC base station as in the previous setup, and an external speaker connected to the NUC to produce audio. The speaker was placed 10cm away from the LSM9DS1 breakout board. A Python program was used to produce audio on the speaker and interface with a custom sensor collection program on the Teensy micro-controller.

Table 3.4: Motion sensor information for tested phones.

Phone Model	IMU Model	Sampling Rate (Hz)	
		Reported	Measured
Google Pixel 1	BMI160	400.00	401.69
Google Pixel 2	LSM6DSM	400.00	409.96
Samsung Galaxy S8	LSM6DSL	400.00	429.27
Samsung Galaxy S9	LSM6DSL	415.97	413.61

Reported inertial measurement unit (IMU) model, which contains both an accelerometer and gyroscope, and sampling rates.

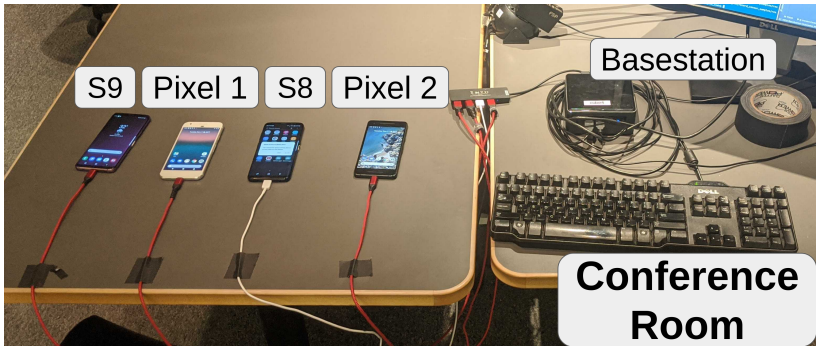


Figure 3.19: Data collection setup in a conference room.

3.9.2.2 Sensor Data Recording

To reduce temporally correlated biases from data collection over a long period of time the *python3* program running on the base station first determines a randomized order for all audio samples to record. The program then ensures the proper setup of all devices for the experiment. It then has the speaker for the experiment play each touchtone audio clip in succession while recording motion sensor data. In the event with multiple devices connected to the base station, only one phone's speaker and sensor were used simultaneously. Motion sensor data was collected at the fastest available sampling rate and saved and sent back to the base station to save the recording to disk.

For each individual setup, we recorded the motion sensor data of the 12 touchtones in Figure 3.16. Each individual dial-tone sample was played for 0.5 s, with each tone being recorded 250 times per setting for a total of 3000 recordings. The data set was divided into training and test sets at 80% and 20% respectively. It was ensured that touchtones were divided equally during the split (e.g. in the test set there were 50 samples of each of 12 touchtones).

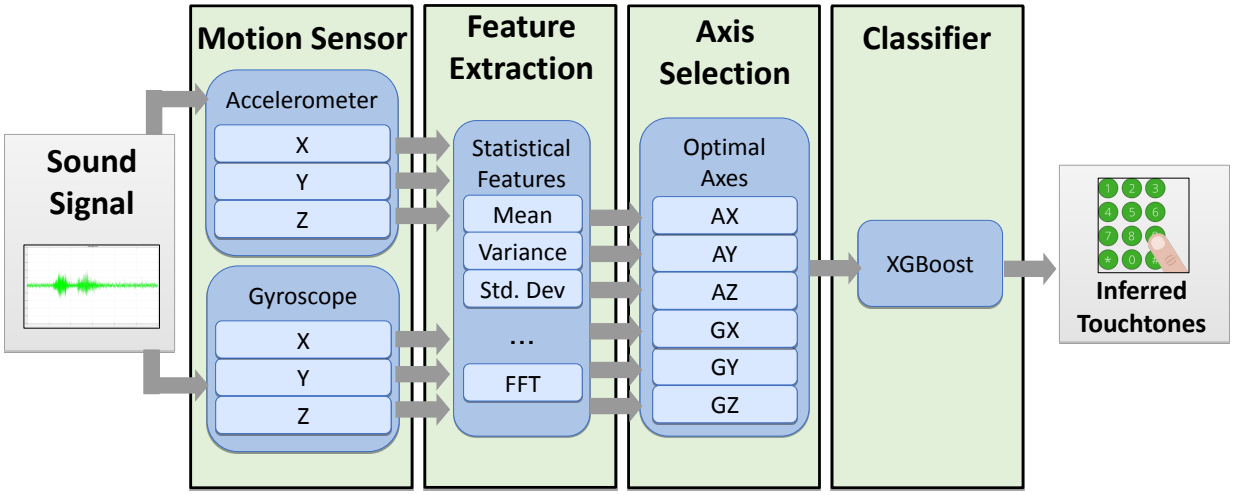


Figure 3.20: Eavesdropping classifier. Our system extract signal features and selectively integrate useful motion sensor data from multiple sensors and axes to better classify touchtones.

3.9.3 Touchtone Classifier

To serve as an evaluation metric we made a machine learning classifer (Figure 3.20) to mimic that of an advanced adversary.

3.9.3.1 Selective Integration of Sensor Data

To emulate a more advanced adversary, we build classifiers that selectively combines feature data from multiple sensors into a single attack model based on the intuition that each sensor axis can be a better or worse predictor for a given touchtone. Previous work has demonstrated classifiers for acoustic leakage onto motion sensor [171, 256, 44], however to our knowledge no previous work has combined data from both sensors simultaneously or selectively integrated axes into a single model. This improvement works as each axis from each sensor carries some measure of unique information. Selectively combining these sources of unique information should yield the best results.

Our method to selectively integrate axes is as follows. First, the system empirically ranks the axes in order of best predictor by building a model for each individual axis and tests its accuracy on validation data. Then the system builds a model with the most accurate two axes, then the top three, etc., until a model with all axes has been tested. Then the system selects the best-performing model among the single-axis and multi-axis models to use in actual testing. Once the best combination of axes has been chosen, the axes will be selected in the "Axis Selection" step shown in Figure 3.20.

Table 3.5: List of statistical features used in classification.

Mean	Median	Kurtosis	Absolute Area	% Mean Crossings
Minimum	Variance	Signal Power	Standard Deviation	Interquartile Range
Range	Maximum	Variation	Spectral Entropy	Fast Fourier Transform
Skew	First, Second, Third Quantiles			

The signal would be split into windows where the above features were calculated.

Table 3.6: Feature settings.

Feature	**Setting**	**Possible Choices**
Statistic	Frame Size (#vals)	10, 20, **50**, 100
	Frame Step (#vals)	**5**, 10, 20
Features MFCC	Window Length (s)	0.025, 0.05, 0.1, **0.2**, 0.3, 0.5
	Window Step (s)	**0.01**, 0.05, 0.01

The optimum feature settings used in the final model are in bold.

3.9.3.2 Features and Classifier Design

We briefly detail the feature extraction and classifier of our touchtone classifier in this section. As a reminder, features are calculated per sensor axis, then features of only the optimal combination of axes are included in the model as described in Section 3.9.3.1.

Time-alignment and Windowing: For feature extraction of a sample, our model first time-aligns signals from different sensors (i.e. sample 1 from one signal correlates with sample 1 of the others). Subsequently, it divides each time-series signal into a series of windows. Each window should correlate with windows of other signals (i.e. window 1 in one signal correlates with window 1 of another signal).

Extract Statistical Features: The system calculates a series of statistics per window per selected sensor axis and concatenates these metrics to produce a single feature vector. The set of statistical measurements, as shown in Table 3.5, are very similar to those used in previous work [44].

Zero-padding: Feature vectors with a different number of time windows, which may happen due to experimental error, must have the same number of features for the classifier to compare properly. The system zero-pads each feature vector to ensure the same length.

XGBoost Classifier: Our system uses *XGBoost* to classify the extracted features from the selected axes. XGBoost is a common classifier that uses gradient boosting and has been shown to effective in several different applications [73].

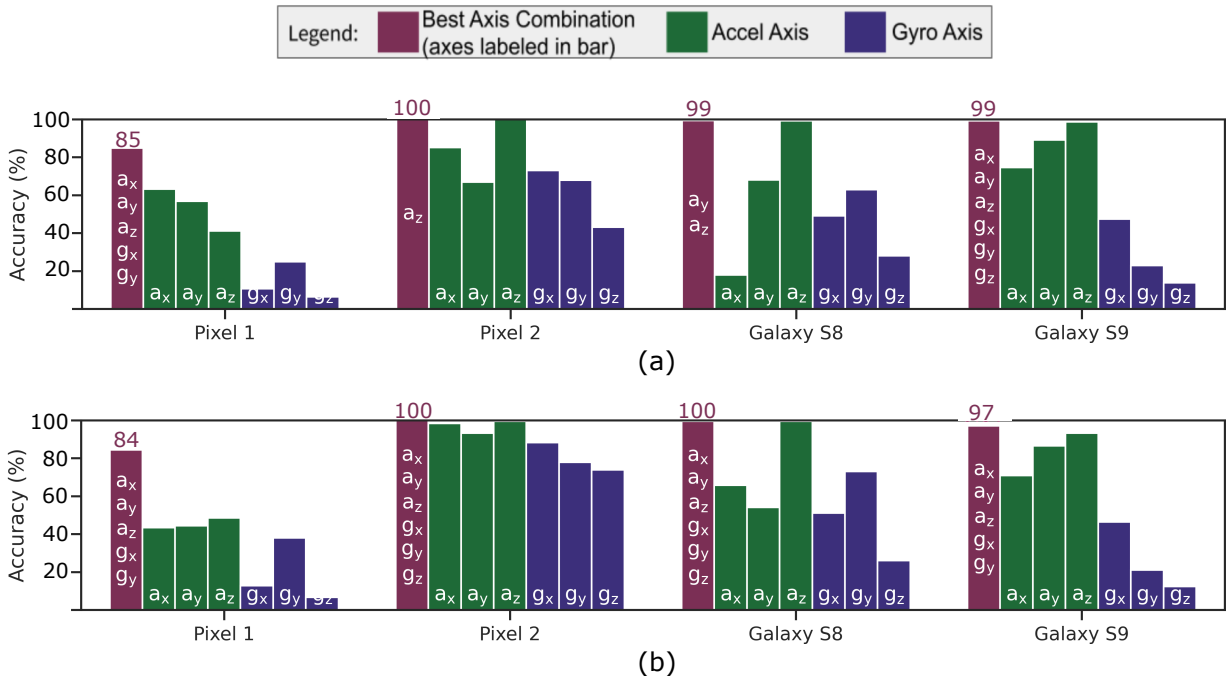


Figure 3.21: Baseline results for the touchtone eavesdropper without any mitigation. (a) Conference room and (b) Server room hardware setups. For each phone, we show the accuracy of classification models trained on individual axes alone, then show the accuracy for the model trained on the optimal combination of axes.

Table 3.7: Classifier settings.

Classifier	Setting	Possible Choices
XGBoost	learning rate	0.05, 0.10, 0.15, **0.20**, 0.25, 0.30
	max depth	3, 4, **5**, 6, 8, 10, 12, 15
	min child weight	1, **3**, 5, 7
	gamma	0.0, **0.1**, 0.2 , 0.3, 0.4
	colsample bytree	0.3, 0.4, **0.5** , 0.7
Random Forest	bootstrap	True, **False**
	max depth	30, 40, 50, 60, 70, **80**, 90, 100, None
	min samples leaf	1, **2**, 4
	min samples split	**2**, 5, 10
	n-estimators	200, 400, 600, 800, 1000, 1200, 1400 **1600**, 1800, 2000

The optimum feature settings used in the final model are in bold.

3.9.3.3 Implementation Details

The system uses a *python3* program to process the sensor recordings and subsequently train and/or test recognition models. We utilize numpy, scipy, and other standard *python3*

libraries to perform feature extraction as described previously. The system then uses *python3* XGBoost implementation with support libraries from Scikit-learn to perform any training, validation, or testing of machine learning models. To select the optimal combination of axes as described previously, the system would first train separate models for individual axis. These axes would then be ranked by individual accuracy performance and axes would be added in order of highest accuracy and evaluated. Last, for these eleven combinations (6 individual and 5 multi-axis) the system would choose the best-performing axis combination and use that for its model.

To choose specific features and model hyper-parameters, we performed a randomized grid search using data collected from a Pixel 2 phone in a conference room to pick parameters. The randomized grid search did not test every possible combination of parameters in the interest of time, and thus it is possible more optimal parameters could be chosen. The possible parameters for features and classifiers are shown in Tables 3.6 and 3.7 respectively with the best, selected parameters shown in bold. We tested these settings against a commonly used feature set for audio classification with Mel Frequency Cepstral Coefficients (MFCCs) [171] and another common classifier with Random Forest [44] to provide a comparison against other commonly used selections. We found that the XGBoost model with the statistic features constantly outperforms the other classifier-feature combinations. We took the highest accuracy result to select feature and classifier settings. These settings stayed the same through all testing.

Specifically, we used the statistical features in Table 3.6, which are calculated with a window size of 50 sensor reading samples and a step of 5 samples. For the XGBoost classifier, it uses a learning rate of 0.2, a max depth of 5, a min child weight of 3, a gamma value of 0.1, and a colsample_bytree value of 0.5. Based on the assumption that the adversary knows the model of the victim's phone and can acquire a duplicate device in advance, we train the classifiers on the data collected with the same phone as the test data to evaluate the upper limits of the recognition accuracies. On average, it takes less than 0.02 seconds to classify an eavesdropped touchtone.

3.9.4 Evaluation Results and Analysis

In this section, we report the attack and mitigation results with the setups described in Section 3.9.2.1. We analyze and summarize the findings of our assessment of the eavesdropping attack and different mitigations. Software low-pass filtering and reducing the sensor sampling rate can only moderately mitigate the attack while significantly hindering data bandwidth (and thereby application functionality). Software and hardware digital anti-aliasing filters

cannot eliminate touchtone eavesdropping but are able to significantly mitigate the threat while also preserving more data bandwidth.

We find that the unmitigated touchtone classifier achieves accuracy exceeding 99% for three of the four phones as shown in Figure 3.21, demonstrating that malicious applications can effectively recover user input.

3.9.4.1 Differences Between Phone Models

One of the phones, the Pixel 1, performs poorest in nearly every test despite similar sampling rates as the other phones. The highest touchtone inference accuracy for Pixel 1 does not exceed 85% while other phones can all achieve over 99%. As shown in Table 3.4, we notice that Pixel 1's IMU is produced by a different manufacturer than the other three phones. This result demonstrates that factors other than sampling rates can vary recognition rates. These factors could include signal propagation path that attenuates the acoustic signal, less sensitive sensors, different frequency responses, or different sensor configurations and MEMS structures. This result also suggests that some motion sensors may be more resistant to touchtone leakage than others. We believe a dedicated future study examining which motion sensors are less susceptible could provide insight into future hardware-based mitigations.

3.9.4.2 Accelerometer vs. Gyroscope Axis Accuracies

Classification based on data from an accelerometer axis achieved higher average accuracy than gyroscope axis data. While the exact reasons remain unclear, we provide a possible assumption. Accelerometers measure linear acceleration while gyroscopes measure angular acceleration. The phone's speakers produce audio through vibration, and then vibration travels through the phone body to affect both the accelerometers and gyroscopes. Vibration acts as linear acceleration in this case, which the accelerometer is designed to measure. While the gyroscope is not designed to measure linear acceleration, its sensing mass(es) still vibrate and these vibrations are quantized. Thus, the intent of each sensor changes the effectiveness of this particular scenario.

3.9.4.3 Selective Combination of Sensor Axes

The selective combination of axis data achieved significantly higher results for one phone model, the Google Pixel 1, and improved accuracy versus a single axis for all but one case. This exception case was the test for the Google Pixel 2 in the conference room, and it could not improve accuracy as accuracy was already 100%. For all phones but the Pixel 1, the improvement was limited because the results were already near 100% accuracy. However, for

the Pixel 1 the selective-axis integration improved as much as 40% over single-axis accuracies. This indicates that in cases with noisier data, the selective axis integration could help a classifier model utilize the various touchtone information in each axis to achieve higher accuracies.

3.10 Conclusion

These two examples provide evidence to support hypothesis **H1**, showing how the increasing resolution and sensitivity of various types of sensors increase the range of $s_{sec} \cap (s_{int} \cup s_{side})$, leaking increasingly more diverse and unexpected secret information to untrusted parties potentially acting as adversaries. Furthermore, the information leakage problem through zero-permission motion sensors explicitly shows that the requirement **KR1** is not met in many popular consumer-orientated IoT platforms. This chapter also investigates several different methodologies of mitigation, demonstrating the importance of not unwittingly providing the highest resolution of sensor data to software applications. Finally, the problems also point out the challenge of clearly defining the boundaries between trusted and untrusted parties, and the boundaries between intended and unintended inputs.

CHAPTER 4

Information Leakage Due to Increasing Sensor Structural Complexity

4.1 Overview

Besides the resolution and sensitivity of sensors, there are other more implicit factors such as complex hardware structures that could contribute to more information acquired by adversaries. Still centered around hypothesis **H1**, this section uses camera sensing examples to introduce two such factors, namely the movable lens and rolling shutter of modern cameras. While they were originally implemented to support useful functionalities that can benefit users such as long exposure time and optical image stabilization, they can actually be exploited to steal information in a security context. More importantly, we find that these features enable adversaries to use cameras to sense audio—another modality of physical information, revealing how the increasing complexity of sensing systems potentially blurs the boundary of sensing modalities [163].

4.2 Threats of Smartphone Cameras Near Audio

Smartphone and Internet of Things (IoT) cameras are increasingly omnipresent near sensitive conversations even in private spaces. Our work introduces the problem of how to prevent the extraction of acoustic information that is unwittingly modulated onto image streams from smartphone cameras. We center our analysis on a discovered point-of-view (POV) optical-acoustic side channel that leverages unmodified smartphone camera hardware to recover acoustic information from compromised image streams. The side channel requires access to an image stream from a smartphone camera whose lens is near the eavesdropped acoustic source emitting structure-borne sound waves. The key technical challenge is how to characterize the limit of partial acoustic information leakage from humanly imperceptible

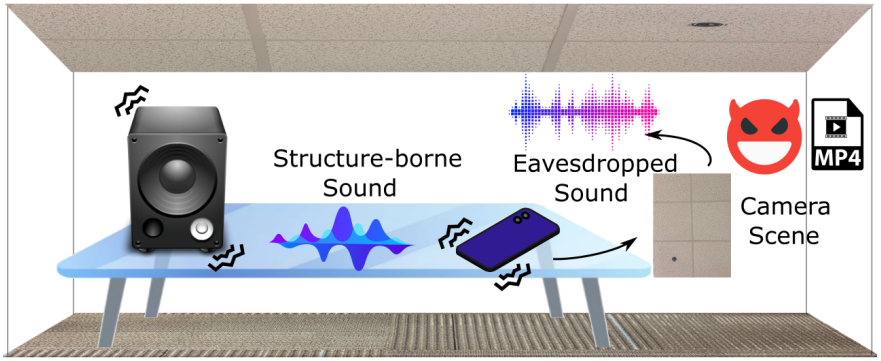


Figure 4.1: Illustration of the POV optical-acoustic side channel when a camera is recording a ceiling or floor. Adversaries can eavesdrop structure-borne sounds emitted by electronic speakers by extracting acoustic signals from artifacts of lens movement and rolling shutter patterns in smartphone cameras that depend on POV rather than objects in the field of view.

image distortions, which is made possible by nearly universal movable lens hardware and CMOS rolling shutters that are sensitive to camera vibrations.

The most related body of research on optical-acoustic side channels involves recording videos of vibrating objects within the field of view with specialized, high-frame rate cameras [42, 85, 261, 254, 255]. However, innovations in privacy-aware camera systems and software can actively detect and hide sensitive objects in camera images to prevent such direct data leakage [240, 83, 232]. In contrast, our work explores the optical-acoustic side channel intrinsic to existing smartphone camera hardware itself, eliminating the need for objects in the field of view or line of sight: an image stream of a ceiling suffices (Figure 4.1). That is, we extract acoustic information from the vibratory behavior of the built-in camera—rather than the behavior of a vibrating object within the field of view of a specially mounted camera.

Our threat model and approach build upon previous research that used smartphone motion sensors for acoustic eavesdropping [171, 43, 51, 45, 63, 216], where structure-borne sound emitted by electronic speakers vibrates motion sensors and also leaks acoustic information. However, cameras do not directly encode acoustics like motion sensors. Instead, our work must demodulate acoustic information unwittingly encoded within image stream artifacts. Assessing the limits of information recovery with this optical-acoustic side channel thus poses the challenge of designing a signal processing pipeline that optimizes (1) the acoustic signal extraction from images and (2) the effective utilization of extracted signals. To tackle the first challenge, we characterize the side channel's signal path and model the rolling shutter pattern formation under sound wave motions as a signal modulation process. Our modeling

reveals the limits of recoverable signal posed by factors such as imaging exposure time that can be optimized. It also reveals the theoretical signal extraction process, which guides us to design a diffusion registration-based extraction algorithm that rapidly and robustly recovers sound signals. Our recovered signals[1] with mainstream smartphones preserve over 600 Hz bandwidth of speech spectrum.

To tackle the second challenge, we observe that the extracted band-limited signals are complex transformations of the original sound and thus difficult for humans to recognize directly. In order to fully utilize the information embedded in the extracted signals, we design a classification model based on the HuBERT Large transformer [122]. Our extensive evaluation with 10 smartphones on a widely used spoken digit dataset [57] suggests that this optical-acoustic side channel powered by our signal processing pipeline allows adversaries to recover acoustic information from the surroundings. Specifically, we observed 80.66% accuracy on speaker-independent 10-digit recognition, 91.28% accuracy on recognizing 20 speakers, and 99.67% on gender recognition when a Google Pixel 3 phone was placed beside a speaker on a desk. In addition to classification, we also used NIST-SNR and Short-Time Objective Intelligibility (STOI) metrics to measure the quality and intelligibility of recovered speech signals, and observed scores up to 28 dB and 0.53, respectively. We further evaluated this side channel's robustness with various speaker volumes and speaker-phone distances as well as its applicability in different structure-borne propagation scenarios, including when the phone and speaker are placed on different desks or in different rooms.

Finally, we systematically investigate the possible defenses from the standpoints of user-based countermeasures and future camera design improvement respectively. For the latter, we propose corresponding hardware modifications to mitigate the two enabling factors of this attack, namely rolling shutter and movable lens. To summarize, the goal of this work is to model, measure, and demonstrate the capability of the POV optical-acoustic side channel on smartphone cameras and help defend against the threat on current and future camera devices.

4.2.1 Related Work

Sound Recovery From Vibrating Objects In Videos. The concept of recovering sound by analyzing vibrating objects in video frames was first introduced by Akutsu et al. [42] in 2013 where they used high-speed cameras (over 6,000 fps) to record the movements of a speaker's face and neck. Davis et al. [85] found it is possible to recover speech by aiming a specialized high frame-rate camera at lightweight objects (e.g., plastic bags) vibrated by

[1]Sample audio and additional materials can be found on our project website https://sideeyeattack.github.io/Website/

sound waves. Follow-up research on this topic mainly focused on improving the efficiency of sound recovery based on Davis's technique using specialized high frame-rate cameras [261, 254, 255]. Some works also discussed the possibility of utilizing the rolling shutter effect to emulate higher frame rates with common cameras, but the discussions remain proof-of-concept in lab settings as it requires a high-end camera on a tripod to focus precisely on the lightweight objects at a very close distance [85, 252, 101, 206]. In comparison, our work exploits the rolling shutter artifacts caused by the movement of smartphone camera lenses that are intrinsic to existing smartphone camera hardware itself. This feature allows our optical-acoustic side channel to work without any vibrating object in the camera's field of view and enabled us to evaluate a wide range of possible sound recovery scenarios including when the speaker and camera are in two different rooms (Section 4.6.2). Furthermore, previous works' recovered signal amplitude is proportional to the lens focal length due to their need of objects in the video frames, which poses the major limitation of requiring short camera-object distance or expensive optics [85]. In comparison, our work addresses this limitation by exploiting the movable lens structure on smartphone cameras as a signal amplifier under structure-borne sound.

Smartphone Motion Sensor Side Channels. In 2014, Gyrophone [171] first proposed the idea of using gyroscopes on smartphones for acoustic eavesdropping. They investigated a structure-borne attack scenario where the smartphone and electronic speaker are on a shared table surface. Following works such as AccelEve [51], Spearphone [45], and [63] proposed a structure-borne threat model of eavesdropping audio played by the smartphone's built-in electronic speakers with accelerometers on the same phone, which is similar to our same-phone scenario evaluated in Section 4.6.1.

Compared to motion sensor side channels, the optical-acoustic side channel proposed in this work opens up a new modality of smartphone acoustic eavesdropping since cameras create an orthogonal space of threat models in cases where motion sensor data is not available or add to the total information extracted when it coexists with motion sensors. Camera side channels provide a high bandwidth while motion sensors often have better sensitivity to vibrations. In addition to the shared surface-coupling and phone body-coupling scenarios, our paper further investigates new scenarios where smartphones are on different surfaces than the speakers such as on a different desk, in a shirt pocket, in a bag, or even in a different room. It is worth pointing out that comparisons between these motion sensor side channels' results and our results may not provide meaningful insights due to the large differences in their threat models, algorithms, evaluation setups, etc.

Physical Acoustic Eavesdropping. Researchers also exploited other physical mechanisms for acoustic eavesdropping. We refer the readers to the SoK paper by Walker et al.

[238] for a relatively comprehensive review. Lamphone [178] and the little seal bug [179] use telescopes and optical sensors to sense the optical changes caused by sound-induced object vibrations remotely. Glowworm [177] finds that the LED light intensity of electronic speakers leaks acoustic information and uses telescopes and photodiodes to eavesdrop on it. Compared to these works, our work does not require specialized devices and light but uses smartphones in private spaces for eavesdropping. LidarPhone [198] inherits the well-studied concept of sound laser vibrometry but uses malware to exploit the lidar sensors on robot vacuum cleaners for eavesdropping. Hard drive of hearing [150] discovers that the read/write head of hard drives can be turned into unintentional microphones for eavesdropping when the head is vibrated by loud sounds.

Camera-based Attacks. Poltergeist [130] by Ji et al. studied the robustness problem of the camera OIS from an almost complementary perspective to our work. They discovered that adversaries can generate intentional ultrasounds to change the gyroscope readings of OIS in similar ways as explored in [209, 228] and thus cause controlled motion blurs in the camera videos to attack computer vision-based autonomous vehicles. They See Me Rollin' by Köhler et al. [142] studied laser-based optical injection attacks against CMOS cameras in autonomous vehicles. They exploited the rolling shutter mechanism of CMOS cameras to inject row-wise fine-grained disruption patterns into camera videos that could hide up to 75% of objects perceived by state-of-the-art computer vision object detectors. While their work studies how rolling shutters can cause robustness and security problems to downstream processing units when subjected to active optical injections, our work investigates how to passively recover ambient acoustic information from rolling shutter cameras vibrated by sound.

4.3 Threat Model & Background

4.3.1 Threat Model

We characterize the threat of POV acoustic information leakage into smartphone cameras through structure-borne sound propagation. The sound generated by a sound source in the vicinity of a camera propagates to the camera and vibrates it, inducing rolling shutter effects in the camera image stream. The rolling shutter pattern thus becomes a function of the acoustic signal. The objective of an adversary is to learn the reverse mapping from the rolling shutter pattern to the privacy-sensitive information in the acoustic signal. Formally,

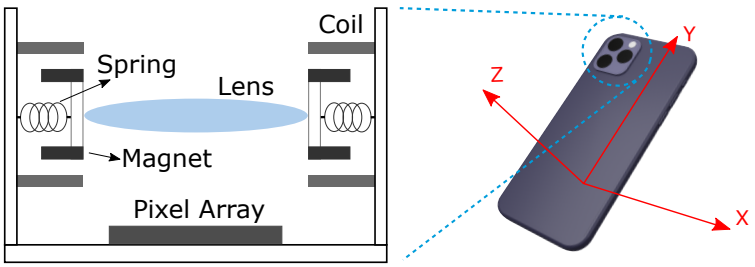


Figure 4.2: The movable lens structure widely exists in smartphone cameras with optical image stabilization (OIS) and auto-focus (AF). When sound waves move the camera lens suspended on the springs, the optical path changes and creates an optical-acoustic side channel.

we define the eavesdropping attack that an adversary $\mathcal{A}$ launches as a function $f_{\mathcal{A}}$:

$$f_{\mathcal{A}}: \quad \left\{P_v(S^l(t), \mathbb{E}),\ \mathbb{E}_{\mathcal{A}}\right\} \longrightarrow \tilde{l}, \quad \tilde{l} \in \mathbb{L}$$

where $S^l(t)$ is the continuous-time acoustic signal generated by the sound source; $l, \tilde{l} \in \mathbb{L}$ are the true and estimated information label of the acoustic signal; $\mathbb{L}$ is the set of all possible information labels and is reasonably assumed to be finite; $\mathbb{E}, \mathbb{E}_{\mathcal{A}}$ are the sets of environmental factors that are present during the attack (e.g., phone-speaker distance) and that are controlled or known by the adversary respectively, and have $\mathbb{E} \supseteq \mathbb{E}_{\mathcal{A}}$; $P_v(\cdot)$ denotes the projection from the acoustic signal to the videos containing the rolling shutter pattern. To measure the threat, we define the advantage of an adversary over random-guess adversaries as a probability margin

$$\mathbf{Adv}_{\mathcal{A}} = \mathbb{P}\left[f_{\mathcal{A}}(P_v(S^l(t), \mathbb{E}),\ \mathbb{E}_{\mathcal{A}}) - l < \epsilon\right] - \frac{1}{|\mathbb{L}|} \tag{4.1}$$

where ϵ is an arbitrarily small number. A successful attack is defined as $\mathbf{Adv}_{\mathcal{A}} > 0$. Although $\mathbf{Adv}_{\mathcal{A}}$ is a theoretical value that requires knowing the probability distributions and functions in Equation 4.1 to calculate, we can estimate this value by obtaining classification accuracies on datasets with equally likely labels as the ones in Section 4.6.

Targeted Information Recovery. We focus on recovering information from human speech signals broadcast by electronic speakers, as this is one of the most widely investigated threat models validated by previous research [171, 43]. In particular, our study investigates the feasibility and limit of recovering acoustic information from smartphone cameras without requiring microphone access. To better assess the limit, we allow the adversary to utilize state-of-the-art signal processing and machine learning techniques. We discuss three types of information recovery with increasing difficulty, namely (1) inferring the human speaker's

gender, (2) inferring the speaker's identity, and (3) inferring the speech contents.

Adversary Characteristics. We consider an adversary in the form of a malicious app on the smartphone that has access to the camera but cannot access audio input from microphones. In common mobile platforms including Android and iOS, the app will have full control over the camera imaging parameters such as focus and exposure controls once the camera access is granted. An adversary can change these parameters for optimal acoustic signal recovery based on their knowledge of the signal modulation process. We assume the adversary captures a video with the victim's camera while the acoustic signal is being broadcast. We also assume the adversary can acquire speech samples of the target human speakers beforehand to learn the reverse mapping to the targeted functions of the original speech signals and they can perform this learning process offline in a lab environment, which have been the standard assumptions in related side-channel research [171, 51, 45].

Attack Scenario. Sounds broadcast by an electronic speaker can reach a smartphone's camera through structure-borne propagation when there exists a propagation path consisting of a single structure or a system of structures such as tables, floors, and even human body. Such a structure-borne model has been frequently used in previous works [171, 43, 239] of smartphone acoustic eavesdropping. Similar to previous works of motion sensor side channels, the malicious app eavesdrops on acoustic information under the general user expectation that no information can be stolen through sound when smartphone microphone access is disabled. Although camera access is usually regarded as being on the same privacy level as microphone access, users aware of the risk of acoustic leakage through microphones are still likely to grant camera access to apps until they realize the existence of the optical-acoustic side channel. We believe this can happen in three major situations. (1) The malicious app requests only camera access without microphone usage in the first place. Apps can disguise themselves as hardware information checking utilities (e.g., the widely used "AIDA64" app [26]) or silent video recording apps that do not record any audio. (2) The malicious app requests both camera and microphone access but a cautious user only grants camera access. We found that filming apps (e.g., the "Open Camera" [36] and "Mideo" [35]) often simply record without audio when microphone access is not granted. (3) The malicious app requests and is granted both camera and microphone access, but a user physically disables the microphone input by using external gadgets such as the Mic-lock microphone blocker [34]. Additionally, malicious apps can record videos stealthily without camera preview or in the background as has been done by existing apps like the "Background Video Recorder" on the Google Play Store [28] and "SP Camera" on the Apple App Store [38].

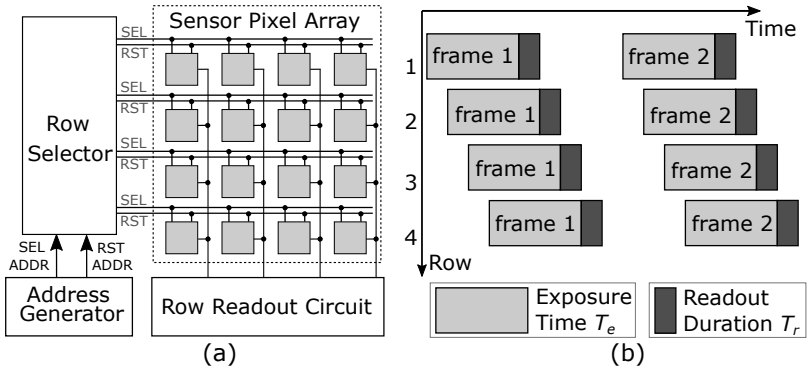


Figure 4.3: (a) CMOS rolling shutter camera's row-wise sampling architecture with a 4×4 sensor pixel array. (b) The sequential readout of rows for two consecutive frames with exposure time T_e and row readout duration T_r.

4.3.2 Rolling Shutter Cameras

Rolling shutter cameras, which feature a *row-wise sampling architecture* (Figure 4.3 (a)), dominate the market of portable electronics including smartphones. Row-wise sampling overlaps the exposure of one row with the read-out of subsequent rows (Figure 4.3 (b)). An address generator controls this process by generating row-reset (RST) and row-select (SEL) signals that start the exposure and read-out of each row respectively. The interval between the two signals is the exposure time T_e. The duration of each row's read-out is denoted as T_r. The row-wise sampling architecture comes at the cost of additional image distortions when the optical paths change while imaging a scene. The optical paths can change when a relative movement happens between the scene, the lens, and the pixel array. The rolling shutter distortions are thus a function of optical path variations.

4.3.3 Movable Lens

While the CMOS photo-sensitive pixel array is mounted on printed circuit boards (PCB) and rigidly connected to the camera body, the lens in most modern CMOS cameras is flexibly connected to the camera body by suspension structures using springs and specialized wires [211]. Such suspension structures allow relative movement between the lens and the pixel array, as shown in Figure 4.2. The movable lens is an essential component of cameras' optical image stabilization (OIS) and auto-focus (AF) systems and is almost ubiquitous in hand-held camera devices including smartphone cameras.

Optical Image Stabilization: OIS is an image stabilization method for mitigating tremor-caused motion blurs. Most OIS systems allow for 2D movements of the lens that are parallel to the pixel array plane, resulting in translational transformation of images. We

only consider 2-DoF OIS movements and term such movements as XY-axes movements. OIS lens stroke is on the order of 100 µm [202].

Auto-focus: Most AF systems support 1-DoF movements of the lens on the axis that is perpendicular to the pixel array plane, which we term as Z-axis movements. Such movements can induce zooming effects that can be viewed as scaling transformations of the 2D image. AF lens stroke is also on the order of 100 µm [104].

Sound Propagation. This work investigates the consequences of movable lenses vibrated by structure-borne sound waves. Sound waves can propagate both through the air by inducing movements of air molecules, and through structures by inducing mechanical deformations in them. Structure-borne propagation can often transmit much higher sound energy than air-borne propagation [81]. In 2018, Anand et al. systematically analyzed the response of smartphone motion sensors to air-borne and structure-borne sound waves [43]. Their experiments show that structure-borne sound generated by electronic speakers causes stronger vibrations of the sensors and thus enables more feasible eavesdropping with motion sensors. Building upon their results, our work explores how structure-borne sounds can affect smartphone cameras.

4.4 Modeling Acoustic Eavesdropping in Cameras

In this section, we seek to answer three key questions regarding the feasibility of the side channel: (1) Why does such a side channel happen? (2) What are the factors deciding the channel's capability? (3) How can adversaries extract high-quality signals from the channel?

4.4.1 Signal Path Causality

Mechanical Subpath. When the electronic speaker on a table plays audio with total kinetic energy E_s, part of the kinetic energy it generates $k_0 E_s$ propagates to the body of the phone in the form of structure-borne sound waves and vibrates the smartphone body. Specifically, longitude waves mainly cause XY-axes motions of the smartphone body while transverse and bending waves mainly cause Z-axis motions [81]. The smartphone body and the camera body, including the sensor pixel array, are rigidly connected and thus have the same motion amplitude and velocity. Viewing them as a single unit separated from the camera lens, we denote the kinetic energy causing vibrations of this unit as E_p. We can approximately model this unit's motions on the table as a spring-mass system [231] with a spring constant c_p and motion amplitude A_p. The camera lens is connected to the camera body through springs and can thus be regarded as a second spring-mass system. A portion

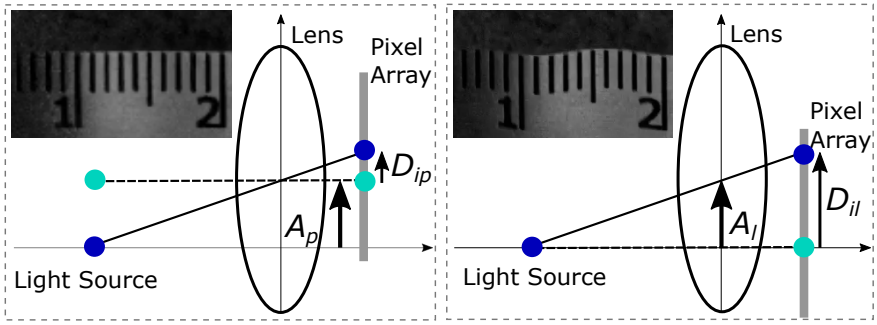


Figure 4.4: The movable lens structure acts as a signal amplifier when structure-borne sound vibrates the smartphone camera. The dotted and solid lines represent the light ray projected before and after vibration. (Left) Without moving lenses, the rolling shutter pattern induces negligible pixel displacements. (Right) When lenses move, pixel displacements get amplified.

of E_p, denoted as $k_1 E_p$, is converted to its elastic potential energy by stretching/compressing the springs. Denote the effective spring constant of the lens suspension system as c_l and the relative motion amplitude between the lens and the smartphone-camera unit as A_l ($A_l < A_p$), we then have

$$k_0 E_s = E_p = \frac{1}{2} c_p A_p^2 = \frac{1}{k_1} \frac{1}{2} c_l A_l^2 \tag{4.2}$$

Note that k_0 and k_1 are frequency-dependent and reflect the physical properties of the mechanical system consisting of the speaker, the table, and the phone. In other words, A_p and A_l can be expanded along the frequency axis to represent the frequency response (transfer function) of the mechanical subpath. Such frequency response is hard to model but can be measured in an end-to-end manner (Section 4.4.4).

