DESIGN AND IMPLEMENTATION OF RECONFIGURABLE ENVIRONMENTS TOWARDS AI-DRIVEN INTEGRATED SENSING AND COMMUNICATION

JINGYU TONG

PhD

The Hong Kong Polytechnic University

2025

The Hong Kong Polytechnic University

Department of Computing

Design and Implementation of Reconfigurable Environments towards AI-Driven Integrated Sensing and Communication

Jingyu Tong

A thesis submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy
January 2025

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

Signature: ____________________

Name of Student: Jingyu Tong

Abstract

Integrated Sensing and Communication (ISAC) has pivotal as a cornerstone technology for future wireless systems, perfectly positioned to integrate sensing and communication functionalities using the same frequency bands and hardware components. Concurrently, smart reconfigurable environments have become crucial in the evolution of contemporary wireless networks, which employ advanced technologies to manipulate the propagation of signals across spatial and temporal dimensions, significantly enhancing network capabilities. This dissertation explores deploying smart reconfigurable environments to augment communication and sensing performance within wireless systems. The challenges and opportunities addressed in this research focus on two distinct applications for reconfigurable environments enhanced ISAC systems: enhancing system efficiency and improving security. The primary objective of this research is to bridge the gap between theoretical advancements and practical applications by developing innovative algorithms that elevate the performance of ISAC systems, thereby ensuring their effectiveness in real-world settings.

In this thesis, we present three fundamental contributions: two aimed at enhancing efficiency and one at improving authentication security. Previous ISAC systems have often been hindered by low energy and communication efficiency. To address these issues, we propose two new methods. The first method, MetaMosaic, constructs Reconfigurable Intelligent Surfaces (RIS) by utilizing RFID tags repurposed as unit cells, significantly boosting signal strength and reducing communication latency. MetaMosaic offers a scalable and cost-effective solution for dynamically optimizing wireless environments, thus improving communication efficiency via a bespoke neural radiance field. Second, we proposed the Radio-Frequency Neural Network (RFNN), which incorporates machine learning directly within wireless sensors. By processing data at light speed close to the source, RFNN drastically reduces both time and energy consumption, minimizing the need for extensive data transmission. It maintains high accuracy across various wireless sensing tasks while adhering to the stringent power limits typical of Artificial Intelligence of Things (AIoT) environments.

Regarding security enhancement, we introduce Voltmark, which leverages ambient AC light flickering to enhance security and traceability. This approach employs natural watermarks produced by ambient lighting, detectable through standard smartphone cameras, to authenticate the spatial and temporal contexts of transactions without additional hardware. Seamlessly integrating into existing lighting infrastructures, Voltmark exemplifies the potential of ISAC with optical wireless (ISAC-OW), a pivotal component of 6G. This integration not only converts every lamp into a sensing element but also significantly enhances the security framework, boosting the robustness and reliability of the emerging network architectures in the 6G era.

This thesis explores the physical properties of environments, combines hardware and software innovations, and introduces novel algorithms drawn from principles in wireless sensing, signal processing, and machine learning. Through various experiments, these systems are implemented and evaluated, demonstrating their capability to support many real-world applications, including behavior and material sensing, signal enhancement, and security enhancement.

Publications Arising from the Thesis

Conference Proceedings

1. Jingyu Tong, Xiaopeng Zhao, Zhicheng Wang, Donghui Dai, Zhenlin An, Lei Yang, "Commercial RFIDs as Reconfigurable Intelligent Surfaces," in Proc. of *IEEE INFOCOM*, 2025.

2. Jingyu Tong, Zhenlin An, Xiaopeng Zhao, Sicong Liao, Lei Yang, "In-Sensor Machine Learning: Radio Frequency Neural Networks for Wireless Sensing," in Proc. of *ACM MobiHoc*, 2024.

3. Zheng Gong, Zhenlin An, Donghui Dai, Jingyu Tong, Shuijie Long and Lei Yang, "Enabling Cross-Medium Wireless Networks with Miniature Mechanical Antennas," in Proc. of *ACM MobiCom*, 2024.

4. Sicong Liao, Zhenlin An, Qingrui Pan, Xiaopeng Zhao, Jingyu Tong, Lei Yang, "XiTuXi: Sealing the Gaps in Cross-Technology Communication by Neural Machine Translation," in Proc. of *ACM Sensys*, 2023.

Demos and Posters

1. Sicong Liao, Jingyu Tong, Zhimin Mei, Donghui Dai, Yuanhao Feng, Qiongzheng Lin, Lei Yang, "Poster: A One-size-fits-all Solution for Cross-Technology Communication via Transformer," in Proc. of *ACM Mobisys*, 2024.

2. Jingyu Tong, Zhenlin An, Xiaopeng Zhao, Sicong Liao, Lei Yang, “Radio Frequency Neural Networks for Wireless Sensing,” in *Proc. of ACM MobiCom*, 2023. (Best Graduate Award)

3. Zheng Gong, Zhenlin An, Jingyu Tong, Donghui Dai, Lei Yang, “Demo: Constructing Smart Buildings with In-concrete Backscatter Networks,” in Proc. of *ACM MobiCom*, 2022. (Best Demo Award Runner-up)

Acknowledgments

Three years ago, when I decided to resign from Tencent to pursue a Ph.D. degree, Prof. Jie Wang recommended that I study under Dr. Lei Yang at the Hong Kong Polytechnic University. Little did I know that the journey ahead would be filled with challenges and immense personal growth. However, the support and companionship of many excellent individuals made this journey not only possible but also deeply meaningful.

Firstly, I would like to express my deepest appreciation to my supervisor, Dr. Lei Yang, for his exceptional guidance, support, and mentorship throughout my Ph.D. journey. His profound knowledge and insightful advice have been invaluable in shaping my research and academic growth. I am grateful for his patience and encouragement and for always challenging me to reach new heights. His dedication to excellence and passion for research have been a constant source of inspiration. Without his unwavering support, this dissertation would not have been possible.

I am deeply grateful to my collaborators and partners throughout my Ph.D. journey. I extend my sincere thanks to Dr. Zhenlin An and Dr. Qingrui Pan for their invaluable assistance during the early stages of my studies, which greatly deepened my understanding of the wireless and optical fields. My appreciation also goes to Mr. Xiaopeng Zhao, Mr. Donghui Dai, Mr. Zheng Gong, Mr. Sicong Liao, Mr. Zhimin Mei, Dr. Yuanhao Feng, Ms. Xuanzhi Wang, and Ms. Xueyuan Yang. Their expertise in various research topics and system design has broadened my academic horizons, and our collaborative discussions have been invaluable to my research. I am also thankful to my research group members, Mr. Zhicheng Wang, Mr. Fengrui Zhang, Mr. Shen Wang, Mr. Guosheng Wang, and all other collaborators, for their shared experiences and friendship throughout this journey. Their support and camaraderie have made this experience truly rewarding.

Finally, I wish to express my deepest gratitude to my beloved parents and my wife, Ms. Chujun Chen, for their unwavering support, encouragement, and love throughout my graduate studies, especially during the challenging times of the COVID-19 pandemic. Their constant support gave me the strength to persevere and believe in myself.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In recent years, the advent of 6G wireless technology has garnered substantial interest from both industry and academia, driven by the requirements of innovative applications demanding unprecedented levels of connectivity and intelligence within network systems. Applications such as autonomous driving, mixed reality, and the infrastructure of smart cities require communication devices or terminals to support not only high data rates but also real-time sensing and active interaction with their environments. Integrated Sensing and Communication (ISAC), a critical technology of 6G, has gained prominence in research due to its dual capability to simultaneously sense the environment and maintain seamless communication [1–5].