Optical & Electronic Subpaths. The movements of the smartphone body and the lens change the optical paths in different ways. Figure 4.4 shows a simplified model of how the two types of movements on the X-axis affect the light ray from a still point source to the sensor pixel array. In Figure 4.4 (a), the smartphone-camera body unit moves by A_p while there exists no relative movement between it and the lens. With a focal length of f (on the order of 5 mm[2]) and a camera-scene distance of d, the light ray projection point on the pixel array shifts by $\frac{f}{d} A_p$. In Figure 4.4 (b), only the lens is moving by A_l while the smartphone-camera unit stays still. In this case, the projection point shifts by $(1 + \frac{f}{d}) A_l$. The optical projections are then sampled by the photo-sensitive pixel array and converted to digital signals, with the shifts of the projection point converted to pixel displacements in the images. Denote the general pixel displacement as D_i, the two types of movements will then result in pixel displacements of $D_{ip} = \frac{f}{d} \frac{A_p}{H} P$ and $D_{il} = (1 + \frac{f}{d}) \frac{A_l}{H} P$, where H and P are

[2]The commonly claimed focal lengths on the order of 20 mm are the values converted to the equivalent of a full-frame camera sensor instead of the true physical values.

the physical sizes and pixel resolution of the sensor pixel array on the X-axis respectively.

The interesting question arises as to whether D_{ip} or D_{il} is the main enabling factor of this side channel. Note that $\frac{f}{d}$ is very small since the camera-scene distance is usually larger than 10 cm. In light of this, we hypothesize D_{il} is the dominant factor assuming A_p and A_l, which cannot be measured directly, are on the same order of magnitude. We then verify our hypothesis experimentally by recording videos while preventing and allowing lens movements using a magnet. Figure 4.4 shows the significantly higher pixel displacement magnitudes when the lens is free to move under a 200 Hz sound wave. With a small distance d of 10 cm, we observed $D_{ip} < 1px$ and $D_{il} \approx 8px$, which translates to $A_p < 63\,\mu m$ and $A_l \approx 22\,\mu m$. We thus ascertain that the lens movement is the main cause of the noticeable pixel displacements in the images. In other words, *the movable lenses act as motion signal amplifiers compared to those cameras that can only move with the smartphone body.* In light of this finding, we model the displacement as a function of the lens movement as

$$D_i \approx D_{il} = (1 + \frac{f}{d})\frac{A_l}{H}P \tag{4.3}$$

4.4.2 Rolling Shutter Modulation

As pointed out in Section 4.3.3, multi-DoF motions of the lens will mainly cause translation and scaling 2D transformations in the image domain. With a rolling shutter, transformations caused by multiple motions will be combined into one image frame because of the row-wise sampling scheme, and consequently produce wobble patterns that can be viewed as the outcome of modulating vibration signals onto the image rows. Furthermore, motion blurs exist due to the finite (namely, not infinitely small) exposure time of each row. For example, Figure 4.5 (b) and (c) show the simulated rolling shutter image (250×250) with an exposure time of 1 ms when 500 Hz sinusoidal motion signals on the X and Z axis are modulated onto the image in Figure 4.5 (a) respectively. In light of these observations, we model the limits of acoustic signal recovery.

4.4.2.1 Imaging Process

We can model the imaging process of each row in a frame as a linear process where the final (row) image is the summation of different views that are 2D transformations of the original/initial view within the exposure time. The summation is actually the accumulation of photons on the CMOS imaging sensors. Consider frames of size M rows, N columns, and the simplest case where the motion only results in a uni-axis translation transformation on the column direction (X axis). We denote the i-th row of the initial view as a column

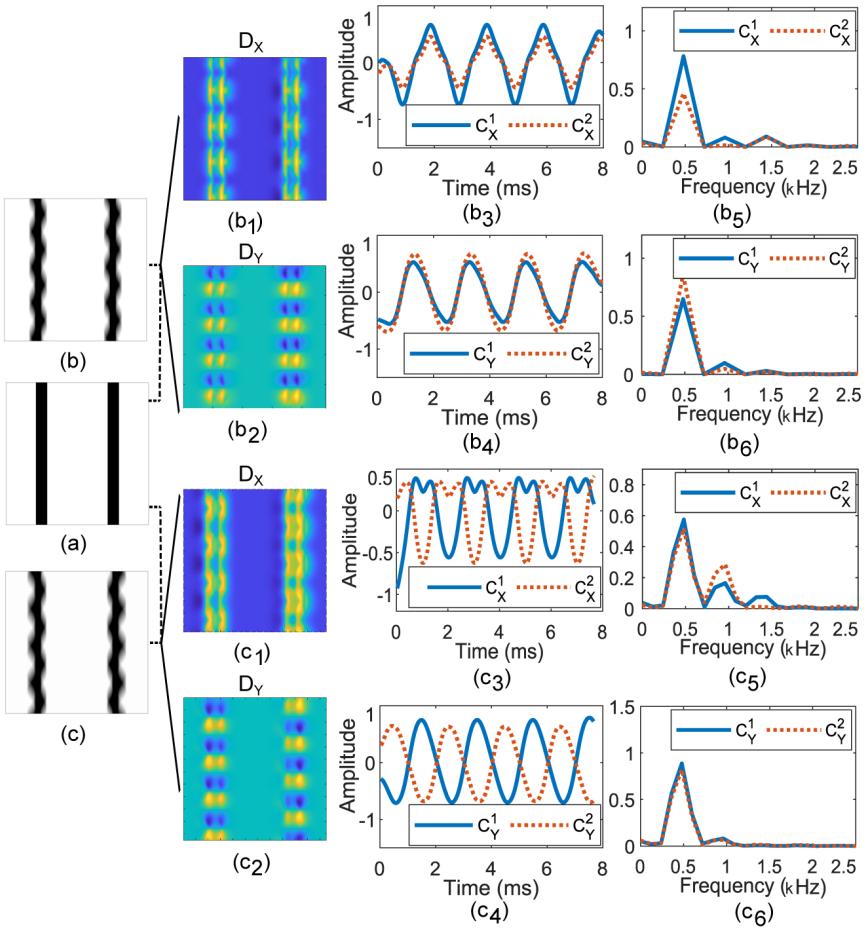


Figure 4.5: The simulated rolling shutter images under a 500 Hz sound wave and the extracted signals with diffusion-based image registration. (a) The original scene. (b, c) The scenes with X and Z-axis motions respectively. (b/$c_{1,2}$) The X and Y-direction displacement fields. (b/$c_{3,4}$) The time domain signals computed from displacement fields with column-wise channels. (b/$c_{5,6}$) The corresponding frequency domain signals.

vector $r(i)$, and the matrix formed by all the possible translated views of $r(i)$ as $R_i = \begin{bmatrix} ... & r^{j-1}(i) & r^j(i) & r^{j+1}(i) & ... \end{bmatrix}$. Theoretically, R_i has an infinite number of columns as the translation is spatially continuous. Considering a more practical discretized model, we let j correspond to the displacement value in pixels in the image domain. For example, $r^{-3}(i)$ denotes the view shifted to the reverse direction along the X-axis by 3 pixels. Allowing negative indexing to R_i for convenience and discretizing the continuous physical time with small steps of δ, the formation of the i-th row in the k-th image frame, which is denoted as $\widetilde{r}(k, i)$, can then be expressed as the summation of different columns of R_i:

$$\begin{cases} \widetilde{r}(k,i) = \sum_{n=n_{k,i}^{start}}^{n \leq n_{k,i}^{end}} R_i[:, s(n\delta)] \\ n_{k,i}^{start} = \frac{T_f^k + (i-1)T_r}{\delta} \\ n_{k,i}^{end} = \frac{T_f^k + (i-1)T_r + T_e}{\delta} \end{cases} \tag{4.4}$$

where T_f^k denotes the imaging start time of the frame and $s(n\delta)$ denotes the discrete motion signal with amplitude D_i (Eq. 4.3) in the image domain. Equation 4.4 shows how rolling shutter exposure modulates the signal onto the images' rows. The objective of the adversary is to recover $s(n\delta)$ from $\widetilde{r}(k,i)$.

4.4.2.2 Limits of Recovery

With the modeling above, we can compute the characteristics of the recoverable signals.

Captured Signal. Signals in time intervals $[n_{k,M}^{end}\delta, n_{k+1,1}^{start}\delta]$, i.e., the gap between different frames, cannot be recovered since no camera exposure happens then. We term this portion as the "lost signal" and the remaining portion as the "captured signal". We can calculate the percentage of the captured signal

$$\eta_{cap} = f_v M T_r \tag{4.5}$$

where f_v is the video frame rate. Higher η_{cap} means the adversary can recover more information from images.

Sample Rate & Bandwidth. For the captured signal, although the adversary wants to infer all the transformed views and thus recover all signals in time intervals $[n_{k,i}^{start}\delta, n_{k,i}^{end}\delta]$, it is impossible to know the order of these views' appearance because the photons from all the views are summed in the exposure time and the loss of order information is irreversible. Without the order information, the adversary can only reformulate Equation 4.4 as

$$\begin{cases} \widetilde{r}(k,i) = R_i x(i) \\ x(i)_j = \sum_{n=n_{k,i}^{start}}^{n \leq n_{k,i}^{end}} \mathcal{I}\left\{s(n\delta) == j\right\} \end{cases} \tag{4.6}$$

where $x(i)$ is a coefficient column vector whose j-th entry $x(i)_j$ represents how many times the translated view $r^j(i)$ appeared within the exposure time; $\mathcal{I}\{\cdot\}$ is the indicator function. Theoretically, with the measurable final image $\widetilde{r}(k,i)$ and the matrix R_i that can be approximately constructed using a still frame, $x(i)$ can be computed by solving the linear system in Equation 5.3. To recover a 1D motion signal that is a function of $s(n\delta)$, the adversary can estimate a synthetic motion data point $a(i)$ from $x(i)$ by taking the weighted average of

j with respect to $x(i)$:

$$a(i) = \frac{\sum_j j \times x(i)_j}{\sum_j x(i)_j} = \frac{1}{T_e/\delta} \sum_{n=n_{k,i}^{start}}^{n \leq n_{k,i}^{end}} s(n\delta) \tag{4.7}$$

The adversary-measurable signal $a(i)$ thus embeds the information of the original motion signal.

Based on Equations 4.4 and 4.7, we can conclude that the measurable signals extracted from the rolling shutter patterns have an effective sample rate of $1/T_r$. Equation 4.7 also shows that the sampling process from a motion-blurred image acts as a moving mean filter whose frequency response is determined by the exposure time T_e.

4.4.3 Motion Extraction Algorithm

Directly using Equation 4.7 for signal extraction faces three real-world challenges: (1) Solving the linear system of Equation 5.3 is computation-intensive. (2) The size of R_i increases exponentially as the motion's DoF increases. (3) Equation 5.3 is mostly underdetermined. We thus designed a motion signal extraction algorithm based on *diffusion-based image registration* [223, 236]. It takes in a reference image I_{ref} and a moving image I_{mov} of size $M \times N$ (number of rows and columns, e.g., 1080×1920), and outputs 2D displacement fields (matrices) for X and Y-direction displacements respectively, i.e., $D_X^{M \times N}$ and $D_Y^{M \times N}$. Figure 4.5 shows the raw displacement fields for (b) and (c). We further apply column-wise averaging to the matrices to reduce data dimensionality as well as the impact of random noise in the imaging process, which improves signal robustness. We assign columns to different groups and take group-wise averages on the X and Y displacement fields respectively. We empirically choose the number of groups n_g to be the nearest integer to $2N/M$ to balance the robustness and the details we want to preserve. After averaging, we reduce D_X and D_Y to $4N/M$ 1D signals of length M (number of rows), and we term each 1D signal as a channel. Let $dir \in \{X, Y\}$ and a^i denote the averaging column vector with its $j-$th entry denoted as a_j^i, the channels are then formally defined as

$$\begin{cases} C_{dir}^i = D_{dir} \cdot a^i, \quad i = 1, 2, \ldots, n_g \\ a_j^i = \frac{N}{n_g} \mathcal{I}\left\{\frac{n_g}{N}(i-1) < j \leq \frac{n_g}{N} i\right\} \end{cases}$$

For the 250×250 images in Figure 4.5, the 4 channels and their spectrums are shown in ($b_{3\text{-}6}$) and ($c_{3\text{-}6}$). When dealing with a video, i.e., a sequence of images, we use the first frame as I_{ref}, and concatenate consecutive frames' channels to get the channel signals of the

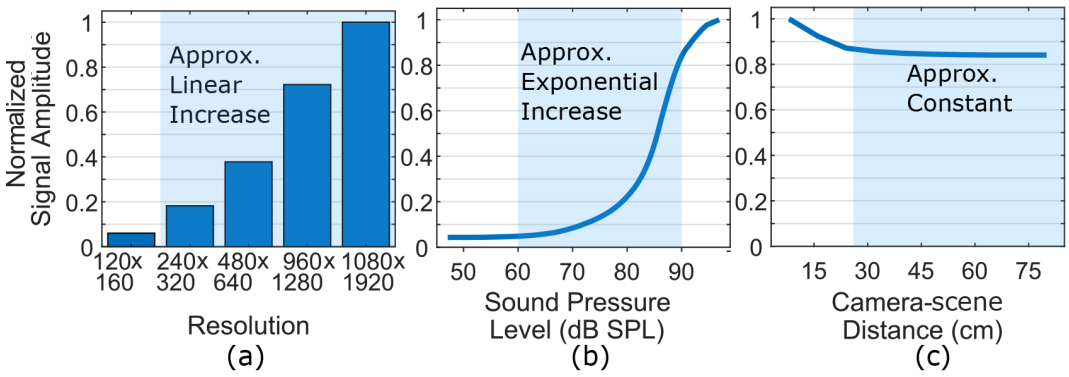


Figure 4.6: The relationship between signal amplitudes (normalized) and different factors. (a) Amplitude increases approximately linearly with video resolution. (b) Amplitude increases approximately exponentially with speaker volume. (c) Amplitude remains approximately constant as the camera-scene distance changes due to the movable lens structure.

video. We use the same notation C^i_{dir} to denote a video's channels.

4.4.4 Feasibility & Attack Characterization

Camera Scene. Most smartphones have both front and rear cameras. Although some smartphone manufacturers such as Vivo have started to equip their front cameras with OIS [24], we focus on rear cameras in this work since more of them are equipped with OIS and AF. The rear camera has a certain scene while imaging. The scene can affect information recovery because their structures, textures, and distance from the camera can modify the characteristics of the light rays entering the camera. The scene changes with the smartphone's placement and location. As depicted in Figure 4.1, a phone on a table with an upward-facing rear camera often records a scene of the ceiling ("Ceiling Scene"); a downward-facing camera on a non-opaque surface such as a glass table often records a scene of the floor ("Floor Scene"). For simplicity we assume there are no moving objects in the scene.

For our preliminary analysis, we use a test setup with a KRK Rokit 4 speaker and a Google Pixel 2 phone held by a flexible glass platform on a table with the phone's rear camera facing downwards to simulate a Floor Scene. We use a customized video recording app that acts as the malicious app to record in MP4 format. We first discuss the choice of adversary-controllable camera parameters and then discuss the environmental factors in order to characterize the envelope of the adversary's capability.

Camera Control Parameters. The frequency response of the side channel is determined by both the mechanical subpath and the camera control parameters of the malicious

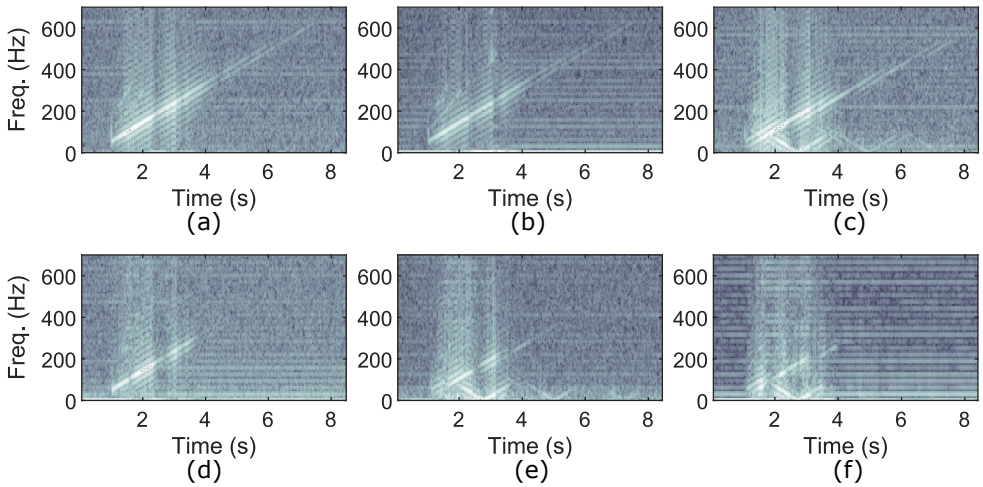


Figure 4.7: The recovered chirp signals (50-650 Hz in 7s) with different camera control parameters and a 30 fps frame rate. (a) Optimized parameters and 1 ms exposure time. (b) OIS is left on. (c) EIS is left on. (d) 10 ms exposure time. (e) OIS, EIS, AF are left on with 10 ms exposure time. (f) Recovered with the phone stock camera app without any optimization.

app that can be optimized by the adversary. We estimate the frequency response by conducting a frequency sweep test where we play the audio of a chirp from 50 to 650 Hz. We then aim to find the optimum response for our Google Pixel 2. Figure 4.7 (a) shows the best response where the maximum recovered chirp frequency is about 600 Hz. Specifically, we optimize the control parameters in the following ways: (1) Disable auto-exposure and reduce the exposure time (Section 4.4.2). (2) Disable optical and electronic image stabilization (OIS and EIS) and auto-focus (AF). (3) Minimize video codec compression. (4) Maximize pixel resolution. (5) Choose appropriate frame rates for each phone. Figure 4.7 (b-f) also show the responses when optimum settings are not achieved.

Configuration Factors. Variations of configuration factors can also affect the recoverable signals. We discuss the impact of three main factors: sound pressure, distance from the scene, and phone orientation.

(1) Sound pressure level. Louder sounds induce larger signal amplitudes, i.e., D_i in Equation 4.3, by increasing E_p and thus A_p. Figure 4.6 (b) shows that discernible signals appear when the SPL is larger than 60 dB. The signal amplitude increases exponentially as the SPL increases until the lens motion approaches the stroke limit of the suspension system around 90 dB. Such an exponential relationship agrees with our modeling in Section 4.4.1 since the SPL is a logarithmic function of E_p and $D_i \propto \sqrt{E_p}$. It suggests the attack might be relatively sensitive to changes in volumes. We will conduct further evaluations in Section 4.6.2.

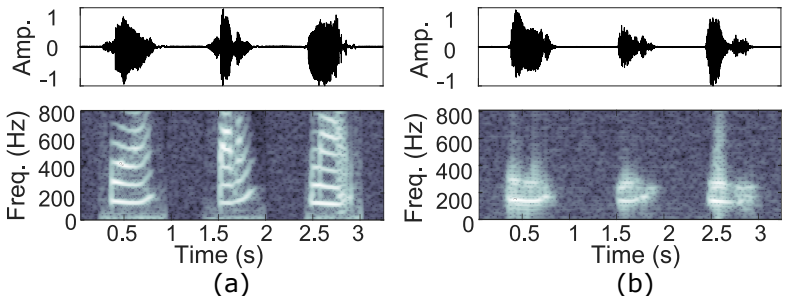


Figure 4.8: The waveform and spectrogram of spoken digits "zero", "seven", and "nine". (a) The original signals. (b) The recovered signals from a 3.2s video with optimized camera parameters.

(2) Camera-scene distance. According to Equation 4.3, the camera-scene distance d has progressively smaller impacts on D_i as it increases. Figure 4.6 (c) shows that the signal amplitude is approximately constant when the distance between the smartphone camera and the object in the scene is larger than 30 cm. Considering that both Ceiling and Floor Scenes often have distances much larger than 30 cm, Figure 4.6 (c) suggests this factor has a relatively small impact on the attack capability.

(3) Phone orientation. The orientation (on the XY-plane) of the phone with respect to the sound source changes the lens motion's directionality. We empirically evaluate how orientation can affect the attack by testing different orientations. We find that phone orientation has a relatively small impact on the extractable acoustic information since most cameras' movable lenses have at least 3 DoF. The lens motions in all directions can thus be effectively captured.

(4) Other factors. Besides the three factors above, other factors such as the phone-speaker distance affect the recovered signal in less quantifiable ways due to the lack of descriptive mathematical models. We will evaluate the impact of these factors in typical settings in Section 4.6.2.

4.5 Learning The Functions of Speech

Figure 4.8 (a) and (b) show the original and recovered speech signals of a human speaking "one", "seven", and "nine". While we could detect clear tones and their dynamics with more than doubled recoverable frequency range compared to that of smartphone motion sensor side channels reported so far (about 250 Hz maximum)[51], the recovered speech audio is still challenging for humans to recognize directly. We believe the reason is that the maximum bandwidth of 600 Hz often only captures the fundamental frequency (F_0) of vowels and

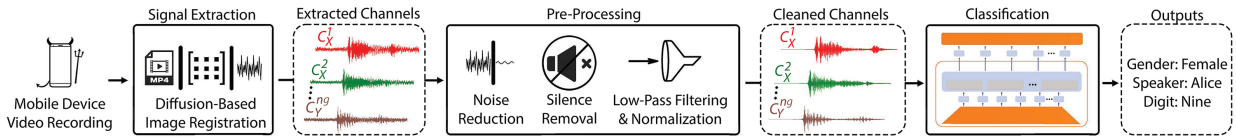


Figure 4.9: Our signal processing pipeline exploits the optical-acoustic side channel on smartphone cameras. The signal extraction stage extracts sound-induced signals from the videos recorded on smartphones. The pre-processing stage cleans up the signals and feeds them into the classification model, where the gender, speaker, and speech content are recognized.

voiced consonants while losing the second and third formants (F_0, F_1); signals from unvoiced consonants (over 2 kHz) such as "f", "s", "k" will also be completely missing [60, 246]. It has been shown that an audio channel with a 1kHz bandwidth could only allow single-word recognition rates of less than 20% by humans [192]. Furthermore, Figure 4.7 shows certain low frequencies such as 200 Hz can generate higher frequency distortions that can contaminate the true high-frequency signals. This suggests a human hearing system-based attack $f_{\mathcal{A}}$ is not likely to succeed. We also found existing machine-based Speech-to-Text engines such as Google Cloud [30], IBM Watson [31], and Apple voice assistant [32] unable to detect speech in the recovered signals. The observation motivated us to construct a more specialized $f_{\mathcal{A}}$ for estimating the information recovery limits.

4.5.1 Signal Processing Pipeline

As shown in Figure 4.9, our $f_{\mathcal{A}}$ is a signal processing pipeline that consists of the following three stages.

(1) Signal extraction. The stage implements the extraction algorithm shown in Section 4.4.3. It accepts videos collected by the malicious app and outputs $2n_g$ (8 in the case of 1080p videos) channels of 1D signals.

(2) Pre-processing. It performs noise reduction, liveness detection, trimming, low-pass filtering, and normalization of the channels. As shown in Figure 4.7, the extracted signals contain non-trivial but spectral-static noise caused by different imaging and image registration noise. We thus first apply a background noise reduction step using the spectral-subtraction noise removal method in [196, 195]. We then conduct a channel-wise amplitude-based liveness detection that determines the start and end index of the contained speech signals. Afterward, we average the start and end indices of the channels and trim them to remove the parts without speech signals. We further apply a digital low pass filter with a cutoff frequency of 4kHz to get rid of the remaining high-frequency disturbances caused by camera imaging noise. Finally, we normalize the channels and pass them to the next stage.

(3) Classification. Our classification stage implements a classification model that builds upon the Hidden-unit Bidirectional Encoder Representations from Transformers (HuBERT) large [122], which is introduced next.

4.5.2 Classification Model

Our HuBERT-based classification model utilizes the advantages of transfer learning, waveform input, and state-of-the-art performance[1]. The model consists of three major components: CNN encoder, transformer, and classifier. To adopt the original HuBERT for our purpose, we change the model by (1) modifying the CNN encoder to allow multiple waveform channels, (2) changing the transformer dropout probability, and (3) adding a classification layer to allow HuBERT to be used for spoken digit classification. We implement all of these changes while preserving as much of HuBERT's pre-training as possible to leverage the benefit of transfer learning. Preserving the pre-trained weights is particularly important for the CNN encoder because it helps avoid the vanishing gradient problem that commonly occurs when training deep neural networks [226]. We use the weights of the first layer for each channel of our input signal $C_X^1, ..., C_Y^{n_g}$ and change the original dropout probability of 0.1 to 0.05 to better regularize the model for our task. We then designed and added our classifier to process the output of the transformer. The classifier averages the non-masked discovered hidden units and outputs label scores for each classification task. In our classification tasks, gender, digit, and speaker output 1, 10, and 20 scores respectively, which are used to obtain the likelihood of each label and thus the final predicted class.

The CNN encoder contains 7 CNN layers, each outputting 512 channels. The first CNN layer inputs a single channel while the remaining layers input 512 channels. Since our input signal, $C_X^1, ...C_X^{n_g}, C_Y^1, ..., C_Y^{n_g}$, consists of multiple waveforms, the CNN encoder is modified accordingly, using all the input channels to discover utterances $Y_1, Y_2, ..., Y_m$. The transformer contains 24 blocks, an embedding size of 1024, and 16 attention heads, which amounts to 317 million trainable parameters. The generated hidden units output by the model can be used for a variety of speech recognition tasks. In the case of classification, the final classifier layer averages the non-masked discovered hidden units $Z_1, Z_2, ..., Z_m$ and outputs label scores for each classification task. We use cross entropy as the objective function for our binary and multi-class classification tasks during training. In hyperparameter tuning, we discovered that the initial learning rate of 1e-4 and a scheduler decaying it every 4 epochs by a factor of 0.8 delivers optimal results.

4.6 Evaluation

To gauge the general capability of the optical-acoustic side channel, we carry out evaluations on a spoken digit dataset used in previous work of smartphone motion sensors acoustic side channel [51]. We first evaluate the structure-borne side channel's performance in shared-surface and different-surface scenarios separately using a Google Pixel 2 to investigate the impact of different structures and structure organizations, and then compare the performance between different phone models. For evaluation metrics, we provide both common speech audio quality metrics including NIST-SNR and Short-Time Objective Intelligibility (STOI), and accuracies of our specialized classification model. The former measures how good the extracted audio signals are and are used in major previous works of acoustic recovery & eavesdropping [178, 179, 177, 85, 150]. The latter measures how well information labels are extracted from audio signals to quantify the limits of information recovery. We found the two systems of metrics generally agree with each other as we observed correlation scores of 0.72 and 0.80 between our model's digit classification accuracies and NIST-SNR and STOI respectively with our evaluation data.

4.6.1 Evaluation Setup

Dataset & Classification Tasks. The dataset is a subset of the AudioMNIST dataset[3] [57] and contains 10,000 samples of signal-digit utterances (digit 0-9) from 10 males and 10 females. We perform three classification tasks, namely speaker gender recognition, speaker identity recognition, and speaker-independent digit recognition. These three tasks correspond to the three levels of information recovery in Section 6.3.1.1 with $|\mathbb{L}| = 2, 20, 10$ respectively. Since all data labels for each task are equally likely in the dataset, the classification accuracies then serve as a statistical indication of $\mathbf{Adv}_{\mathcal{A}}$.

Experimental Setup & Data Collection. As our baseline setup, we place the smartphones and a KRK Classic 5 speaker side by side on a glass desk (Floor Scene). The speaker volume measures 85 dB SPL at 1 m [218]. The impact of smaller volumes including normal conversation volumes will be discussed in Section 4.6.2. For each evaluation case, we collect the whole 10k samples in our dataset using Python automation. We randomize the order of collected samples to avoid biased results due to unknown temporal factors. All phones use an exposure time of 1 ms.

Training & Classification Metric. To train the HuBERT large model for classification, we randomly split the 10k-sample dataset into training, validation, and test sets with 70%, 15%, and 15% splits, respectively. For each device or scenario evaluation, we train 3

[3]https://github.com/soerenab/AudioMNIST

Table 4.1: Performance in shared-surface scenarios

Scenario	Case	Avg. SNR	Avg. STOI	G (%)	S (%)	D (%)
Scene	Floor Scene 1	18	0.51	99.87	91.02	79.69
	Floor Scene 2	13	0.48	99.54	83.85	70.05
	Ceiling Scene	9	0.38	99.87	86.27	67.64
Volume	85 dB	18	0.51	99.87	91.02	79.69
	75 dB	11	0.44	99.80	89.13	76.95
	65 dB	4	0.18	98.83	76.11	68.16
	55 dB	2.4	0.13	80.27	34.77	27.67
	45 dB	2.3	0.15	54.49	8.92	13.28
	35 dB	2.3	0.14	54.23	6.84	15.95
Glass Desk, Distance, Volume	10 cm, 85 dB	9	0.38	99.87	86.27	67.64
	10 cm, 65 dB	1.9	0.25	81.25	37.17	32.03
	110 cm, 85 dB	9.3	0.35	99.74	84.24	64.13
	110 cm, 65 dB	1.8	0.32	81.12	36	31.12
Wooden Desk, Distance, Volume	10 cm, 85 dB	4.4	0.19	98.37	73.11	57.55
	10 cm, 65 dB	1.8	0.25	60.29	13.22	17.25
	130 cm, 85 dB	5.2	0.22	99.48	83.59	69.53
	130 cm, 65 dB	1.8	0.21	75.2	30.08	28.26
Wooden CR TBL, Distance, Volume	10 cm, 85 dB	8.8	0.33	99.02	79.82	66.6
	10 cm, 65 dB	2.4	0.19	76.76	42.58	32.49
	200 cm, 65 dB	2.3	0.19	70.75	33.53	26.43
	300 cm, 65 dB	2.6	0.19	83.2	41.86	30.99

TBL - Table, CR - Conference room, G - Gender, S - Speaker, D - Digit

HuBERT large models, one for each classification task. We trained all the models from the original pre-trained HuBERT large to allow for better comparison and used the same test set for final evaluations of all the models, where we report classification accuracies on the test set. The validation set is used for hyperparameter tuning and final model selection. During training, the model with the highest Receiver Operating Characteristic Area Under Curve (ROC-AUC) score is selected as the final model.

NIST-SNR and STOI. NIST-SNR [5] (referred to as SNR hereafter) measures the speech-to-noise ratio by calculating the logarithmic ratio between the estimated speech signal power and the noise power. A higher SNR score indicates better signal quality. STOI [219] is a widely used intelligibility metric. A higher STOI score indicates the speech audio is more comprehensible to humans. For all evaluation cases, we measure the SNR and STOI over the 1536-sample test set to make it comparable to the classification accuracies reported. We also utilize SNR and STOI to measure signal quality in certain test cases that do not present

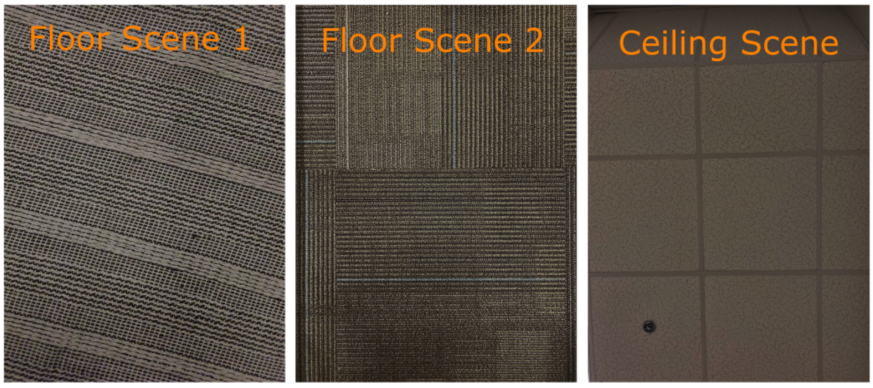


Figure 4.10: The three scenes evaluated.

a unique evaluation dimension by using a 100-sample signal testing subset consisting of 100 speech samples randomly sampled from the test set. In comparison to a full evaluation case, using the signal testing subset allows us to assess signal quality in a large number of different test cases in an efficient way. According to the sample size selection guideline from NIST [1], a sample size of 100 allows us to estimate the change in the average SNR and STOI scores with a 99% confidence level at a resolution of 0.5 times the standard deviation of the test set population's scores. With all the evaluation data we collected, this gives us a resolution of about 1.6 for SNR and 0.1 for STOI.

4.6.2 Shared-surface Scenarios

Shared-surface scenarios include the phones and speakers on the same surface, usually a table. In different scenarios, the quality of the recovered signals varies with configuration changes as shown in Section 4.4.4. We first study the impact of camera scenes and speaker volumes individually, and then investigate several representative scenarios that incorporate different combinations of the key factors of surface structure and phone-speaker distance.

Camera Scene. Table 4.1 shows the classification results under three scenes as shown in Figure 4.10. The first scene (Floor Scene 1) is with a downward-facing camera on the glass desk imaging the floor covered by a carpet. The second scene (Floor Scene 2) uses the same table and downward-facing camera but contains a different carpet on the floor. The third scene (Ceiling Scene) is with the camera upward-facing on the same table imaging the ceiling. Floor Scene 1 produces the highest accuracies in all three classification tasks, which we believe is due to the following reasons. First, the carpet in Floor Scene 1 has a lighter color than the carpet in Floor Scene 2, enabling more photons to be reflected and enter the camera and thus increasing the signal-to-noise ratio. Second, the image scene of Floor Scene

Table 4.2: Performance with different speaker devices

Speaker Device	**Avg. SNR**	**Avg. STOI**	**Gender (%)**	**Speaker (%)**	**Digit (%)**
KRK Classic 5	18	0.51	99.87	91.02	79.69
Logitech Z213	18	0.44	99.09	88.8	77.67
Laptop G9-593	3.3	0.12	94.92	57.03	36.78
Samsung S20+	6.4	0.15	89.00	53.91	32.36

1 has larger contrast than that of the Ceiling Scene due to the more abundant textures of the carpet compared to the ceiling.

Volume. Different speaker volumes represent different daily scenarios. Figure 4.6 (b) shows that the speaker volume has a strong impact on the signal amplitude. We found, however, the sharp decrease in signal amplitude does not lead to a proportional decrease in the classification accuracies. Table 4.1 shows the result with 4 typical conversation volumes and 2 whisper/background volumes: 85, 75, 65, 55, 45, and 35 dB often represent shouting, loud conversation, normal conversation, quiet conversation, whisper, and background noise respectively [29]. The results indicate that for volumes of 55 dB and above, the $f_{\mathcal{A}}$ designed still has a significant advantage over a random-guess adversary, demonstrating the side channel's effectiveness in quiet conversation volumes. The accuracies appear to be in the random-guess range at 45 and 35 dB.

Surfaces Structure and Phone-speaker Distance. Besides the glass desk, we evaluated a wooden desk in the same office and a 3m-long wooden conference room table. The Ceiling Scene was used for this set of evaluations. Table 4.1 shows the results with two different distances on the wooden and glass desks at 85 and 65 dB. The first distance is 10 cm and represents the scenario of placing the phone right beside the speaker; the other distance is the maximum achievable distance on each table (110 and 130 cm) by placing the phone on one edge and the speaker on the other edge, as shown in Figure 4.11. With the glass desk, a 3% decrease was observed for digit recognition when the distance increases from 10 cm to 110 cm. For the wooden table, the accuracies increased when the distance increased from 10 cm to 130 cm. Although this may seem counterintuitive at first, a closer look at the desks' mechanical structures suggests it is due to the smaller effective thickness on the edge of the table. At 65 dB, the glass and wooden desks show larger drops in accuracies than those in the volume experiments, which we believe is due to the ceiling scene having a more uniform color spectrum compared to Floor Scene 1, making smaller vibration amplitudes a more significant factor on classifier performance.

To further evaluate the side channel's robustness with larger phone-speaker distances, we

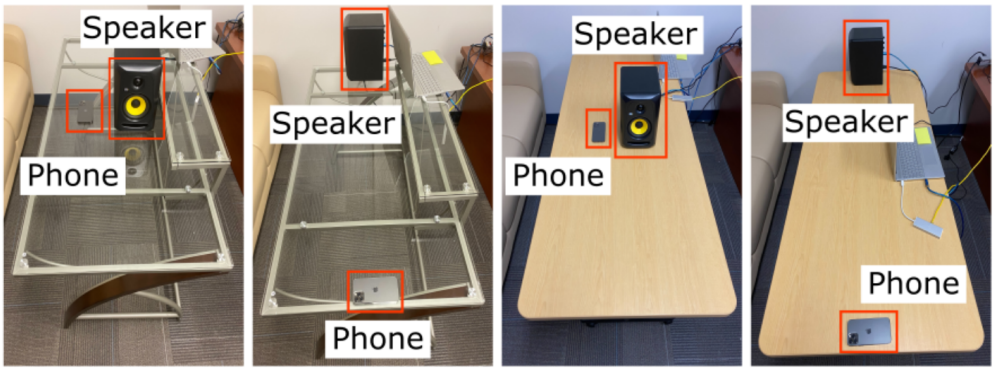


Figure 4.11: Setups of glass and wooden desks with the camera facing the ceiling. From the left. (a) 10 cm phone-speaker distance (b) 110 cm phone-speaker distance (c) 10 cm phone-speaker distance (d) 130 cm phone-speaker distance.

conducted experiments with a 3m-long wooden conference room table. As shown in Table 4.1, the classifiers' accuracies remain larger than random-guess accuracies, indicating the side channel's effectiveness at distances larger than 100 cm at normal conversation volumes.

Speaker Device. To uncover the potential impact of speaker devices on the side channel, we tested 4 different speaker devices including two standalone speakers (KRK Classic 5 and Logitech Z213), a laptop speaker (Acer Laptop G9-593), and a smartphone speaker (Samsung S20+). Table 4.2 shows that all 4 speaker devices allow for performance better than a random-guess adversary. We found even smaller internal speakers of portable devices including the laptop speaker vibrating a nearby phone's camera and the Samsung S20+'s speaker vibrating its own onboard camera could induce discernible signals. The variation in accuracies over the 4 devices is mainly due to the different maximum output volumes they can achieve; while the KRK Classic 5 and the Logitech Z213 speakers can output 85 and 75 dB respectively, the Laptop G9-593 and Samsung S20+ speakers are limited to 60 dB output.