To effectively implement ISAC systems, extensive research has been conducted on various multi-input multi-output (MIMO) architectures, which are pivotal as they offer waveform diversity for sensing and spatial multiplexing for communications concurrently [6]. Many research efforts have focused on optimizing transmit beamforming and waveforms [2,7,8] and developing sophisticated receive beamforming strategies [9]. Despite these advancements, ISAC systems still encounter performance degradation, particularly in complex propagation environments. To address these issues, smart reconfigurable environments have been proposed [10]. This approach is regarded as a key technology for 6G wireless networks and beyond, attributed to its capability to dynamically alter the propagation environment, thereby enhancing system resilience and efficiency.

Smart reconfigurable environments leverage technologies such as Reconfigurable Intelligent Surfaces (RIS) [11–13], Digital Micromirror Devices (DMDs) [14, 15], and Active Frequency Selective Surfaces (AFSS) [16, 17], which can dynamically alter the phase, amplitude, frequency, and polarization of input wireless signals, thereby shaping the signal propagation to enhance communication and sensing functionalities. This adaptive capability is poised to overcome traditional challenges in wireless communication, such as interference and signal fading, and is envisioned as a key enabler for realizing the ISAC systems in complex, real-world settings.

1.2 Motivation

While smart reconfigurable environments have significantly advanced ISAC systems in recent years, the majority of these advancements have been demonstrated through simulations or small-scale experiments. This limitation is mainly due to the reliance on wired connections in current designs [18–20], which restricts system scalability, and the high power requirements of machine learning-based methods [21–23] in real-world experiments. For scalability, most existing designs that utilize smart reconfigurable environments to enhance communication efficiency incorporate only a limited number of reconfigurable elements. This is often due to their dependency on wired connections for power supply and state control, which severely limits their scalability and flexibility. Furthermore, the effectiveness of such systems is contingent upon the number of unit cells integrated and their distribution, underscoring the need for designing scalable and flexible smart reconfigurable systems. On the other hand, most existing ISAC systems employ machine learning methods to contend with challenges such as multipath effects, ambient interference, frequency offsets, and hardware diversity. However, the implementation of Electric Neural Networks (ENNs) within ISAC systems is fraught with challenges. The power-intensive and computation-sensitive demands of ENNs are not compatible with the cost-effective and energy-efficient requirements of wireless nodes. This challenge motivates us to develop a machine-learning system characterized by high power efficiency.

As a next-generation wireless communications technology, 6G will continue to utilize multiple frequency bands, encompassing low-band, mid-band, and high-band spectrums. Beyond millimeter waves, 6G technology will extend into terahertz (THz) frequencies and even the optical spectrum for both home and industrial applications. However, current ISAC research still focuses mainly on RF signal, and Integrated Sensing and Communication with optical wireless (ISAC-OW) remains unexplored, which can be seamlessly integrated into existing lighting and display systems to transform every lamp and screen into active elements of the 6G ISAC network. Recently, some ISAC-OW systems have been proposed for applications such as localization [24, 25] and vital signs monitoring [26]. Despite these advancements, the security issues within ISAC-OW systems remain underexplored, highlighting a critical area for further research and development.

1.3 Research Scope and Contribution

The dissertation delineates its contributions in two primary segments: the evolution of smart reconfigurable environment technology for both radio frequency and optical signals. The contributions are summarized below.

- **Scalable and flexible reconfigurable intelligent surface:** Reconfigurable Intelligent Surfaces (RISs) represent a transformative advancement in wireless system technologies, capable of modifying the amplitude or phase of incoming signals. These surfaces have been extensively utilized to enhance both sensing accuracy and communication efficiency. Traditional RIS designs often rely on wired connections, which restrict scalability and flexibility, thereby limiting the performance of current RIS-based ISAC systems. To overcome these constraints, our approach involves using a collective array of RFID tags to construct a large-scale RIS. Each RFID tag serves as an individual unit cell and is wirelessly controlled by a remote RFID reader. Specifically, we have re-engineered a dual-frequency microstrip patch antenna for each RFID-addressed unit cell to receive commands from the RFID reader and reflect RF signals at 2.4 GHz or higher. Furthermore, to circumvent the need for frequent feedback from mobile devices to measure channel variations under different settings, we propose a specialized neural radiance field for RIS. This approach estimates channel variations and swiftly pinpoints optimal RIS configuration parameters, enhancing the system's overall efficacy and responsiveness.

- **Radio frequency neural networks:** The rapid development of Electronic Neural Networks (ENNs) has significantly advanced wireless communication and sensing systems, demonstrating the successful integration of ENNs [21, 27]. However, implementing ENNs directly within sensors presents significant challenges, primarily due to their high power and computational demands, which are infeasible for cost-effective and energy-efficient sensory nodes. To address this, we propose Radio Frequency Neural Networks (RFNNs), which are designed to perform neural network computations on RF signals at the speed of light. Specifically, we introduce a full-forward-propagation method that calculates the loss value without requiring an accurate model of the RFNN using standard mathematical techniques. Furthermore, instead of relying on absolute and predetermined labels, we employ contrastive learning, which enables the RFNN to autonomously learn from a pair of similar or dissimilar samples, enhancing its ability to recognize unknown RSS labels adaptively.

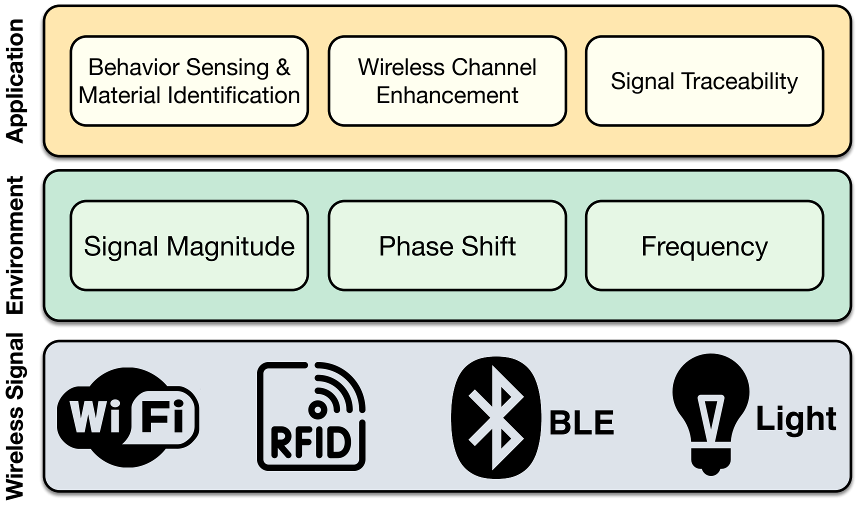


Fig. 1.1: Research Framework of the dissertation.

- **Enhancing temporal-spatial traceability through AC flickering:** In the realm of optical wireless communications, security and traceability are paramount, especially in applications involving sensitive data transfers such as mobile payments. Traditional methods that employ static optical codes, such as QR codes, are fraught with significant security vulnerabilities due to their lack of spatial-temporal traceability. Therefore, we explore the use of environmental AC flickering signals controlled by the electric network as a unique fingerprint to enhance the spatial-temporal traceability of optical wireless systems. Our studies demonstrate that voltmarks, derived from these AC flickering signals, inherently possess unique temporal and spatial characteristics of the AC cycle and the lighting fixtures. This capability allows for the precise verification of the time and location at which optical signals are captured, significantly bolstering the security and reliability of such systems.

The overview of the research scope and framework for this thesis is illustrated in Fig. 1.1. The research framework is structured along three dimensions. Initially, wireless signals, including WiFi, RFID, BLE, and visible light signals, are collected, serving as the medium for the communication and sensing systems. Subsequently, the transmission environment is reconfigured using our proposed methods, where the signal magnitude, phase shift, or frequency of the environmental signals is modified. This reconfiguration facilitates the realization of applications such as power-efficient wireless sensing, enhancement of wireless channels, and signal traceability.

1.4 Organization of the Dissertation

The rest of the dissertation is organized as follows:

- In Chapter 2, we delve into the domain of smart reconfigurable environments, focusing on their impact on wireless sensing and communication technologies. We review the evolution of wireless sensing systems, emphasizing both RF-based and optical-based technologies. Furthermore, we also give an introduction to smart reconfigurable environments, particularly the role of Reconfigurable Intelligent Surfaces (RIS) in manipulating wave properties for improved communication efficacy. We discuss the design of RIS systems and their applications in enhancing wireless communication security and signal quality.
- In Chapter 3, we propose to address the limitations imposed by the wired architecture prevalent in current RIS systems by introducing a novel approach that harnesses commercially available RFID tags to construct a cost-effective, scalable, and adaptable RIS. Traditional RIS designs typically rely on wired connections and specially customized unit cells, which inherently restrict scalability and escalate costs. Our approach employs RFID tags as dynamic unit cells, which are controlled wirelessly by a remote RFID reader. This strategy significantly simplifies the system architecture while drastically cutting costs. Additionally, our system is capable of modulating the phase of reflected electromagnetic waves, thereby enhancing communication efficiency and signal strength in diverse environments. The flexibility and scalability of our approach are substantiated through its successful

deployment in various settings, ranging from conventional office spaces to complex virtual reality environments.

- In Chapter 4, we propose to introduce an in-sensor machine learning system based on Radio Frequency Neural Networks (RFNNs). This system capitalizes on RF signals as a computational medium, enabling processing at the speed of light. Due to the unique architecture of RFNN, we implement the full-forward propagation in conjunction with contrastive learning techniques for training our model. Through this approach, we enable dynamic and efficient computation without the need for conventional computing devices.
- In Chapter 5, we propose a novel method to enhance the spatial and temporal traceability of optical communication systems. We utilize the ambient AC flickering signals as a unique spatial-temporal fingerprint. These signals are captured and extracted using Vision Transformers, which effectively encode the temporal dynamics and spatial characteristics of the flickering patterns. The extracted signals are then compared with measurements from the electric power grid to authenticate and trace the communication signals, providing a robust method for ensuring the integrity and provenance of optical communications.
- In Chapter 6, we summarize the research challenges addressed and the contributions made by this thesis, which include the development of a scalable and flexible RIS system, the implementation of Radio Frequency Neural Networks to augment computation efficiency within wireless systems, and the introduction of a spatial-temporal tracing model aimed at enhancing security for optical wireless systems. Additionally, we outline prospective future directions, emphasizing the realization of wireless digital twins for smart reconfigurable environments. We also discuss alternative architectures for physical neural networks, which could further revolutionize the integration of state-of-the-art machine learning techniques with physical network infrastructures.

Chapter 2

Literature Review

In recent years, the concept of the smart reconfigurable environment has garnered significant interest from the research community due to its remarkable capability to manipulate the propagation characteristics of signals dynamically. This technological innovation presents substantial opportunities to enhance the effectiveness and efficiency of wireless sensing and communication systems. In this chapter, we explore the evolution of wireless sensing and communication technologies, followed by a detailed examination of smart reconfigurable environments, from their conceptual design to practical applications.

2.1 Wireless Sensing and Communication

In this section, we provide a comprehensive overview of the existing achievements in the field of wireless sensing and communication. We begin by examining systems that utilize radio frequency signals, exploring the diverse applications and technological advancements that have shaped this area. Subsequently, we will delve into systems based on optical signals.

2.1.1 RF-based Wireless Sensing and Communication

Over the past several decades, RF-based wireless sensing has undergone substantial growth, propelled by its increasing integration into a variety of applications across multiple sectors. This technology has been effectively deployed in fields such as localization [23, 28–31], human activity recognition [32, 33], fruit sensing [22], material identification [34], liquid sensing [35], and imaging [36]. These diverse applications highlight the versatility and critical role of RF-based sensing in tackling complex challenges within healthcare, agriculture, industrial monitoring, and security domains. Traditionally, the processing of these signals has relied on powerful server-based GPUs [23] and increasingly on edge devices [37]. Limited by the power-intensive and computationally demanding nature of ENNs, the deployment of in-sensor or near-sensor wireless sensing systems poses significant challenges.

Beyond these varied sensing applications, RF-based sensing technologies also contribute significantly to enhancing communication efficiency. Recent advances have seen the application of Neural Radiance Fields (NeRF) [38] to represent wireless channel information, thereby improving both communication and sensing capabilities. The introduction of NeRF has spurred interest well beyond its initial applications in computer vision [39–41], with successful implementations now observed in wireless communication systems [38]. A notable innovation is NeRF^2, which adapts NeRF technology from optical to electromagnetic waves, enabling precise channel prediction. Qualcomm's WiNeRT [42] utilizes neural representation to model ray-surface interactions, which aids in the accurate estimation of RF signal propagation along transmit-receive paths, exemplifying the convergence of neural network methodologies and traditional RF sensing technologies to achieve groundbreaking improvements in wireless communication systems. However, these systems are primarily designed for static environments, necessitating further exploration into how smart environment systems can be effectively incorporated into the current neural radiance field framework to enhance their performance.

2.1.2 Optical-based Wireless Sensing and Communication

As a next-generation wireless communication technology, 6G is expected to extend its frequency bands into terahertz (THz) frequencies and even the optical spectrum for both residential and industrial applications. Consequently, optical-based wireless sensing and communication systems have attracted increasing attention in recent years [43–45]. This technology has been effectively deployed in fields such as localization [46, 47], motion capture [48, 49], and vital signs monitoring [26]. Among these, the Quick Response (QR) code stands out as one of the most widely used optical communication systems. QR codes encode information into visual patterns for transmission via light and decode it through optical sensors like cameras. They play pivotal roles in commercial applications such as mobile payments, authentication, and online ordering, where maintaining high security is crucial. Despite their extensive use, securing QR code images against unauthorized reproduction presents significant challenges, as highlighted in studies by Krombholz et al. [50], Vidas et al. [51], and Dabrowski et al. [52]. Some solutions have been explored to bolster QR code security, including the embedding of device-specific information to thwart replay attacks. One such implementation, ScreenID [53], integrates screen PWM frequency data into QR codes, allowing systems to detect and prevent the use of fraudulently reproduced codes on different screens. However, this method does not address security concerns for QR codes that are printed and physically distributed, leaving a gap in protections for these widely used formats.

To enhance the security of printed QR codes, researchers have turned to advanced cryptographic techniques such as visual cryptography. This method involves decomposing the QR code into multiple obscured layers, as demonstrated by Thamer et al. [54] and Lu et al. [55]. The reconstruction of the original code requires all layers, thereby protecting against unauthorized access. While effective, the necessity of multiple images can complicate usability in individual cases. Another approach employs Moiré patterns to obscure QR codes by manipulating spatial frequencies to make the codes readable only under specific conditions, as researched by Pan et al. [56]. However, this technique requires controllable display technologies, limiting its application to digital screens and excluding printed materials. These methods highlight ongoing efforts and challenges in enhancing QR code security, underscoring the need for solutions adaptable to both digital displays and printed formats in securing sensitive transactions and data exchanges. Therefore, enhancing QR code security without the need for dedicated hardware remains a challenging problem.