Additional Objects on Surface. Thus far, most experiments were conducted with the speaker and the phone as the only objects present on the surface. Theoretically, the presence of additional objects on the surfaces propagating sound waves will only have a small impact on the side channel because structure-borne sound vibrates the entire structures which are often much heavier than the objects on the surfaces. To further investigate this factor, we conducted experiments with a daily occurring scenario of a cluttered desk with a varying set of common objects placed on the desk including a speaker, a laptop, a monitor, and a printer. Despite the slight change in SNR and STOI scores, full evaluations of the least and most cluttered scenarios reported similar classification accuracies: the least cluttered

Table 4.3: Performance in different-surface scenarios

Scenario	Avg. SNR	Avg. STOI	G (%)	S (%)	D (%)
Monitor Stand 85 dB	11	0.45	99.09	80.53	60.42
Monitor Stand 65 dB	2.6	0.09	84.05	42.32	32.1
Two Desks 85 dB	2.6	0.08	75.72	19.6	14.26
Two Rooms 85 dB	2.3	0.06	66.93	15.17	15.17
Shirt Pocket 85 dB	2.5	0.19	95.9	66.37	45.7
Bag Pocket 85 dB	4.1	0.23	93.1	40.1	55.34

G - Gender, S - Speaker, D - Digit

scenario achieved 94.86%, 70.44%, and 50.98% for gender, speaker, and digit classification accuracy respectively while the most cluttered desk scenario achieved 91.41%, 69.27%, and 56.25%. The results suggest cluttered surfaces with heavy objects allow for similar side channel performance.

4.6.3 Different-surface Scenarios

We evaluated several different-surface scenarios including (1) the speaker on the desk and the phone on the desk's monitor stand; (2) the speaker on the floor and the phone in the pocket of a shirt and a backpack worn by a mannequin; (3) the speaker and phone on different desks; (4) the speaker and phone in separate rooms. Table 4.3 indicates the side channel's performance over a random-guess adversary in these scenarios. With the exception of monitor stand experiments, we believe the decrease in performance can be attributed to the fact that the same speaker energy E_s now vibrates structures of much larger weight and stiffness (in this case the concrete floor) as opposed to a wooden floor structure [253] or wooden/glass surface. This makes it more difficult to create oscillation of structures with larger amplitudes to produce higher SNR. Additional causes of performance degradation could be due to the contact point between the desk, the mannequin's foot, and the transfer medium, i.e., the floor, moving relative to each other and causing frictional losses of the vibration energy E_s and thus also result in a lower SNR.

4.6.4 Different Smartphones

To evaluate the capability and robustness of our side channel on different phones, we analyzed the classification accuracies of 10 phones in the Floor Scene 1 setup. Table 4.4 shows the results from three smartphone families, namely the Google Pixel, Samsung Galaxy, and

Table 4.4: Performance with different smartphone models

Device	**Avg. SNR**	**Avg. STOI**	**Gender (%)**	**Speaker (%)**	**Digit (%)**
Pixel 1	18	0.46	99.61	81.84	69.53
Pixel 2	18	0.51	99.87	91.02	79.69
Pixel 3	17	0.49	99.67	91.28	80.66
Pixel 5	22	0.49	99.48	84.51	70.25
Samsung S7	21	0.49	99.54	82.94	66.08
Samsung S8+	17	0.45	99.61	79.30	57.29
Samsung S20+	22	0.49	99.80	83.92	61.07
iPhone 7	28	0.53	99.87	85.09	65.23
iPhone 8+	26	0.50	99.41	81.64	66.67
iPhone 12 Pro	28	0.52	99.22	76.56	62.30

Apple iPhone. We also show the estimated recoverable frequency ranges, the rear camera modules, and their key characteristics in [163]. To measure the key characteristics, we generate a 200 Hz tone for 3 seconds. We then find $1/T_r$ by changing it to align the recovered signal with 200 Hz. With $1/T_r$, we calculate η_{cap} according to Equation 4.5. We further measure η_{cap} by dividing the length of the recovered tone by 3 seconds. The measured and calculated η_{cap} match well with each other which shows the correctness of our modeling. We used 30 fps for the Android phones because that is what most Android manufacturers currently provide to 3rd party apps while iPhone used 60 fps.

As shown in Table 4.4, the Google Pixel phones generate the highest accuracies for all three classification tasks. The iPhones generate slightly better results than Samsung phones. Samsung S8+ generated the worst accuracies. We notice the videos of Samsung S8+ suffer from missing frames potentially due to internal processing issues. We observe that η_{cap} has the strongest correlation where lower η_{cap} provides the adversary with less information and consequently lower accuracies. We also notice that there exists a trend of newer camera modules having lower T_r, i.e., higher rolling shutter frequency, and thus lower η_{cap}. The question of this trend being usable as a mitigation technique is further analyzed in section 4.7. All the phones we tested achieved at least 99.22%, 76.56%, and 61.07% accuracies on gender, speaker, and digit recognition respectively. This suggests that the adversary is able to perform successful side channel attacks with high $\mathbf{Adv}_{\mathcal{A}}$ (Section 6.3.1.1) on a large portion of phones available on the market at the time of writing.

Multi-device Scalability. To investigate the feasibility of cross-device attacks, we conducted four multi-device studies: (1) with the most recent phones from the three phone families (2) the four models of the Pixel family (3) the three models of the Samsung family

(4) the three models of the iPhone family. As shown in [163], most cross-device cases show advantages over a random-guess adversary, demonstrating the existence of common information across different phone models. It is worth noting that when the classification model is trained on Pixel 5 and iPhone 12 Pro and tested on Samsung S20+, the accuracies for gender and digit recognition (highlighted in green in Table X of [163]) are higher than training on S20+ itself. Similarities in recovered signals across different models are determined by various sources, such as similar image sensors, rolling shutter frequencies, and image signal processing units (ISPs). For example, Samsung S7 and Pixel 1 have the same rolling shutter frequency and very similar ISP and processor. In contrast to the IMX260 sensor used in Samsung S7, the IMX378 sensor used in Pixel 1 does not support OIS [37]. The results suggest our side channel has the potential to be generalized for unseen devices, especially when the adversary trains with data from smartphone models with similar camera systems. The bolded numbers in [163] indicate there is often no accuracy loss in testing on a specific phone when data from other phones are added to the training set. When training on all three or four phones and testing on a specific phone, the accuracies are almost ubiquitously better than or similar to training on that phone alone. This suggests our classification model is capable of representing data distribution from multiple phone models with minimal to no information loss.

4.7 Mitigation

This section analyzes immediate countermeasures that may be carried out by users and more informed protections for manufacturers that aim to secure future camera devices.

4.7.1 User-based Countermeasures

Lower-quality Cameras. Users can use lower-quality cameras to limit information embedded in videos by reducing video resolution and frame rate. However, these measures cannot degrade eavesdropping performance without significantly sacrificing overall video quality. Figure 4.6 (a) shows that reducing video resolution from 1080×1920 to 480×640 reduces the signal amplitude by about 60%. However, Figure 4.6 (b) and Table 4.1 show that when the volume decreases by 10 dB, the signal amplitude decreases by about 75% which only reduced digit classification accuracy from 79.69% to 76.95%.

Phones Away From Speakers. A straightforward yet effective approach for privacy-aware users is to place phones away from electronic speakers. As shown in Table 4.3, removing phones from the same surface as the speaker immediately reduces attack performance.

Adding Dampening Materials. Another possible method is to add vibration-isolation dampening materials between the phone and the surface in the hope to lower k_0 in Equation 4.2. Using the evaluation baseline setup and Pixel 2, we tested specialized vibration reduction mats made of visco-elastic polyurethane [39] with varying degrees of hardness. Three mats were used with common type OO durometers of 30, 50, and 70[27]. Our tests show the three materials produced similar effects in mitigating our attack (Table 4.3). A classification evaluation shows adding such dampening materials reduced digit classification accuracies by 14.33% (Table 4.6).

4.7.2 Camera Design Improvement

Fundamentally, the side channel arises because of movable lenses that modulate smartphone motion onto video streams and rolling shutters that increase the available sample rate of adversarial signal recovery. We thus investigate the possible ways to mitigate these two sub-problems from the perspective of future camera designs.

4.7.2.1 Rolling Shutter Mitigation

Besides a plain approach of replacing rolling shutters with global shutters, we identify two methods to tackle the problem by increasing rolling shutter frequencies or adding randomization.

Higher Rolling Shutter Frequency. As mentioned in Section 4.6.4, we observed a trend of higher rolling shutter frequencies in newer camera sensors. We believe this trend shows camera designers' intention to approximate global shutters, which also led to lower attack performance as a byproduct. It is thus worth investigating the effectiveness of utilizing this trend as a defense. Basically, higher rolling shutter frequencies reduce the amount of intra-frame motion signals captured by adversaries (Section 4.4.2). We generated model-based predictions[1] of the side channel adversary's success with increasing rolling shutter frequencies and used the evaluation samples of Pixel 2 as the baseline.

Table 4.5 shows the tested η_{cap}, the required rolling shutter frequency, and the classification accuracies. The result suggests that further increasing the sample rate does reduce classification accuracies, but the adversary still has a large advantage over random-guess adversaries even if they can only recover 0.1% of the signals at 32,400 kHz. Furthermore, the accuracy decay sees an asymptotic trend, suggesting a potential lower bound of the accuracies even when the sample rate approaches infinite. We believe this lower bound is posed by the inter-frame information retained. In other words, adversaries may recover a large amount of information even from a global shutter camera just by measuring variations

Table 4.5: Recognition accuracy with different η_{cap}

$\eta_{\mathbf{cap}}$ (%)	**Sample Rate (kHz)**	**Gender (%)**	**Speaker (%)**	**Digit (%)**
95	34	99.87	91.02	79.69
50	65	99.54	80.86	59.77
10	324	95.51	68.55	50.59
5	648	93.29	62.89	48.89
1	3,240	86.78	48.37	41.21
0.5	6,480	86.85	43.88	38.87
0.1	32,400	83.20	42.25	38.54

between frames.

Random-coded Rolling Shutter. If higher rolling shutter frequencies cannot be achieved, another method is to scramble the intra-frame signals by randomly mapping $s(n\delta)$ to $a(i)$ in Equation 4.7. Simply put, we can potentially randomize the order of each row's exposure and readout within each frame. This method only has a small impact on video quality because it only affects rolling shutter patterns in the videos which are already considered as distortions. Our simulation shows random-coded rolling shutter is able to produce defense effectiveness as good as increasing the rolling shutter frequency from 34 to about 100 kHz for Pixel 2. We conjecture this is because the intra-frame motion signals are only scrambled instead of completely removed and our classification model is able to utilize statistical information (e.g., max/min/mean) of the scrambled signals.

To implement random-coded rolling shutters, the address generator (Figure 4.3) needs to output randomly ordered instead of sequential addresses. Existing research shows manufacturers can already make the address generator output designated control sequences by changing camera firmware [112, 212]. The remaining cost of implementation is for adding a random number generator (RNG) that communicates with the address generator. In fact, imaging sensors themselves are a good source of entropy and have been already used in research and industry for generating true random numbers [258, 153, 8].

4.7.2.2 Lens Movement Mitigation

Our experiments show that addressing problems caused by rolling shutters alone cannot eliminate the threats due to the upper bound of protection effectiveness posed by the inter-frame motion information that still resides in the videos. It appears that the main cause of this side channel is the design flaw in existing smartphone camera sensors that leaves the lens dangling and free to move in the lens suspension system. Below, we propose two possible

Table 4.6: Effectiveness of single and combined defenses

Defense	**Gender (%)**	**Speaker (%)**	**Digit (%)**
None (Baseline)	99.87	91.02	79.69
① Rubber Mat Dampening	98.64	80.11	65.36
② Higher RS Freq. (648 kHz)	93.29	62.89	48.89
③ Random-coded RS	98.18	76.56	60.22
①+②	75.65	43.88	33.14
①+③	72.66	46.03	37.63
④ Tough Spring/Lens Locking	65.23	16.73	16.67
②+④	53.91	8.66	16.73
③+④	54.36	8.46	13.93

methodologies in an attempt to mitigate this.

Tougher Springs. Our signal path modeling reveals that increasing the elastic force of the lens suspension springs (c_l in Equation 4.2) makes it more difficult for sound waves to vibrate the lens. There are several possible modifications designers can make to achieve this as suggested by the model of smartphone camera lens voice coil motor (VCM) systems [74]:

$$\begin{cases} S = \frac{R}{V^2} \cdot \left(\frac{F_e - f_{\text{fric}} - xc_l - mg}{m}\right)^2 \\ F_e = Nil_wB_g = N\frac{VA}{\rho L} l_wB_g \end{cases}$$

S is the VCM's sensitivity that designers want to optimize; F_e is the electromagnetic actuation force; x is the lens displacement. To keep S the same so that users do not experience degradation in camera functionality and usability, we identify the following straightforward ways to compensate for the impact of higher c_l along with their costs. (1) Increase the number of coil windings N, the coil length l_w, or coil area A. This will increase the size of the camera modules. (2) Increase the magnetic flux density B_g by using better permanent magnet materials. This will add to the budget. Other parameters such as coil voltage V and resistance R may also be adjusted but can lead to higher camera power consumption. While different camera products are subjected to specific manufacturing constraints, we believe our analysis above provides a starting point that designers can consider.

Lens Locking. We envision the ultimate solution to the lens movement problem is to have a locking mechanism that completely prevents lenses from moving when they are not supposed to. Such a mechanism may be achieved by adding controllable pillars around the lens. The pillars contract when OIS and AF are enabled to make space for lens movement and expand to fix the lens in place otherwise.

Simulation of Effectiveness. To demonstrate the potential of these solutions, we

simulated tougher springs and lens locking by using an external magnet to prevent the lens from moving in the same way as Section 4.4.1. The decreasing attack accuracies are shown in Table 4.6. The remaining non-random classification accuracies are likely due to a combination of (1) the residual lens movements that the magnet cannot completely remove, and (2) the tiny movements of the smartphone body. Finally, combining multiple methods of defense can further bring attack performance down to the random-guess range as shown in Table 4.6.

4.8 Conclusion

This chapter provides evidence that besides very explicit factors such as resolution and sensitivity, other implicit factors in the underlying physical construction of the sensors also contribute to the increase of $s_{sec} \cap (s_{int} \cup s_{side})$, invalidating **KR1**. The investigation of possible mitigations shows that while it is not feasible to directly remove the complex structures creating side channels due to their useful benign functionalities, there exist systematic ways to reduce the leakage by improving existing designs such as applying randomness to the sensing process.

CHAPTER 5

Information Leakage due to Unprotected Sensor Data Transmission

5.1 Overview

In a sensing system, d_{sensor} (Equation 2.1) needs to be further collected, parsed, and distributed to downstream hardware components. As mentioned, there is no encryption in this process. This implies that if someone can eavesdrop on the plain data, they should be able to partially recover the secret information. Investigating hypothesis **H2** in the camera sensing settings, this chapter shows how the unprotected transmission between different parts of a camera system, specifically the CMOS imagers and downstream processors such as Graphics Processing Units (GPUs) produces physical electromagnetic side channel leakage that can be picked up by external adversaries using radio antenna to reconstruct the confidential camera inputs in real-time [162].

5.2 Threats of Camera Data Leakage

Cameras, being one of the highest-entropy sensors, are becoming omnipresent even in private spaces. Recent advances in the miniaturization of semiconductor electronics have spurred the wide integration of cameras into various embedded and mobile systems ranging from smartphones to IoT gadgets such as smart locks and home monitors. For smart home security cameras alone, the number of families owning such devices is predicted to grow from 99 million to 180 million between 2023 and 2027 [215]. Given the near-universal adoption of embedded cameras and the critical information they could capture such as the private activities and personnel information in offices and households, it is imperative to prevent unauthorized access to camera data. While previous research examined the data eavesdropping vulnerabilities in networked IP cameras' software stack [41, 156, 118, 221], the hardware

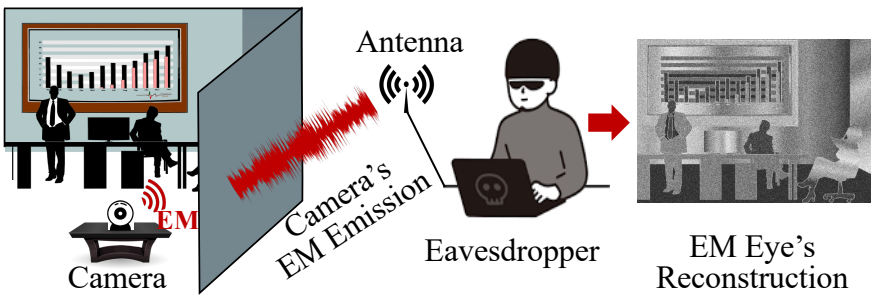


Figure 5.1: Embedded cameras leak EM signals in operation, allowing eavesdroppers to visually spy on private spaces by reconstructing camera images.

design of these embedded camera devices has not been scrutinized yet. To understand the threats more thoroughly, our work investigates a new dimension of the problem by asking *how may adversaries eavesdrop on camera data by exploiting the side-channel byproducts generated by the cameras' physical operations?*

Our work draws inspiration from recent works showing that embedded cameras' electromagnetic (EM) emissions allow people to detect the presence of cameras [160, 251, 200]. While these works simply use the existence of EM emissions as an on/off indicator of camera operations, essentially extracting a single bit of information, our work further investigates how much information of camera data is leaked from such EM emissions[1], and how adversaries can eavesdrop on the camera image streams by reconstructing synthesized images from the EM signals. Through experiments with the open-source Raspberry Pi camera, one of the most used embedded camera prototyping platforms, we observe highly predictable correlations between the EM emission patterns and the camera image contents. Nevertheless, mapping the 1D EM signals to 2D images is conceptually challenging without further knowledge of the EM generation process. Our investigations unveil that the primary EM leakage source is the digital image data transmission interface between the image sensor chips and the downstream image processing components. We carry out a detailed analysis of the physical layer of the embedded camera's data transmission interface. We find that RAW sensor data represented in bits are transmitted in a deterministic way following a frame-by-frame, row-by-row, and column-by-column order. By understanding the serialized data transmission scheme and reverse-engineering the transmission parameters, adversaries can directly generate eavesdropped image streams in real-time using portable equipment including an antenna, a software-defined radio receiver, and a laptop.

Despite the ability of direct image reconstructions, our experiments reveal additional challenges that limit adversaries' capability of retrieving intelligible information from the re-

[1]Demo and tutorial are available at https://emeyeattack.github.io/Website/

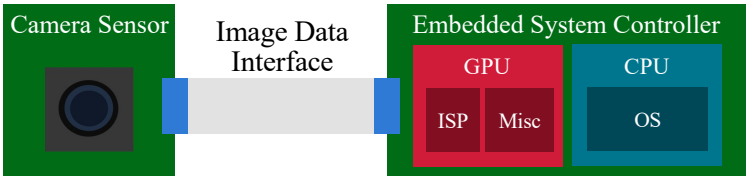


Figure 5.2: The typical architecture of embedded camera systems.

constructions. For example, the eavesdropped images suffer from loss of colors and incorrect gray-scale values, as well as significant noise that causes degradation of image quality. We thus develop a model to characterize the physical leakage process of digital image transmission and analyze the root cause of these distortions. Our analysis shows that the limited EM signal bandwidths that could be afforded by adversaries in practical settings cause irreversible loss of image data structures in the EM signals, which then manifests itself as very structured distortions in the reconstructions. An adversary aiming to get a high-quality image then faces the challenge of partially recovering the data structures by leveraging their prior knowledge of the physical leakage process. To explore to what extent an adversary can achieve this, we develop an enhanced eavesdropping pipeline to strategically combine available EM signals and infer high-quality images using a supervised image-to-image translation network that learns the structured mappings between original and distorted images. We find the pipeline capable of removing most distortions, recovering authentic gray-scale images, and even producing colored images that well-approximate the camera scenes.

To examine the scope of EM Eye's risks, we conducted experiments with popular IoT camera development platforms including Raspberry Pi 3B+/4B, Nvidia Jetson Nano, Asus Tinkerboard 2S, and 12 commercial-off-of-the-shelf (COTS) devices with embedded cameras. With middle-end EM receiving equipment, our evaluations show that smartphone camera EM emissions could be received from up to 30 cm away, allowing adversaries to install low-profile hidden antennas to eavesdrop on smartphone photography. Dash cams and smart home cameras could be eavesdropped on from up to 5 m away, allowing adversaries to spy on physically-isolated spaces such as the interiors of cars, households, and offices through doors and walls as shown in Fig. 5.1. Our investigation of camera EM side-channel further uncovers the underlying physical vulnerability of unprotected image data baseband transmission. We note that this vulnerability is also shared by the well-known TEMPEST and acoustic side-channel eavesdropping attacks against computer displays. Despite the past 40 years of computer display eavesdropping research, our work shows that there still exists a semantic gap between the understanding of TEMPEST vulnerabilities and how modern sensors process and transmit data. Finally, we analyze how to protect embedded cameras by improving

the data transmission protocols and discuss how future adversaries may apply the same eavesdropping methodology to other types of sensor data.

5.2.1 Related Work

Computer Display Side-channel Eavesdropping. It has been widely acknowledged that computer displays generate side-channel leakages in operation that allow adversaries to eavesdrop on the displayed contents. The most known research is TEMPEST attacks where EM leakage is used to reconstruct computer screens. Following the first work by Wim van Eck in 1985 [234] that proved the feasibility of reconstructing video display contents using non-military commercial-grade equipment, extensive research has been carried out over the last 40 years. Some notable works include Markus Kuhn's efforts to develop low-cost techniques to eavesdrop on analog CRT [143] and digital LCD flat panel displays [145]. While earlier works only investigated standalone computer display units which often generate stronger EM emissions, Hayashi et al. showed it is possible to eavesdrop on smaller tablet and laptop screens from 2m away [116]. Recently, Liu et al. [159] extended this attack to smartphone displays. However, due to the very weak EM emissions generated by the small smartphone circuits, the researchers had to use machine learning classifiers to recognize the humanly-unintelligible reconstructions at a distance of 1 cm. Besides EM emissions, Genkin et al. [106] showed that acoustic side-channel signals generated by computer display circuits when processing different pixel data also allow adversaries to detect screen contents using machine learning classifiers. In all these works, texts on screens have been the sole target of eavesdropping.

Cameras work in similar ways as computer displays in that they both have to transmit streams of 2D images in a serialized manner. Our work shows that a more fundamental analysis framework for 2D digital image transmission leakage can be developed to model and generalize these attacks. Compared to previous works, our research bridges the gap between such information leakage mechanisms and a broad range of emerging sensor systems. From the standpoint of technical advances, this work shows that the camera image contents are significantly more complex and diverse than those of computer displays, causing new challenges such as light gradient distortions that increase the difficulty of using existing TEMPEST techniques to reconstruct high-quality recognizable images. We thus design and apply new computational techniques to address these unique challenges.

IP Camera Hijacking & Sniffing. With a similar purpose of accessing the outputs of unauthorized cameras, several works have found that networked IP cameras can be hijacked or sniffed by adversaries when there exist vulnerabilities in the network configurations. For

example, Abdalla et al. showed that many cameras use default passwords and unencrypted communications [41]. Ling et al. demonstrated the feasibility of performing an online brute-force attack to uncover IP camera's password because many cameras only have only four-digits long passwords [156]. Herodotou et al. found that a generic camera module used by many spy camera manufacturers can be controlled by adversaries over the internet as long as the serial number of the camera is known [118]. Tekeoglu et al. successfully reconstructed 253 JPEG images from about 20 hours of video track by sniffing an IP camera's unencrypted network traffic [221]. While these works show the feasibility of eavesdropping on IP cameras when there exist software vulnerabilities, our work explores the complementary aspect of physical vulnerabilities of camera designs. This allows an adversary to eavesdrop on not only networked cameras but also locally-operated cameras as well as systems with strong software security such as smartphones and home security devices.

Camera Electromagnetic Leakage. This work builds upon the main hypothesis that the EM leakage of cameras is correlated with camera contents and can be used to infer or even reconstruct camera outputs. This hypothesis is motivated by recent research discoveries of the EM characteristics of embedded cameras. Several works have shown that smartphone cameras and hidden spy cameras produce EM emissions when they are turned on, allowing people to detect forbidden malicious operations of these cameras [251, 160, 200]. Essentially, these works only extract a single bit of entropy (on/off) from camera EM emissions. It also remains unclear how the EM emissions are generated by cameras. In the opposite direction, Jiang et al. [131] demonstrate the feasibility of injecting EM interference to partially control CMOS camera's outputs with an image row-level granularity; Köhler et al. [141] demonstrate a pixel-level injection granularity with Charge-Coupled Device (CCD) cameras which are less common in modern consumer electronics. Their results suggest there is significantly more entropy embedded in camera EM characteristics that can be harvested. Building upon these insights, our work seeks to characterize the feasibility, causality, and limits of eavesdropping on pixel-level information from the EM leakage of cameras in embedded systems.

5.3 Threat Model & Background

5.3.1 Threat Model

We characterize the threat of passive eavesdropping on the confidential camera data of embedded systems by exploiting the unintentional EM emissions from camera sensors, the image data transmission interfaces, and image signal processors. The goal of the adversary is to reconstruct an image stream that approximates the authentic camera output as closely

as possible. We assume the adversary uses a set of readily available commercial hardware equipment that is able to collect the EM emissions generated by the cameras. This often includes an antenna, a low-noise amplifier (LNA), a software-defined radio (SDR) device such as a USRP [97], and a laptop that runs the image reconstruction algorithms. We consider various camera-antenna distances and two corresponding categories of eavesdropping scenarios, namely the hidden-antenna (HA) and physical-isolation (PI) scenarios. In the former scenario, we assume the adversary manages to install a low-profile antenna near the target camera to receive stronger EM emissions. In the latter scenario, we assume the camera is located in a physically isolated space such as a private room and the adversary's antenna can only be placed outside the room to receive EM emissions through walls or doors. In both cases, the camera scenes contain private information that is supposed to be visible only to the legitimate camera owner.

5.3.2 Embedded Cameras

Embedded system devices are increasingly equipped with camera peripherals. Compared to traditional cameras such as digital single-lens reflex (DSLR) cameras, embedded cameras often feature open-standard designs that allow them to interface with a wide range of controllers. Fig. 5.2 shows the architecture of a typical embedded camera system. The camera's semiconductor image sensors convert photons hitting the semiconductors into proportional electrical signals. Each image sensor contains millions of sensing units corresponding to "pixels" in the digital image domain. The electrical signals are amplified, conditioned, digitized by analog-to-digital converters (ADCs), and transmitted to the computation units such as the image signal processor (ISP) in GPUs. The GPU then produces the final images after debayering (also known as demosaicing), image corrections, and miscellaneous post-processing. Like most sensor peripherals, embedded camera modules are often supplied by third parties and integrated by consumer electronics manufacturers.

RAW Images and Debayering. RAW images refer to the unprocessed data generated by image sensors. Since each semiconductor sensing unit only captures a single channel of RGB color that is selected by a color filter array, each pixel only has one color in the RAW images. To get a normal color image that users are familiar with which has all three RGB color channels, the ISP needs to perform a debayering step to interpolate the missing RGB channels for each pixel based on available colors from its neighbors [113].

Pixel Data Transmission. Image sensors and ISPs are connected by a pixel data transmission interface that transmits the RAW pixel data. Some examples of such interfaces include the High-speed Serial Pixel Interface (HiSPi) [46], the Digital Video Port (DVP) [125],

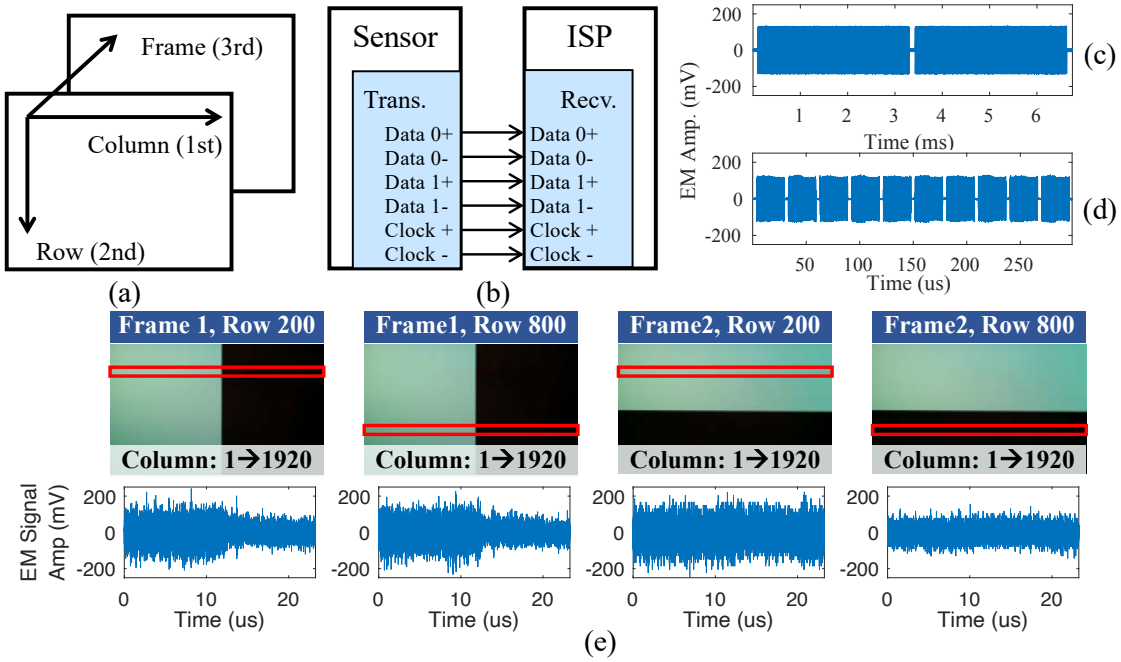


Figure 5.3: How embedded cameras' operations generate EM signals that leak camera image information. (a) Each video frame is transmitted row by row and column by column. (b) The MIPI CSI-2 interface transmits image data with multiple lanes of differential data wires and clock wires, all generating EM leakage. (c) EM signals of two consecutive frames. (d) EM signals of ten consecutive rows. (e) EM signals of transmitting different frames, rows, and columns, showing clear correlations with the image contents.

the Low-voltage Differential Signaling (LVDS) [245], and the MIPI Camera Serial Interface 2 (MIPI CSI-2) [173]. MIPI CSI-2 has been widely adopted for its good usability, dedicated EM anti-interference designs, and capacity to support a variety of camera applications. It has become the de-facto standard for embedded cameras due to the rising demand for higher throughput and compatibility between hardware and software from different vendors. Same as most digital image transmission interfaces, MIPI CSI-2 transmits videos frame by frame. For each frame which is a 2D matrix, the camera transmits each row sequentially from top to bottom; for each row, each column (pixel) is also transmitted sequentially from left to right as shown by Fig. 5.3 (a). There often exists blanking between the transmission of consecutive frames and rows where the data transmission interface stays in an idle state without active transmissions. On the physical layer, MIPI CSI-2 uses high-speed differential signaling wires with up to four data lanes and a shared clock lane. Fig. 5.3 (b) demonstrates a MIPI CSI-2 interface with two data lanes.

5.4 Modeling Electromagnetic Eavesdropping on Cameras

Adversaries are to eavesdrop on the camera images by analyzing the electromagnetic signals that are converted from the optical signals captured by the camera's image sensor. This section investigates the feasibility, model, and characteristics of these optical EM side channels.

5.4.1 Feasibility

We use a Raspberry Pi camera V1 (RPi V1) to record a computer monitor displaying two simplified black/white scenes. The top row of Fig. 5.3 (e) shows the two scenes recorded by the camera. Meanwhile, we collect the EM signals around the camera using a near-field EM probe connected to an oscilloscope. The camera records with a frame rate of 30 fps. At various center frequencies including different multiples of 51 MHz, we receive periodic signals at 30 Hz matching the camera frame rate. Fig. 5.3 (c) shows such signals at 204 MHz with two consecutive frames and blanking between them. We have confirmed that the received signals are from the camera instead of the computer monitor which has a refresh rate of 120 Hz. When zooming in, we can also see the transmission of different rows with blanking in between, as shown by Fig. 5.3 (d). Inspecting the EM signals corresponding to different frames, rows, and columns, we found obvious correlations between the shape of the EM signals and the pixel values of the camera image, as shown in Fig. 5.3 (e).

EM Leakage Source. To determine where the EM leakage comes from, we use a tiny near-field magnetic probe to collect the EM emissions from each component of the camera device while shielding the other components. We find that the EM signals have significantly better signal-to-noise ratios (SNRs) when the probe is placed near the image data transmission cable that connects the image sensor and downstream image processing components. We thus conclude that the cable for image data transmission is the main EM leakage source.

Basic Image Reconstruction. To reconstruct an image, the adversary needs to map the one-dimension EM signals received by the antenna within a certain frequency band back to a two-dimension matrix by associating each segment of the EM signals to specific pixels of the image. This requires the adversary to model key parameters including the pixel transmission rate, row transmission rate, image height and width, blanking periods, etc. The adversary then needs to convert 1D vectors of EM signals to scalar pixel values of the reconstructed image, essentially demodulating the EM signals that are modulated by the

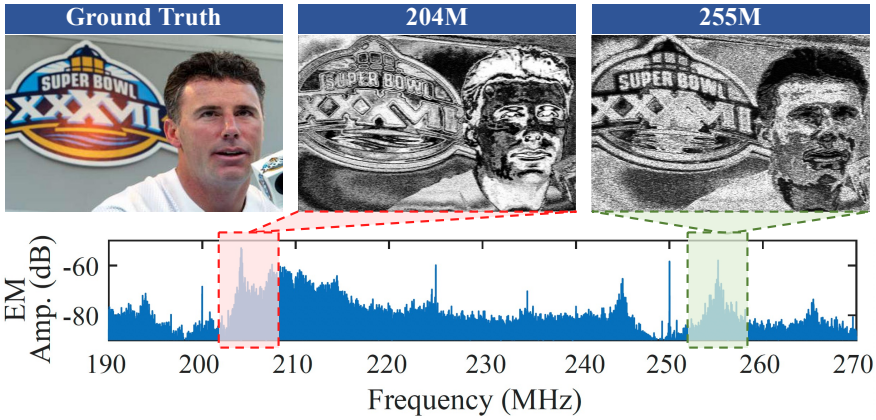


Figure 5.4: Illustrations of EM emission's spectrum and two reconstructed images using signals around 204 and 255 MHz.

image contents. Since the EM emission process is an unintentional communication channel, we believe simpler modulation schemes such as amplitude and frequency modulation are more appropriate than other sophisticated man-made schemes. A closer look at the temporal-spectral variations of the EM signals reveals that only very wide-band and rapid variations exist in the frequency components of the emissions, which could require a GHz-level sampling bandwidth to provide sufficient coverage and is thus not feasible. We thus hypothesize that amplitude demodulation is the most appropriate method based on our observations in Fig. 5.3 and use the amplitudes of EM signals as the gray-scale values of the pixels. We denote this reconstruction process as $\mathcal{R}_{base}$ and provide further details in Section 5.5.1. With $\mathcal{R}_{base}$, we are able to reconstruct images that share very similar structures as the camera ground truths in real time. Fig. 5.4 provides an example of such reconstructed images and the spectrum of the corresponding EM signals.

5.4.2 Digital Image Transmission Leakage Model

To understand why the reconstruction method above can recover an image similar to the camera ground truth and the potential ways to further improve the reconstruction performance, we analyze the fundamental information leakage model that unpins the optical EM side channels in embedded cameras. We use one of the most popular data transmission protocols, MIPI CSI-2 with RAW10 image data format and two data lanes, as an example for developing the model. This protocol is also used by RPi V1. Nevertheless, we note that the modeling and analysis methodology also applies to other digital image transmission interfaces.

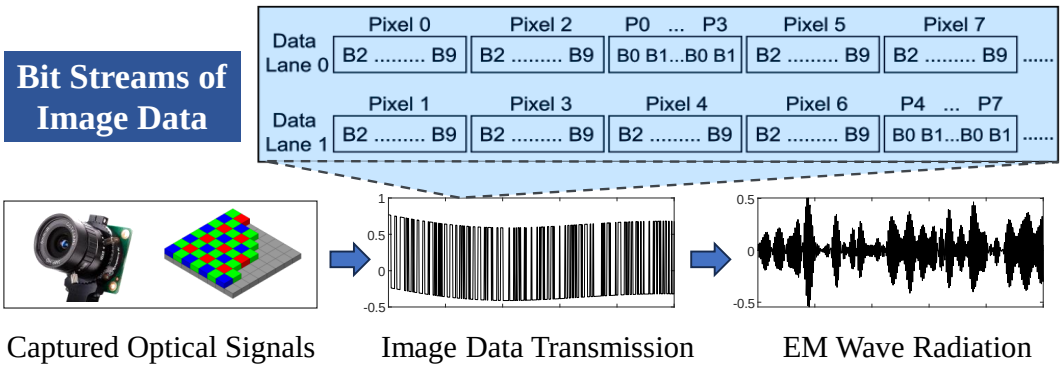


Figure 5.5: The information flow of camera EM leakage. Optical signals captured by image sensors are converted to bit streams shown on the top. The transmission cables act as unintentional antennas that convert the bits into radiated EM waves.

5.4.2.1 Fundamental Principle

Fig. 5.5 demonstrates how the optical information received by a camera sensor is transformed into EM signals that adversaries can capture. The process can be divided into two stages. In the first stage, the camera sensor transmits image data represented by digital bits row by row. The alternating currents/voltages caused by bit flips produce EM waves in the camera environment according to Maxwell's equation. In the second stage, the cable between the image sensor and ISP acts as an unintentional transmission antenna and propagates the EM waves to the adversary's receiving antenna. The EM signals are subjected to various environmental noises. With an off-of-the-shelf USRP device, the adversary can then sample the EM signals in specific frequency bands.

Fig. 5.5 also demonstrates how MIPI CSI-2 of image sensors transmits RAW10 images in the form of digital bits with two data lanes. Each pixel/column is represented by 10 ordered bits B0 to B9 (least significant bit (LSB) to most significant bit (MSB)) with the least significant bits transmitted first. The sensors treat a byte as a transmission element, although there is often no blanking between bytes during transmission. Since each pixel has 10 bits, RAW10 has to pack four consecutive pixels into a unit of five bytes where the two LSBs of the four pixels are packed into the last byte. Two units (8 pixels) are further grouped together. Using the dual data rate (DDR) technique, the clock f_{clk} frequency is twice the frequency of transmitting a bit f_b. For RPi V1, f_{clk} is measured to be 204 MHz, which means the byte transmission frequency is 51 MHz. When more than one data lane is used, consecutive byes are distributed to the lanes sequentially. It is worth pointing out that each wire of the transmission system, including the data and clock wires generate its own EM signals and the final signal the adversary receives is a mixture of them.

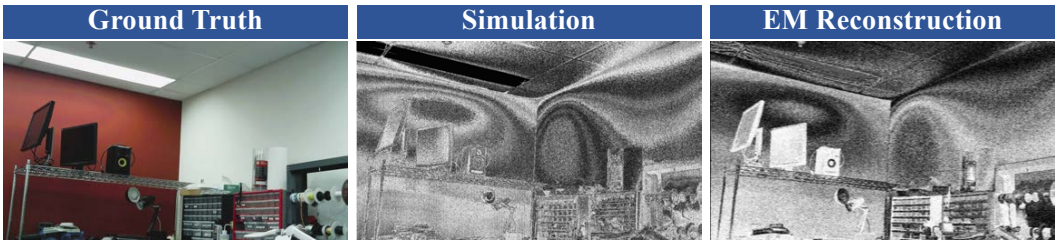


Figure 5.6: The camera ground truth, simulated, and actual EM reconstruction. Distortions such as the amplification of light gradients and high-frequency noises appear.