2.2 Smart Reconfigurable Environment

In this section, we review the existing work on smart reconfigurable environments. We begin by providing a brief introduction to the design of current smart reconfigurable environment systems. Subsequently, we explore several advanced applications that have emerged in this field.

2.2.1 Smart Reconfigurable Environment System Design

Smart reconfigurable environments are engineered spaces where the propagation of electromagnetic waves can be dynamically controlled to enhance communication and sensing capabilities. At the core of these environments lies the concept of Reconfigurable Intelligent Surfaces (RIS), which have recently attracted significant attention due to their unique ability to manipulate electromagnetic waves in ways that conventional materials cannot achieve. RIS are artificial planar structures composed of a large number of sub-wavelength elements, whose electromagnetic properties can be dynamically adjusted to control the amplitude, phase, and polarization of incident waves. Research has demonstrated that RIS can effectively control wave amplitude [18–20], phase [57–59], and polarization [60], offering unprecedented control over the propagation of electromagnetic signals within a smart reconfigurable environment.

This ability to tailor wavefronts enables the environment itself to become an active participant in communication systems, rather than just a passive medium. Unlike passive metasurfaces [61–63], which have fixed electromagnetic responses, RIS support real-time adaptability and optimization of wavefront properties. This dynamic control is achieved through the integration of tunable elements, such as varactors, PIN diodes, or micro-electro-mechanical systems (MEMS), that can adjust the surface's response based on external control signals. Consequently, RIS can adapt to changing environmental conditions or user requirements, making them ideal for implementing smart reconfigurable environments.

Although recent developments in one-bit RIS systems have been promising for their balance of complexity and performance [64–66], these systems typically rely on wired connections for power and control, which constrains their scalability and flexibility. Efforts to overcome these limitations include exploring RFID systems for RIS control using specialized chips in proof-of-concept unit cells [67]. However, these unit cells are predominantly controlled by a central controller through wired connections. Innovations such as MetaSight [68] and IMS [69], which employ WISPs or RFIDs to manage whole surfaces, underscore the ongoing challenges in achieving a truly wireless, scalable, and flexible RIS system.

2.2.2 Smart Reconfigurable Environment Applications

Reconfigurable Intelligent Surfaces (RIS) have been explored in numerous applications, including wireless channel security enhancement [64, 70, 71], signal quality improvement [72, 73], eavesdropping [74], and RF imaging [75]. Recently, preliminary attempts have been made to utilize RIS to enhance the accuracy and coverage of traditional wireless sensing systems [64].

As one of the earliest and most well-established backscatter communication systems [76–78], Radio-Frequency Identification (RFID) technology plays a crucial role in the development of smart reconfigurable environments. RFID systems alternate their internal impedance between reflective and non-reflective states to relay information to a reader. This fundamental communication method uniquely positions RFID as the minimal unit for a one-bit reflective RIS. Researchers have leveraged RFID technology for various applications, such as localization [31,79–81], image annotation [82], human activity sensing [83,84], and spectrum mapping [85].

By leveraging the ability to alter propagation coefficients, researchers have also begun to employ optical and RF RIS to construct Physical Neural Networks (PNNs). These systems utilize the dynamic control of electromagnetic wave properties provided by RIS to perform neural network computations directly in the physical domain. PNNs have garnered significant attention in deep learning research [86–90], with most existing work focusing on image classification tasks [86, 87, 90]. Recent studies have also demonstrated that PNNs can be applied to wireless communication coding and decoding, as well as multi-beam focusing [89]. Preliminary efforts have utilized RIS to construct convolution operations for modulation classification [91], highlighting the potential of integrating PNNs within smart reconfigurable environments to enable advanced computation capabilities.

Chapter 3

Scalable and Flexible Reconfigurable Intelligent Surface

3.1 Introduction

Reconfigurable Intelligent Surfaces (RISs) are a groundbreaking development in the realm of wireless communication, responding to the critical need for improved network efficiency and capacity in an era of increasing connectivity. These innovative surfaces, capable of actively modifying the behavior of electromagnetic waves, are reshaping wireless network operations [65, 72, 92, 93]. The purpose of RISs extends to redefining network infrastructure, paving the way for more sustainable, adaptable, and efficient solutions, particularly crucial in the rollout of 5G networks and beyond. In essence, RISs are a response to the growing demands and complexities of wireless communication, offering a novel approach to meet these challenges head-on.

Nevertheless, the realization of a scalable, minimally burdensome, and economically viable RIS faces significant challenges. First, the prevalent designs of RIS are typically tailored to specific use cases with specialized unit cells, which limits their scalability in terms of production and cost. The lack of standardization precludes these

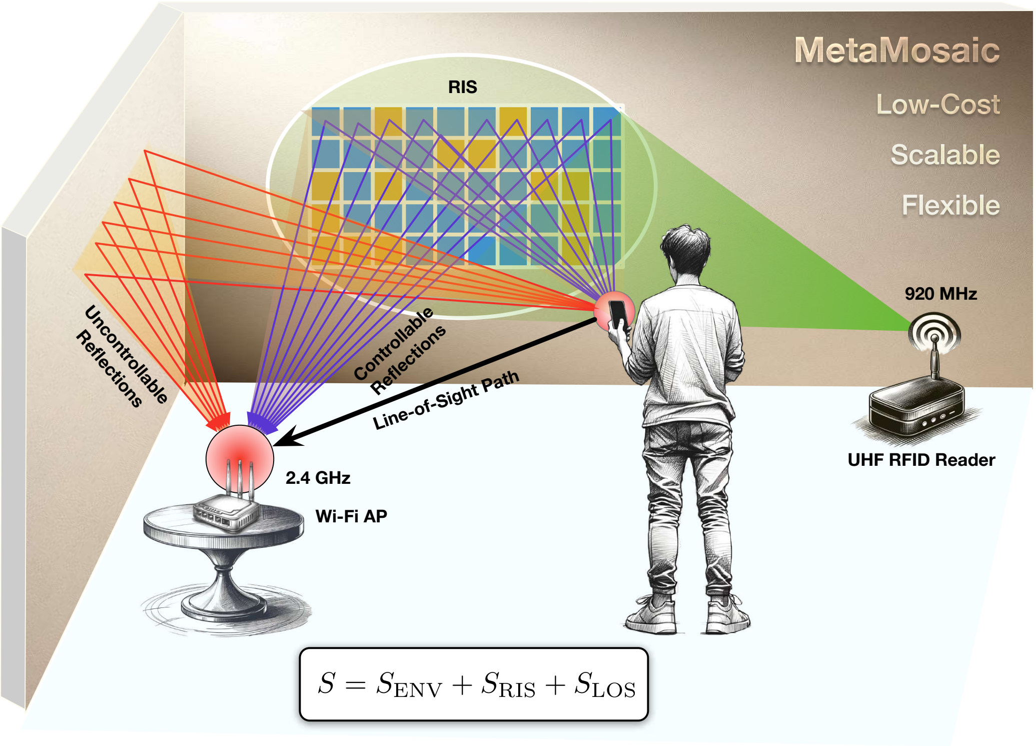


Fig. 3.1: Usage Scenario. The Wi-Fi AP or mobile device captures RF signals through direct line-of-sight propagation, uncontrollable reflections from the surrounding environment, and controlled reflections from the RIS. The goal of our system is to use UHF RFID tags to create a cost-effective, scalable, and adaptable RIS. This system is energized and managed by a remote RFID reader.

systems from reaping the benefits of mass production, consequently escalating the manufacturing costs. Secondly, the current design paradigm of RIS often involves wired connections between the unit cells and the centralized controller. This wired framework imposes a constraint on scalability, particularly in terms of the number of unit cells that can be feasibly integrated. Lastly, the operation of RIS systems necessitates a power source to sustain the operational parameters of its unit cells. However, the maintenance expenses for such power requirements become prohibitively high, especially in the context of an extra-large RIS. This creates a substantial economic burden, further complicating the spread of RIS systems.