5.4.2.2 Modeling

Based on the understanding of the leakage process, we develop a mathematical model that can explain and simulate the physical leakage process's key characteristics. Assume the adversary tries to reconstruct an image that approximates the ground-truth camera image I_{GT} from the EM signals in the frequency band $[f_{lo}, f_{hi}]$ with a function $\mathcal{R}_{base}\{\cdot\}$, the EM reconstruction image can be calculated by

$$I_{EM}^{[l,h]} = \mathcal{R}_{base}\Big\{z + b_{clk} + \mathcal{F}_{filt}\left[l, h, \mathcal{F}_{data}(I_{GT})\right]\Big\}, \tag{5.1}$$

where z represents the noise, b_{clk} represents a constant signal offset produced by the clock wire's emissions given that clock amplitudes are stable, $\mathcal{F}_{filt}\left[l, h, \cdot\right]$ represents the EM energy transfer function in the frequency band $[l, h]$, and $\mathcal{F}_{data}(\cdot)$ is the digital data transmission function that maps a 2D ground-truth image to a 1D bit stream transmitted by the data wires [162]. Although theoretically, all the non-deterministic functions and variables in Eq. (5.1) are dependent on the environment and challenging to measure and model accurately, we found that simplified approximations (e.g., setting $\mathcal{F}_{filt}$ to a constant in the sampled frequency range) can produce simulated images that have very close quality and characteristics to the actual EM reconstructions. Fig. 5.6 provides some examples of the simulated and actual reconstructions using $\mathcal{R}_{base}$.

5.4.2.3 Key Characteristics

Based on the model, we then investigate several key observations of the eavesdropped images and analyze their causalities.

Baseband Leakage Frequency Dependency. The emitted EM signals are baseband signals of the digital bits instead of narrow-band signals that are modulated onto certain carriers such as clock frequencies of the system, which are more common for intentional communication systems. Since the baseband signal is wideband, every frequency band can

contain different information about the ground-truth image. For example, Fig. 5.4 shows how 204 MHz and 255 MHz better capture the edge and gray-scale of the ground truth respectively. In practice, the adversary can only sample a subset of the digital wide-band information at a time. Advanced adversaries may thus need to combine information from different frequency bands. Besides the different information contained, each frequency band also has its unique EM wave propagation efficiency (transfer function) that leads to different SNRs for the adversary's received signals. We find that frequency components near the fundamental and harmonic frequencies of the digital transmission byte frequency (51 MHz) have the strongest signal strengths and lead to the best-quality reconstructions. This is because of the strong periodicity of transmitted bytes, leading to high EM amplitudes at these frequencies that can tolerate environmental noise better.

Multi-wire Signal Polarity Inversion. Another key phenomenon is that at certain frequency bands that contain f_{clk} and its harmonics, the amplitude of the EM signals could be inverted when the antenna moves relative to the cameras, leading to inversion of the reconstructed image's grayscale polarity. Based on this observation, we hypothesize that the inversion of polarity is caused by the superposition of EM signals emitted by the data and clock wires. We then verified our hypothesis by measuring emissions from the clock and data lines separately (see [162] for details). Essentially, the clock emissions can interfere with the EM emissions from data wires. When the antenna is placed at a position that receives EM signals as a mixture of the data clock wire signals, the two signals can cancel each other out, producing an image that approximates a white image subtracted by the data line image. This image thus has an inverted polarity compared to the data line-only reconstructions.

Practical Sampling Distortion. We observe well-structured distortion patterns in all reconstructions, including:

- Loss of color information. Only gray-scale information remains in the reconstructions.
- Shuffled gray-scale mapping. The original and reconstructed images have different but correlated gray scales.
- Light gradient & high-frequency noise. Light gradients result in ellipse/contour-like shapes that are not visible in the original camera images, e.g., in Fig. 5.6. The reconstructions also have additional high-frequency noise.

Such distortion patterns are caused by the imperfect sampling of the EM leakage signals that adversaries could achieve in practice. The imperfection is two-fold. First, adversaries often can only sample an EM signal bandwidth on the order of 10 MHz with common USRPs and laptops while digital image transmissions have bandwidths on the order of 1 GHz. This

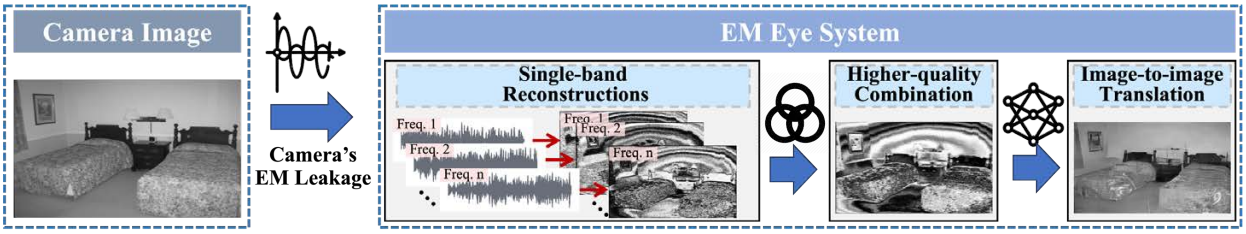


Figure 5.7: The image eavesdropping pipeline of EM Eye.

causes the loss of a significant amount of information. Second, even if a hypothetical adversary can sample the whole bandwidth, e.g., by using multiple USRPs or sampling multiple times, it is still impractical for them to recover the original bit stream transmitted because of the added noise during EM propagation and the requirement of perfect synchronization for determining which bit is being transmitted. With these problems in mind, we can analyze the causality of the distortions above.

To recover the RGB colors of images using debayering, the adversary needs to know the original gray-scale value of each pixel precisely which requires perfect sampling of the digital bits and is thus impractical. In the original image, the gray-scale values represent an ordered array of bits; in the reconstructions, the gray-scale values represent the EM signal amplitudes which approximately correspond to the numbers of bit flips in the array. As a result, gray-scale values of the camera outputs are mapped to different values in the reconstructions in a shuffled but deterministic way. For example, bright lights and windows in the original images are often mapped to dark polygons in the EM reconstructions (see the first two columns of Fig. 5.6 for example). This is because the saturated bright pixels in the original image are mapped to constant ones in the transmitted digital data and cause significantly lower EM amplitudes due to the few bit flips.

The high-frequency noise exists everywhere in the reconstructed images while the light gradient distortions appear mostly on single-color surfaces in the scene. The culprit of light gradient and high-frequency noise is the loss of data structure due to imperfect sampling. Specifically, it is because *the EM emissions of different bits get combined without correct bit ordering.* In the original digital transmission protocol of cameras, each bit has its own weight and the ground-truth pixel value is calculated by $v_{GT} = \sum_{i=0}^{9} 2^i Bi$. The adversary, however, can only calculate the pixel values while losing bit-ordering information in practice, because it is challenging to determine the current bit being transmitted. Practically, all bits are considered equivalent whose EM emissions are added up without weights assigned. Conceptually, this can be modeled as $v_{EM} = \sum_{i=0}^{9} Bi$ which amplifies the light intensity variations and high-frequency noises that are often embedded in the least significant bits.

5.4.2.4 Insights

Our investigation reveals several challenges and opportunities for adversaries to reconstruct higher-quality images compared to the basic reconstructions presented above. (a) The frequency dependency problem calls for a method for integrating information in different frequency bands in order to harvest more entropy from the original camera outputs. (b) Although the multi-wire signal polarity inversion does not affect human visual perception significantly, it can cause additional noise to automated data processing and pattern recognition pipelines and thus needs to be mitigated to improve the eavesdropping performance. (c) The practical sampling distortions cause obvious degradation of the images' visual quality and intelligibility. As a result, the adversaries need to employ additional techniques for correcting these distortions. We will introduce the improved eavesdropping design that supports adversaries to extend their performance limits in the next section.

5.4.3 Relationship with Computer Display Eavesdropping

We discover that the eavesdropping vulnerability of embedded cameras shares the same physical principle as previous computer display eavesdropping attacks (Section 5.2.1) where the transmitted plain digital image data leaks in the form of EM waves. Furthermore, we confirm that all the key phenomena above are also observable when we replicate computer display eavesdropping attacks following previous research. However, many of these phenomena such as light gradient amplification and polarity inversions have not been reported and analyzed before. We believe this is because computer display eavesdropping only investigated simple screen contents of uniform texts on uniform backgrounds (e.g., no light gradients), which do not suffer significantly from the practical sampling distortions. In contrast, camera image scenes have more complex and diverse structures and textures, posing greater challenges for adversaries to reconstruct intelligible images. In addition, our survey shows that amplitude demodulation has also been the state-of-the-art method for mapping 1D EM signals to scalar pixel values in display eavesdropping attacks, which confirms our design choice in Section 5.4.1.

5.5 Eavesdropping System Design

To support the evaluation of eavesdropping limits and factors, we design a system that employs the signal processing pipeline shown in Fig. 5.7. The adversary first finds at least one frequency band that contains the EM leakage of transmitted digital image data. For each frequency band, the adversary reconstructs a single-band EM image from the received EM

signals in this band. The adversary then strategically combines the images from different available frequency bands using a distortion-guided combination algorithm. The output of this algorithm, i.e., the multi-band EM image, is then input into an image-to-image translation network to acquire a final reconstructed image.

5.5.1 Single-band Image Reconstruction

The single-band image reconstruction process $\mathcal{R}_{base}$ on each frame can be formulated as

$$\begin{cases} I_{EM}^{[l,h]}[i_r, i_c] = \frac{1}{n_{samp}} \sum_{n=n_1}^{n_2} a[n] \\ n_{samp} = n_2 - n_1 + 1, a[n] = \mathcal{F}_{amd}[m[n]] \\ n_1 = \lfloor f_s(i_f T_f + i_r T_r + i_c T_c) \rfloor \\ n_2 = \lfloor f_s(i_f T_f + i_r T_r + (i_c + 1) T_c) \rfloor, \end{cases} \tag{5.2}$$

where i_f, i_r, i_c are the frame, row, and column indexes, T_f, T_r, T_c are the frame, row, and column transmission duration that needs to be estimated by the adversary through EM measurements, $m[n]$ is the discrete IQ measurements output of USRP with a sampling rate f_s, and $\mathcal{F}_{amd}[\cdot]$ is the amplitude demodulation function. Apparently, when f_s is on the order of 10 MHz in practical settings, n_1 and n_2 will be the same which is also the same for multiple consecutive i_c. This means the actual column resolution W_{EM} of the reconstructed image is smaller than the transmitted image and is determined by $W_{EM} = f_s T_{fd}/H_{EM}$ where H_{EM} is the row resolution that remains the same as the original transmitted image and T_{fd} is the actual frame data transmission duration excluding inter-frame blanking. As a result, T_c degrades to T_r/W_{EM} in most cases and does not need to be estimated separately. To improve the signal quality, we also perform frame averaging on the consecutive frames of camera outputs, which aims to mitigate the random noise in the EM wave propagation process and help the useful signals stand out. It is worth noting that this reconstruction process is also the current state-of-the-art (SOTA) used in computer display eavesdropping attacks, which we use as a building block as well as a baseline for our enhanced image reconstruction pipeline. We conduct an additional polarity-correction step that compares single-band reconstructions with data wire-only simulations and inverts the polarity if inversion is detected. We then apply histogram equalization to the image to further reduce the impact of clock signal offset b_{clk} (Eq. (5.1)) on image contrast.

5.5.2 Distortion-guided Multi-band Combination

We design a combination criterion based on the heuristic that the best combination can mitigate the light gradient distortions on single-color surfaces to the largest degree. As Section 5.4.2.3 points out, the light gradient distortions arise because the bit-ordering information is lost. For example, both B2 and B6 have a periodicity of 8-bit cycles in RAW10 (Fig. 5.5), producing the same EM frequency that cannot be separated apart. Nevertheless, we observe that different frequency bands could still contain some inter-bit information. For example, if the 8-bit cycle frequency is a Hz, then the frequency of $2a$ Hz embeds the variation between B2 and B6. Similarly, we know that *different frequency bands embed different inter-bit information.* As a result, we propose that an adversary who can perform multi-band combination effectively should be able to minimize the light gradient distortions to restore the single-color surfaces. In our experiments, we empirically formulated this as

$$\hat{I}_{EM} = \sum_{i=0}^{N} w_i \cdot I_{EM}^{[l_i,h_i]}, \;\; s.t. \;\; [w_i] = \min_{[w_i]} \; ||c - S(\hat{I}_{EM})||, \tag{5.3}$$

where N is the number of available bands, w_i is the weight of band i, $S[\cdot]$ is a segmentation function that allows the adversary to manually select a subarea of the image that is likely a single-color surface, and c is a constant that the adversary can select to represent the color (gray scale) of the surface. Note that such an operation is possible because the single-band EM images often contain important structural information about the scene and experienced adversaries are able to hypothesize some key objects in the scene such as the walls of a room (see Fig. 5.7 for example). When selecting the frequency bands to combine, we also employ a thresholding criterion similar to [76] in order to remove components that are too noisy. Fig. 5.7 shows an example of this process. Typical values of N are in the range of 1-3 in our evaluations.

5.5.3 Image-to-image Translation

To further mitigate the remaining image distortions, we employ a supervised image-to-image translation process. This is inspired by our observation that additional semantic information in the image domain can be utilized to reconstruct images that are closer to the ground-truth image. For example, when observing the remaining light gradient distortion patterns, experienced human adversaries are able to understand that these distorted areas are likely to be single-color surfaces (which have the strongest light gradients) in the original camera output and thus manually correct the images. Another example is that the dark polygons in the EM reconstructions often map to the bright lights and windows in the original images. Given

the very structured mappings, we hypothesize that it is possible to automate this process of correcting structured distortions in the EM reconstructions using machine learning-based approaches.

To verify this hypothesis, we formulate the task as an image-to-image translation problem from the EM-reconstructed image space to the original camera output space. We adopt pix2pix [128], an aligned image translation model based on a conditional generative adversarial network (GAN) to reconstruct a higher-quality image I_{EM} from $\hat{I}_{EM}$. Fig. 5.7 demonstrates an example of the translated reconstruction image in comparison with the gray-scale ground truth. We find the translation process capable of removing almost all remaining distortions when the testing images are within a reasonable range of variation compared to the training images. Although the generative model can also recover similar colors, color information is often less useful for image pattern recognition tasks. In addition, the color recovery problem only relies on image semantic information and is completely detached from the EM leakage physics. We thus focus on gray-scale images in our following evaluations.

5.6 Evaluation

5.6.1 Overview

Our evaluation seeks to measure the limits of the embedded camera eavesdropping risks under various camera designs and environmental conditions.

Experimental Setup. To provide reproducibility and scalability over multiple devices, we use the same setup as Section 5.4.1 where images of different scenes are displayed by a monitor screen and recorded by the cameras under test. We utilize two existing datasets to cover the common camera scenes pertinent to the threat model. The first dataset is a subset of the Face Detection Data Set and Benchmark [129] and has 3000 randomly selected images, each containing at least one person in the scene. The second dataset is a subset of the MIT Indoor Scenes Benchmark [189] that also has 3000 randomly selected images. Since the supervised image-to-image translation requires a training phase, we use 2700 images' corresponding $\hat{I}_{EM}$ from each dataset for training. In Section 5.6.2, we calculate the quantitative metrics over all 600 test images to evaluate the performance of the eavesdropping pipeline. To support scalable tests with fine-grained variations in the evaluation of factors and COTS devices, we also use a randomly-selected test subset of 35 images for each dataset which provides a confidence level of 90% at a resolution of 0.5 times the standard deviation of the test set population's scores [1]. For the training of the image-to-image network, we use the default hyper-parameters of the model [128] with 100 training epochs. We then use

the last epoch's model as the final network. By default, we use the same model trained on a base case (Section 5.6.2) to test various test sets to examine the generalizability of this supervised network over different factors. The only exception is the evaluation of different camera sensors and controllers (Section 5.6.2) where we also train models using their own EM reconstructions as a comparison to investigate the improvement of dedicated image translation models. In total, we have collected 32400 training images and 10460 test images. We use an EM sampling rate (f_s) of 8 MHz in all experiments.

Quantitative Metrics. To quantify the impact of different factors on the eavesdropped information on both the EM signal and the image perception levels, we use the following metrics:

1. Unintentional signal-to-noise ratio (USNR) calculates the ratio of the unintentional EM emission power to the background noise power[76].

2. Structural similarity index measure (SSIM) measures the similarity between the eavesdropped and ground-truth camera images.

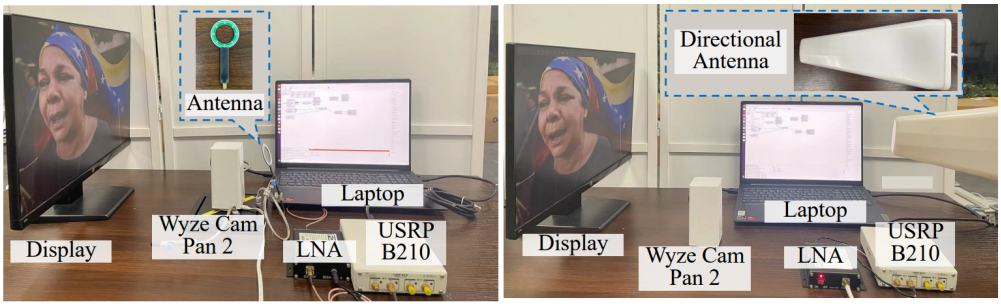
3. Face detection rate (Fdetect) calculates the ratio between the number of faces detected in the eavesdropped and ground-truth face dataset images.

4. Indoor scene captioning rate (Icaption) calculates the ratio of the longest common subsequence between the descriptional caption texts generated from the eavesdropped and ground-truth indoor dataset images.

SSIM, Fdetect, and Icaption range from 0 to 1, with larger values representing closer replicates of the ground truth. Apparently, the meanings of the absolute values are less intuitive. We thus also show example images corresponding to different values in our evaluations. Nevertheless, the variations of these metrics can still inform us of how different factors affect the quality of reconstructed images. Different from previous computer display eavesdropping research whose targets are simple texts, Fdetect and Icaption are specifically designed by us to measure how machines/humans perceive the complex visual information in camera scenes.

5.6.2 Sensor and Controller

As pointed out in Section 5.3.2, the camera data transmission interface can connect various camera sensors and controllers from different manufacturers. Given that different models of sensors and controllers could change the image data processed and transmitted, we first evaluate the impact of them on EM Eye's performance (Table 5.1). We employ Raspberry Pi 3B+ and Cam V1 (#1) as the base case for collecting $\hat{I}_{EM}$ to train a base model (TrainA).



(a) (b)

Figure 5.8: Experiment setups of using (a) a near-field probe within 10 cm and (b) a directional antenna beyond 10 cm.

We then change the sensors and controllers and collect corresponding $\hat{I}_{EM}$ to train their own models (TrainB).

The TrainA results in Table 5.1 suggest that sensors have a larger impact on the EM reconstructions than controllers. When the sensors change (e.g., [#1, #3, #6]), we observe larger degrees of variations in the image quality than when the controllers change (e.g., [#1, #2] and [#3, #4, #5]). This can be explained by the fact that it is often the sensors that decide the image data's format, amount, transmission speed, etc. The signals that EM Eye eavesdrops on are all produced by the camera sensors while the downstream processors mostly perform post-processing of the image data. Besides sensor hardware that determines the maximum supported image capacity, each camera sensor can also be configured to have various software/firmware settings such as resolution, frame rate, and sensor mode. Our tests show that setting the camera resolution does not change the transmitted data and EM emissions because the sensor always transmits the full resolution supported by a certain sensor mode and lets ISPs to down-sample the images in software. A different frame rate will change the number of frames transmitted per second and require the adversary to adjust the eavesdropping frame rate setting accordingly. Different sensor modes [187], which are combinations of camera firmware settings that decide the actual resolutions used by the sensor chips, will change the width and height of transmitted images and require the change of eavesdropping parameters.

Fig. 5.9 compares some examples of direct EM reconstructions using state-of-the-art (SOTA) techniques and enhanced reconstructions using the EM Eye pipeline. Overall, obvious improvements in the visual quality are observed. The only caveat is that the image-to-image translation network can sometimes distort certain details of the images such as small textual objects. In these cases, the adversary may refer to the untranslated images $\hat{I}_{EM}$ to capture such information. Table 5.1 show the percentage of improvement in the

Table 5.1: Evaluation results of EM Eye on 6 sets of sensor and controller.

#	Sensor Module (Reconstruction Parameters)†	Controller Module	USNR (dB)	$W_{EM} \times H_{EM}$	TrainA(Improvement)* SSIM	Fdetect	Icaption	TrainB(Improvement)* SSIM	Fdetect	Icaption
1	Cam V1: OV5647	Raspberry 3B+ (Base)	39.68	186 × 1080	0.58(↑235.0%)	0.80(↑78.2%)	0.33(↑21.6%)	N/A	N/A	N/A
2	(T_f: 33.31 ms, T_r: 29.58 us)	Raspberry 4B	40.30	186 × 1080	0.45(↑221.8%)	0.75(↑50.7%)	0.29(↑19.1%)	0.55(↑298.8%)	0.78(↑57.7%)	0.32(↑30.9%)
3	Cam V2: IMX219	Raspberry 3B+	41.34	84 × 1290	0.29(↑186.9%)	0.51(↑95.5%)	0.23(↑80.8%)	0.45(↑349.4%)	0.70(↑168.3%)	0.27(↑115.5%)
4	(T_f: 33.84 ms, T_r: 18.90 us)	Nvidia Jetson Nano	42.51	84 × 1080	0.30(↑132.4%)	0.35(↑102.4%)	0.21(↑71.5%)	0.43(↑240.1%)	0.69(↑298.2%)	0.27(↑117.5%)
5		Asus Tinkerboard 2S	40.47	144 × 2466	0.39(↑112.5%)	0.60(↑79.5%)	0.26(↑49.6%)	0.53(↑193.0%)	0.76(↑129.6%)	0.31(↑78.7%)
6	Cam V3: IMX708 (T_f: 33.24 ms, T_r: 26.72 us)	Raspberry 3B+	43.54	104 × 1080	0.34(↑110.4%)	0.52(↑27.1%)	0.20(↑70.5%)	0.48(↑199.6%)	0.68(↑65.0%)	0.24(↑100.7%)

† The frame duration T_f and row duration T_r need to be estimated to decode the eavesdropped EM emission to reconstruct the images.
* EM Eye is evaluated on TrainA (base model) and TrainB (retrained model), and the percentage represents the improvement over the SOTA approach.

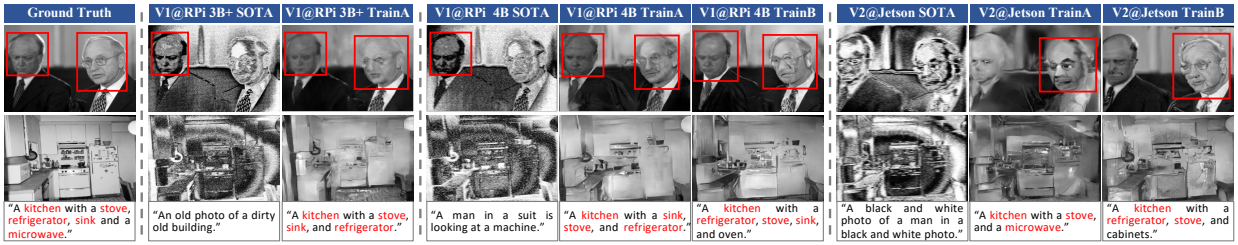


Figure 5.9: Examples of eavesdropped images from three camera-controller systems using the SOTA and EM Eye pipelines, where A is the camera and B is the controller in A@B. Training dedicated models for each camera-controller combination (TrainB) provides better results than the base case model (TrainA). The detected faces of the face dataset images and the generated captions of the indoor dataset images are shown.

quantitative image quality metrics compared to the SOTA results. On average, we observe 166.5%, 72.2%, and 52.2% increases in the SSIM, Fdetect, and Icaption scores for TrainA. The average values increase to 256.2%, 143.7%, and 88.7% for TrainB. The comparison between the metrics in TrainA and TrainB also shows that dedicated image translation models trained with each sensor-controller combination's EM data can indeed improve the quality of the eavesdropped images. The EM emissions of RPi 4B with Cam V1 are the most similar to the base case while those of Nvidia Jetson Nano with Cam V2 are the most dissimilar. The non-trivial metrics of all cases show that the base case model has a reasonable level of generalizability to process data from various sensors and controllers.

Summary. Different sensors and controllers can affect the EM signals while the EM Eye pipeline is able to reconstruct images with various sensor and controller settings. It also provides sufficient generalizability to allow the reconstructed images to outperform the SOTA results of direct EM reconstructions in most cases. In addition, resourceful adversaries may train dedicated models on each target camera system to further improve the eavesdropping performance.

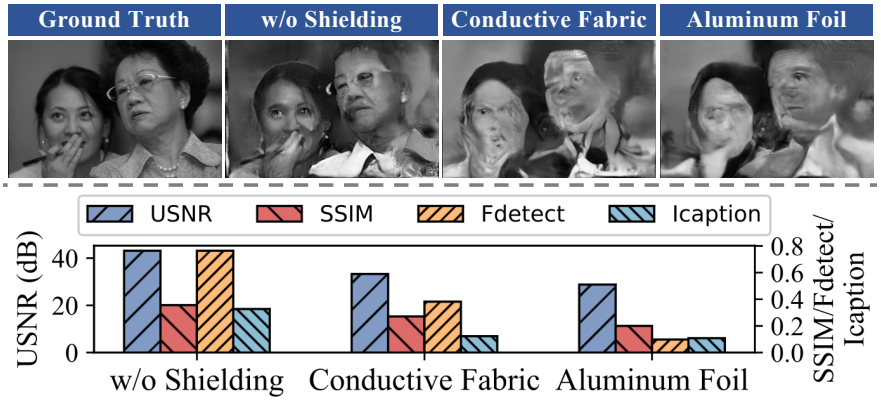


Figure 5.10: Illustrations of (bottom) the impact of different cable EMI shielding, and (top) the same image reconstructed with different cable EMI shielding.

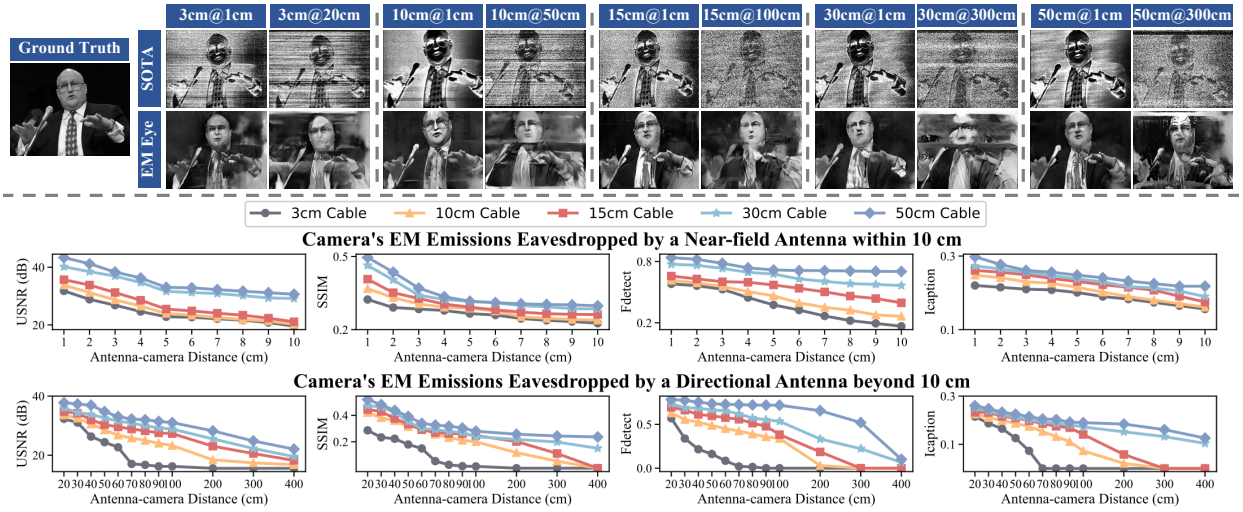


Figure 5.11: Illustrations of (bottom) the impact of distances with different cable lengths, and (top) the same image reconstructed at different distances with different cable lengths, where A is the cable length and B is the distance in A@B.

5.6.3 Transmission Cable & Environmental Factors

Next, we measure the limits of EM Eye under various physical factors of the transmission cable and environment.

Cable EM Shielding. EM shielding uses special cable shield materials to block or reduce the propagation of EM waves. We evaluate its impact using 15 cm cables in three forms, namely the default cable of Raspberry Pi cameras without shielding, a cable shielded with conductive fabric, and one with aluminium foil. We use a near-field antenna to capture the EM emissions at a distance of 1 cm, and compare the values of USNR, SSIM, Fdetect, and Icaption for each cable with the same experimental setup. We depict the results in Fig. 5.10

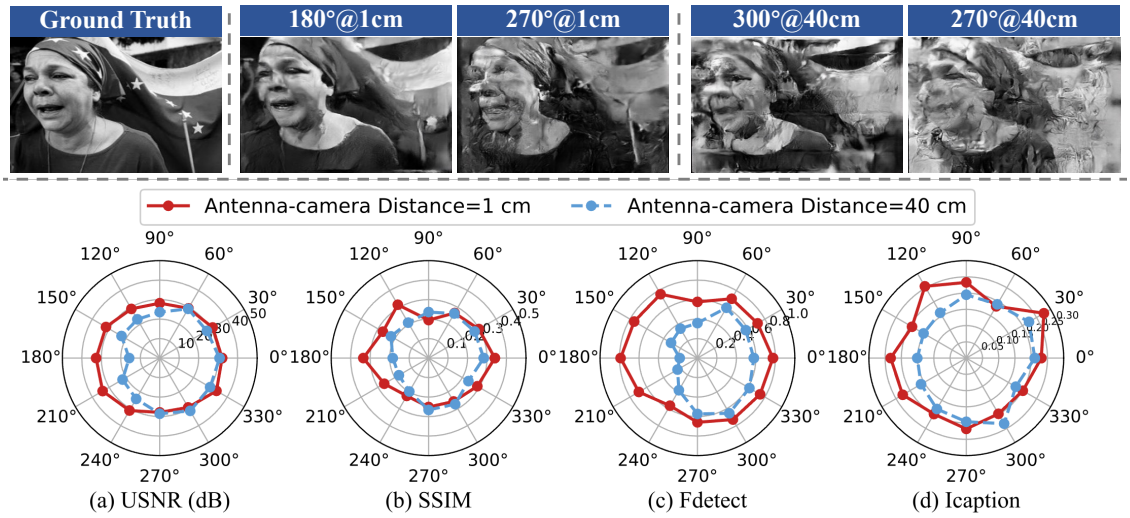


Figure 5.12: Illustrations of (bottom) the impact of angles at 1 cm and 40 cm, and (top) the same image reconstructed at different angles at these two distances, where A is the antenna-camera angle and B is the distance in A@B.

(bottom). The cable shielded with conductive fabric and aluminium foil material significantly reduces the intensity of the EM emission radiated by the cable. We observe 9.84 dB and 14.33 dB decrease in USNR values respectively. Nevertheless, it is still possible to reconstruct images with acceptable SSIM and Fdetect values on these two shielded cables as shown in Fig. 5.10 (top).

Antenna-camera Distance and Cable Length. With the transmission cable acting as an unintentional antenna, the strength of EM emission attenuates with the antenna-camera distance. To quantify the impact, we measure the metrics under different distances with five typical cable lengths, namely 3, 10, 15, 30, and 50 cm. We use a near-field probe in Fig. 5.8 and a directional antenna in Fig. 5.8 with the same experimental setup to capture the EM Emission within and beyond 10 cm. The results are shown in Fig. 5.11. Notably, USNR, SSIM, Fdetect, and Icaption values gradually decrease with increasing distances, and longer cables often have higher values for these metrics at the same distance in our experiments. As shown in Fig. 5.11 (top), we observe almost unanimously better-quality images with longer cables. This is because the gains of different cable lengths vary, and longer cables provide a larger effective area, resulting in greater efficiency in radiating EM waves [54]. The maximum distances we could achieve for 3 cm, 10 cm, standard 15 cm, 35 cm and 50 cm cables are 50 cm, 200 cm, 270 cm, 400 cm, and 450 cm respectively. We note that the distance can be further increased by employing a professional antenna with superior directionality and gain.

Antenna-camera Angle. To examine the impact of camera-antenna angles, we change

Table 5.2: Evaluation results of EM Eye on 12 COTS camera devices.

#	COTS Camera Devices		Reconstruction Parameters†			EM Eye Performance						Scenarios	
	Manu. and Model	Year	T_f (ms)	T_r (us)	Freq.	USNR	$W_{EM} \times H_{EM}$	SSIM	Fdetect	Icaption	Max. Dist.*	HA	PI
1	Google Pixel 1	2013	33.45	21.49	600,1649 MHz	42.17 dB	168×1140	0.30	0.44	0.19	30 cm	✓	✗
2	Google Pixel 3	2018	33.27	10.89	515,680 MHz	41.81 dB	74×2840	0.24	0.36	0.19	2 cm	✓	✗
3	Samsung S6	2015	33.32	10.50	527,1054 MHz	38.92 dB	184×3000	0.31	0.70	0.19	5 cm	✓	✗
4	ZTE Z557	2019	41.70	17.00	522,1740 MHz	35.09 dB	310×1940	0.28	0.68	0.14	1 cm	✓	✗
5	Wyze Cam Pan 2	2019	49.98	29.63	890,1185 MHz	42.39 dB	164×1080	0.31	0.43	0.23	350 cm	✓	✓
6	Xiaomi Dafang	2019	66.66	29.63	322,890 MHz	39.06 dB	190×1080	0.35	0.67	0.17	500 cm	✓	✓
7	Baidu Xiaodu X9	2023	66.67	53.33	204,1470 MHz	35.86 dB	460×1080	0.24	0.23	0.15	200 cm	✓	✓
8	TeGongMao	2023	66.00	44.00	763,1144 MHz	40.58 dB	190×720	0.19	0.24	0.14	120 cm	✓	✓
9	Goov V9	2022	33.00	44.00	546,656 MHz	33.79 dB	190×720	0.31	0.32	0.18	70 cm	✓	✓
10	QiaoDu	2021	66.48	29.61	293,1191 MHz	38.79 dB	84×1080	0.22	0.38	0.17	50 cm	✓	✓
11	360 M320 Dashcam	2020	40.00	22.00	450,1261 MHz	39.71 dB	142×1440	0.29	0.17	0.22	250 cm	✓	✓
12	Blackview Dashcam	2022	33.22	27.78	155,1015 MHz	34.38 dB	190×1080	0.30	0.21	0.24	300 cm	✓	✓

† We only report two frequencies of the strongest emission.
* The maximum distance can be further increased by using higher-end EM receiving equipment such as professional direction antennas and analog filters.

the angle from 0 o 360 ith a step of 30 12 angles in total). The angle is defined as the angle between the centerline of the camera cable and the antenna. We conduct two sets of experiments using a near-field probe at a distance of 3 cm and a directional antenna at 40 cm respectively. Fig. 5.12 shows the impact of angles with the quantitative metrics. The angle has a small impact on EM Eye's performance at a close distance while some angles slightly outperform others. Due to the nature of the directional antenna, the angle has more impact on the eavesdropped images at a larger antenna-camera distance. As shown in Fig. 5.12, when the angle is between 90 nd 270 t 40 cm, the values of these three metrics are significantly lower than when the angle is between 0 o 90 r 270 o 360

Interference from Electrical Devices and Background Noises. *(a) Electrical Devices.* The interference from displays of some electrical devices (such as TV, monitor, smartphone, etc.) is the most likely to affect EM Eye since the EM emission pattern of these displays is similar to that of the camera. However, modern displays offer refresh rates of 60, 120, or even 240 fps [213], whereas embedded cameras' frame rates are often limited to 30 fps. Therefore, adversaries can distinguish camera emissions from the display's interference by setting the center frequency at those frequencies with no repetitions above 30 Hz to minimize the interference. We have verified this through experiments. Besides, the EM emission pattern of cameras is very different from that of earbuds [76], recorders [260], wireless eavesdroppers [205, 69], etc. *(b) Background Noises.* Since EM Eye works at various frequencies, adversaries can improve image quality by avoiding selecting eavesdropping frequencies that conflict with common communication frequency bands. It is also effective to use analog filters to filter out background noises.

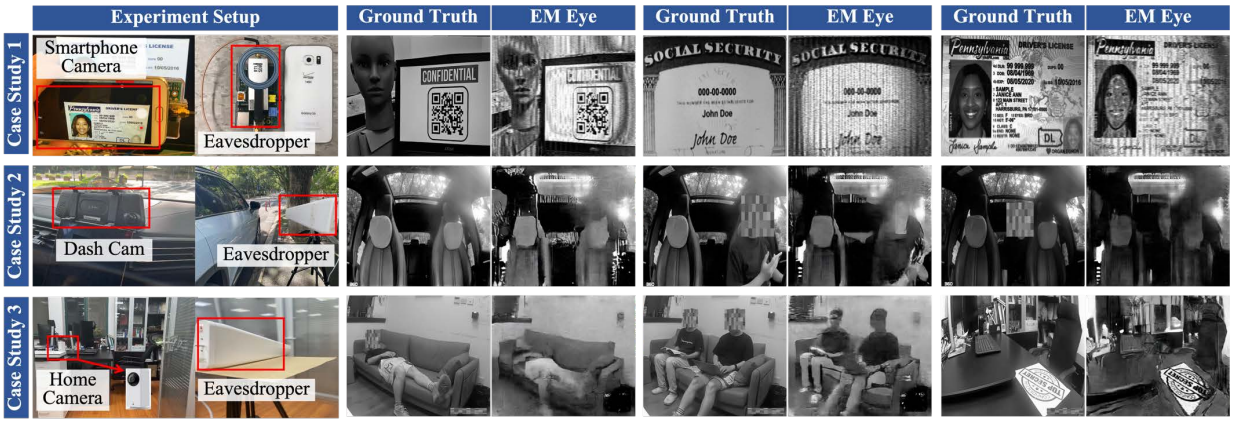


Figure 5.13: Three case studies of how EM Eye poses eavesdropping threats against smartphones, dash cams, and home security cameras. For each case, the experimental setup and three examples of ground truths and eavesdropped images are shown.