As one of the earliest and most well-established backscatter communication systems, RFIDs alternate their internal impedance between two states — reflective and non-

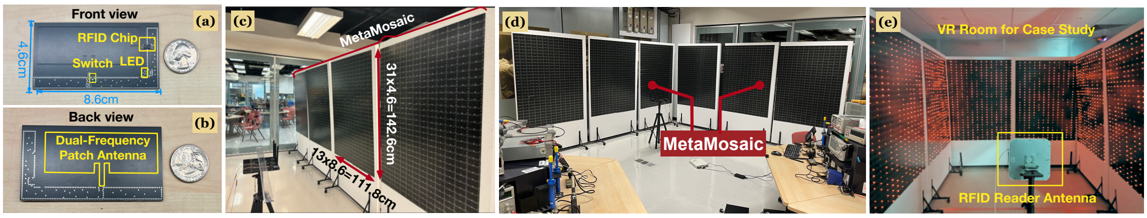


Fig. 3.2: Prototypes of MetaMosaic. (a) and (b) the unit cells, which are constructed from RFID tags, seen from the front and back, respectively. (c) and (d) illustrate the surfaces of MetaMosaic, each arranged with a grid of 13 × 31 unit cells. (e) displays a practical application within a VR room, an enclosed area encircled by six MetaMosaic surfaces. The unit cells lit by LEDs have been designated to flip the phase of the reflected EM waves by 180°, while the remaining unit cells reflect unaltered waves.

reflective — to relay information to a reader. This fundamental communication method uniquely positions RFID as the minimal unit for a one-bit reflective RIS. When “on”, the RIS reflects the signal; otherwise, the signal is absorbed. Inspired by this insight, we can employ a collective array of RFID tags to construct a large-scale RIS, on which each RFID tag, functioning as an individual unit cell, is wirelessly controlled by a remote RFID reader.

We term our innovative Reconfigurable Intelligent Surface as MetaMosaic, termed after the intricate art of mosaic. MetaMosaic is an assembly of numerous individual RFID tags, collectively forming an expansive and unified surface. MetaMosaic stands out by innovatively converting commercially available passive RFID tags into dynamic unit cells to forge a one-bit, phase-modulated reflective RIS. As depicted in Fig. 3.1, the MetaMosaic setup is strategically mounted on a wall. It operates under the control of a distant RFID reader, functioning at a frequency of 920MHz. Concurrently, MetaMosaic possesses the capability to reflect incoming electromagnetic waves at 2.4GHz, either maintaining their phase or introducing a 180-degree phase shift. This selective phase modulation enables the controllable reflections to synchronize in phase with the aggregate signal derived from various paths, enhancing overall signal coherence and efficacy. This approach not only redefines the application of RFID technology but also paves the way for more adaptable and efficient wireless communication environments.

MetaMosaic offers a new direction in RIS design with notable advantages. (1) *Cost-Effectiveness*: It uses mass-produced UHF RFID chips to reduce costs to under 50 cents per unit. (2) *Flexibility*: The wireless nature allows for versatile and straightforward deployment. (3) *Scalability*: Its wireless framework facilitates seamless scale-up, embracing expansive and intricate environments without the limitations of wiring. (4) *Efficiency*: Units operate with Gen2 RFID tags for simultaneous control, improving responsiveness. (5) *Sustainability*: Battery-free design ensures lower operational and maintenance costs, making it an eco-friendly solution.

However, adapting MetaMosaic for practical, real-world applications presents two primary challenges:

• First, UHF RFID systems generally function at around 920MHz, restricted to a narrow bandwidth of 20MHz, which results in less efficient reflection at the target 2.4GHz band. To bridge this gap, we re-engineered a dual-frequency microstrip patch antenna for each RFID-addressed unit cell to achieve two main goals. The first is to optimize these antennas to efficiently harvest energy and receive signals from a 920MHz RFID reader's continuous wave. The second is to adapt these antennas to effectively reflect RF signals at 2.4GHz or above, thereby aiding corresponding frequency wireless communication.

• Second, a crucial feature of a RIS is its ability to reconfigure the reflective coefficients of its unit cells, thus orchestrating constructive interference of signals at the receiver's location. In a one-bit RIS that includes M elements, the number of possible states for the system can reach 2^M, giving rise to a vast array of configuration possibilities. This complexity has led to the reliance on heuristic algorithms in earlier systems, which incrementally adjust configurations using immediate feedback from mobile devices, such as variations in RSS [65]. This hard-landing approach, while effective, can be time and energy-intensive. To mitigate this issue, we propose the creation of a specialized neural radiance field for RIS, designated as RIS-NeRF^2. This framework would undergo pre-training through a finite collection of RF measurements. It is designed to swiftly pinpoint the optimal RIS configuration parameters, streamlining the reconfiguration protocol, and thereby elevating the system's efficacy and operational performance.

Summary of Results: We developed six MetaMosaic surfaces by using a total of 2,418 unit cells, as shown in Fig. 3.2. Our comprehensive evaluation in ten diverse settings, ranging from real-world office spaces to laboratory rooms, yielded the following insights:

- The RIS-NeRF^2 model consistently achieved an average median deviation of just 0.74% in signal prediction, exemplifying its high accuracy. This precision forms the cornerstone for employing a soft-landing algorithm for optimal reconfiguration, substantially reducing the reconfiguration time to merely one-tenth of that required by traditional hard-landing methods.
- Across the ten different scenes, MetaMosaic realized an average median gain of 18.82dB, peaking at a median gain of 25dB. This performance notably surpasses existing state-of-the-art large-scale RIS by an additional 8.51dB. The system's adaptability was also demonstrated, proving its effectiveness as a general-purpose RIS solution and scalability in different settings and layouts.
- In virtual reality (VR)-based field study, where an enclosed space was surrounded by MetaMosaic surfaces, the system facilitated a data throughput of 14.86Mbps and 4.63ms, showcasing its practical application and effectiveness in a controlled, immersive environment.