5.6.4 COTS Camera Devices & Case Study

We have evaluated EM Eye on 12 commercial-off-the-shelf (COTS) camera devices from three different categories to investigate the common use cases of embedded cameras. These include 4 smartphones, 6 smart home cameras, and 2 dash cams. All of these devices are intact with their original packaging. Table 5.2 shows the specifications and eavesdropping parameters of these devices. Besides evaluating the eavesdropped image quality at 1 cm, we also measured the approximate maximum eavesdropping distance for each device at which we can still recover intelligible images. The maximum distances vary from 1 cm to 500 cm and with significant differences across devices. While all devices can be eavesdropped on in hidden-antenna scenarios where the antenna is close to the camera, we also observe that 8 out of the 12 devices allow adversaries to perform physical-isolation eavesdropping through windows, doors, and walls. We believe the variations in eavesdropping distances are mostly decided by the length and shielding materials used by these devices. For example, we found that smartphones often use short cables with better shielding designs to minimize the EM interference between the onboard components. Dash cams and home security cameras, on the other hand, tend to use cheap unshielded cables to reduce the manufacturing cost and longer cables to support different form factors of the mechanical structures. Despite the variations in these devices' designs, we note that the EM Eye vulnerability is a shared problem in common embedded camera devices, and we have reported our findings to the camera vendors. Based on the results above, we carry out case studies on three typical attack scenarios that we envision to be applicable to the threat model.

Smartphone Camera Eavesdropping. Since smartphone camera emissions only allow

adversaries to eavesdrop from a close distance, we envision a hidden-antenna scenario where the antenna and EM signal receiver could be installed in modified power banks. Such power banks may either be tampered with from the supply chain as distributed products or provided by shared power bank rentals that are common in shopping malls. Existing COTS products of miniaturized low-cost SDR receivers such as the RTL-SDR dongles [193] suggest the possibility of manufacturing such power banks. Fig. 5.13 (top) showcases an envisioned prototype and three sensitive images eavesdropped when the victim takes photos of private documents, including a QR code, a social security card, and a driver's license, with a Samsung S6 phone.

In-car Peeking. When victims park their cars with their interior dash cams on, an adversary may be able to peek at the inside of the cars using EM Eye eavesdropping from nearby. Fig. 5.13 (middle) shows an example setup with a 360 M320 dashcam [40] on the dash board of the car. The adversary sets up an antenna 50 cm away from the car (100 cm antenna-camera distance) to capture the EM emissions. Three eavesdropped images reveal no one in the car, one person in the driver's seat using his phone, and one in the back seat. When needed, the eavesdropping equipment can also be made portable as a suitcase, as has been demonstrated in previous research [116], to avoid further drawing the attention of the cars' owners.

Through-wall Room Spying. Another typical physical-isolation eavesdropping scenario involves an adversary spying on a private household or office room through the EM emissions of the IoT home security camera. The convention of installing such security cameras near the room's walls, windows, and doors could allow the adversary to receive the camera's EM emissions from only a few meters away. Fig. 5.13 (bottom) demonstrates a case where the antenna is placed 70 cm away outside an office room (150 cm antenna-camera distance). The adversary can see a person sleeping on a couch, two people sitting on the couch, and a confidential document on a desk by eavesdropping on a Xiaomi Dafang home camera [103].

5.7 Mitigation

We analyze the possible countermeasures from the standpoint of camera and system designers.

5.7.1 Naive Protections

EM Jamming. Jamming is a common technique used to disrupt intentional communication systems. However, we believe jamming is less suitable for mitigating camera eavesdropping given that the leaked signals are wide-band, requiring an expensive device to cover such a wide bandwidth. Furthermore, jamming can easily compromise the legitimate camera data stream itself as has been demonstrated by [131, 141]. Jamming devices can either be installed by camera manufacturers or users. The challenge is it needs to cover a large space as the EM field distribution can be unpredictable and varying. This is based on our observations that different probe positions and orientations will lead to very different results.

Shorter Cables & Better Shielding. Our evaluation shows that short cables often produce weaker EM emissions, especially in the far field. Device manufacturers are thus encouraged to employ shorter cables in their designs. However, we note that such changes may also require a complete redesign of the devices' mechanical structures since it requires the camera lens to be very close to the controller boards. Otherwise, the manufacturers can consider using better-shielded data transmission cables, which have been shown to be capable of reducing the EM signal strength by over 10 dB.

5.7.1.1 Interface Design Improvment

Increase and Randomize Transmission Blanking. With the same frame rate and resolution of the transmitted images, increasing the blanking between frames and rows will reduce the effective resolution of the eavesdropped images under a certain eavesdropping sampling rate. This requires the transmission interface to have higher bit rates. Furthermore, adding intentional jitters to randomize the blanking duration can prevent adversaries from easily performing frame averaging and thus reduce the leakage USNR they receive.

Grouped Pixel Smoothing Protocol Improvement. We argue that the current image data transmission protocols are flawed and can be improved to mitigate EM leakage. Essentially, the EM emissions originate from the periodic bit flips. Ideally, the order of transmitted rows, columns, and even bits should be randomized, eliminating all the periodicity. However, we also realize such randomization requires a complete hardware redesign and could be expensive for manufacturers. We thus seek to improve the protocol by keeping the overall architecture but minimizing the number of periodic bit flips. We achieve this by simply rearranging the bits. Specifically, we observe that adjacent pixels (columns) have similar values in their bits, especially the MSBs. By putting the same bits from adjacent pixels in a byte as shown in [162], we can smooth out many bit transitions and reduce the EM emission amplitudes. In addition, the more adjacent pixels grouped together in this

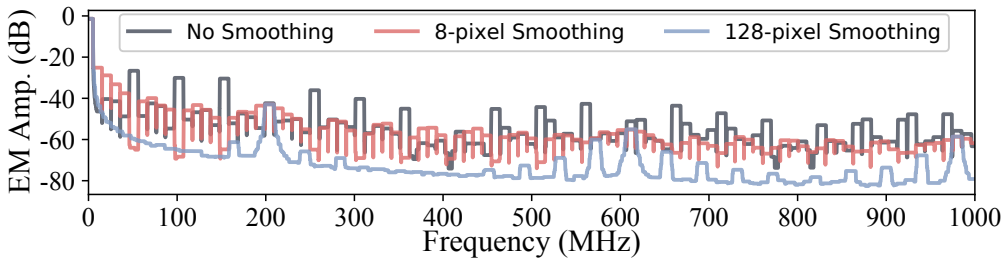


Figure 5.14: The simulated EM emission strengths with no defense and with the proposed grouped pixel smoothing in the transmission protocol design.

way, the fewer emissions there will be. Fig. 5.14 demonstrates the EM emission spectrum calculated by the simulation model (Eq. (5.1)) when there is no such defense and when 8 and 128 pixels are grouped together for smoothing, respectively. Most of the strong emission peaks at the multiples of the byte frequency (51 MHz) are mitigated by over 10 dB. Note that the original protocol already groups 8 pixels together in transmission, so supposedly 8-pixel smoothing requires minimal modifications to the interface designs.

5.7.2 Discussion: Other Sensing Devices

We believe the threat of EM side-channel eavesdropping may be further extended to other sensing devices.

Encoded Video Data Transmission. Although embedded systems widely use open-standard image data transmission interfaces that send uncompressed RAW data, many traditional camera devices such as USB webcams still use proprietary interfaces that send encoded (e.g., h264) video data. With such devices, the adversary cannot use the eavesdropping method of this work to directly reconstruct images. However, it is possible to use machine-learning-based classifiers to recognize human-unintelligible EM signals because image patterns can be recognized as long as the corresponding EM signals have sufficient separability. We experimented with a Logitech C920x HD Pro webcam which transmits compressed video data. We tried to classify 100 different face images recorded by the webcam using its EM emissions. We simply use the EM signals' Fast Fourier Transform coefficients processed by Linear Discriminant Analysis and a self-built three-layer neural network classifier. Even with the crude features, we could achieve a test accuracy of 90.12% for the 100-class classification. This makes us believe the eavesdropping threat can affect a wider range of cameras even if the adversary doesn't understand how data is transmitted.

General Sensors. Every sensor peripheral has to transmit data to the central processors. Most sensors transmit unencoded plain data. Given that the data throughput of most

sensors is much smaller than cameras, we believe the EM side-channel eavesdropping on other sensors could be achieved even with less sophisticated equipment and data reconstruction algorithms. In addition, eavesdropping on sensors used in industrial settings may not require the adversaries to be physically isolated from the sensors. For example, an employee trying to steal the secret specifications of a product that is being measured by a benchmarking device may physically approach the automated device to collect EM signals.

5.8 Conclusion

This chapter provides experimental evidence to support **H3**, focusing on electromagnetic side-channel leakage of camera data transmission interfaces. It further suggests that the requirement of **KR2** may not be met in most of the existing sensing systems because there are no protection mechanisms for various types of sensor data transmission in existing systems. It remains an important direction to further characterize the physical leakage-enabled eavesdropping risks against sensor data inputs. This chapter also shows how this problem can be connected to previous TEMPEST research and the obvious semantic gap that needs to be addressed in future research to apply the TEMPEST analysis framework to emerging sensing systems. Finally, our investigation shows that there is a large room for designers to improve both the hardware and interface protocol designs for better sensor data security.

CHAPTER 6

Injecting False Information Through Sensors Side Channels

6.1 Overview

Previous chapters have shown how secret information can be unintentionally captured by sensors, especially when s_{side} keeps increasing due to the higher sensitivity of emerging sensing systems. In another threat model concerning sensor data integrity, adversaries may be able to change how d_{sensor} reflects the value of s_{int} that the authentic users care about by intentionally generating physical signals to interfere with s_{side}. This chapter investigates this hypothesis **H3** with two case studies. In the first one [164], adversaries could use intentional electromagnetic interference (IEMI) to change the output of temperature sensors used in vaccine temperature cold chains, leading to spoiled vaccines and safety concerns (Section 6.2). In the second example [133], IEMI could trigger keyboard inputs by interfering with the signals perceived by the analog sensing circuits of both wired and wireless keyboards, leading to DoS or targeted keystroke injections (Section 6.3).

6.2 Case Study: Controlling Temperature Sensor Readings using Electromagnetic Interference

Protecting the global human population against COVID-19 depends on complex logistics and transportation of vaccines, often at unusually low, cryogenic temperatures. Moreover, malicious cybersecurity actors, both individuals and nation states, exist and have disrupted the vaccine supply chain.

In January 2021, a large U.S. healthcare system asked for help to protect its refrigeration systems from radiofrequency (RF)-based analog cybersecurity threats against the temperature sensors used in COVID-19 vaccine cold chain transportation and storage. It is

well-known in the security research community that intentional electromagnetic interference (EMI) can not only disrupt but also control the output of temperature sensors [230, 102].

With the goal of assessing potential RF-based risks facing COVID-19 vaccine cold chain and deriving accessible methods for protection, the authors conducted experimental and theoretical analyses that led to the following lessons learned:

- The experiments confirmed that EMI can disturb temperature sensors in cryogenic freezers.
- Precautions of simple physical and administrative controls can considerably reduce the risk of electronic tampering of the vaccine cold chain transportation and storage to ensure safety and effectiveness.
- Interdisciplinary research between the fields of biomedical engineering and embedded security results in discoveries that protect the health and safety of patients.

Multiple reports indicated that the U.S. quarantined more than 3,000 doses from Pfizer and 16,000 doses from Moderna vaccine shipments after the sensors reported unexplained anomalies in temperature readings [135]. During this event, which at the time of this writing remained under investigation, a question arose of how to defend sensors from potential analog cybersecurity threats.

Cybersecurity exploits can cause sensors monitoring the vaccine temperatures to detect falsely higher and/or lower readings, leading to deceptively incorrect excursions from critical temperature ranges. To ensure public confidence in the efficacy of the vaccines, it's important that cooling and monitoring systems operate within correct temperature ranges, even when sensors are malfunctioning or subjected to the threats. Moreover, automated regulatory compliance based on sensor readings could cause unintended, self-inflicted disruptions to the supply chain: Vaccines with temperature excursions in sensor readings are required to be recalled and analyzed by the manufacturer [100], causing further disruption to a vaccine in short supply.

6.2.1 Threat Model & Background

6.2.1.1 Threat Model

This work assumes a threat model where a physical external adversary generates IEMI signals in the vicinity of vaccine temperature monitors to control the readings of the temperature sensor. The adversary aims to induce falsely higher or lower temperatures to endanger the safety of the vaccines or disrupt the vaccine supply.

6.2.1.2 Known Threats & Regulations

Intentional EMI used against off-chip temperature sensors has been shown to affect sensor readings and thus disrupt the temperature monitoring and control of commercial devices that use such sensors. For example, research has shown that intentional EMI can be used to change the temperature readings of an infant incubator from a distance of 5 m or induce a temperature excursion of up to 40°C in a shielded hybridization oven used in laboratories [230].

The susceptibility of these devices depends on various factors, including the signal-conditioning circuit used to process the sensor signal and convert it into readable values for the users, the electronic components and materials used to fabricate the sensors, and the control system that regulates the behavior of the device in the case of closed-loop systems. These types of vulnerable temperature sensors also are widely used in vaccine cold chain transportation and storage [78]. Digital temperature loggers, which contain such sensors, are suggested by the Centers for Disease Control and Prevention (CDC) for COVID-19 vaccine handling [100].

These sensors consist of sensing units made of thermocouples, resistance temperature devices, or thermistors that transduce temperatures to electric signals. Subsequent signal conditioning circuits then convert the electric signals (voltages) into digital temperature readings (numbers). The threat arises because EMI can cause electric distortions on the wires between temperature sensors and embedded computer systems.

Today, embedded systems cannot distinguish between the authentic electric signals generated by the temperature and those by intentional EMI. Thus, the embedded computer systems will unknowingly accept false temperature readings from sensors fooled by intentional EMI. In other words, a malicious party can use EMI to drive the temperature readings for the vaccines higher or lower than its real value and cause false temperature excursions. Because EMI essentially refers to radio waves that can penetrate walls, malicious parties may launch this attack stealthily by generating EMI even in a different room from where vaccines are kept.

Sensor device manufacturers typically use methods such as metal shielding of the circuits and sensor probes to reduce the susceptibility to the interference. However, the real-world effectiveness of these practices is difficult to predict. The authors conducted preliminary tests of popular digital temperature loggers from two manufacturers that meet manufacturing practices and guidelines for cold chain transportation (one compliant with the EN 12830 standard and the other compliant with the 21 CFR part 11 standard). We found that both devices were susceptible to intentional EMI. In addition, we found that a real-time temperature monitor used in hospital settings can be attacked, causing the sensors to falsely

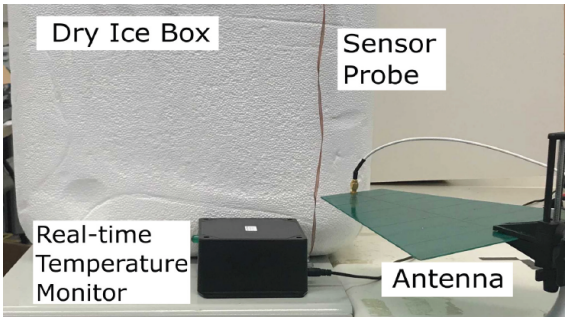


Figure 6.1: Experimental setup for measuring the temperature variation under intentional electromagnetic interference attack with a foam box filled with dry ice.

report both higher and lower temperatures.

Effective EMI frequencies range from 350 to 1,100 MHz, which can be easily generated with commercially available radio devices. With a maximum EMI output intensity of just 30 dBm (close to the maximum intensity of 3G mobile phones) and a distance of 0.1 m between the EMI output device and the target temperature sensors, the maximum temperature reading increase of the temperature-monitoring devices was +6°C and the maximum decrease was –38°C. In comparison, the EN 12830 standard enforces a ±1°C measurement error tolerance for temperature-monitoring devices and the CDC recommends ±0.5°C or less.

This degree of change in temperature readings can cause a critical temperature excursion and compromise vaccine shipments and storage. Because the EMI output intensity decides how large the electric distortion will be in the target sensor and how far the EMI signal can travel, a higher degree of change in temperature readings or a longer attack distance also can be achieved by a malicious party via use of higher-power radio devices. In an extreme case, previous research has shown that a high-power microwave generated with civilian equipment has the potential to perform a kilometer-range sabotage.

To show the impact of intentional EMI, we conducted a demonstrative experiment using the real-time temperature monitor used in hospital settings to measure the cryogenic temperature generated by dry ice in a foam box (Figure 6.1). Figure 6.2 shows how a malicious attacker can control the temperature readings of the real-time temperature monitor. In the tests, the malicious attacker causes controlled positive and negative temperature offsets by using different EMI frequencies and increases the offsets by using higher EMI intensity.

6.2.2 Temperature Sensing Security Analysis

The difficulty of mitigating intentional EMI threats against off-chip temperature sensors is threefold:

- Engineering efforts (e.g., RF shielding, EMI filters, twisted-pair cables) that make de-

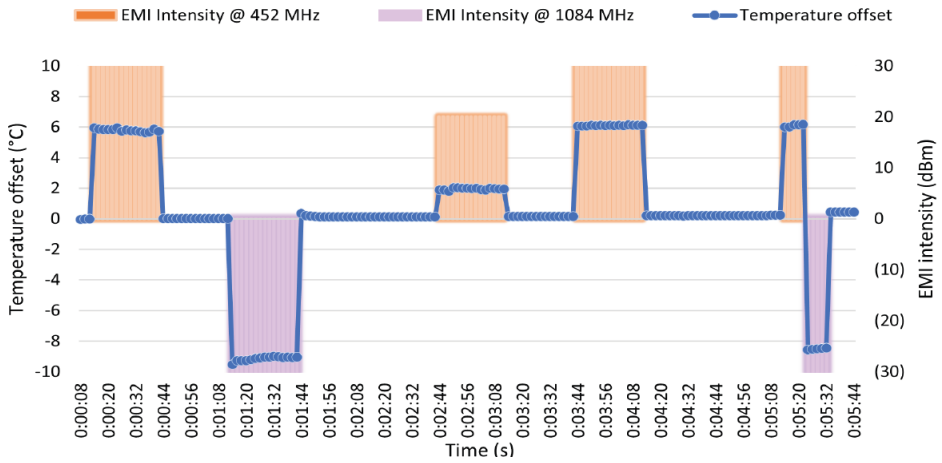


Figure 6.2: Real-time temperature monitor readings offsets under intentional electromagnetic interference (EMI) attack with dry ice (at –77°C) in three different scenarios. Test 1 (left): controlled positive and negative offsets resulting from 30 dBm EMI for 30 seconds; test 2 (center): controlled offset with increasing EMI intensity (20 and 30 dBm, respectively); test 3 (right): controlled rapidly changing offset.

vices pass the standard industrial electromagnetic compatibility tests have been shown to be insufficient for preventing an intentional EMI attack [230, 99].

- Temperature sensors that already are designed, manufactured, and deployed cannot be easily modified in a timely manner to mitigate intentional EMI threats because this often requires sophisticated hardware/circuit component modifications (e.g. modifying the signal conditioning circuit).

- Other countermeasures (e.g., sensor redundancy) might not effectively mitigate EMI threats because closely located sensors (e.g., as found with small refrigerators or vaccine transport boxes) can suffer from intentional EMI disruption during the attack and distantly located sensors cannot measure the accurate temperature in the vicinity of the vaccines. In addition, no standard technique currently exists for using redundant temperature sensors to prevent sensor attacks.

As a result of conventional countermeasures being insufficient, a substantial gap exists for mitigating intentional EMI threats against the vaccine cold chains in accessible and nonintrusive ways. In the current analysis, we address this gap by proposing a few simple measures that can effectively reduce the risk of malicious tampering with intentional EMI to near zero through approaches such as physical administrative controls.

The effort that the attacker needs to exert and the type of attack model are the two key points to consider when designing mitigation schemes. A malicious party needs to find certain frequencies for the EMI signals that can most successfully affect the target temperature

sensor. Some frequencies may increase the temperature reading, whereas others may decrease it; this depends on the specific sensor device model and the deployment scenario, which affects the electrical coupling path between the EMI source and target sensor.

To find the vulnerable frequencies for a particular temperature sensor, the malicious party needs to attempt different frequencies and observe corresponding changes in temperature readings. Without a feedback system, adversaries will have a difficult time guessing how their EMI is affecting a sensor's output. An adversary may consider a brute-force, wide-spectrum attack in an effort to eliminate the need for finding the vulnerable frequencies, but it comes at the cost of using considerably more expensive and rarer radio equipment that supports a very wide band (hundreds of megahertz) of RF.

Because only certain vulnerable frequencies (e.g., the resonant frequencies of a target device's circuit) can be exploited by the adversary to cause traceable changes in temperature readings, previous research on intentional EMI has focused primarily on vulnerable-frequency attacks. In the current analysis, we also address mitigation of vulnerable-frequency attacks.

Generally speaking, two types of threat models exist: off-site and on-site exploitation. In off-site exploitation, the attacker would know in advance the model of sensor devices being used. The attacker could acquire the same equipment and find the vulnerable frequencies in an off-site setting, then bring portable devices (e.g., walkie talkies, which are widely known to emit strong EMI) customized at these vulnerable frequencies to the proximity of vaccines and change temperature readings.

On-site exploitation, on the other hand, does not require prior knowledge of the sensor device model. The attacker can set up a laptop with radio antennas and software- defined radio devices, then tune the frequencies and observe corresponding temperature reading changes on the spot. Of course, on-site exploitation requires more risk by the adversary, who might be noticed to be in possession of radio equipment.

Finally, if an attacker does not know the exact model of the sensor device used, they may use a combined approach in which they guess and buy similar products and obtain a list of the vulnerable frequencies of these devices via off-site testing. Then, they can perform an on-site exploitation by first trying those frequencies and observing whether the target device has the same vulnerable frequencies. However, no guarantee exists that the devices will share a similar range of vulnerable frequencies and, depending on the devices' complexity, the approach will require greater time and effort on the part of the adversary.

6.2.3 Mitigation

The key to mitigating such threats is to increase the effort and time the attacker needs to exert in order to find the devices' vulnerable EMI frequencies and the appropriate EMI output intensity. Several precautions can be easily taken to reduce the risks to a minimal level: (1) cutting off the feedback, (2) keeping the sensor device model confidential, (3) hiding/randomizing the location of the temperature-monitoring devices, (4) carefully selecting sensors with a desired sampling rate, and (5) using temperature indicators that are less or not susceptible to EMI.

Cutting Off the Feedback. The attackers cannot easily know if the EMI frequencies used are the vulnerable frequencies if they cannot observe the change in temperature readings. The feedback cutoff can be achieved by eliminating easily snooped monitor screens and real-time temperature display on the temperature-monitoring devices. For instance, a small blinder on the temperature display (similar to a gas station payment pump or voting machine) can make snooping more difficult.

If easily snooped visual feedback cannot be eliminated, stand-off distances from the monitoring devices should be enforced to prevent nonauthorized people from observing the readings. A larger stand-off distance will also require a higher-power EMI output device in order to affect the sensor, which increases the cost incurred by the malicious party.

Further, the temperature data should only be accessible to trustworthy parties when necessary. Some temperature-monitoring systems also provide wireless communication functionality and monitoring software, which expose additional attack surfaces for the attacker to acquire the temperature- reading feedback. In this case, enforcing strong passwords and authentication schemes is crucial.

Of note, although feedback cutoff is the most effective method to prevent on-site exploitation, technically it cannot prevent off-site exploitation, in which case the attacker is the administrator of the duplicate target device that was acquired and therefore has unlimited access to sensor readings. However, avoiding temperature sensor devices with real-time temperature display can also greatly increase the effort of an attacker conducting an off-site exploitation due to the burden of reading the data repeatedly in an asynchronous fashion.

Keeping the Sensor Device Model Confidential. Keeping the sensor device model confidential is the most effective way to prevent off-site exploitation because, in this case, the attacker cannot acquire a duplicate device to find the vulnerable frequencies in advance. But similarly, this strategy alone cannot defend against on-site exploitation where the attacker can test the real device on the spot if the device's temperature-reading feedback is not cut off or well protected.

Hiding/Randomizing the Location of Temperature-Monitoring Devices. Hid-

ing/randomizing the location of the temperature-monitoring devices can reduce risk. After the attacker finds the vulnerable frequencies, the degree of change in temperature readings depends on the output intensity of the EMI source, as well as the distance and coupling path between the EMI source and target sensors. The attacker faces the risks of using too low intensity (so that no temperature excursions are caused) or too high intensity (so that the temperature excursions appear as artificial, which could reveal the attacker's existence). Hiding/ randomizing the sensor locations can prevent attackers from knowing the distance and coupling paths, greatly increasing the effort of the attackers for deciding the appropriate output intensity and thus lowering risks of this threat.

Carefully Selecting Sensors with a Desired Sampling Rate. Carefully selecting sensors with a desired sampling rate will reduce risk. The sample rate of a temperature sensor is the frequency of updating the temperature readings. The lower the sample rate, the slower the attacker will be able to identify the vulnerable frequencies because of the slow feedback update. To maximize the effort the attacker needs to put forth, it is advisable to select a temperature sensor/device whose maximal supported sample rate is closest to the minimal sample rate necessary to effectively monitor vaccine conditions and ensure vaccine safety.

For example, if the vaccine monitoring requires reading the temperature every 10 minutes, choosing a sensor/device with the highest supported sample rate of one sample per 10 minutes is recommended over using one that supports one sample per second and setting the device to read the temperature every 10 minutes. Otherwise, the attacker could easily conduct an off-site attack in which they set the duplicate device acquired to the highest supported sample rate and thus find the vulnerable frequencies quickly. In the above example, the time that the attacker needs to find the vulnerable frequencies can be in creased 600 times (10 min/1 s) by selecting the right sensor device.

6.3 Case Study: Injecting Phamton Keystrokes using Electromagnetic Interference

Keyboard has been an indispensable input component of any computer setup since the 1970s [84]. As a fundamental peripheral input device, keyboard has been an object of security research for decades. The majority of studies focused on how to eavesdrop on the typed keystrokes (also known as keylogging) and their countermeasures [175, 64, 50, 115, 157, 79, 56, 169, 147, 237, 163, 158, 167, 182, 55, 194, 70, 210, 184]. Others have tried injecting malicious fake keystrokes with reprogrammed USB devices masquerading as

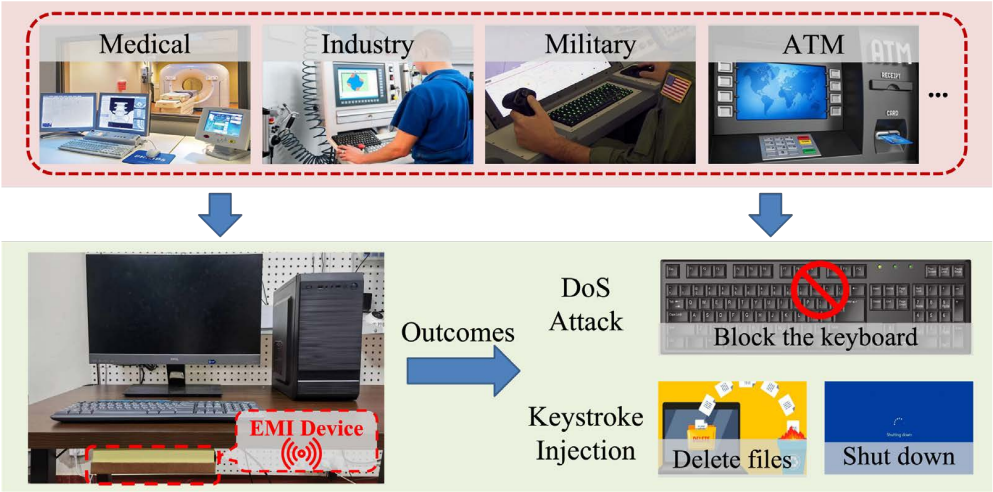


Figure 6.3: Keyboards are widely used in medical, industry, military, ATM, and other applications. Exploiting the vulnerabilities of the keyboard sensing mechanisms, GhostType can perform DoS attacks to block the keyboard or inject random keystrokes and certain targeted keystrokes.

keyboards [138, 180, 183, 67]. To prevent these fake keystrokes from directly manipulating computers, keyboards are recommended to be vetted or authenticated in security-sensitive applications [224, 77, 111, 117, 181, 140, 82]. Our study[1] revisits keyboard security and asks one more fundamental question: *to which degree can we trust the keystroke sensing of unaltered legitimate keyboards?*

Trustworthy keystroke sensing lays the foundation to secure computer operations in various critical application scenarios, including medical [109], industry [139], military [80], ATM [222], etc. Untrusted keystroke inputs could disrupt the operation of downstream computers and result in unexpected consequences. However, the security of keystroke sensing mechanisms has hardly been investigated by the security community. The main reason is that keyboards are known for their high reliability, especially in comparison with the touchscreen alternatives, which have been demonstrated to be vulnerable to electromagnetic interference (EMI) [170, 241, 203]. Keyboards sense keystrokes based on a simple principle—a keypress turns on/off a physical switch and therefore changes the received voltage level indicator which is usually 3.3 or 5 V. The high voltage level is naturally more resistant to conductive and radiative interference and keyboards are normally designed and tested for electromagnetic compatibility (EMC). In addition, the long history of keyboard manufacturing has given birth to false keystroke-prevention designs such as debounce and anti-ghosting mechanisms. These factors seem to suggest a reduced attack surface of malicious exploits.

[1]Demos: https://sites.google.com/view/ghosttype-demo

Our work aims to perform a security analysis of the overarching keystroke sensing mechanisms on modern keyboards and keypads. Specifically, we investigate whether unaltered keyboards/keypads may sense adversary-controlled fake keystrokes other than the authentic physical keystrokes from human inputs. As illustrated in Fig. 6.3, if an adversary is able to inject fake keystrokes into a legitimate keyboard without touching it, she may stealthily manipulate the computer by disrupting normal user operations, deleting documents, shutting the computer down, etc., depending on the specific keystrokes that can be injected by the adversary.

6.3.1 Threat Model & Background

6.3.1.1 Threat Model

Adversary's Goal. The adversary aims to contactlessly inject keystrokes into a keyboard through intentional electromagnetic interference (IEMI), thus blocking keyboard inputs or input keys to manipulate the connected computer. Our work considers two types of attack outcomes:

(1) **Denial-of-Service (DoS) Attack**, where the adversary can completely block the sensing of authentic keystrokes to disable user operations.

(2) **Keystroke Injection**, where the adversary can inject random keystrokes to make the computer unresponsive and even crash, or inject certain targeted keystrokes of the attacker's choice.

We make the following assumptions for the adversary to achieve the aforementioned attack outcomes:

Capability of the Adversary. We assume it is only feasible for the attacker to inject keys using external EMI signals contactlessly. This happens when the attacker has no on-site controls over the target keyboard's hardware/software and cannot take apart or tap into the keyboard or physically connect a malicious USB device in the form of BadUSB.

Knowledge of the Victim Keyboard. We assume the adversary knows the target keyboard's model, and she may obtain a similar keyboard for assessment beforehand. For example, she may disassemble the keyboard to systematically analyze the matrix circuit and scanning characteristics to retrieve the specifications by reverse engineering.

Attack Setup. We assume the adversary can hide the injection equipment by attaching it under the keyboard's desk or placing it at a distance from the keyboard. We also assume the adversary can control the equipment remotely.

6.3.1.2 Keyboard Sensing Mechanism

Keyboards are the most prevalent computer input device. There are several types of keyboards, including membrane, mechanical, dome, capacitive, buckling-spring, hall-effect, and optical keyboards. Membrane keyboards have been the most popular since the mid-1990s because they are cheap and easy for mass production. Keyboards can have different numbers of keys depending on the vendor and model, with most keyboards having 80 to 110 keys.

The typical workflow of a keyboard consists of three steps: keystroke sensing, scancode transmission, and task execution. The keyboard processor employs the scanning algorithm to scan the matrix circuit and sense keystrokes. Each key is assigned a unique identifier called a "scancode" with a translation table stored in the keyboard processor's memory. When the processor detects a key being pressed, it compares the key's coordinate on the matrix circuit to the scancode translation table. It then reports the scancode to the host computer via standard communication protocols such as PS/2, USB, and Bluetooth. After receiving the scancode, the computer raises an interrupt to process the scancode and register a keystroke. Finally, the operating system (OS) passes the keystroke information to applications.

Matrix Circuit Scanning. The keyboard is often designed in a special architecture known as the matrix circuit. The matrix circuit is built by arranging switches/keys in a grid-like array of M scanning lines (TXs) and N receiving lines (RXs) with one switch/key at every intersection. There is no established standard for the design of the matrix circuit layout, and a matrix circuit with M TXs and N RXs can support up to $M * N$ keys in theory. Each RX is pulled up to remain in the logical-high state ("1") in the idle state, and the keyboard processor drops each TX to the logical-low state ("0") in sequence to scan the matrix circuit. When a key is pressed, as shown in Fig. 6.4, the circuit of the corresponding TX-RX pair is closed. The scanning signal on TX is received by RX, resulting in RX being dropped to the logical-low state.

Keystrokes Sensing. The majority of keyboards sense keystrokes on the principle of detecting the logical state on the input GPIO. The keyboard processor employs a Schmitt Trigger at the input GPIO to determine the input logic state. The Schmitt Trigger determines the input logic state by applying two threshold voltages: the high threshold voltage V_{IH}, and the low threshold voltage V_{IL}. The keyboard processor detects a key as pressed when an RX is dropped below the low threshold voltage V_{IL} when a TX is scanned. The generic values for V_{IH}, and V_{IL} are 2.0-2.5 V and 1.2-1.5 V for a 5 V system, 1.2-1.5 V and 0.6-0.8 V for a 3.3 V system respectively. The exact thresholds depend on the processor's electrical characteristics.

Capability of Handling Simultaneous Keystrokes. *Key Rollover* is the term used to describe how many keys can be pressed simultaneously. A keyboard with n-key rollover

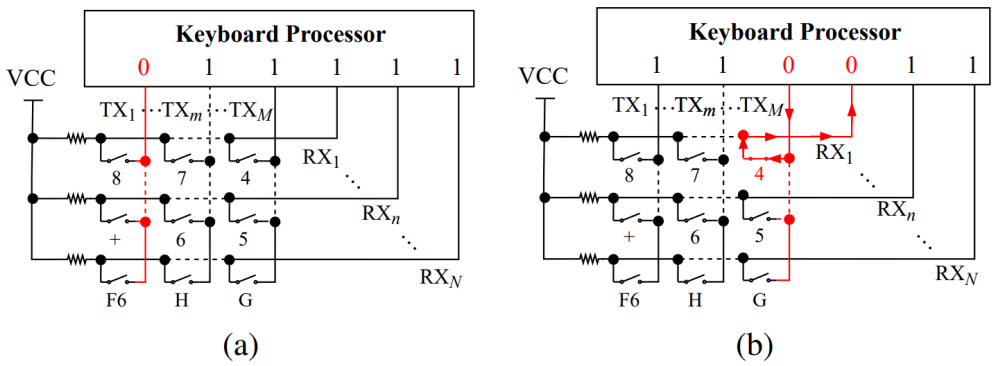


Figure 6.4: (a) The keyboard arranges switches/keys in a grid-like array. (b) When a key is pressed, a closed circuit is formed, and a corresponding RX is dropped to the logical-low state.

(NKRO) can correctly detect and handle all keys being pressed simultaneously. Typical general-purpose keyboards are 3-KRO to 6-KRO, and gaming keyboards usually support NKRO. *Keyboard Ghosting* is the problem that some keyboard keys don't work when multiple keys are pressed simultaneously. This happens when three or more keys sharing rows and columns are pressed simultaneously, and the connected circuit permits the current to flow incorrectly. Keyboards typically use filtering logic to detect and block keystrokes before this happens in software or employ diodes at each key to prevent the incorrect current flow in hardware.

6.3.2 Keyboard Sensing Security Analysis

In this section, we perform a systematic security analysis of 15 off-the-shelf keyboards through reverse engineering and uncover three vulnerabilities of keyboard sensing mechanisms.

6.3.2.1 Vulnerabilities of Matrix Scanning

We disassemble 15 off-the-shelf membrane keyboards. An example of their internal structures includes a keyboard processor board and a three-layered plastic matrix circuit board. The keyboard processor board is linked to a USB cable with a magnetic ring to shield high-frequency electromagnetic interference. The processor's GPIO pins are connected to traces on the matrix circuit board through physical contact.

Keyboard Sensing Characteristics Revealing. We use an oscilloscope to monitor the signal on each GPIO pin of the keyboard processor. The signals on TX and RX without and with a key pressed are illustrated in Fig. 6.5. The signal on each RX remains high in

the idle state when no key is pressed, while the scanning signal on each TX is a pulse signal with width w and scanning period T_S. Thus, we first determine whether it is TX or RX by measuring whether there is a pulse or DC signal on each GPIO pin. The results in Table 6.1 indicate that the most commonly used keyboard matrix circuit employs 18 TXs and 8 RXs. We then measure each keyboard's scanning characteristics, including idle state voltage V_{Idle}, pulse width w, scanning period T_S and time difference ΔT between two adjacent TXs. The results are summarized in Table 6.1, indicating that the scanning characteristics vary with the keyboard vendor and model. *We then hypothesize that the keyboard processor may be spoofed by replaying the scanning signal into an RX according to the following two reasons:* (1) the processor determines whether there is a key by sensing RX's logic state without authenticating the received signals, and (2) the scanning signal is a simple negative pulse signal that is not encrypted.

Wired Keystroke Injection. To validate our hypothesis above, we wire into an arbitrary RX and utilize a signal generator to inject a pulse signal with the same scanning characteristics as revealed in Table 6.1. Several keys were successfully injected, indicating that the keyboard processor does not validate the authenticity and legitimacy of the received signals. According to the keystroke sensing mechanism, keystrokes can be injected by dropping the RX's voltage to the low threshold V_{IL} when a TX is scanned. Thus, we change the values of the replayed pulse signal's amplitude V_{in}, period T_{in}, and pulse width w_{in} to test more diverse injection signals. We first decrease the amplitude V_{in} of the injection signal in a 0.1 V step from V_{Idle}. The result shows that keystrokes can be injected when V_{in} satisfies Eq. (6.1) to drop the voltage at an RX across the GPIO's low threshold voltage V_{IL}. For example, keystrokes can be injected into Cherry KC1000 keyboard and Logitech MK235 when V_{in} is higher than 3.4 V and 3.6 V, respectively.

$$V_{Idle} - V_{in} \leq V_{IL} \tag{6.1}$$

We then change the value of injection period T_{in} and pulse width w_{in}. The results indicate that keystrokes can be injected only when T_{in} satisfies Eq. (6.2).

$$T_{in} = \frac{T_s}{k}, k \in \mathbb{N}^* \tag{6.2}$$

This is because keyboards usually employ a *debounce delay* to ensure only one signal is acted upon each key-down or key-up event to prevent spurious keystrokes, i.e., a key press/release is only determined to be a keystroke if it is detected by two consecutive scanning cycles. Thus, the injected signals must hold for at least two consecutive scanning cycles. Additionally, keystrokes are injected at a higher speed when increasing the value of k. We also notice

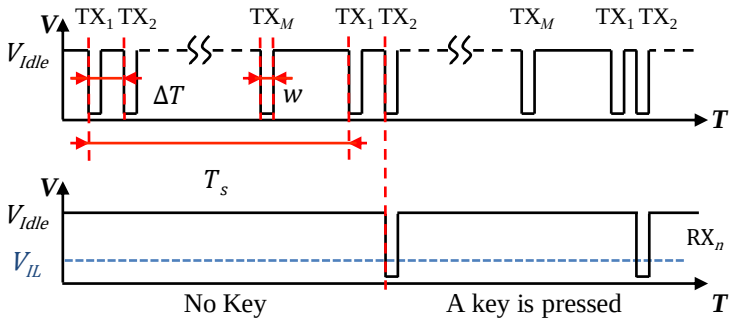


Figure 6.5: The keyboard processor continuously pulses each TX for a short duration in sequence, and the scanning signal on the TX flows through the switch to RX when a key is pressed.

multiple keystrokes are injected simultaneously when we increase w_{in}, and keystrokes are injected occasionally or even not injected when we decrease w_{in}. The keyboard is blocked when the number of injected keystrokes exceeds the keyboard's key rollover capacity (Section 3.9.1.1).

6.3.2.2 Vulnerabilities of Contactless Keystroke Injection

Potential Coupling Path for EMI Injection. The keyboard matrix circuit board is a three-layered plastic sheet board with dense traces exposed on the upper and lower sheets and cavities at each key location on the middle sheet. These TX and RX traces are irregular and vary from keyboard vendor and model. Although keyboards are consumer electronics that are expected to have adequate EM shielding, we found no anti-interference design on the matrix circuit board for almost all keyboards except the magnetic ring of the USB cable protecting the USB instead of the sensing circuit. *We hypothesize these long exposed traces can be exploited as potential EM coupling paths for EMI injections.*

Feasibility Study of Contactless Keystroke Injection. We conducted a frequency sweep test on a Cherry KC1000 keyboard to test the hypothesis. We employ a signal generator (SIGLENT SDG6032X), a power amplifier for EMI signal generation, and an antenna for EMI signal transmission. We place the antenna under the keyboard matrix circuit and conduct a frequency sweep test with a sinusoidal signal from 10 MHz to 100 MHz with a step of 10 MHz and an amplitude of 1 Vpp. During the test, we can randomly inject keystrokes into the keyboard at 30, 50, 60, 70, and 80 MHz *only* and block the keyboard at 20, 90, and 100 MHz. These results demonstrate that the matrix circuit traces act as the EM coupling path for contactless keystroke injections, resulting in different attack outcomes.

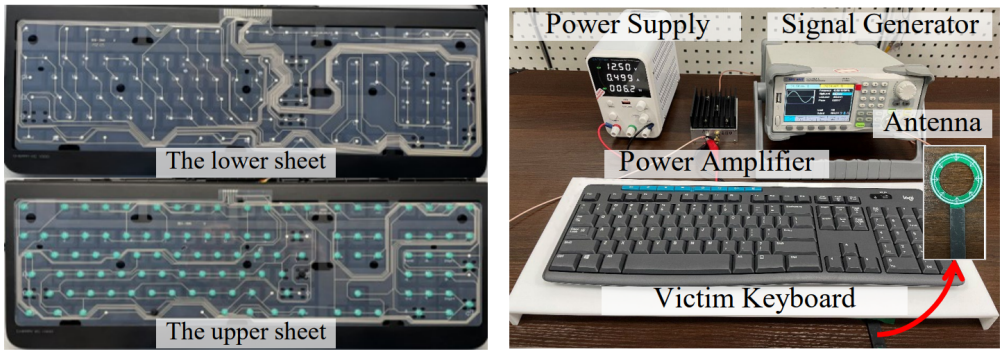
Table 6.1: Characteristics of matrix circuits and scanning signals retrieved through reverse engineering.

Vendor & Model	Num. of Keys (TXs, RXs)	Scanning Characteristics			
		V_{Idle}	T_S	w	ΔT
Cherry KC1000	108 (18,8)	5 V	3.6 ms	15 us	200 us
ACER YKB913	104 (18,8)	5 V	4.0 ms	120 us	120 us
ACER KM41-2K	104 (18,8)	3.3 V	8 ms	35 us	35 us
A4TECH MK100	104 (18,8)	5 V	3.8 ms	110 us	130 us
A4TECH FG1010	98 (18,8)	3.3 V	2.4 ms	45 us	130 us
Logitech MK235	104 (12,11)	3.3V	4.0 ms	8.5 us	10 us
Logitech MK220	100 (12,11)	3.3 V	4.0 ms	8.0 us	11 us
Rapoo K150	104 (18,8)	3.3 V	8.2 ms	142 us	250 us
Rapoo X125S	104 (18,8)	3.3 V	7.9 ms	140 us	250 us
Dell KB522P	116 (18,8)	5 V	3.2 ms	10 us	30 us
Dell KM2123D	104 (18,8)	3.3 V	7.8 ms	120 us	160 us
Lenovo KM4800S	107 (18,8)	5 V	7.8 ms	230 us	250 us
BOW MK610	79 (16,8)	3.3 V	7.2 ms	180 us	300 us

6.3.2.3 Vulnerabilities of Hidden Keys

During the experiments, we observed an interesting phenomenon: keys that don't exist on the keyboard's physical layout are injected. We call these keys "hidden keys". For example, we injected several hidden keys on a Cherry KC1000 keyboard, including function keys to open the file browser, turn the volume up/down, make the media play/previous, and possible ASCII codes for debugging such as "171", "233", "255", etc.

Prerequisites of Hidden Keys. We found that the hidden key phenomenon occurs because the keyboard matrix circuit is designed with a key at the intersection of TX and RX without a physical switch. The key sets of keys on the matrix circuit and the keyboard's physical switches are M_p and M_k, respectively. Theoretically, M_k should be equal to M_p, but in practice, M_k is a proper subset of M_p and $M_p - M_p \cap M_k \neq 0$, which is the prerequisites of hidden keys. The keyboard processor could handle all the input keys in M_k, and the set of hidden keys M_{hidden} can be expressed as $M_{hidden} = M_p - M_k$. We believe that hidden keys exist due to keyboard designers' negligence in inspecting and removing the non-existent keys from the keyboard processing firmware. This could be because it is more cost-effective for manufacturers to develop one matrix for various products. Under normal circumstances, these hidden keys will not be triggered because a human cannot close a nonexistent switch. However, the adversary could inject every key on the keyboard matrix circuit to trigger hidden keys, which may cause unexpected consequences to the downstream OS system and software.



(a) (b)

Figure 6.6: Illustrations of (a) the traces on the upper and lower sheets, and (b) the experiment setup of contactless keystroke injection via EMI. The keyboard is placed on a 5 mm-thick acrylic sheet, and the antenna is hidden under the sheet.

6.3.2.4 Effective Keystroke Injection via EMI

Fig. 6.7 illustrates the injection signal we designed for GhostType, where four parameters can be configured, including frequency f_{in}, amplitude v_{in}, pulse width w_{in} and period T_{in}. We design the injection signal as a pulse-modulated sinusoidal signal for two reasons: (1) the pulse-modulated sinusoidal signal is the most commonly used in state-of-the-art EMI injections [132, 241, 134, 201, 89, 88, 87, 86, 243], and (2) the feasibility of changing the reading of GPIO pins by injecting sinusoidal signals has been demonstrated in [243, 201, 86] and our preliminary study in Section 6.3.2.2. As briefly mentioned in Section 6.3.2.1, keystrokes can be injected when the injection signal satisfies two constraints:

- **Constraint 1:** The induced sinusoidal voltage $V_{emi}(t)$ at an RX needs to drop below V_{IL} when a TX is scanned to be sensed as a keystroke.
- **Constraint 2:** The injection signal needs to be injected during at least two consecutive scanning cycles because of the debounce mechanism.

To understand how to satisfy these constraints, we investigate the voltage and timing requirements of the injection signal to establish the theory of effective keystroke injections via EMI. First, we analyze the requirements of frequency f_{in} and amplitude v_{in} to inject keystrokes effectively. Then, we analyze the requirements of width w_{in} and period T_{in} to perform the single- and multiple-keystroke injections.

Requirements of frequency f_{in} and amplitude v_{in}. For a sinusoidal signal with frequency f_{in} and amplitude v_{in}, the induced voltage coupled into the keyboard's RX is an

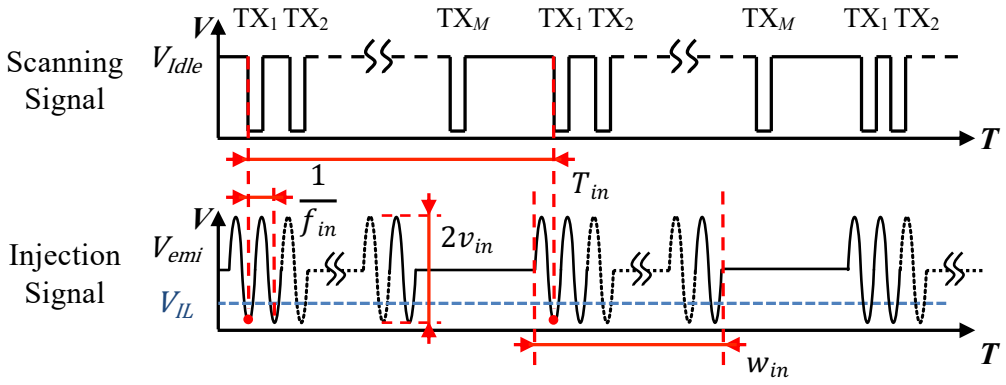


Figure 6.7: The injection signal designed for effective keystroke injections is a pulse-modulated sinusoidal signal with frequency f_{in}, amplitude v_{in}, pulse width w_{in} and period T_{in}.

AC signal $V_{emi}(t)$ that varies with time t, which can be expressed as Eq. (6.3).

$$V_{emi}(t) = E_c v_{in} sin(2\pi f_{in} t + \varphi_0) \tag{6.3}$$

where E_c is the coupling efficiency since the injection signal is coupled into the victim matrix circuit by means of the magnetic coupling mechanism. And the signal S_{in} in Fig. 6.7 can be expressed as Eq. (6.4).

$$S_{in} = \begin{cases} V_{emi}(t) & kT_{in} < t \leq w_{in} + kT_{in} \\ 0 & otherwise \end{cases} \tag{6.4}$$

where S_{in} is a pulse-modulated sinusoidal signal and T_{in} is the period of the injection signal and $k \in \mathbb{N}$.

To meet the first constraint to inject a keystroke when the m-th TX is scanned, the voltage requirement of the injection signal $V_{emi}(m\Delta T)$ can be expressed as Eq. (6.5) by combining Eqs. (6.1) and (6.3).

$$E_v v_{in} sin(2\pi f_{in} m\Delta T + \varphi_0) \geq V_{Idle} - V_{IL} \tag{6.5}$$

where $m\Delta T$ represents the scanning time of the m-th TX, $m = 1, 2, ..., M$, $V_{Idle} - V_{IL}$ is a constant, and M is the number of TXs. The specific values of V_{Idle} and V_{IL} are keyboard-dependent, which can be measured through reverse engineering. Since the frequency of injection signal f_{in} (several MHz) is more than three orders of magnitude greater than the frequency of the scanning signal f_s (several kHz), the timing relationship between the attack

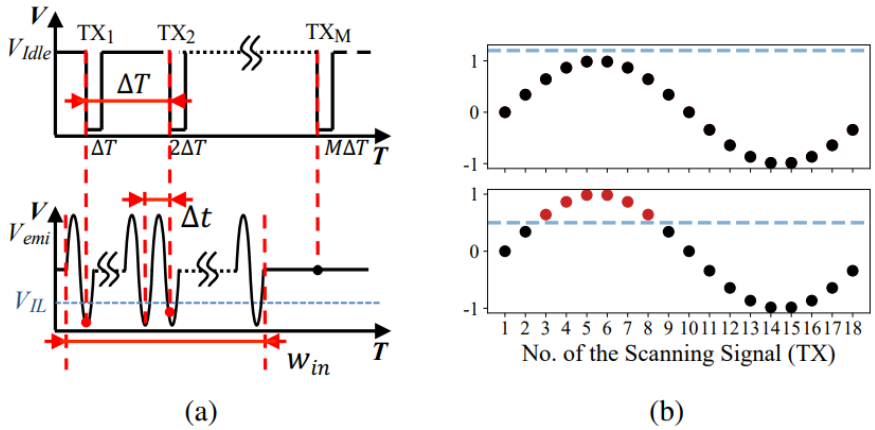


Figure 6.8: (a) The timing relationship between the injection and scanning signal. (b) Illustration of the requirements of the injection signal.

and each TX's scanning signal can be expressed as Eq. (6.6).

$$\Delta T = kt_{in} + \Delta t, 0 \leq \Delta t < t_{in} \; and \; k \in \mathbb{N} \tag{6.6}$$

Substitute Eq. (6.6) into Eq. (6.5) and simplify, the voltage requirement of the injection signal can be expressed as Eq. (6.7).

$$sin(2\pi f_{in}\Delta t m + \varphi_0) \geq \frac{V_{Idle} - V_{IL}}{E_c v_{in}}, \; m = 1, 2, ..., M \tag{6.7}$$

where kt_{in} vanishes because $f_{in}t_{in} = 1$, and $0 \leq f_{in}\Delta t \leq 1$. Thus, $sin(2\pi f_{in}\Delta t m + \varphi_0)$ becomes a discrete function of m. The solid dots represent the injected voltage on the RXs when a TX is scanned, and the blue dashed line represents the threshold $(V_{Idle} - V_{IL})/E_c v_{in}$. Since $max(sin(\cdot)) = 1$, the sinusoidal signal has no intersection with the blue dashed line when $V_{Idle} - E_c v_{in} > V_{IL}$, i.e., Eq. (6.7) is unsolvable and there is no keystroke injected. When $V_{Idle} - E_c v_{in} \leq V_{IL}$, the blue dashed line gradually moves down, and more keystrokes are injected as v_{in} increases. Thus, the minimum injection voltage is $v_{in} = V_{Idle} - V_{IL}$. The red dots in Fig. 6.8 (b) represent successful keystroke injections when the corresponding TX is scanned. Besides, we can change the value of φ_0 to inject keystrokes from different TXs. Faraday's law of induction states that the coupling efficiency E_c strongly depends on the injection signal frequency f_{in}. Thus, we must choose an appropriate combination of f_{in} and v_{in} to satisfy Eq. (6.7) to inject keystrokes contactlessly.

Prior works usually try to maximize E_c by analyzing the resonant coupling frequency f_{res}. However, analyzing f_{res} is both difficult and unnecessary for our attack for the following two reasons: *(1) Difficult:* The resonant coupling frequency f_{res} is determined by the geometry

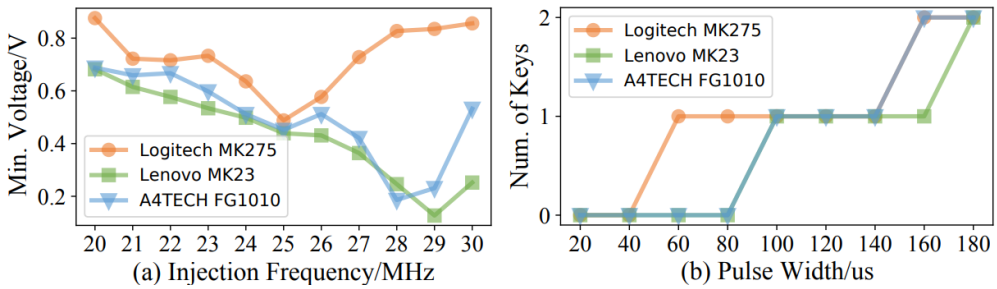


Figure 6.9: (a) The minimum voltage v_{in} required for keystroke injections at different frequencies f_{in}. (b) The number of simultaneously injected keys with different pulse widths w_{in}.

of traces and the keyboard's matching network impedance, which is difficult to calculate theoretically because traces are complex and different on each keyboard, and there are no publicly available high-frequency keyboard processor models. *(2) Unnecessary:* Keystrokes can be injected at a wide range of frequencies, not just one. Although the coupling efficiency E_c fluctuates with frequency, it can be easily overcome by increasing the amplitude v_{in}. Therefore, we can sweep the frequency across a wide band, determine the injection frequency candidates $\{f_1, f_2, \dots\}$ for relatively high power transfer, and then increase v_{in} to satisfy Eq. (6.7). It is important to note that the high-power EMI attack is power-hungry and may interfere with other devices to make them easily detectable. Thus, to make the injection energy-efficient and undetectable, attackers must select optimal injection frequency candidates with higher E_c to perform keystroke injections with a relatively low v_{in}.

We validate this by conducting a frequency sweep experiment on three keyboards and changing the amplitude v_{in} at different injection frequencies to determine the minimum required amplitude under a successful keystroke injection frequency. The results in Fig. 6.9(a) show that keystrokes can be injected at a wide range of frequencies and the minimum required amplitude varies in a non-linear pattern at different frequencies, indicating that some frequencies are more advantageous to an EMI injection than others. We can choose these more advantageous frequency candidates to perform a more powerful keystroke injection.

To meet the second constraint, Eq. (6.5) and Eq. (6.8) must be satisfied simultaneously.

$$V_{emi}((m+M)\Delta T) \geq \frac{V_{Idle} - V_{IL}}{E_c v_{in}} \tag{6.8}$$

where M is the number of a keyboard's TXs. Keystrokes are continuously injected when satisfying Eq. (6.9).

$$sin(2\pi f_{in}\Delta t m + \varphi_0) = sin(2\pi f_{in}\Delta t(m+M) + \varphi_0) \tag{6.9}$$

When $\Delta t = 0$, i.e., $\Delta T = kt_{in}$, Eq. (6.9) holds. The injection constraint of the debounce mechanism is automatically satisfied. When $\Delta t \neq 0$, a sufficient and necessary condition to satisfy Eq. (6.9) is

$$f_{in} \cdot \Delta t \cdot M = C,\ C \in \mathbb{N} \tag{6.10}$$

Combining Eqs. (6.6) and (6.10), t_{in} and ΔT need to satisfy the relationship in Eq. (6.11) to inject a keystroke.

$$\frac{\Delta T}{t_{in}} = k + \frac{C}{M},\ k \in \mathbb{N}^* \tag{6.11}$$

Since ΔT is two orders of magnitude greater than t_{in}, many solutions exist for Eq. (6.11). This conclusion is further demonstrated in [133], where keystrokes can be injected at a wide range of frequencies with the 48 off-the-shelf keyboards. When $V_{emi}(m\Delta T) \neq V_{emi}((m + M)\Delta T)$, keystrokes can only be injected periodically because Eq. (6.5) and Eq. (6.8) can be partially satisfied simultaneously in a sinusoidal signal period, which is inefficient and not the goal in this paper.

Requirements of width w_{in} and period T_{in}. We investigate the requirements of w_{in} and T_{in} to perform single- and multiple-keystroke injections. We can configure the value of width w_{in} and period T_{in} to change the number of injected TXs. w_{in} can be expressed as Eq. (6.12) and T_{in} satisfies Eq. (6.2).

$$w_{in} = kt_{in},\ k \in \mathbb{N} \tag{6.12}$$

where k is the cycle of a sinusoidal wave. When $T_{in} = T_s$, single and multiple keystrokes can be injected by satisfying Eq. (6.13) and Eq. (6.14), respectively.

$$Single\ Keystroke:\ w \leq w_{in} < \Delta T \tag{6.13}$$

$$Multiple\ Keystrokes:\ w_{in} \geq w + k\Delta T \tag{6.14}$$

where ΔT is the time difference between two adjacent TXs and $k \in \mathbb{N}$. When $T_{in} = T_s/k$ and $k = 2, 3, \ldots$, multiple keystrokes are injected . We validate this by conducting keystroke injections on three different keyboards. The results in Fig. 6.9(b) demonstrate that we can configure the value of w_{in} to inject single or multiple keystrokes simultaneously. The larger w_{in} is, the more keystrokes are injected simultaneously.

6.3.3 Mitigation

To mitigate the vulnerabilities of keystroke sensing mechanisms, we provide insights into potential hardware and software mitigations gleaned from our investigations.

Shield Keyboards with Metal Materials. Keyboards with a steel plate underneath the matrix circuit are less susceptible to EMI injections when the injection antenna is placed underneath the keyboard. It is worth noting that adversaries can still use the antenna above the keyboard to attack keyboards shielded with merely a metal plate underneath. We recommend that keyboard manufacturers employ metal enclosures as a straightforward countermeasure to protect both sides of the keyboard from EMI injections.

Enhance the Keystroke Sensing Mechanism. We believe keyboard manufacturers could improve the keystroke sensing mechanism in four ways. (1) Randomize the scanning signal waveform. The keyboard sensing mechanism can be spoofed primarily because the keyboard processor does not verify whether the received keystroke scanning signals came from the keyboard's TX. To ensure trustworthy keystroke sensing, we propose that the keyboard randomize the scanning signal waveform to be employed as the "verification signal". When a pressed key completes a circuit, the keyboard controller checks if that, and only that signal, is received on the appropriate RX pin. (2) Redesign the scanning signal's parameters. Our simulations revealed that decreasing the value of time difference ΔT between the two adjacent TXs considerably reduced the success rate of phantom keystroke injections. As a result, keyboard engineers can design appropriate scanning parameters to make the keyboards less vulnerable to GhostType. (3) Randomize the scanning sequence to make it difficult for adversaries to predict when and which TX is scanned to inject specific keystrokes into the targeted RX. (4) Detect and remove hidden keys using the proposed test method to avoid unexpected consequences.

6.4 Conclusion

This chapter provides evidence and analysis for hypothesis **H3**, showing how adversaries can affect the integrity of sensor data by exploiting side channels in sensor hardware to inject false information. It focuses on IEMI, one of the most generic types of threat against electronic sensing systems, and reveals the lack of verification mechanisms in existing sensing systems. Simple mitigations can leverage the fact that such physical signal-based manipulations are generally sensitive to the variations of attacker-target relative positions. However, more fundamental solutions that verify sensor signal authenticity need to be further researched and implemented on the hardware-software interfaces to provide fundamental protections.

CHAPTER 7

Utilizing Sensor Side Channels for Multimodal Sensing

7.1 Overview

While previous chapters have extensively investigated how the side channels in sensors (i.e. s_{side} in Equation 2.1) could negatively affect the security and privacy of sensing systems, this chapter investigates how such channels can be utilized by system defenders to improve the security of these systems. Centered on hypothesis **H4**, this chapter proposes the concepts of virtual sensor synthesis from sensor side channels and investigates how it enables multimodal sensing with a single hardware sensor for authentication [161]. To that end, we revisit the concept of sensor side channels and extend the generic model in Section 2.2 into authentication settings. Using a camera-sensing example, we show how the motion side channel information in videos can be used to protect existing smartphone face authentication systems from silicon mask spoofing attacks.

7.2 Synthesizing Virtual Sensors from Side Channels

Sensor side channels enable an adversary to violate integrity of sensor outputs by influencing or controlling the sensor with transduction attacks [247, 107], or to eavesdrop on sensitive information and compromise confidentiality by exploiting flaws in sensor and system designs [172, 63, 45, 208]. For example, the eavesdropping example PIN Skimmer [208] shows that adversaries can infer smartphone touchscreen inputs by exploiting side channel motion information captured by smartphone cameras. While the security research community invested significant effort identifying and mitigating analog sensor side channels, our work argues that *it can be beneficial to embrace, understand, and control analog sensor side channels instead of simply eliminating them.* This is motivated by our observation that such side

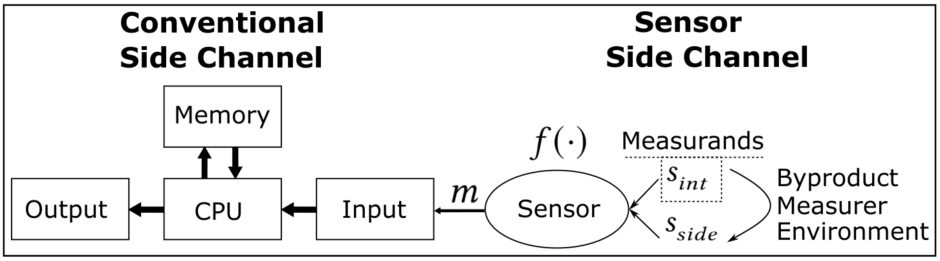


Figure 7.1: Sensor side channels are different from conventional side channels as they measure the measurement processes instead of computation processes. Sensor side channels can measure the byproduct, measurer, and environment to verify authenticity of intended sensor measurands.

channel information may also be used for authentication. For example, extensive research has been conducted on using dedicated motion sensors to capture smartphone touch dynamics for continuous implicit user authentication [220]. Relating it to PIN Skimmer, a natural question arises as to whether cameras support such authentication when dedicated motion sensors are not available. We thus propose and investigate the problem of how to utilize sensor side channels for defensive purposes such as multimodal authentication by synthesizing virtual sensors from them.

Side channels are inherent to analog sensors' physics. There exist a considerable number of potential sensor side channels besides those revealed by transduction and eavesdropping attacks. However, most of these side channels are deliberately "closed" in the design phase by employing mitigation mechanisms such as calibration and noise reduction. It is foreseeable that sensor and system designers will also try to mitigate newly discovered side channels. This work argues a different perspective and approach to embrace such sensor side channels. If these side channels can be used in a beneficial way, we envision future designs allowing mitigation mechanisms to be strategically disabled or downgraded when needed such as during authentication sessions.

We provide a preliminary analytical framework for modeling analog sensor side channels and explaining the origins and characteristics of them. The framework categorizes sensor side channels according to their separability from intended signals and whether they have controllable mitigation mechanisms. Based on the framework, we define the problem of virtual sensor synthesis for multimodal measurand authentication and summarize three possible ways of applying this approach (Figure 7.1). First, by verifying signatures of signal byproducts and asking the question "What is the probability that Alice generated both the measurands and byproducts?" Second, by verifying the person performing the measurement and asking "What is the probability that Alice generated the measurands if Bob was the

measurer?" Third, by verifying the environment of the measurement process and asking "What is the probability that Alice generated the measurands if the measurement was taken in location B?"

A proof-of-concept case study further concretizes the concepts and related considerations by studying a camera motion side channel that enables cameras to sense out-of-sight motion. This side channel is caused by mechanical connections between camera devices and adjacent objects in motion such as a hand holding the camera. We propose a methodology for synthesizing virtual inertial measurement units (IMUs) from this side channel that can extract both inter-frame low-frequency and intra-frame high-frequency motion information. The case study discusses this side channel's potential application in helping facial recognition systems defend against 3D silicon mask spoofing attacks by verifying postural hand tremor motion of the person holding the camera device. Preliminary test with 4 people suggests the camera motion side channel help reduce false positive rates by up to 87.5%. It also shows that disabling video stabilization enables higher performance, emphasizing the benefits of strategically disabling side channel mitigation mechanisms. Finally, we discuss the possible issues of temporarily opening sensor side channels during authentication and the directions future works may take to address the issues.

7.3 Problem Formulation

This section defines the problem of using sensor side channels for measurand authentication. Our paper proposes the concept of using sensor side channels for authentication as a new direction of research for the community. We also fill a gap by suggesting a mathematical definition of sensor side channels, beginning with a framework for defining and categorizing sensor side channels. We then introduce the problem of synthesizing virtual sensors and using them for multimodal sensor measurand authentication.

7.3.1 Sensor Side Channel Analytical Framework

Following the model in Section 2.2, a sensor can be modeled as a function that maps physical measurands to digital measurements over time. A measurand is a quantity that a sensor intends to measure [110]. Different types of sensors are designed to measure different modalities of measurands such as sound, temperature, motion, etc. Users who are informed of the apparent purpose and specifications of sensors often see a sensor as the following function

over a single variable of the measurand:

$$m = f(s_{int}) \tag{7.1}$$

where m and s_{int} denote the digital measurements and analog measurand respectively and $f(\cdot)$ denotes the sensor.

7.3.1.1 Sensor Side Channels

Although Equation 7.1 provides average sensor users a clean and easy abstraction, actual sensor implementations are much "dirtier' and introduce numerous hidden variables to the equation that result in unintended components in measurement m. For instance, every conductor wire can be regarded as an unintentional antenna, leading to side channels that convert electromagnetic energy to measurements of non-electromagnetic sensors [99, 230]. In this case, hidden variables related to electromagnetic energy in the environment should be added to Equation 7.1. Another example of such variables is temperature. Semiconductors made of silicon are inherently sensitive to heat due to its ability to excite electrons. So technically, Equation 7.1 should also include temperature as a variable. Electromagnetic energy and temperature are just examples of hidden variables associated to the underlying physical characteristics of devices. There are also hidden variables caused by design flaws and uncontrollable variations in the manufacture processes. Thus, Equation 7.1 should be modified to enable a side channel-aware modeling of sensors:

$$m = f(s_{int}, s_{side}), \quad s_{side} = [s_{v_1}, s_{v_2}, ...] \tag{7.2}$$

where s_{side} represent the set of all these hidden variables that can potentially lead to side channels attacks.

The comparison between Equation 7.1 and 7.2 shows that the gap between users' understanding and sensors' actual implementation gives birth to sensor side channels. On a high level, we believe the gap can also be attributed to the insufficient specifications of legitimate and illegitimate sensor behaviors in the existing system's security policies. Note that this differs from conventional non-sensor side channels where side channels bypass the clearly specified security policies [108]: there are often no dedicated security policies for sensors yet in existing systems.

Sensor side channels are sometimes more conceptually difficult to recognize than conventional non-sensor side channels such as differential power analysis channels. The reason is that non-sensor side channels are used to mainly measure computation processes where there exists a clear boundary between computation and measurement whereas sensor side

channels are used to measure the measurement processes themselves (Figure 7.1).

A possible way of identifying sensor side channels is to test the hypothesis that the analog signal of a variable v_i correlates with m with certain significance, i.e.,

$$|Corr(m, s_{v_i})| > \alpha, \quad s_{v_i} \in s_{side} \tag{7.3}$$

where α is a threshold value. Note that this work does not discuss the actual choice of threshold values and correlation functions since they can be flexible depending on the actual application scenarios and security requirements. In cases where it is challenging to project m and s_{v_i} to the same vector space in order to compute correlation scores, other methods such as supervised classification can also be used if s_{v_i} can be converted into data labels.

7.3.1.2 Separability and Controllability

The unintended components in the measurements are caused by the existence of s_{side} and can be either separable or inseparable from the intended components. The separability between the intended and unintended components is the key that decides whether a side channel can be mitigated and controlled or not. Conceptually, separable components can be defined as the following: there exists at least one function $\tilde{f}(\cdot)$ that can break m down into intended and unintended components such that those components only correlate (with significance) with the measurand and other hidden variables respectively, i.e.,

$$\begin{aligned} \exists \tilde{f}(\cdot) \quad s.t. \quad & \tilde{f}(m) = [m_{int}, m_{side}], \quad m_{side} = [m_{v_1}, m_{v_2}, ...], \\ & |Corr(m_{int}, s_{int})| > \alpha_{i1}, |Corr(m_{v_i}, s_{v_i})| > \alpha_{i2}, \\ & |Corr(m_{int}, s_{v_i})| < \beta_{i1}, |Corr(m_{v_i}, s_{int})| < \beta_{i2} \end{aligned} \tag{7.4}$$

When a sensor side channel has separable components, we say it is a *separable side channel*. Separability is decided by sensor implementation $f(\cdot)$. Side channels with inseparable components in existing sensor implementations led to the various unsolvable attacks against sensors because designers cannot extract only the intended components.

Theoretically, those with separable components can be mitigated by mechanisms referred to as compensation, calibration and noise reduction. Such mitigation mechanisms can be abstracted as another function $g(\cdot)$ that suppresses the unintended components in the output of $\tilde{f}(\cdot)$, i.e., $g(\tilde{f}(m)) = m_{int}$. If the mitigation mechanisms can be both turned on and off, the user of the sensor system then have full control of the sensor side channel. We call such a sensor side channel *controllable*:

- *A controllable sensor side channel is one whose corresponding unintended measure-*

ment component is separable from the intended component and can be suppressed by a mitigation mechanism that can be enabled and disabled.

7.3.1.3 Examples

We provide some existing examples of each category of sensor side channels to shed light on the differences and possible future evolution.

Inseparable. The Gyrophone eavesdropping attack [172] and its follow-up works [63, 45] use an aliasing-enabled inseparable acoustic side channel in smartphone IMUs to recover speech. These IMUs have intended acceleration and angular velocity measurands mostly under the frequency range of human speech. However, due to the lack of effective analog low-pass filtering before the ADC, aliases of the high-frequency speech signals exist in the output of ADC and enable adversaries to recover speech information. Furthermore, the aliases cannot be separated from the intended motion signals since they are in the same frequency range. Intuitively, adding analog filters to the sensors make this acoustic side channel separable. In order to be controllable, the sensor API may further allow CPU to enable and disable the filters.

Separable But Uncontrollable. Those seemingly intact sensors that have not been reported vulnerable to side channel-based attacks also have inherent side channels, but just in a suppressed manner thus these channels are not exposed to attackers. Take sensors' heat sensitivity mentioned in Section 7.3.1.1 as an example. MEMS humidity sensors, gyroscopes, accelerometers, etc., are widely equipped with temperature-compensated designs or online thermal calibration procedures [248, 105, 71]. It can be anticipated that if the compensation and calibration mechanisms can be temporarily disabled, these sensors' measurements will exhibit significant correlation with the ambient temperature. In this way, the separable side channel becomes controllable.

Controllable. There already exist sensors with controllable side channels. A good example is handheld cameras getting equipped with video stabilization mechanisms. Camera motion is often regarded as side effects that degrade the quality of the intended signal, i.e., the scene in the field of view of the camera [250]. Video stabilization mechanisms, including electronic image stabilization (EIS) and optical image stabilization (OIS), etc., are implemented to mitigate these side effects by optically or electronically reducing the unwanted image scene movements caused by camera motion. Many operating systems such as Android allow app developers to choose if these video stabilization mechanisms will be turned on or off when the underlying camera hardware offers the API to control it. However, it is worth noting that such existing controllable side channels are most likely byproducts of OS designers' conventions of providing more fine-grained interfaces, especially for open-

source OS like Android which allows users to control EIS and OIS separately. In contrast, iOS does not allow explicit and separate control of EIS and OIS. Such a large degree of control is provided to support more potential use cases and enhance usability. For example, users may want to disable smartphone's built-in optical image stabilization when using an external gimbal because the two can interfere with each other and produce extra image distortions [6]. To the best of our knowledge, these existing controllable side channels have not been explored to enhance the security of systems.

7.3.1.4 Summary

It is possible to convert existing inseparable or uncontrollable side channels into controllable side channels by improving sensor designs, as has been suggested by the increasing popularity of video stabilization in cameras. Thus, it is important to think from a perspective of technology development when considering benefits of sensor side channels. Furthermore, protecting physical sensors from side channel attacks often already means transforming inseparable side channels to be separable. With some additional effort of making mitigation mechanisms controllable instead of forever-on, sensor side channels can be used in a beneficial and controlled manner. The following discussions assume sensors have controllable side channels.

7.3.2 Measurands Authentication Using Synthesized Virtual Sensors

7.3.2.1 Virtual Sensor Synthesis.

A virtual sensor is a function that maps m to m_{v_i}. Ideally, the construction of $\tilde{f}(\cdot)$ in Equation 7.4 already presents such an overarching function that can measure both the intended and side channel components. Such construction is apparently challenging since it needs to consider all possible side channels. Actual implementations can reduce the level of challenge by focusing on maximizing $|Corr(m_{v_i}, s_{v_i})|$ and $-|Corr(m_{v_i}, s_{_{int}})|$ for only the set of targeted hidden variable $\{v_i\}$. We denote such a function specifically crafted for $\{v_i\}$ as $\tilde{f}_{\{v_i\}}$ and call them virtual sensor functions.

7.3.2.2 Problem Definition

We define the problem as a binary hypothesis test in a comparative manner by first referencing to the unimodal authentication on the physical sensor's measurand alone. Without virtual sensors, objects in Equation 7.1 including m, $s_{_{int}}$, and f are all that the designer of

the authentication system can perceive. Let there be a measurand with a true identity L and a claimed identity $\tilde{L}$. The H_1 and H_0 hypotheses are $\tilde{L} = L$ and $\tilde{L} \neq L$ respectively. Denote the unimodal authentication system as $\mathcal{A}_u : m \to \{1, 0\}$, where it declares H_1 and H_0 when outputting 1 and 0 respectively. We can then define the total error of the unimodal system E_u as

$$\begin{aligned} E_u &= c_1 \mathbb{P}[\text{declare } H_1 | H_0] + c_2 \mathbb{P}[\text{declare } H_0 | H_1] \\ &= c_1 \mathbb{E}[\mathcal{A}_u(m) | H_0] + c_2 \mathbb{E}[1 - \mathcal{A}_u(m) | H_1] \end{aligned} \tag{7.5}$$

where $\mathbb{P}[\cdot|\cdot]$ and $\mathbb{E}[\cdot|\cdot]$ denotes conditional probability and expectation respectively, c_1 and c_2 denote the cost coefficients for false positive and false negatives respectively.

Similarly, a multimodal authentication system with n synthesized virtual sensors can be denoted as $\mathcal{A}_m : [m_{int}, m_{v_1}, ..., m_{v_n}] \to \{1, 0\}$. The total error E_m is defined as

$$\begin{aligned} E_m = c_1 \mathbb{E}[\mathcal{A}_m([m_{int}, m_{v_1}, ..., m_{v_n}]) | H_0] \\ + c_2 \mathbb{E}[1 - \mathcal{A}_m([m_{int}, m_{v_1}, ..., m_{v_n}]) | H_1] \end{aligned} \tag{7.6}$$

As a result, the problem of synthesizing virtual sensors to authenticate the measurand in a multimodal manner can be defined as:

- *Constructing virtual sensor functions $\tilde{f}_{\{v_i\}}$ and multimodal authentication system $\mathcal{A}_m$ such that better performance is achieved for measurand authentication, i.e., $E_m - E_u < 0$.*

7.3.2.3 Security Properties

Although multimodal authentication using synthesized virtual sensors look similar to that using multiple physical sensors, it provides two different security properties.

First, it works with existing devices and media that only have a single physical sensor's data. Although high-end devices like smartphones are equipped with multiple physical sensors, there still exist lower-end devices that only serve a single purpose such as ultrasonic proximity detectors and humidity monitors. Furthermore, sometimes it is needed to verify the identity of an object such as a photograph that has already been generated with only a single sensor. In this case, synthesized virtual sensors can extract additional information in a retrospective way.

Second, it potentially provide more robustness against spoofing attacks on individual sensors. The level of attack difficulty depends on the complexity of $\tilde{f}$, i.e., how difficult it is to decouple and then modify different measurement components. Using multiple individual

sensors such as cameras, accelerometers, etc., is equivalent to having a $\tilde{f}$ that does not need to decouple anything at all since the inputs already separated. Conceptually, if we regard the measurements corresponding to different virtual or physical sensors as random variables, we can then regard their variances and covariances as the entropy provided for authentication [176]. Virtual sensors potentially provides more entropy because the coupling between them adds to the covariances. Such entropy originates from the intrinsic physics of sensors.