Contribution. MetaMosaic stands out as the first instance of a battery-free, low-cost, and RFID-addressed RIS. This design brings two advancements to the domain of intelligent communications. First, the application of repurposed RFID technology enables each unit cell to be wirelessly controlled by an RFID reader. Second, the

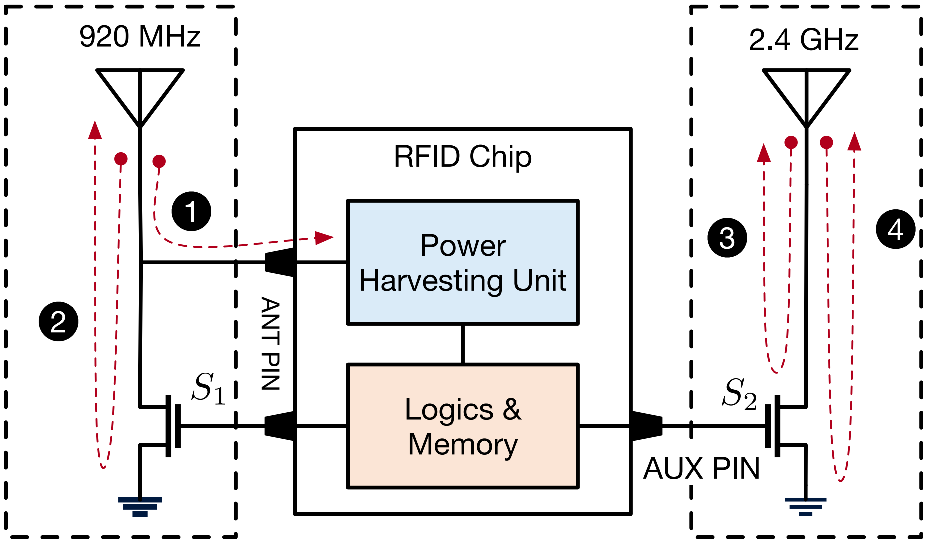


Fig. 3.3: Design of the unit cell. The unit cell of MetaMosaic is composed of a commercial RFID chip and two antennas operating at 920MHz and 2.4GHz, respectively.

implementation of bespoke neural radiance fields marks the inception of the first soft-landing reconfiguration algorithm, which is a leap forward in the field.

3.2 Repurposing RFID into Unit Cell

This section begins with an introduction to the fundamentals of RFID technology and then progresses to explain how an RFID tag can be converted into a unit cell.

3.2.1 RFID Background

In Ultra-High-Frequency (UHF) RFID systems, a device known as the reader emits a strong RF signal. Nearby RFID tags detect and respond to this emitted signal. Each RFID tag comprises two primary components:

- **Microchip**: The heart of the tag, storing data like the unique `EPC` and possibly other information. Its memory can be read-only or writable, and it controls tag operations.

- **Antenna**: Specifically designed for UHF frequency operation, the antenna draws energy from the reader's signal to power the microchip, negating the need for a battery.

An RFID tag represents the simplest instance of a one-bit RIS, functioning as a solitary unit. Benefiting from the economies of scale, with around 40 billion UHF tags reportedly sold in 2023 [94], the per-unit cost of these tags has significantly reduced, currently standing at as low as 5 cents. This widespread adoption and mass production have contributed to making RFID tags more affordable.

3.2.2 From Tag To Unit Cell

Microchips and antennas are typically produced by separate manufacturers in the market. These components are then combined by downstream suppliers for various applications. Following this practice, the unit cell of MetaMosaic incorporates two antennas into existing commercial microchips, as shown in Fig. 3.3.

The first antenna, situated on the left, is tuned to 920MHz. It interfaces with an impedance switch, labeled S_1, alongside a power harvesting module. This switch is regulated by the RFID's internal logic and toggles between two modes:

❶ **Absorptive State**. When the switch S_1 is off, it renders an open circuit. With the power harvesting unit's input impedance tailored to the antenna's, it allows the inbound signal to penetrate the circuit, optimizing power absorption. In this state, the cell primarily functions as an energy absorber.

❷ **Reflective State**. When the switch S_1 is on, it creates a closed circuit, effectively grounding the antenna. This disrupts impedance matching, resulting in the complete reflection of the incident RF signal.

By default, the antenna's principal role is to gather energy or accept instructions from an external reader.

The second antenna, situated on the right, operates at 2.4GHz. It is linked directly to ground through an additional impedance switch S_2. This switch offers a separate backscattering mechanism. Unlike the 920MHz configuration, the RF signals in this subsystem do not enter the chip but are reflected constantly. The corresponding reflection coefficient is described by:

$$\Gamma = \frac{Z_L - Z_A}{Z_L + Z_A} \tag{3.1}$$

where Z_L represents the impedance when the switch is engaged, and Z_A denotes the antenna's impedance at 2.4GHz. The logic system governs the switch S_2, resulting in two distinct configurations:

❸ **Open-Circuit Configuration:** When the switch S_2 off, the switch circuit's impedance, Z_L, effectively becomes infinite. As a result, the reflection coefficient reaches a maximum value of 1, i.e., $\Gamma = \frac{\infty - Z_A}{\infty + Z_A} = 1$. This causes the incident RF signals to be completely reflected by the 2.4GHz antenna, with no alteration in phase.

❹ **Short-Circuit Configuration:** Conversely, when the switch S_2 is closed, the switch circuit's impedance, Z_L, to approximately zero, indicating a short-circuit condition. The reflection coefficient then becomes $\Gamma = \frac{0 - Z_A}{0 + Z_A} = -1 = e^{\mathbf{J}\pi}$. Under these circumstances, the incident wave S is fully reflected with a phase shift of π, as $Se^{\mathbf{J}\pi}$, implying a 180-degree phase change.

Therefore, the reflective coefficient of a unit cell Γ at 2.4GHz is either 1 or -1. Our unit cell effectively functions as a one-bit RIS, capable of phase modulation on the incoming signals.

3.2.3 Controlling Reflective Coefficients

The logic circuits within standard RFID chips are primarily designed to adhere strictly to the EPCglobal Gen 2 air protocol [95]. This raises the question: how can we utilize a commercial RFID chip to control the reflection coefficient at 2.4GHz? With the increasing demand for more versatile applications, modern RFID chips are being equipped with additional pins that can output the tag's internal state. For instance, the EM4325 chip from EM Microelectronic [96] includes an auxiliary (AUX) pin capable of indicating the tag's status as either selected or not selected according to the Gen 2 protocol stipulations.

Before initiating the inventory process, an RFID reader can employ a `Select` command to specify which tags should participate in the communication. The AUX pin outputs a high voltage signal when the tag is in the selected state and a low voltage when it is not selected. By connecting this AUX pin to the switch S_2 in our design, as illustrated in Fig. 3.3, we can effectively control the reflection coefficient of the unit cell. Essentially, changing the reflection coefficient becomes a matter of tag selection: when the tag is selected, switch S_2 is activated (opened), altering the reflection properties; when not selected, the switch remains inactive (closed). This approach allows us to manipulate the reflection coefficients of the unit cells using simple `Select` commands, eliminating the need for complex circuitry or additional power sources.

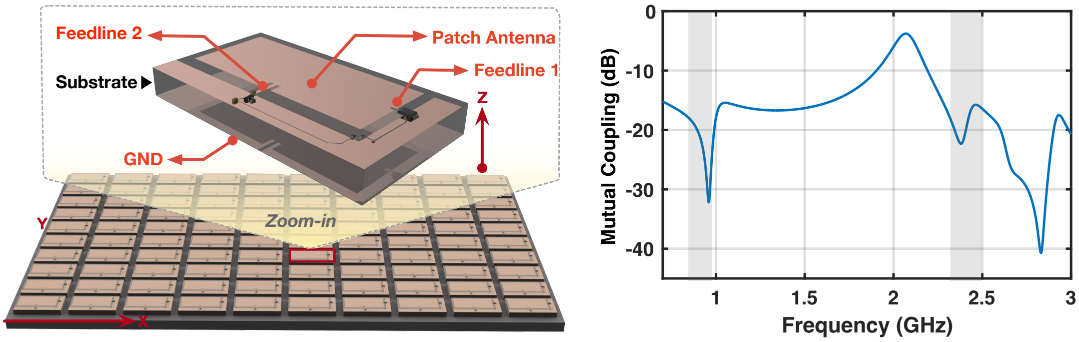
Traditional RIS designs, especially in large-scale systems with potentially thousands of unit cells, face substantial overhead due to the need for sequential adjustments of each element, a process that can take seconds. This is particularly problematic in mobile communications, where frequent changes in reflective states are required. In contrast, the RFID-addressed MetaMosaic operates on a federated model, with independently functioning units under a unified framework. This setup allows for quick alterations in the reflective coefficients of selected tags through a single, concise `Select` command, significantly reducing the overhead to microseconds and thereby boosting the system's efficiency and responsiveness in practical communication scenarios.

3.2.4 Dual-Frequency Patch Antenna

While it might appear that incorporating two antennas would necessitate additional space, in reality, the narrow-band microstrip patch antenna is innately capable of resonating and emitting at two distinct frequencies, f_1 and f_2, along its two edges, respectively. As Fig. 3.4(a) shows, the resonant frequencies of the antenna are highly dependent on its height H and width W. Formally, $f_1 = \frac{c}{2H\sqrt{\epsilon_e}}$ and $f_2 = \frac{c}{2W\sqrt{\epsilon_e}}$, where c is the speed of light and ϵ_e is the effective dielectric constant. Notably, each frequency is characterized by a different polarization direction, orthogonal to each other. By attaching two feedlines to distinct edges of the patch antenna, it is possible to utilize both frequencies simultaneously without mutual interference. In our configuration, dimensions of $W = 8.6$cm and $H = 4.6$cm have been selected to facilitate operation at frequencies of 920MHz and 2.4GHz, respectively. Fig. 3.4(b) illustrates the mutual coupling (S_{21}) between two feed ports. As it can be seen, at the two operational bands, the coupling coefficient remains below -15dB, indicating a mere 0.3% energy transfer from one port to the other. Consequently, this design permits the modulation of the phase of the 2.4GHz signal without affecting the 920MHz EM waves' state. This design strategy enables the antenna to maintain a size that is on par with conventional commercial RFID tags, despite its dual-frequency functionality.

3.3 RIS Reconfiguration

In this section, we take advantage of neural radiance fields to optimize the reconfiguration algorithm.



(a) Patch Antenna

(b) S_{21}

Fig. 3.4: Dual-frequency patch antenna. (1) we leverage the dual polarization of the patch antenna at the two edges to achieve the dual-frequency antenna. (2) shows the mutual coupling coefficient S_{21} of two feed ports.

3.3.1 Hard-landing Reconfiguration

The primary objective of RIS is to modify surface reflections in a manner that allows controllable reflections to constructively align with the LoS propagation at the receiver, thereby enhancing the signal strength. This process necessitates the mobile device to constantly provide feedback on signal strength each time there is a shift in the RIS configuration. This method of collaborating with the mobile device to evaluate the effectiveness of a RIS configuration is termed “hard landing", as shown in Fig. 3.5. Notably, the hard landing is laborious and time-intensive. This is not only due to the nature of the feedback acquisition but also because of a staggering 2^M potential configurations for a M-cell RIS. Traditional RIS systems often lean on heuristic algorithms for the optimization of reconfiguration [65,97], which necessitates multiple iterations to ascertain the optimal configuration.

3.3.2 Scene Representation

The necessity for a "hard landing" in reconfiguration arises from the inability to accurately predict channel behavior. To tackle this issue, it’s crucial to develop a model for the current scene that enables a more refined, "soft-landing" reconfiguration ap-

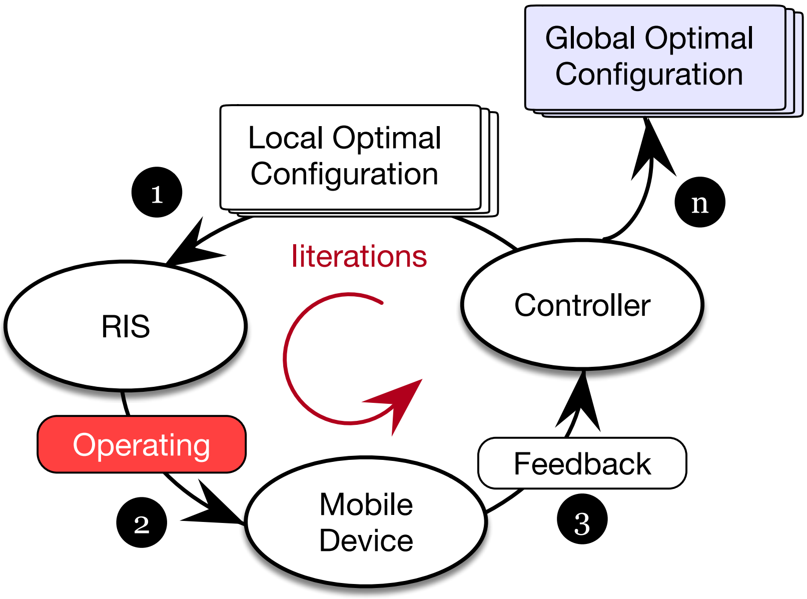


Fig. 3.5: Hard-landing based Configuration strategy. Past RIS systems adopt the heuristic search algorithm to identify the optimal configuration parameters after multiple iterations. In each iteration, when a new configuration is assumed, the controller necessitates the RX device to constantly provide feedback on the signal strength.

proach, which can effectively simulate the performance of a proposed configuration. This concept is inspired by the neural radiance fields (NeRF) [39], a technique designed for the detailed modeling of complex 3D scenes. Extending NeRF's principles, NeRF^2 broadened its scope from the optical realm to encompass electromagnetic aspects, particularly for simulating radio-frequency radiance fields [38]. In line with these developments, we introduce RIS-NeRF^2 to represent the scene where the RIS operates. This model breaks down the scene into a myriad of small, continuous voxels, each characterized by a 3D coordinate and two critical properties: attenuation and radiance properties.

- **Attenuation Property**: Each voxel V_i full of air or other medium attenuates the propagation of the RF signals that travel through the voxel. The attenuation property of voxel V_i is quantified as a complex number denoted by $h(V_i)$:

$$h(V_i) = \Delta a(V_i) e^{\mathbf{J}\Delta\theta(V_i)} \tag{3.2}$$

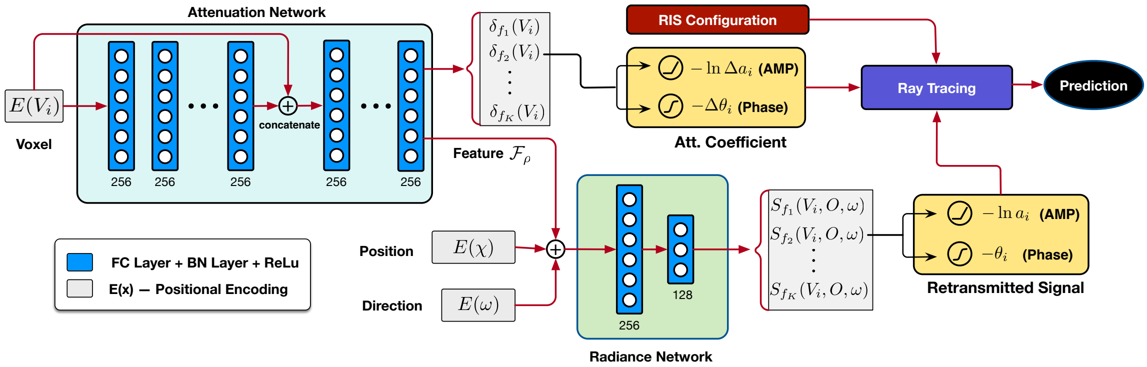


Fig. 3.6: Network Architecture of RIS-NeRF^2

where $\Delta a(V_i)$, normalized between 0 and 1, indicates amplitude reduction, while $\Delta\theta(V_i)$, ranging from 0 to 2π, represents the phase shift. To facilitate computation, this coefficient is usually converted into a negative logarithmic form denoted by $\delta(V_i)$:

$$\begin{aligned}\delta(V_i) &= -\ln(h(V_i)) = -\ln(\Delta a(V_i)e^{\mathbf{J}\Delta\theta(V_i)}) \\ &= -\ln \Delta a(V_i) - \mathbf{J}\Delta\theta(V_i)\end{aligned} \tag{3.3}$$

The use of negative logarithms allows for easier confinement of these ranges through the application of ReLU and Sigmoid functions in the neural network layers.