7.3.2.4 Application

The general problem definition can be applied to different sources of side channel variables whose signatures correlate with the claimed identity of the measurands. Depending on the sources, we believe synthesized virtual sensors can be applied in the following three ways to verify authenticity of measurands.

Byproduct Verification. A physical process generating intended measurands is likely to generate other forms of energy as byproducts. Let us explore the example of a loudspeaker that replays a person Alice's speech recordings while a nearby microphone is listening to this replay. Say there is someone claiming the speech audio collected by the microphone is coming from Alice herself speaking live and an investigator tries to verify this claim. The investigator finds out that the loudspeaker also generates unintended, secondary byproducts in the form of structure-borne vibrations, electromagnetic emission, heat, etc., which may be sensed by virtual sensors synthesized from the microphone's side channels. So, if these byproducts exist, the investigator knows it is not likely a legitimate recording of Alice's voice. In this case, the core authentication question can be summarized as *"What is the probability that Alice generated both the measurands and byproducts?"*

Measurer Verification. A Measurer is the person who makes measurements with a physical sensor. Measurers themselves generate unintended emissions taking the form of physical signals containing certain signatures that correlate with the identity of measurands. For example, say there exists an unmodified photo of a person who is claimed to be Alice and an investigator tries to verify this claim. The investigator managed to find out that the camera operator who took this photo, i.e., the measurer was Bob because Bob was speaking when he took the photo and his speech induced identifiable image blurs through a camera motion side channel. If the investigator also knows that Bob has never been in the vicinity of Alice, then the investigator knows the person in the photo is not Alice. Obviously, measurand authentication through measurer verification may require higher-level contextual information compared to byproduct verification. The core authentication question is *"What is the probability that Alice generated the measurands if Bob was the measurer?"*

Environment Verification. Similar to measurer verification, verifying the environment surrounding measurands also allows one to authenticate the measurands. Take the same example above. Say the photo has a temperature side channel that shows the ambient temperature was 104°F/40°C at the time of generating the photo, pointing to a location B. If the investigator knows Alice has never been in location B, then the investigator knows the person in the photo is not Alice. The core authentication question is *"What is the probability that Alice generated the measurands if the measurement was taken in location B?"*

7.4 Case Study

The case study demonstrates how to use camera motion side channels (Section 7.3.1.3) to synthesize virtual IMUs that can collect postural hand tremor information for measurand authentication in facial recognition applications. It can be regarded an example of both byproduct and measurer verification.

7.4.1 Primer

7.4.1.1 Postural Tremor Information.

Tremor is the involuntary rhythmic movement of a human body part caused by reciprocal innervations of muscles. Such involuntary movements are present in all people, with those found in healthy people and disease conditions (e.g., Parkinson disease) classified as physiological and pathological tremor respectively [225]. Clinical research finds that tremors measured by accelerometers can effectively predict the category of tremors. Some works further show that hand tremors measured by accelerometers and gyroscopes are unique to an individual and stable over time, suggesting the feasibility of using tremors as a biometric for personal identification [174, 93].

7.4.1.2 Threat Model.

We study a threat model of spoofing attack against smartphone facial recognition systems where imposters are assumed to launch a silicone face mask spoofing attack [190]. To better show the effectiveness of the synthesized IMUs, we further assume the silicone mask perfectly mimics the face of the victims. During the attack, the imposter wears the silicone mask and holds the victim's smartphone for authentication. Our objective is to extract camera motion from videos that represents the postural hand tremor of users to defend against such perfect silicone mask attacks.

It is worth noting this particular case study's threat model requires users to hold their phones in their hands during facial recognition as the contact between their phones and hands provides a propagation path for the vibration information of hand tremor. We believe this is also the most frequent situation seen in smartphone-based facial recognition applications. Nevertheless, there do exist some circumstances where users may want to place their phone on a table during authentication. Our tremor recognition with synthesized virtual IMUs will not work in this case due to the lack of camera motion. Similarly, a spoofing attacker cannot authenticate successfully in this case without providing the camera the correct motion. To enable users to authenticate without holding their phones, we believe future works may look into other sensor side channels that acquire a different type of user biometric information such as body-radiated electromagnetic/heat energy without requiring direct contact with the phone.

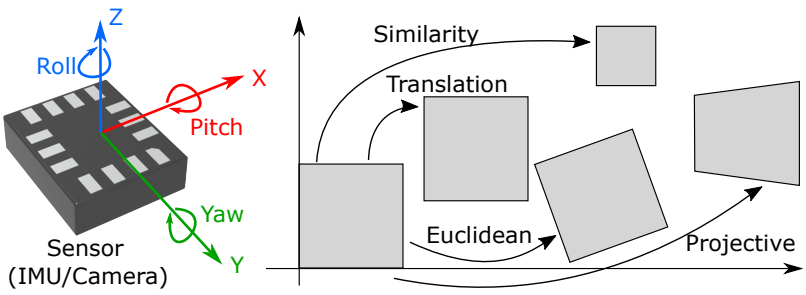
7.4.2 Synthesis Methodology

Different methodologies can be used to synthesize virtual IMUs from camera motion side channels. For example, a completely model-based methodology requires understanding $f(\cdot)$ and $\tilde{f}(\cdot)$. Although the most accurate, it requires thorough understandings of every targeted camera system and is challenging. Another possible methodology is to completely rely on neural network to process the raw videos and let the network figure out $\tilde{f}_{\{v_i\}}$, which is similar to previous work of inferring sounds from object motions in videos [185]. This methodology requires intensive computation resources and data collection. This work focuses on the middle ground by investigating a model-informed methodology that constructs $\tilde{f}_{\{v_i\}}$ based upon the concepts of image registration. Image registration is the process of overlaying two or more images of the same scene that are taken at different times, from different viewpoints, and/or by different sensors [262]. The methodology aims to extract both inter-frame motions and intra-frame motions.

7.4.2.1 Understand Motion Modulation.

To construct $\tilde{f}_{\{v_i\}}$, the first step is to understand how motion signals are modulated onto image streams. We analyze the motion modulation process from two different perspectives.

Frame Transformation. The frame transformation perspective considers changes of the frames subjected to camera motions as 2D image transformations. Figure 7.2 shows the possible image transformations corresponding to motion on each one of the six real-world axes and the measurements of physical IMUs. As a result, motions that can be measured by IMUs can also be mapped to inter-frame variations of the camera videos.



Sensor \ Motion	X	Y	Z	Pitch	Yaw	Roll
IMU Output	Accl X	Accl Y	Accl Z	Gyro X	Gyro Y	Gyro Z
Cam Output	Trans.	Trans.	Sim.	Proj.	Proj.	Euc.

Figure 7.2: Types of 2D image transformations corresponding to the type of camera motion and motion readings measured by physical IMUs.

Rolling Shutter. Besides inter-frame variations, the rolling shutter property of most cameras on portable devices can generate intra-frame variations that embed high-frequency motion. Rolling shutter is the shutter mechanism of commercial CMOS cameras, which exposes and samples the rows of an image sensor sequentially instead of simultaneously as in a global shutter [154]. If viewing the possible 2D image transformations as bases, rolling shutter combine multiple transformations into a single frame. It increases the effective sample rate of the motion signals provided by the camera side channel.

Based on the knowledge of how camera motion is modulated onto images, two corresponding categories of virtual IMU synthesis methods are introduced next to measure low-frequency and high-frequency information respectively.

7.4.2.2 Low-frequency Information Measurement

The frame transformation perspective enables measurements of low-frequency components. It perceives the difference between two frames as the result of a single motion vector composed of single-axis motions (Figure 7.2) within the period of one frame. The camera imaging process thus becomes the sampling process of the measurable motion signals with a sample rate that is the same as the video frame rate, e.g., 30 Hz in case of 30 fps videos. Theoretically, all image registration methods are applicable to extract inter-frame variations. We discuss one possible construction.

Image Transformation Estimation (ITE). A straightforward way of extracting the frame differences is registering the frames with respect to a reference frame by estimating the 2D image transformations needed to warp the reference frame to the other frames as has been explored in [208]. Each 2D transformation estimation generates a 3-by-3 transformation matrix. By concatenating each entry of different transformation matrices chronologically, it

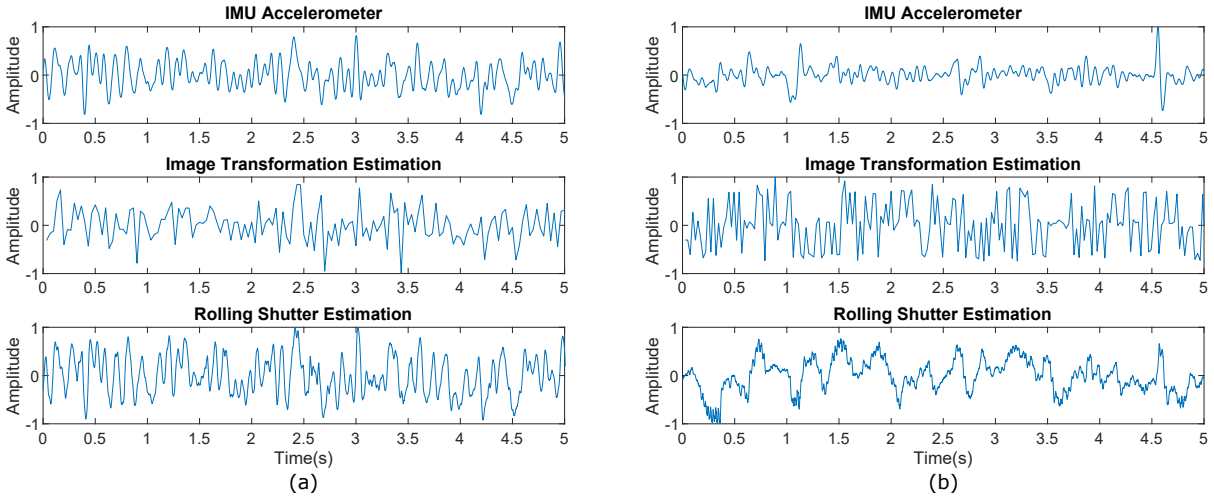


Figure 7.3: Measurements of physical IMU accelerometer (408 Hz) and virtual IMU synthesized with the ITE and RSE methods from videos (30 fps frame rate, 1080p resolution) in 5 seconds. Amplitudes are normalized to compared different measurement approaches. (a) Videos stabilization is off. (b) Videos stabilization is turned on. Strategically disabling sensor side channel mitigation mechanisms boosts up virtual sensors' capability for measurand authentication.

produces 9 vectors that represent the output of $\tilde{f}_{\{v_i\}}$. Diverse algorithmic implementations of this method are possible. This works uses an image registration implementation based on phase correlation [191].

7.4.2.3 High-frequency Information Measurement

The rolling shutter perspective allows for the extraction of intra-frame high-frequency variations. It perceives the difference between two frames as the result of multiple sequential motion vectors. The number of motion vectors is the same as the number of rows of the camera imaging sensor as each row is exposed and sampled sequentially. The effective sample rate is thus the row-scanning rate of the rolling shutter, which is higher than 30 kHz for most commercial cameras. Nevertheless, not all signals within its Nyquist frequency can be recovered, as the non-zero exposure time causes motion blurs and attenuate the higher-frequency signals [85]. Similarly, a possible construction is introduced below.

Rolling Shutter Estimation (RSE). Methods of rolling shutter estimation still compares different frames, but performs such comparison on the even smaller granularity level of rows or individual pixels. Then, the methods concatenate the values generated by the comparison first across different rows of a single frame, and then across different frames to form the motion signal vectors. With the proposed methodology, this work converts rolling shutter estimation into a pixel-level image registration problem. Algorithms capable

of pixel-level registration often generate displacement fields, i.e., matrices of the same size as the registered images, on the X and Y directions. The produced matrices are apparently high-dimension and difficult to process. We can then group the matrices column-wise and average the columns in each group to produce easily understandable signals. This work uses a diffeomorphic image registration method [236] to implement RSE.

7.4.2.4 Demonstration

Figure 7.3 shows the motion signals measured by a physical IMU (408 Hz sample rate) and virtual sensors using ITE and RSE methods. A Google Pixel 2 smartphone held by a person recorded the physical IMU readings and camera videos simultaneously, where the postural hand tremor of the person caused the camera motion. The ITE and RSE methods have sample rates of 30 Hz and 34 kHz respectively. The figure only displays a single vector of the physical and virtual sensor measurements respectively that represents the horizontal motion to simplify the visualization.

Figure 7.3 (a) and (b) shows the measured signals with the video stabilization functionality being off and on respectively. When video stabilization is off, the virtual sensor outputs of both the ITE and RSE method show strong correlation with the physical IMU measurements. It is also clear that a 30 Hz sample rate is not sufficient to capture all the motion, as the ITE method's signal shows larger distortions than that of the RSE method. When video stabilization is turned on, the camera motion signals deviate more from the IMU readings as expected. Although the signal of RSE method still shows observable correlation with the IMU signal, ITE produces seemingly uncorrelated signals.

7.4.3 Experiment

We conduct preliminary tests with 4 people and a Google Pixel 2 smartphone. The 4 participants are all healthy males with similar ages, heights, and weights. As a proof-of-concept instead of an actual system product, we regard facial recognition and tremor recognition as two decoupled problems and test them separately. The tremor recognition mechanism can be regarded as an additional layer of protection besides the existing facial recognition system. We investigate the impact of disabling and enabling video stabilization in both of the two tests.

The objective of testing tremor recognition is to verify the effectiveness of the synthesized IMUs. To that end, we also recorded the physical IMU readings for comparison. The objective of testing facial recognition is two-fold. First, it is important to inspect if the postural hand tremor of different people can already make a difference in the original fa-

cial recognition systems without synthesis of virtual sensors. This verifies the necessity of constructing dedicated virtual IMUs. Second, since turning off video stabilization may lead to better virtual sensor performance, it is also necessary to inspect if it would degrade the performance of facial recognition given that the videos are more shaky due to unmitigated camera motion.

7.4.3.1 Data Collection.

The 4 participants act as the legitimate user in turn and the remaining 3 participants act as the imposters. During the legitimate user sessions, each legitimate user holds the phone and records his own face for 30 times. We refer to these videos as legitimate videos. During the spoofing attack sessions, each of the 3 imposters holds the phone but records the face of the legitimate user standing beside the imposter for 6 times to mimic a perfect silicone mask as assumed in Section 7.4.1. We refer to these videos as imposter videos. Each video recording is about 6s in length and the physical IMU readings are recorded simultaneously. The procedure is carried out first with video stabilization disabled. At the end, each participant recorded 48 videos when he held the phone with 30 of them being legitimate videos and the other 18 being imposter videos. We then repeat the procedure with video stabilization enabled. The total 384 videos (192 videos each set) are used for testing facial recognition and tremor recognition.

7.4.3.2 Test Procedure & Result

We generalize the authentication problem as an identification problem and use classification models to measure the effectiveness of the two authentication schemes against the spoofing attack.

Facial recognition Procedure. We tested MobileFaceNets [72] as the classification model which is a widely used facial recognition model designed for mobile platforms. 80% of each person's legitimate videos are used to enroll their faces. The remaining 20% of legitimate videos together with all imposter videos that contain faces of the legitimate users are used as the authentication test data.

Facial recognition Result. Both the legitimate users and imposters' videos authenticated with 100% success rate no matter the video stabilization was enabled or disabled. As expected, the results suggest that existing face authentication systems are mostly likely not designed to utilize camera motion side channel information. Mapping it to Equation 7.5, it suggests $\mathbb{E}[\mathcal{A}_u(m)|H_0] \to 1$ and $\mathbb{E}[1 - \mathcal{A}_u(m)|H_1] \to 0$ for the system under this specific spoofing attack. The results also show that disabling video stabilization to allow for more

capable virtual IMUs did not affect the performance of the original facial recognition system.

Tremor Recognition Procedure. For each video, we generate virtual IMU measurements using both the ITE and RSE methods. We extract common time-domain and frequency-domain features as the ones used in [174, 63]. As a simple proof-of-concept, we did not use sophisticated machine learning models but directly utilized Matlab's implementation of support vector machine (SVM) with a quadratic kernel and the default hyper-parameters [33]. 5-fold cross validation was performed in the training phase along with a one-vs-one multi-class classification method. Similar to facial recognition, for each legitimate user we use 80% of the legitimate videos (24 videos) in the training phase and the remaining 20% legitimate videos (6 videos) together with all imposter videos (18 videos, 6 from each of the three imposters) as authentication test data. We then calculate the true positive and true negative rates on the test set. To provide comparisons, we repeat the same procedure also for the physical IMU data.

Tremor Recognition Result. Table 7.1 shows the results of tremor recognition. Virtual IMU using RSE had performance approaching that of the physical IMU. It suggests that under this specific spoofing attack, $\mathbb{E}[\mathcal{A}_m([m_{_{int}}, m_{_{v_1}}])|H_0] \rightarrow 0.125$ and $\mathbb{E}[1 - \mathcal{A}_m([m_{_{int}}, m_{_{v_1}}])|H_1] \rightarrow 0.083$ if using an AND logic to combine facial and tremor recognition decisions. This results in $E_m - E_u \rightarrow -0.875c_1 + 0.083c_2$, which is highly likely to be smaller than 0. It is also clear that disabling video stabilization improves the performance of virtual IMUs.

7.4.3.3 Summary & Implication

Our preliminary tests indicate a high probability that integrating user postural hand tremor information from camera motion side channels will help existing facial recognition systems defend against visual spoofing attacks. Test results show MobileFaceNets could recognize legitimate users with 100% accuracy but could not detect (with 0% accuracy) a powerful silicone mask spoofing attack that almost perfectly replicates visual features of users. This behavior is not a design defect of existing facial recognition systems, but an anticipated outcome of only using visual information during an authentication process. On the other hand, virtual IMUs synthesized from camera motion channel were able to detect such a visual spoofing attack with over 87.5% accuracy at a cost of reducing true positive rate to 91.7%. The simplest approach of integrating virtual sensor into existing facial recognition systems is to have a standalone tremor recognition module that processes camera motion information in the videos, and have the system declare a legitimate user only when both this tremor recognition module and the original facial recognition module declare it simultaneously. In this way, the overall system's security performance increases in the face of facial spoofing

attacks even with a lower true positive rate. This result also suggests *when a physical sensor system has poor performance on a security task, it is easy to produce an obvious marginal benefit on the system's performance by integrating sensor side channel information.* Of course, a more sophisticated decision system can tune its weights on the facial and tremor recognition modules to strike a better balance between usability and security.

Beyond camera motion side channels, our tests also provide one viable data point for the general concept of utilizing sensor side channels and reveal some common problems it faces. For example, we expect the same problem of usability-security trade-off in using virtual sensors synthesized from sensor side channels alongside the original physical sensors. Essentially, physical sensors and synthesized virtual sensors provide two streams of information, each one of which is more reliable in one task but also unreliable in another task. The design trade-off appears when the overall system needs to complete both tasks to achieve its functionality.

7.4.3.4 Limitation & Future Work.

With the goal of showing a proof-of-concept example, our experiment provides empirical statistical evidence for the benefit of utilizing camera motion side channels only based on a very limited data distribution. The limitations of tested data lie in the following 4 main dimensions.

First, the 4 young male participants may not provide a high enough degree of demographic diversity, especially for evaluating postural hand tremors which are highly dependent on age, gender, and health conditions [124]. While we based our choice of the 4 participants on the hypothesis that more similar participants produce less distinct tremor patterns and thus help us estimate a lower bound of tremor recognition performance, we believe studying more diverse groups of people will generate new insights into recognition performance variability and possible strategies of recognition algorithm design.

Second, we collected 30 samples of legitimate-user videos and 18 spoofing attack videos for each legitimate user's authentication session within a single day. We find this initial set of samples provided evidence to suggest the potential of utilizing hand tremor information from camera side channels to enhance existing facial recognition system's security. It is possible that tremor patterns can change with time. Although previous research shows hand tremor remains stable after 78 days [93], a longer duration needs to be investigated in future complete. The recognition system may need to periodically update its database if tremor pattern is found to vary over time.

Third, we emulated perfect silicone masks by using the real faces of legitimate users. This only provides an estimate of the upper bound of the overall recognition system's performance

Table 7.1: Test accuracy of tremor recognition

	Physical IMU		Virtual ITE		Virtual RSE	
	TPR	TNR	TPR	TNR	TPR	TNR
Stab. OFF	95.8%	94.4%	62.5%	65.3%	91.7%	87.5%
Stab. ON	95.8%	93.1%	45.8%	41.7%	70.8%	72.2%

improvement when tremor recognition is used. Specifically, the benefit of including tremor recognition may get lower when a worse-quality silicone mask is used because the damage the attack can do to the original unimodal authentication system is lower while tremor recognition still causes a decrease in the true positive rate. As a result, we suggest future works test different qualities of silicone masks on popular facial recognition systems to better assess the benefit of including virtual IMUs for tremor recognition.

Fourth, the decoupling of facial recognition and tremor recognition problems in this proof-of-concept case study prevents us from utilizing the temporal correlation between the facial and camera motion signals and investigating the impact of the correlation information. Intuitively, systems that inspect such temporal correlation information require spoofing attackers to further achieve synchronization between the physical and virtual sensors' data streams and thus provide additional protection. We envision real-world products building upon the virtual sensors authentication concept to utilize deep-learning approaches for processing temporally-correlated physical and virtual sensors' information.

7.5 Discussion

Below we discuss the major areas of possible future work and interesting research questions.

Sensor Side Channel Models. To support future applications of sensor side channels, we believe more concrete and computable mathematical models than the framework proposed in Section 7.3 are needed as the current framework relies on abstract concepts instead of rigorous mathematical derivations. We envision future models to have the following features. First, they need to enable exact definitions and determination of different types of sensor side channels by providing the algorithms for calculating signal correlations and threshold values. Second, they need to provide quantitative metrics for measuring the usability-security trade-off mentioned in Section 7.4.3.3. Third, they need to delineate mechanisms for measuring the available signal quality and bandwidth of side channel measurement components.

Security for Sensor Side Channel Authentication. Technically, inseparable sensor side channels also provide the information needed for measurand authentication. We advocate the use of separable and controllable sensor side channels because they are protected

from adversaries that exploit unmitigated side channels. Nevertheless, risks of malicious exploitation still exist within authentication time. It is thus necessary for future works to consider how to ensure that side channels benefit the defender, but not adversaries that attempt eavesdropping and transduction attacks, during authentication.

We believe an access control and permission system that is similar to existing systems managing physical sensors on mobile platforms (e.g., Android) can be employed to prevent eavesdropping attacks. Virtual sensor entries can potentially be created and integrated into existing permission systems so that knowledge and methodology of solving physical sensors' problems can also benefit virtual sensors. Transduction attacks, on the other hand, are harder to address. In the context of sensor side channel based measurand authentication, transduction attacks can be generalized as authentication spoofing that tries to modify perceived characteristics of the byproducts, measurers, and environments. As a result, existing methodologies of spoofing detection may be applied. In summary, we believe there are opportunities to address the problems of virtual sensors by reflecting on existing methodology for physical sensors.

Side Channels vs. Legitimate Channels. We believe there will be an interesting phenomenon that sensor side channels are turned into legitimate communication channels when active controls and dedicated APIs are developed to support as well as regulate the use of sensor side channels in the future. After all, the key difference between side channels and legitimate channels is whether the channels are designed, intended, and allowed by the system's security policy or not. When such side channels are regarded as legitimate channels, however, new side-channel information may again be discovered to be embedded in such "legitimate" information as hardware and computation technologies keep advancing and extending the boundary of recoverable physical signals. We thus believe it is necessary for researchers to take a development perspective and periodically examine the security implications of sensor side channels.

Fewer Sensors via Sensor Repurposing. In a broader context, we believe the technique of synthesizing virtual sensors from sensor side channels aligns with the general idea of repurposing sensors for different sensing tasks. Essentially, we are trying to shift sensor hardware functionalities to the software space by understanding the transformation between different forms of signal energy and carrying out additional model-based computations. In contrast to the current trend of deploying more and more sensors in the Internet of Things era, we cannot help thinking if such sensor repurposing ideas would allow us to reduce the number of physical sensors and achieve more abstract and manageable sensor peripheral systems that are subjected to smaller attack surfaces.

Besides reducing the number of physical sensors, the technique could also be applied

to enhance existing systems that require new functionalities but have harsh environmental conditions where a hardware update is challenging. This idea is revealed in the example of NASA's Voyager 1 spacecraft which needed to measure plasma density in order to determine its location relative to the heliosphere. Voyager 1's plasma spectrometer stopped working in 1980, making a direct plasma density measurement impossible. However, the operation team learned that our sun sometimes emits shock waves that can cause the plasma surrounding the spacecraft to oscillate. The team then measured the oscillation using Voyager 1's onboard plasma wave sensing system as a proxy of the plasma density [114], essentially synthesizing a virtual plasma density sensor by understanding the energy transformations.

7.6 Conclusion

This chapter provides evidence that analog sensor side channels can benefit defenders by providing an opportunity to authenticate the sensor measurands. It provides an analytical framework for this problem and defines several key conditions that need to be met for the sensor side channels to be utilized for good purposes. Synthesizing virtual sensors from the side channels of physical sensors formulates a mechanism for repurposing existing sensor hardware to harvest extra modalities of information. We believe the applications of this mechanism can potentially span a much larger scope than authentication. Going forward, we envision that virtual sensor synthesis could develop into a new research area that actively interacts with the existing research areas of digital forensics, sensor fusion, multimodal deep learning and perception, etc. The fundamental research question we will need to explore is how to model the transformations between the energies of different information modalities.

CHAPTER 8

Conclusion

This thesis has shown how a side channel-based framework can be developed and applied to analyze the security and privacy problems in sensing. On a high level, it has characterized three major types of threat models, including (1) a software-domain adversary with access to sensor data trying to infer secret physical information, (2) a physical-domain adversary without access to sensor data trying to eavesdrop on the data by analyzing physical side-channel leakage from the sensing circuits, and (3) a physical-domain adversary trying to manipulate sensor data by coupling intentional electromagnetic signals into the analog sensing circuits. The thesis has provided three major case studies using camera sensing, and several other case studies using the examples of IMUs, temperature sensors, and keyboards. These case studies provide evidence that side channel problems widely exist around the sensor peripherals and hardware-software interfaces in present-day and potential future computing systems. The thesis has also highlighted how the increasing resolution & sensitivity, structural complexity, and unprotected but standardized data distribution of sensors are predicted to result in more serious sensor side channel problems in the near future.

8.1 Future Research

This thesis is the first step in bringing the security and privacy problems of sensing under a side-channel analysis framework. While Section 2.2 and Chapter 7 have provided a coarse-grained generic view of this framework and each of the following chapters has provided some detailed modeling to substantiate the framework, a complete mathematical model that connects the overall framework and the individual instances needs to be further developed. I also envision that the hypotheses **H1**-**H4** can be rigorously proved by having more detailed probabilistic models for the random variables.

Another important line of future research is to innovate modeling, computation, and experimental methods for automating the vulnerability prediction and verification process. For

example, more detailed computable data structures need to be synthesized from the literature on existing sensing security and privacy problems to enable the prediction of unknown vulnerabilities and threat models.

While this thesis presents the discovery and characterization of several zero-day vulnerabilities, it does not seek to verify whether these vulnerabilities have already been exploited in practice. Connecting security analysis to real-world impact is an important aspect that needs to further addressed. For example, static and dynamic analysis methods may be used to investigate to which level commercial applications on smartphones may be exploiting side-channel sensor data for malicious purposes.

Finally, it is crucial to further connect the system-centric problems of sensing security and privacy with existing secure computing and privacy theories. Future research needs to investigate the gaps between these theories and existing sensors' physical constraints to identify the appropriate systematic solutions to the problems.

BIBLIOGRAPHY

[1] Engineering Statistics Handbook: Sample Sizes Required. https://www.itl.nist.gov/div898/handbook/prc/section2/prc222.htm, 2012.

[2] Converting diagonal field of view and aspect ratio to horizontal and vertical field of view. http://vrguy.blogspot.com/2013/04/converting-diagonal-field-of-view-and.html, 2013.

[3] Webcam Field of View . https://www.telehealth.org.nz/assets/Uploads/1511-webcam-field-of-view.pdf, 2015.

[4] Approximate Focal Length for Webcams and Cell Phone Cameras. https://learnopencv.com/approximate-focal-length-for-webcams-and-cell-phone-cameras/, 2016.

[5] NIST Speech Signal to Noise Ratio Measurements. https://www.nist.gov/itl/iad/mig/nist-speech-signal-noise-ratio-measurements, 2016.

[6] Trick: Switching off the optical image stabilization of iPhone X, XS, XS Max, XR. https://www.sir-apfelot.de/en/switch-off-optical-image-stabilization-iphone-x-xs-xs-max-xr-23970/, 2019.

[7] Cisco Annual Internet Report (2018–2023) White Paper. https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html, 2020.

[8] ID Quantique and SK Telecom announce the world's first 5G smartphone equipped with a Quantum Random Number Generator (QRNG) chipset. https://www.idquantique.com/id-quantique-and-sk-telecom-announce-the-worlds-first-5g-smartphone-equipped-with-
2020.

[9] Schott AG: Transmittance of optical glass. https://www.schott.com/d/advanced_optics/5b1f5065-0587-4b3f-8fc7-e508b5348012/, 2020.

[10] The most maddening part about working from home: video conferences. . https://www.washingtonpost.com/technology/2020/03/16/remote-work-video-conference-coronavirus/, 2020.

[11] Acer Predator 15. https://www.acer.com/ac/en/IN/content/predator-model/NH.Q1YSI.001, 2021.

[12] Alexa SEO and Competitive Analysis Software. https://www.alexa.com/, 2021.

[13] Amazon Mechanical Turk. https://www.mturk.com/, 2021.

[14] Big Type Websites. https://www.siteinspire.com/websites?categories=22, 2021.

[15] Blue Light Blocking Glasses Market Size 2021 with a CAGR of 7.7% , Research by Business Opportunities, Top Companies data report covers, globally Market Key Facts and Forecast to 2025. https://www.wboc.com/story/43536337/blue-light, 2021.

[16] Blue Light Blocking Glasses on Amazon. https://www.amazon.com/gp/product/B07VBFSY33/, 2021.

[17] Cheese. https://wiki.gnome.org/Apps/Cheese, 2021.

[18] Default style sheet for HTML 4. https://www.w3.org/TR/CSS2/sample.html, 2021.

[19] For better or worse, working from home is here to stay. https://www.cnbc.com/2021/03/11/one-year-into-covid-working-from-home-is-here-to-stay.html, 2021.

[20] Let's Talk About Base Curves. https://opticianworks.com/lesson/lets-talk-base-curves/, 2021.

[21] Nikon Z7. https://www.nikonusa.com/en/nikon-products/product/mirrorless-cameras/z-7.html, 2021.

[22] Samsung Notebook 9. https://www.samsung.com/hk/pc/notebook-9-np900x5m-k03/, 2021.

[23] Shot Noise. https://en.wikipedia.org/wiki/Shot_noise, 2021.

[24] Vivo V21 introduces the first selfie camera with optical stabilisation. https://www.techadvisor.com/news/mobile-phone/vivo-v21-ois-selfie-camera-3804069/, 2021.

[25] Zoom. https://zoom.us/, 2021.

[26] AIDA64, Google Play. https://play.google.com/store/apps/details?id=com.finalwire.aida64&hl=en_US&gl=US, 2022.

[27] AVCam: Building a Camera App. https://developer.apple.com/documentation/avfoundation/cameras_and_media_capture/avcam_building_a_camera_app, 2022.

[28] Background Video Recorder 1, Google Play. https://play.google.com/store/apps/details?id=com.camera.secretvideorecorder&hl=en_US&gl=US, 2022.

[29] Engineering ToolBox, (2005). Required Voice Level at Distance. https://www.engineeringtoolbox.com/voice-level-d_938.html, 2022.

[30] Google Cloud Speech-to-Text. https://cloud.google.com/speech-to-text/, 2022.

[31] IBM Watson Speech to Text. https://speech-to-text-demo.ng.bluemix.net/, 2022.

[32] iOS App Dev Tutorials: Transcribing Speech to Text. https://developer.apple.com/tutorials/app-dev-training/transcribing-speech-to-text, 2022.

[33] Matlab templateSVM. https://www.mathworks.com/help/stats/templatesvm.html, 2022.

[34] Mic-Lock Microphone Blocker. https://www.amazon.com/Mic-Lock-Microphone-Blocker-Pack-Surveillance/dp/B078Z11LSG, 2022.

[35] Mideo, Apple App Store. https://apps.apple.com/us/app/mideo-record-video-with-music/id1358135284, 2022.

[36] Open Camera, Google Play. https://play.google.com/store/apps/details?id=net.sourceforge.opencamera&hl=en_US&gl=US, 2022.

[37] Pixel phone hardware tech specs. https://support.google.com/pixelphone/answer/7158570?hl=en#zippy=, 2022.

[38] SP Camera, Apple App Store. https://apps.apple.com/us/app/sp-camera/id603037910, 2022.

[39] Unique Urethane Dampens Shocks and Noise. https://web.archive.org/web/20090903100610/http://www.designnews.com/article/327542-Unique_Urethane_Dampens_Shocks_and_Noise.php#, 2022.

[40] 360. 360 m320 dashcam. https://shopee.sg/-Local-Seller-Brand-360-M301-M320-M320C-HD-Car-Dash-Camera-(Front-Rear)-(SD-Card-included)-i.585005157.12446270348, 2022. [Online; accessed 27-June-2023].

[41] Peshraw Ahmed Abdalla and Cihan Varol. Testing IoT security: The Case Study of an IP Camera. In *Proceedings of the 2020 8th International Symposium on Digital Forensics and Security (ISDFS)*. IEEE.

[42] Mariko Akutsu, Yasuhiro Oikawa, and Yoshio Yamasaki. Extract voice information using high-speed camera. In *Proceedings of Meetings on Acoustics ICA2013*, volume 19, page 055019. Acoustical Society of America, 2013.

[43] S Abhishek Anand and Nitesh Saxena. Speechless: Analyzing the threat to speech privacy from smartphone motion sensors. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 1000–1017. IEEE, 2018.

[44] S Abhishek Anand, Chen Wang, Jian Liu, Nitesh Saxena, and Yingying Chen. Spearphone: A speech privacy exploit via accelerometer-sensed reverberations from smartphone loudspeakers. *arXiv preprint arXiv:1907.05972*, 2019.

[45] S Abhishek Anand, Chen Wang, Jian Liu, Nitesh Saxena, and Yingying Chen. Spearphone: a lightweight speech privacy exploit via accelerometer-sensed reverberations from smartphone loudspeakers. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 288–299, 2021.

[46] Aptina. High-speed serial pixel (hispi) interface protocol. https://files.niemo.de/aptina_pdfs/High-Speed_Serial_Pixel_%28HiSPi%29_Interface_Specification.pdf, 2011. [Online; accessed 12-June-2023].

[47] Aries Arditi. Adjustable typography: an approach to enhancing low vision text accessibility. *Ergonomics*, 47(5):469–482, 2004.

[48] Aries Arditi and Jianna Cho. Serifs and font legibility. *Vision research*, 45(23):2926–2933, 2005.

[49] Melanie Arntz, Sarra Ben Yahmed, and Francesco Berlingieri. Working from home and covid-19: The chances and risks for gender gaps. *Intereconomics*, 55(6):381–386, 2020.

[50] Dmitri Asonov and Rakesh Agrawal. Keyboard acoustic emanations. In *IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004*, pages 3–11. IEEE, 2004.

[51] Zhongjie Ba, Tianhang Zheng, Xinyu Zhang, Zhan Qin, Baochun Li, Xue Liu, and Kui Ren. Learning-based practical smartphone eavesdropping with built-in accelerometer. In *NDSS*, 2020.

[52] Michael Backes, Tongbo Chen, Markus Dürmuth, Hendrik PA Lensch, and Martin Welk. Tempest in a teapot: Compromising reflections revisited. In *2009 30th IEEE Symposium on Security and Privacy*, pages 315–327. IEEE, 2009.

[53] Michael Backes, Markus Dürmuth, and Dominique 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.

[54] Constantine A Balanis. *Antenna theory: analysis and design*. John wiley & sons, 2016.

[55] Davide Balzarotti, Marco Cova, and Giovanni Vigna. Clearshot: Eavesdropping on keyboard input from video. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 170–183. IEEE, 2008.

[56] Andrea Barisani and Daniele Bianco. Sniffing keystrokes with lasers/voltmeters. *Black Hat USA*, 2009.

[57] Sören Becker, Marcel Ackermann, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Interpreting and explaining deep neural networks for classification of audio signals. *CoRR*, abs/1807.03418, 2018.

[58] Stephen Beeby, Graham Ensel, Neil M White, and Michael Kraft. *MEMS mechanical sensors*. Artech House, 2004.

[59] Alexander Bick, Adam Blandin, and Karel Mertens. Work from home after the covid-19 outbreak.*CEPR Discussion Paper*. 2020.

[60] Peter Birkholz. Modeling consonant-vowel coarticulation for articulatory speech synthesis. *PloS one*, 8(4):e60603, 2013.

[61] Hristo Bojinov, Yan Michalevsky, Gabi Nakibly, and Dan Boneh. Mobile device identification via sensor fingerprinting. *arXiv preprint arXiv:1408.1416*, 2014.

[62] Connor Bolton, Kevin Fu, Josiah Hester, and Jun Han. How to curtail oversensing in the home. *Communications of the ACM*, 63(6):20–24, 2020.

[63] Connor Bolton, Yan Long, Jun Han, Josiah Hester, and Kevin Fu. Touchtone leakage attacks via smartphone sensors: mitigation without hardware modification. *arXiv preprint arXiv:2109.13834*, 2021.

[64] Connor Bolton, Yan Long, Jun Han, Josiah Hester, and Kevin Fu. Characterizing and Mitigating Touchtone Eavesdropping in Smartphone Motion Sensors. In *Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses*, 2023.

[65] Connor Bolton, Sara Rampazzi, Chaohao Li, Andrew Kwong, Wenyuan Xu, and Kevin Fu. Blue note: How intentional acoustic interference damages availability and integrity in hard disk drives and operating systems. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 1048–1062. IEEE, 2018.

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

[67] Benjamin Cannoles and Ahmad Ghafarian. Hacking experiment by using usb rubber ducky scripting. *Journal of Systemics*, 2017.

[68] Stefano Cecconello, Alberto Compagno, Mauro Conti, Daniele Lain, and Gene Tsudik. Skype & type: Keyboard eavesdropping in voice-over-ip. *ACM Transactions on Privacy and Security (TOPS)*, 22(4):1–34, 2019.

[69] Anadi Chaman, Jiaming Wang, Jiachen Sun, Haitham Hassanieh, and Romit Roy Choudhury. Ghostbuster: Detecting the presence of hidden eavesdroppers. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018.

[70] Bo Chen, Vivek Yenamandra, and Kannan Srinivasan. Tracking Keystrokes Using Wireless Signals. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, 2015.

[71] Lung-Tai Chen, Chia-Yen Lee, and Wood-Hi Cheng. Mems-based humidity sensor with integrated temperature compensation mechanism. *Sensors and Actuators A: Physical*, 147(2):522–528, 2008.

[72] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *Chinese Conference on Biometric Recognition*, pages 428–438. Springer, 2018.

[73] Tianqi Chen and Carlos Guestrin. Xgboost. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug 2016.

[74] Chi-Wei Chiu, Paul C-P Chao, Nicholas Y-Y Kao, and Fu-Kuan Young. Optimal design and experimental verification of a magnetically actuated optical image stabilization system for cameras in mobile phones. *Journal of Applied Physics*, 103(7):07F136, 2008.

[75] Yun Chan Cho and Jae Wook Jeon. Remote robot control system based on dtmf of mobile phone. In *2008 6th IEEE International Conference on Industrial Informatics*, pages 1441–1446. IEEE, 2008.

[76] Jieun Choi, Hae-Yong Yang, and Dong-Ho Cho. Tempest Comeback: A Realistic Audio Eavesdropping Threat on Mixed-signal Socs. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS)*.

[77] Jun Choi. Countermeasures for BadUSB Vulnerability. In *Proceedings of the Conference on Information Security and Cryptology*, 2015.

[78] Michal J Chojnacky, WM Miller, and GF Strouse. Methods for accurate cold-chain temperature monitoring using digital data-logger thermometers. In *AIP Conference Proceedings*, volume 1552, pages 1014–1019. American Institute of Physics, 2013.

[79] Alberto Compagno, Mauro Conti, Daniele Lain, and Gene Tsudik. Don't skype & type! acoustic eavesdropping in voice-over-ip. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pages 703–715, 2017.

[80] Cortron. Rugged keyboards for military industrial applications. https://www.cortroninc.com/rugged-keyboards-for-military-and-industrial-applications/, 2023. [Online; accessed 17-April-2023].

[81] Lothar Cremer and Manfred Heckl. *Structure-borne sound: structural vibrations and sound radiation at audio frequencies*. Springer Science & Business Media, 2013.

[82] Patrick Cronin, Xing Gao, Haining Wang, and Chase Cotton. Time-print: Authenticating usb flash drives with novel timing fingerprints. In *Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP)*.

[83] Anupam Das, Martin Degeling, Xiaoyou Wang, Junjue Wang, Norman Sadeh, and Mahadev Satyanarayanan. Assisting users in a world full of cameras: A privacy-aware infrastructure for computer vision applications. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1387–1396. IEEE, 2017.

[84] Das Keyboard Blog. Typing through time: Keyboard history. https://www.daskeyboard.com/blog/typing-through-time-the-history-of-the-keyboard/, 2011. [Online; accessed 1-Dec-2022].

[85] Abe Davis, Michael Rubinstein, Neal Wadhwa, Gautham J Mysore, Fredo Durand, and William T Freeman. The visual microphone: Passive recovery of sound from video. 2014.

[86] Gökçen Yılmaz Dayanıklı, Abdullah Zubair Mohammed, Ryan Gerdes, and Mani Mina. Wireless Manipulation of Serial Communication. In *Proceedings of the 2022 ACM Asia Conference on Computer and Communications Security (ACM ASIACCS 22)*.

[87] Gökçen Yilmaz Dayanikli. *Electromagnetic Interference Attacks on Cyber-Physical Systems: Theory, Demonstration, and Defense*. PhD thesis, Virginia Tech, 2021.

[88] Gökçen Yilmaz Dayanikli, Rees R Hatch, Ryan M Gerdes, Hongjie Wang, and Regan Zane. Electromagnetic Sensor and Actuator Attacks on Power Converters for Electric Vehicles. In *Proceedings of the 2020 IEEE Security and Privacy Workshops (SPW)*.

[89] Gökçen Yılmaz Dayanıklı, Sourav Sinha, Devaprakash Muniraj, Ryan M Gerdes, Mazen Farhood, and Mani Mina. Physical-Layer Attacks Against Pulse Width Modulation-Controlled Actuators. In *Proceedings of the 31st USENIX Security Symposium (USENIX Security 22)*, 2022.

[90] Zechuan Deng, René Morissette, and Derek Messacar. Running the economy remotely: Potential for working from home during and after covid-19. *Statistics Canada*. 2020.

[91] Android Developers. *Android Debug Bridge*, 2023. https://developer.android.com/studio/command-line/adb.

[92] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. Image quality assessment: Unifying structure and texture similarity. *arXiv preprint arXiv:2004.07728*, 2020.

[93] Kelsey Dun. Master's thesis: Replicability and uniqueness of tremor characteristics in parkinson's disease. 2019.

[94] Rodger J Elble. Tremor. In *Neuro-geriatrics*, pages 311–326. Springer, 2017.

[95] Rodger J Elble, Helge Hellriegel, Jan Raethjen, and Günther Deuschl. Assessment of head tremor with accelerometers versus gyroscopic transducers. *Movement Disorders Clinical Practice*, 4(2):205–211, 2017.

[96] Fürkan Elibol, Uğur Sarac, and Işin Erer. Realistic eavesdropping attacks on computer displays with low-cost and mobile receiver system. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 1767–1771. IEEE, 2012.

[97] Ettus Research. Ettus research usrp products. https://www.ettus.com/products/, 2023. [Online; accessed 12-June-2023].

[98] Sina Farsiu, M Dirk Robinson, Michael Elad, and Peyman Milanfar. Fast and robust multiframe super resolution. *IEEE transactions on image processing*, 13(10):1327–1344, 2004.

[99] Denis Foo Kune, John Backes, Shane S Clark, Daniel Kramer, Matthew Reynolds, Kevin Fu, Yongdae Kim, and Wenyuan Xu. Ghost talk: Mitigating EMI signal injection attacks against analog sensors. In *Proceedings of the 34th IEEE Symposium on Security and Privacy (SP)*, pages 145–159. IEEE, 2013.

[100] Centers for Disease Control and Prevention. Vaccine Storage and Handling Toolkit. www.cdc.gov/vaccines/hcp/admin/storage/toolkit/index.html, 2021.

[101] Lena Franklin and David Huber. Exploiting camera rolling shutter to detect high frequency signals. In *Applications of Digital Image Processing XLII*, volume 11137, page 111370H. International Society for Optics and Photonics, 2019.

[102] Kevin Fu and Wenyuan Xu. Risks of trusting the physics of sensors. *Communications of the ACM*, 61(2):20–23, 2018.

[103] Gadget Freakz. Xiaomi dafang 1080p smart monitor camera review. https://gadget-freakz.com/xiaomi-dafang-1080p-smart-monitor-camera-review/, 2018. [Online; accessed 27-June-2023].

[104] Ahmed Abdurabu Nasser Galaom. *Integration of a MEMS-based Autofocus Actuator into a Smartphone Camera*. PhD thesis, University of Toronto (Canada), 2016.

[105] Jesús A García, Evangelina Lara, and Leocundo Aguilar. A low-cost calibration method for low-cost mems accelerometers based on 3d printing. *Sensors*, 20(22):6454, 2020.

[106] Daniel Genkin, Mihir Pattani, Roei Schuster, and Eran Tromer. Synesthesia: Detecting screen content via remote acoustic side channels. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 853–869. IEEE, 2019.

[107] Ilias Giechaskiel and Kasper Rasmussen. Taxonomy and challenges of out-of-band signal injection attacks and defenses. *IEEE Communications Surveys & Tutorials*, 22(1):645–670, 2019.

[108] Virgil D Gligor. *A guide to understanding covert channel analysis of trusted systems*, volume 30. National Computer Security Center, 1994.

[109] Gloria Cascarino. Medical equipment for outpatient care. https://www.hfmmagazine.com/articles/1349-medical-equipment-for-outpatient-care, 2014. [Online; accessed 17-April-2023].

[110] W Goepel, J Hesse, and JN Zemel. Sensors–a comprehensive survey, fundamentals and general aspects, 1994.

[111] Federico Griscioli, Maurizio Pizzonia, and Marco Sacchetti. USBCheckIn: Preventing BadUSB Attacks by Forcing Human-device Interaction. In *Proceedings of the 2016 14th Annual Conference on Privacy, Security and Trust (PST)*. IEEE.

[112] Jinwei Gu, Yasunobu Hitomi, Tomoo Mitsunaga, and Shree Nayar. Coded rolling shutter photography: Flexible space-time sampling. In *2010 IEEE International Conference on Computational Photography (ICCP)*, pages 1–8. IEEE, 2010.

[113] Bahadir K Gunturk, John Glotzbach, Yucel Altunbasak, Ronald W Schafer, and Russel M Mersereau. Demosaicking: Color Filter Array Interpolation. *IEEE Signal processing magazine*, 22(1):44–54, 2005.

[114] DA Gurnett, WS Kurth, LF Burlaga, and NF Ness. In situ observations of interstellar plasma with voyager 1. *Science*, 341(6153):1489–1492, 2013.

[115] Tzipora Halevi and Nitesh Saxena. Keyboard acoustic side channel attacks: exploring realistic and security-sensitive scenarios. *International Journal of Information Security*, 14(5):443–456, 2015.

[116] Yuichi Hayashi, Naofumi Homma, Mamoru Miura, Takafumi Aoki, and Hideaki Sone. A Threat for Tablet Pcs in Public Space: Remote Visualization of Screen Images Using EM Emanation. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)*.

[117] Grant Hernandez, Farhaan Fowze, Dave Tian, Tuba Yavuz, and Kevin RB Butler. Firmusb: Vetting USB Device Firmware Using Domain Informed Symbolic Execution. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (ACM CCS 17)*.

[118] Samuel Herodotou and Feng Hao. Spying on the spy: Security analysis of hidden cameras. *arXiv preprint arXiv:2306.00610*, 2023.

[119] Atsuki Higashiyama, Yoshikazu Yokoyama, and Koichi Shimono. Perceived distance of targets in convex mirrors. *Japanese Psychological Research*, 43(1):13–24, 2001.

[120] Jan Malte Hilgefort, Daniel Arp, and Konrad Rieck. Spying through virtual backgrounds of video calls. In *Proceedings of the 14th ACM Workshop on Artificial Intelligence and Security*, pages 135–144, 2021.

[121] Brien A Holden, Timothy R Fricke, David A Wilson, Monica Jong, Kovin S Naidoo, Padmaja Sankaridurg, Tien Y Wong, Thomas J Naduvilath, and Serge Resnikoff. Global prevalence of myopia and high myopia and temporal trends from 2000 through 2050. *Ophthalmology*, 123(5):1036–1042, 2016.

[122] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021.

[123] Pengfei Hu, Hui Zhuang, Panneer Selvam Santhalingam, Riccardo Spolaor, Parth Pathak, Guoming Zhang, and Xiuzhen Cheng. Accear: Accelerometer acoustic eavesdropping with unconstrained vocabulary. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1757–1773. IEEE, 2022.

[124] JP Hubble, KL Busenbark, R Pahwa, K Lyons, and WC Koller. Clinical expression of essential tremor: effects of gender and age. *Movement disorders*, 12(6):969–972, 1997.

[125] Infinite Reality. Digital video port (dvp) specification. https://irix7.com/techpubs/007-3594-001.pdf, 2011. [Online; accessed 12-June-2023].

[126] Intel. *Intel NUC*, 2023. https://www.intel.com/content/www/us/en/products/boards-kits/nuc.html.

[127] Mohammad Moinul Islam, Vijayan K Asari, Mohammed Nazrul Islam, and Mohammad A Karim. Video super-resolution by adaptive kernel regression. In *International Symposium on Visual Computing*, pages 799–806. Springer, 2009.

[128] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image Translation with Conditional Adversarial Networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[129] Vidit Jain and Erik Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010.

[130] Xiaoyu Ji, Yushi Cheng, Yuepeng Zhang, Kai Wang, Chen Yan, Wenyuan Xu, and Kevin Fu. Poltergeist: Acoustic adversarial machine learning against cameras and computer vision. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 160–175. IEEE, 2021.

[131] Qinhong Jiang, Xiaoyu Ji, Chen Yan, Zhixin Xie, Haina Lou, and Wenyuan Xu. GlitchHiker: Uncovering Vulnerabilities of Image Signal Transmission with IEMI. In *Proceedings of the USENIX Security 23*, 2023.

[132] Qinhong Jiang, Xiaoyu Ji, Chen Yan, Zhixin Xie, Haina Lou, and Wenyuan Xu. GlitchHiker: Uncovering Vulnerabilities of Image Signal Transmission with IEMI. In *Proceedings of the 32st USENIX Security Symposium (USENIX Security 23)*, 2023.

[133] Qinhong Jiang, Yanze Ren, Yan Long, Chen Yan, Yumai Sun, Xiaoyu Ji, Kevin Fu, and Wenyuan Xu. Em eye: Characterizing electromagnetic side-channel eavesdropping on embedded cameras. In *Network and Distributed Systems Security (NDSS) Symposium*, 2024.

[134] Yan Jiang, Xiaoyu Ji, Kai Wang, Chen Yan, Richard Mitev, Ahmad-Reza Sadeghi, and Wenyuan Xu. WIGHT: Wired Ghost Touch Attack on Capacitive Touchscreens. In *Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP)*.

[135] Lovelave B Jr. U.S. quarantines Pfizer vaccine shipments in California and Alabama after transit 'anomaly' left vials too cold. www.cnbc.com/2020/12/16/covid-vaccine-us-quarantines-pfizer-shipments-in-california-alabama-after-transit-html, 2020.

[136] Marc Juarez, Sadia Afroz, Gunes Acar, Claudia Diaz, and Rachel Greenstadt. A critical evaluation of website fingerprinting attacks. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 263–274, 2014.

[137] Katherine A Karl, Joy V Peluchette, and Navid Aghakhani. Virtual work meetings during the covid-19 pandemic: The good, bad, and ugly. *Small Group Research*, 53(3):343–365, 2022.

[138] Nohl Karsten, Kribler Sascha, and Lell Jakob. Badusb-on accessories that turn evil. *Black Hat USA*, 2014.

[139] Keetouch. What challenges does industrial equipment solve? https://keetouch.eu/en/news/what-challenges-does-industrial-touch-equipment-solve.html, 2020. [Online; accessed 17-April-2023].

[140] Amin Kharraz, Brandon L Daley, Graham Z Baker, William Robertson, and Engin Kirda. USBESAFE: An End-Point Solution to Protect Against USB-Based Attacks. In *Proceedings of the 22nd International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, 2019.

[141] Sebastian Köhler, Richard Baker, and Ivan Martinovic. Signal injection attacks against ccd image sensors. In *Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security*, 2022.

[142] Sebastian Köhler, Giulio Lovisotto, Simon Birnbach, Richard Baker, and Ivan Martinovic. They see me rollin': Inherent vulnerability of the rolling shutter in cmos image sensors. In *Annual Computer Security Applications Conference*, pages 399–413, 2021.

[143] Markus G Kuhn. Optical Time-domain Eavesdropping Risks of CRT Displays. In *Proceedings of the 2002 IEEE Symposium on Security and Privacy (SP)*.

[144] Markus G Kuhn. Electromagnetic eavesdropping risks of flat-panel displays. In *International Workshop on Privacy Enhancing Technologies*, pages 88–107. Springer, 2004.

[145] Markus G Kuhn. Electromagnetic Eavesdropping Risks of Flat-panel Displays. In *Proceedings of the International Workshop on Privacy Enhancing Technologies*. Springer, 2004.

[146] Markus G Kuhn. Security limits for compromising emanations. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 265–279. Springer, 2005.

[147] Markus G Kuhn and Ross J Anderson. Soft Tempest: Hidden Data Transmission Using Electromagnetic Rmanations. In *Proceedings of the International Workshop on Information Hiding*. Springer, 1998.

[148] Rohit Kulkarni. A Million News Headlines, 2018.

[149] Chao-Hsien Kuo and Zhen Ye. Sonic crystal lenses that obey the lensmaker's formula. *Journal of Physics D: Applied Physics*, 37(15):2155, 2004.

[150] Andrew Kwong, Wenyuan Xu, and Kevin Fu. Hard drive of hearing: Disks that eavesdrop with a synthesized microphone. In *2019 IEEE symposium on security and privacy (SP)*, pages 905–919. IEEE, 2019.

[151] Tuljappa M Ladwa, Sanjay M Ladwa, R Sudharshan Kaarthik, Alok Ranjan Dhara, and Nayan Dalei. Control of remote domestic system using dtmf. In *International Conference on Instrumentation, Communication, Information Technology, and Biomedical Engineering 2009*, pages 1–6. IEEE, 2009.

[152] Michael Li. I studied the fonts of the top 1000 websites. Here's what I learned. https://dribbble.com/stories/2021/04/26/web-design-data-fonts, 2021.

[153] Rongzhong Li. A true random number generator algorithm from digital camera image noise for varying lighting conditions. In *SoutheastCon 2015*, pages 1–8. IEEE, 2015.

[154] Chia-Kai Liang, Li-Wen Chang, and Homer H Chen. Analysis and compensation of rolling shutter effect. *IEEE Transactions on Image Processing*, 17(8):1323–1330, 2008.

[155] Tony Lindeberg. Scale invariant feature transform. 2012.

[156] Zhen Ling, Kaizheng Liu, Yiling Xu, Yier Jin, and Xinwen Fu. An End-to-end View of IoT Security and Privacy. In *Proceedings of the 2017 IEEE Global Communications Conference (GLOBECOM)*.

[157] Jian Liu, Yan Wang, Gorkem Kar, Yingying Chen, Jie Yang, and Marco 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*, 2015.

[158] Xiangyu Liu, Zhe Zhou, Wenrui Diao, Zhou Li, and Kehuan Zhang. When Good Becomes Evil: Keystroke Inference with Smartwatch. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (ACM CCS 15)*, 2015.

[159] Zhuoran Liu, Niels Samwel, Léo Weissbart, Zhengyu Zhao, Dirk Lauret, Lejla Batina, and Martha Larson. Screen Gleaning: A Screen Reading TEMPEST Attack on Mobile Devices Exploiting an Electromagnetic Side Channel. In *Proceedings of the 28th Annual Network and Distributed System Security Symposium (NDSS)*, 2021.

[160] Ziwei Liu, Feng Lin, Chao Wang, Yijie Shen, Zhongjie Ba, Li Lu, Wenyao Xu, and Kui Ren. CamRadar: Hidden Camera Detection Leveraging Amplitude-modulated Sensor Images Embedded in Electromagnetic Emanations. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(4):1–25, 2023.

[161] Yan Long and Kevin Fu. Side auth: Synthesizing virtual sensors for authentication. In *Proceedings of the 2022 New Security Paradigms Workshop*, pages 35–44, 2022.

[162] Yan Long, Qinhong Jiang, Chen Yan, Tobias Alam, Xiaoyu Ji, Wenyuan Xu, and Kevin Fu. Em eye: Characterizing electromagnetic side-channel eavesdropping on embedded cameras. In *Network and Distributed Systems Security (NDSS) Symposium*, 2024.

[163] Yan Long, Pirouz Naghavi, Blas Kojusner, Kevin Butler, Sara Rampazzi, and Kevin Fu. Side Eye: Characterizing the Limits of POV Acoustic Eavesdropping from Smartphone Cameras with Rolling Shutters and Movable Lenses. In *Proceedings of the 2023 IEEE Symposium on Security and Privacy (SP)*.

[164] Yan Long, Sara Rampazzi, Takeshi Sugawara, and Kevin Fu. Protecting covid-19 vaccine transportation and storage from analog cybersecurity threats. *Biomedical Instrumentation & Technology*, 55(3):112–117, 2021.

[165] Yan Long, Chen Yan, Shilin Xiao, Shivan Prasad, Wenyuan Xu, and Kevin Fu. Private eye: On the limits of textual screen peeking via eyeglass reflections in video conferencing. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 3432–3449. IEEE, 2023.

[166] Mark W Maciejewski, Harry Z Qui, Iulian Rujan, Mehdi Mobli, and Jeffrey C Hoch. Nonuniform sampling and spectral aliasing. *Journal of Magnetic Resonance*, 199(1):88–93, 2009.

[167] Anindya Maiti, Oscar Armbruster, Murtuza Jadliwala, and Jibo 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 (ACM ASIA CCS 16)*, 2016.

[168] Robert J. Marks, II. *Introduction to Shannon Sampling and Interpolation Theory*. Springer-Verlag, Berlin, Heidelberg, 1991.

[169] Philip Marquardt, Arunabh Verma, Henry Carter, and Patrick Traynor. (sp)iphone: Decoding vibrations from nearby keyboards using mobile phone accelerometers. In *Proceedings of the 18th ACM conference on Computer and Communications Security (ACM CCS 11)*, 2011.

[170] Seita Maruyama, Satohiro Wakabayashi, and Tatsuya Mori. Tap'n ghost: A Compilation of Novel Attack Techniques against Smartphone Touchscreens. In *Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP)*.

[171] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. Gyrophone: Recognizing speech from gyroscope signals. In *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pages 1053–1067, 2014.

[172] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. Gyrophone: Recognizing speech from gyroscope signals. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 1053–1067, 2014.

[173] MIPI Alliance. Mipi csi-2 specifications. https://www.mipi.org/specifications/csi-2, 2023. [Online; accessed 12-June-2023].

[174] Oana Miu, Adrian Zamfir, and Corneliu Florea. Person identification based on hand tremor characteristics. *arXiv preprint arXiv:1606.06840*, 2016.

[175] John V Monaco. SoK: Keylogging Side Channels. In *Proceedings of the 2018 IEEE Symposium on Security and Privacy (SP)*.

[176] Debabrata Mukher jee and Makarand V Ratnaparkhi. On the functional relationship between entropy and variance with related applications. *Communications in Statistics-Theory and Methods*, 15(1):291–311, 1986.

[177] Ben Nassi, Yaron Pirutin, Tomer Galor, Yuval Elovici, and Boris Zadov. Glowworm attack: Optical tempest sound recovery via a device's power indicator led. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 1900–1914, 2021.

[178] Ben Nassi, Yaron Pirutin, Adi Shamir, Yuval Elovici, and Boris Zadov. Lamphone: Real-time passive sound recovery from light bulb vibrations. *Cryptology ePrint Archive*, 2020.

[179] Ben Nassi, Ras Swissa, Yuval Elovici, and Boris Zadov. The little seal bug: Optical sound recovery from lightweight reflective objects. Cryptology ePrint Archive, Report 2022/227, 2022. https://ia.cr/2022/227.

[180] Surya Michrandi Nasution, Yudha Purwanto, Agus Virgono, and Girindra Chandra Alam. Integration of Kleptoware as Keyboard Keylogger for Input Recorder Using Teensy USB Development Board. In *Proceedings of the 8th International Conference on Telecommunication Systems Services and Applications (TSSA)*. IEEE, 2014.

[181] Sebastian Neuner, Artemios G Voyiatzis, Spiros Fotopoulos, Collin Mulliner, and Edgar R Weippl. Usblock: Blocking USB-based Keypress Injection Attacks. In *Proceedings of the IFIP Annual Conference on Data and Applications Security and Privacy*. Springer, 2018.

[182] Ajaya Neupane, Md Rahman, Nitesh Saxena, et al. Peep: Passively Eavesdropping Private Input via Brainwave Signals. In *Proceedings of the International Conference on Financial Cryptography and Data Security*. Springer, 2017.

[183] Marc Newlin. Mousejack, keysniffer and beyond: Keystroke sniffing and injection vulnerabilities in 2.4 ghz wireless mice and keyboards. *DEFCON*, 2016.

[184] Gabriele Oligeri, Savio Sciancalepore, Simone Raponi, and Roberto Di Pietro. Brokenstrokes: On the (in) security of wireless keyboards. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2020.

[185] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H Adelson, and William T Freeman. Visually indicated sounds. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2405–2413, 2016.

[186] Tatsiana Palavets and Mark Rosenfield. Blue-blocking filters and digital eyestrain. *Optometry and Vision Science*, 96(1):48–54, 2019.

[187] picamera. Camera hardware: Sensor modes. https://picamera.readthedocs.io/en/latest/fov.html#sensor-modes, 2023. [Online; accessed 27-June-2023].

[188] Rahul Potharaju, Andrew Newell, Cristina Nita-Rotaru, and Xiangyu Zhang. Plagiarizing smartphone applications: attack strategies and defense techniques. In *International symposium on engineering secure software and systems*, pages 106–120. Springer, 2012.

[189] Ariadna Quattoni and Antonio Torralba. Recognizing Indoor Scenes. In *Proceedings of the 2009 IEEE conference on computer vision and pattern recognition (CVPR)*.

[190] Raghavendra Ramachandra, Sushma Venkatesh, Kiran B Raja, Sushil Bhattacharjee, Pankaj Wasnik, Sebastien Marcel, and Christoph Busch. Custom silicone face masks: Vulnerability of commercial face recognition systems & presentation attack detection. In *2019 7th International Workshop on Biometrics and Forensics (IWBF)*, pages 1–6. IEEE, 2019.

[191] B Srinivasa Reddy and Biswanath N Chatterji. An fft-based technique for translation, rotation, and scale-invariant image registration. *IEEE transactions on image processing*, 5(8):1266–1271, 1996.

[192] Jeff Rodman. The effect of bandwidth on speech intelligibility. *Polycom inc., White paper*, 2003.

[193] RTL-SDR. Buy rtl-sdr dongles (rtl2832u). https://www.rtl-sdr.com/buy-rtl-sdr-dvb-t-dongles/, 2023. [Online; accessed 26-June-2023].

[194] Mohd Sabra, Anindya Maiti, and Murtuza Jadliwala. Zoom on the keystrokes: Exploiting video calls for keystroke inference attacks. *arXiv preprint arXiv:2010.12078*, 2020.

[195] Tim Sainburg. timsainb/noisereduce: v1.0, June 2019.

[196] Tim Sainburg, Marvin Thielk, and Timothy Q Gentner. Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires. *PLoS computational biology*, 16(10):e1008228, 2020.

[197] Jerome H Saltzer and Michael D Schroeder. The protection of information in computer systems. *Proceedings of the IEEE*, 63(9):1278–1308, 1975.

[198] Sriram Sami, Sean Rui Xiang Tan, Yimin Dai, Nirupam Roy, and Jun Han. Lidarphone: acoustic eavesdropping using a lidar sensor. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pages 701–702, 2020.

[199] Mehul P Sampat, Zhou Wang, Shalini Gupta, Alan Conrad Bovik, and Mia K Markey. Complex wavelet structural similarity: A new image similarity index. *IEEE transactions on image processing*, 18(11):2385–2401, 2009.

[200] Ariel Schwarz, Yosef Sanhedrai, and Zeev Zalevsky. Digital Camera Detection and Image Disruption Using Controlled Intentional Electromagnetic Interference. *IEEE transactions on electromagnetic compatibility*, 54(5):1048–1054, 2012.

[201] Jayaprakash Selvaraj. *Intentional Electromagnetic Interference Attack on Sensors and Actuators*. PhD thesis, Iowa State University, 2018.

[202] ROHM Semiconductor. Optical image stabilization (ois) white paper, 2015.

[203] Haoqi Shan, Boyi Zhang, Zihao Zhan, Dean Sullivan, Shuo Wang, and Yier Jin. Invisible Finger: Practical Electromagnetic Interference Attack on Touchscreen-based Electronic Devices. In *Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP)*.

[204] C.E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, Jan 1949.

[205] Cheng Shen and Jun Huang. Earfisher: Detecting wireless eavesdroppers by stimulating and sensing memory emr. In *Proceedings of the 18th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2021.

[206] Hiroki Shindo, Koichi Terano, Kenta Iwai, Takahiro Fukumori, and Takanobu Nishiura. *Noise-reducing sound capture based on exposure-time of still camera*. Universitätsbibliothek der RWTH Aachen, 2019.

[207] Ilia Shumailov, Laurent Simon, Jeff Yan, and Ross Anderson. Hearing your touch: A new acoustic side channel on smartphones. *arXiv preprint arXiv:1903.11137*, 2019.

[208] Laurent Simon and Ross Anderson. Pin skimmer: Inferring pins through the camera and microphone. In *Proceedings of the Third ACM workshop on Security and privacy in smartphones & mobile devices*, pages 67–78, 2013.

[209] Yunmok Son, Hocheol Shin, Dongkwan Kim, Youngseok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, and Yongdae Kim. Rocking drones with intentional sound noise on gyroscopic sensors. In *24th USENIX Security Symposium (USENIX Security 15)*, pages 881–896, 2015.

[210] Dawn Xiaodong Song, David Wagner, and Xuqing Tian. Timing Analysis of Keystrokes and Timing Attacks on {SSH}. In *Proceedings of the 10th USENIX Security Symposium (USENIX Security 01)*, 2001.

[211] Myeong-Gyu Song, Young-Jun Hur, No-Cheol Park, Young-Pil Park, Kyoung-Su Park, Soo-Cheol Lim, and Jae-Hyuk Park. Development of small sized actuator for optical image stabilization. In *2009 International Symposium on Optomechatronic Technologies*, pages 152–157. IEEE, 2009.

[212] Toshiki Sonoda, Hajime Nagahara, Kenta Endo, Yukinobu Sugiyama, and Rin-ichiro Taniguchi. High-speed imaging using cmos image sensor with quasi pixel-wise exposure. In *2016 IEEE International Conference on Computational Photography (ICCP)*, pages 1–11. IEEE, 2016.

[213] Josef Spjut, Ben Boudaoud, Kamran Binaee, Jonghyun Kim, Alexander Majercik, Morgan McGuire, David Luebke, and Joohwan Kim. Latency of 30 ms benefits first person targeting tasks more than refresh rate above 60 hz. In *SIGGRAPH Asia 2019 Technical Briefs*, pages 110–113. 2019.

[214] Raphael Spreitzer. Pin skimming: exploiting the ambient-light sensor in mobile devices. In *Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*, pages 51–62, 2014.

[215] Statista. Number of households with smart security cameras worldwide from 2016 to 2027. https://www.statista.com/forecasts/1301193/worldwide-smart-security-camera-homes, 2023. [Online; accessed 16-June-2023].

[216] Weigao Su, Daibo Liu, Taiyuan Zhang, and Hongbo Jiang. Towards device independent eavesdropping on telephone conversations with built-in accelerometer. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(4):1–29, 2021.

[217] Takeshi Sugawara, Benjamin Cyr, Sara Rampazzi, Daniel Genkin, and Kevin Fu. Light commands: laser-based audio injection attacks on voice-controllable systems. In *29th USENIX Security Symposium*, pages 2631–2648, 2020.

[218] Jan G Švec and Svante Granqvist. Tutorial and guidelines on measurement of sound pressure level in voice and speech. *Journal of Speech, Language, and Hearing Research*, 61(3):441–461, 2018.

[219] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136, 2011.

[220] Pin Shen Teh, Ning Zhang, Andrew Beng Jin Teoh, and Ke Chen. A survey on touch dynamics authentication in mobile devices. *Computers & Security*, 59:210–235, 2016.

[221] Ali Tekeoglu and Ali Saman Tosun. Investigating Security and Privacy of a Cloud-based Wireless IP camera: NetCam. In *Proceedings of the 2015 24th International Conference on Computer Communication and Networks (ICCCN)*. IEEE.

[222] The Fintech Times. Three ways to get the most out of your atm: a spotlight on lac. https://thefintechtimes.com/3-ways-to-get-the-most-out-of-your-atm-a-spotlight-on-lac/, 2023. [Online; accessed 17-April-2023].

[223] J-P Thirion. Image matching as a diffusion process: an analogy with maxwell's demons. *Medical image analysis*, 2(3):243–260, 1998.

[224] Dave Jing Tian, Adam Bates, and Kevin Butler. Defending against Malicious USB Firmware with GoodUSB. In *Proceedings of the 31st Annual Computer Security Applications Conference*, 2015.

[225] J Timmer, M Lauk, and G Deuschl. Quantitative analysis of tremor time series. *Electroencephalography and Clinical Neurophysiology/Electromyography and Motor Control*, 101(5):461–468, 1996.

[226] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[227] General Tools. *DSM403SD*, 2023. https://generaltools.com/class-1-sound-level-meter-with-excel-formatted-data-logging-sd-card.

[228] Timothy Trippel, Ofir Weisse, Wenyuan Xu, Peter Honeyman, and Kevin Fu. Walnut: Waging doubt on the integrity of mems accelerometers with acoustic injection attacks. In *2017 IEEE European symposium on security and privacy (EuroS&P)*, pages 3–18. IEEE, 2017.

[229] Yannis Tsividis. Digital signal processing in continuous time: a possibility for avoiding aliasing and reducing quantization error. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages ii–589. IEEE, 2004.

[230] Yazhou Tu, Sara Rampazzi, Bin Hao, Angel Rodriguez, Kevin Fu, and Xiali Hei. Trick or heat? manipulating critical temperature-based control systems using rectification attacks. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 2301–2315, 2019.

[231] Yu-Chih Tung and Kang G Shin. Expansion of human-phone interface by sensing structure-borne sound propagation. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, pages 277–289, 2016.

[232] Ries Uittenbogaard, Clint Sebastian, Julien Vijverberg, Bas Boom, Dariu M Gavrila, et al. Privacy protection in street-view panoramas using depth and multi-view imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10581–10590, 2019.

[233] International Telecommunication Union. Technical features of push-button telephone sets. *General Recommendations on Telephone Switching and Signalling*, 11 1988. https://www.itu.int/rec/T-REC-Q.23-198811-I/en.

[234] Wim Van Eck. Electromagnetic radiation from video display units: An eavesdropping risk? *Computers & Security*, 4(4):269–286, 1985.

[235] Matt Vasilogambros. Voting by phone is easy. but is it secure? https://gcn.com/articles/2019/07/18/vote-by-phone.aspx, 2019.

[236] Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Diffeomorphic demons: Efficient non-parametric image registration. *NeuroImage*, 45(1):S61–S72, 2009.

[237] Martin Vuagnoux and Sylvain Pasini. Compromising Electromagnetic Emanations of Wired and Wireless Keyboards. In *Proceedings of the 18th USENIX Security Symposium (USENIX Security 09)*, 2009.

[238] Payton Walker and Nitesh Saxena. Sok: assessing the threat potential of vibration-based attacks against live speech using mobile sensors. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 273–287, 2021.

[239] Chen Wang, S Abhishek Anand, Jian Liu, Payton Walker, Yingying Chen, and Nitesh Saxena. Defeating hidden audio channel attacks on voice assistants via audio-induced surface vibrations. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pages 42–56, 2019.

[240] Junjue Wang, Brandon Amos, Anupam Das, Padmanabhan Pillai, Norman Sadeh, and Mahadev Satyanarayanan. A scalable and privacy-aware iot service for live video analytics. In *Proceedings of the 8th ACM on Multimedia Systems Conference*, pages 38–49, 2017.

[241] Kai Wang, Mitev Richard, Chen Yan, Xiaoyu Ji, Sadeghi Ahmad-Reza, and Wenyuan Xu. GhostTouch: Targeted attacks on touchscreens without physical touch. In *Proceedings of the 31st USENIX Security Symposium (USENIX Security 22)*.

[242] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[243] David A Ware. *Effects of Intentional Electromagnetic Interference on Analog to Digital Converter Measurements of Sensor Outputs and General Purpose Input Output Pins*. PhD thesis, Utah State University, 2017.

[244] Zachary Weinberg, Eric Y Chen, Pavithra Ramesh Jayaraman, and Collin Jackson. I still know what you visited last summer: Leaking browsing history via user interaction and side channel attacks. In *2011 IEEE Symposium on Security and Privacy*, pages 147–161. IEEE, 2011.

[245] Wikipedia contributors. Low-voltage differential signaling. https://en.wikipedia.org/w/index.php?title=Low-voltage_differential_signaling&oldid=1021691966, 2021. [Online; accessed 12-June-2023].

[246] Chen Yan, Yan Long, Xiaoyu Ji, and Wenyuan Xu. The catcher in the field: A fieldprint based spoofing detection for text-independent speaker verification. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1215–1229, 2019.

[247] Chen Yan, Hocheol Shin, Connor Bolton, Wenyuan Xu, Yongdae Kim, and Kevin Fu. Sok: A minimalist approach to formalizing analog sensor security. In *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2020.

[248] Haotian Yang, Bin Zhou, Lixin Wang, Haifeng Xing, and Rong Zhang. A novel tri-axial mems gyroscope calibration method over a full temperature range. *Sensors*, 18(9):3004, 2018.

[249] Jianchao Yang and Thomas Huang. Image super-resolution: Historical overview and future challenges. In *Super-resolution imaging*, pages 1–34. CRC Press, 2017.

[250] Junlan Yang, Dan Schonfeld, and Magdi Mohamed. Robust video stabilization based on particle filter tracking of projected camera motion. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(7):945–954, 2009.

[251] Baki Berkay Yilmaz, Elvan Mert Ugurlu, Milos Prvulovic, and Alenka Zajic. Detecting Cellphone Camera Status at Distance by Exploiting Electromagnetic Emanations. In *2019 IEEE Military Communications Conference (MILCOM)*, 2019.

[252] Atsushi Yoshida, Hiroki Shindo, Koichi Terano, Takahiro Fukumori, and Takanobu Nishiura. Interpolation of acoustic signals in sound capture with rolling-shuttered visual camera. In *Proc. Forum Acusticum*, 2020.

[253] Berndt Zeitler, Ivan Sabourin, and Stefan Schoenwald. Wood or concrete floor?-a comparison of direct sound insulation. In *Proceedings of 40th International Congress and Exposition on Noise Control Engineering*, pages 2237–2243, 2011.

[254] Dashan Zhang, Jie Guo, Yi Jin, et al. Efficient subtle motion detection from high-speed video for sound recovery and vibration analysis using singular value decomposition-based approach. *Optical Engineering*, 56(9):094105, 2017.

[255] Dashan Zhang, Jie Guo, Xiujun Lei, and Chang'an Zhu. Note: sound recovery from video using svd-based information extraction. *Review of scientific instruments*, 87(8):086111, 2016.

[256] Li Zhang, Parth H Pathak, Muchen Wu, Yixin Zhao, and Prasant Mohapatra. Accelword: Energy efficient hotword detection through accelerometer. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, pages 301–315, 2015.

[257] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

[258] Xuping Zhang, Li Qi, Zhiqiang Tang, and Yixin Zhang. Portable true random number generator for personal encryption application based on smartphone camera. *Electronics Letters*, 50(24):1841–1843, 2014.

[259] Yang Zhang, Peng Xia, Junzhou Luo, Zhen Ling, Benyuan Liu, and Xinwen Fu. Fingerprint attack against touch-enabled devices. In *Proceedings of the second ACM workshop on Security and privacy in smartphones and mobile devices*, pages 57–68, 2012.

[260] Ruochen Zhou, Xiaoyu Ji, Chen Yan, Yi-Chao Chen, Wenyuan Xu, and Chaohao Li. DeHiREC: Detecting Hidden Voice Recorders via ADC Electromagnetic Radiation. In *Proceedings of the 2023 IEEE Symposium on Security and Privacy (SP)*.

[261] Ge Zhu, Xu-Ri Yao, Zhi-Bin Sun, Peng Qiu, Chao Wang, Guang-Jie Zhai, and Qing Zhao. A high-speed imaging method based on compressive sensing for sound extraction using a low-speed camera. *Sensors*, 18(5):1524, 2018.

[262] Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and vision computing*, 21(11):977–1000, 2003.