- **Radiance Property**: Guided by the Huygens-Fresnel principle, each voxel within a scene becomes an auxiliary source of radiance when it encounters incoming signals. We envision this scenario as each voxel V_i re-radiating RF signals. The process of this re-emission is encapsulated as follows:

$$S(P_{\text{TX}}, V_i, \omega) = a(V_i)e^{\mathbf{J}\theta(V_i)} \tag{3.4}$$

where P_{TX} signifies the TX's location, and ω represents the direction in which the signal is re-emitted. The terms $a(V_i)$ and $\theta(V_i)$ denote the initial amplitude and phase of the signal emitted by voxel V_i, respectively. The emission direction ω is defined as a two-dimensional variable, including the azimuthal and elevation angles. This mathematical representation implies that the RF signals are re-emitted by the voxel V_i toward the direction ω, considering the TX's position at P_{TX}. By adopting this model, we can abstract the intricate and complex interplays and radiative properties of voxels within the scene. Such an abstraction is pivotal for an in-depth and accurate analysis of RF signal propagation patterns and behaviors, enhancing our understanding of signal dynamics within various environmental contexts.

3.3.3 Neural Radiance Fields

How could we estimate the two properties of each voxel? Following the practice of NeRF^2, we also use two MLPs (called RIS-NeRF^2) to model the attenuation and radiance properties of a voxel, as shown in Fig. 3.6. Specifically, one MLP called *attenuation network* is to model the attenuation property of a voxel, while the other one called *radiance network* is to model the radiance property.

• **Attenuation Network**. This network, symbolized as $\mathbf{F}_\delta$, takes the 3D coordinates of a voxel $V_i(x_i, y_i, z_i)$ and computes both the attenuation coefficient $\delta(V_i)$ and a feature vector $\mathcal{F}(V_i)$. The formulation is as follows:

$$\mathbf{F}_\delta : (V_i) \rightarrow (\delta(V_i), \mathcal{F}(V_i)) \tag{3.5}$$

The output includes the attenuation coefficient $\delta(V_i)$, a complex number, and a 256-dimensional feature vector $\mathcal{F}(V_i)$. The real part of $\delta(V_i)$ is refined using a ReLu activation function, ensuring that $-\ln(\Delta a(V_i)) \geqslant 0$. The imaginary part is modified with a $2\pi \times$sigmoid function to limit the phase shift within 0 to 2π. The feature vector $\mathcal{F}(V_i)$, processed through eight fully connected layers with ReLU activations and 256 channels, then serves as input to the radiance subnetwork. Notably, the attenuation characteristic is dictated by the voxel's intrinsic density and the structural makeup of the scene, making it independent of the incoming RF signals.

• **Radiance Network**. Represented by $\mathbf{F}_\Upsilon$, this subnetwork predicts the properties of the RF signal re-emitted by a voxel. Its input parameters include the voxel's attenuation feature vector $\mathcal{F}(V_i)$, the observation direction ω, and the TX's position

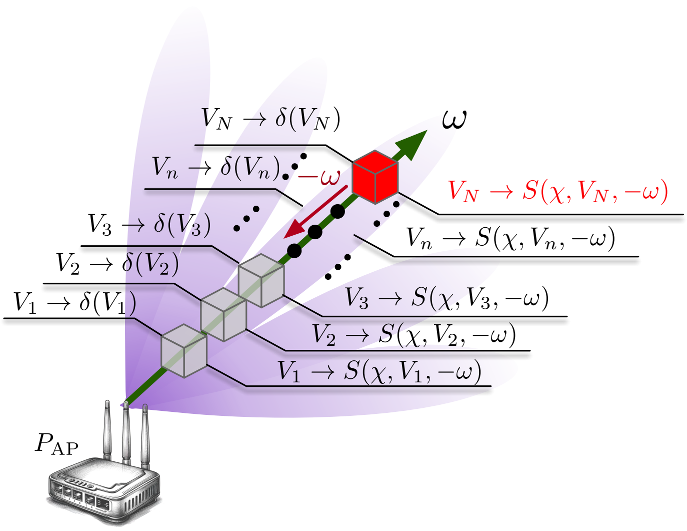

$V_N \to \delta(V_N)$
ω
$V_n \to \delta(V_n)$
$-\omega$
$V_N \to S(\chi, V_N, -\omega)$
$V_3 \to \delta(V_3)$
$V_n \to S(\chi, V_n, -\omega)$
$V_2 \to \delta(V_2)$
$V_1 \to \delta(V_1)$
$V_3 \to S(\chi, V_3, -\omega)$
$V_2 \to S(\chi, V_2, -\omega)$
$V_1 \to S(\chi, V_1, -\omega)$
P_{AP}

Fig. 3.7: Ray Tracing

P_{TX}. The subnetwork's role is described by:

$$\mathbf{F}_{\Upsilon} : (P_{\text{TX}}, \mathcal{F}(V_i), \omega) \to a(V_i)e^{\mathbf{J}\theta(V_i)} \tag{3.6}$$

The radiance subnetwork, comprising two fully connected layers with ReLU activations (256 channels in the first layer and 128 in the second), outputs the direction-dependent re-emitted RF signal $a(V_i)e^{\mathbf{J}\theta(V_i)}$. The amplitude $a(V_i)$ and phase $\theta(V_i)$ components of this signal are also fine-tuned using ReLu and Sigmoid activation functions, respectively.

3.3.4 Ray Tracing

To utilize RIS-NeRF^2 for forecasting the RF signal received at the AP, we employ a divide-and-conquer strategy in our ray tracing methodology. As illustrated in Fig. 3.7, the RF signal reception at the AP is systematically deconstructed. In particular, ❶ the total RF signal is a synthesis of contributions from all possible directions → ❷ the RF signal from any given direction is a cumulative result of signals emanating from all voxels along that directional path → ❸ the signal re-emitted from each individual voxel on the path. By tracing and incrementally integrating these individual voxel signals, we can accurately estimate the final signal that the AP receives.

• **Tracing from a Single Voxel**. As Fig. 3.7 shows, our objective is to ascertain the contribution of voxel V_N to the RF signal at the AP. We establish a ray trajectory from the AP targeting V_N. This ray, oriented along ω, intersects a sequence of $N-1$ voxels, denoted as $V_1, V_2, \ldots, V_{N-1}$, with V_1 closest to the AP and V_{N-1} at the furthest extent. The signal re-emitted by voxel V_N is represented as $S(\chi, V_N, -\omega)$, which is derived from the radiance network F_Υ. This signal represents the RF output from V_N directed oppositely to ω. This signal undergoes attenuation due to the presence of voxels $\{V_{N-1}, V_{N-2}, \ldots, V_1\}$ situated between the AP and V_N. Therefore, the final RF signal received by the AP, contributed by voxel V_N, is articulated as follows:

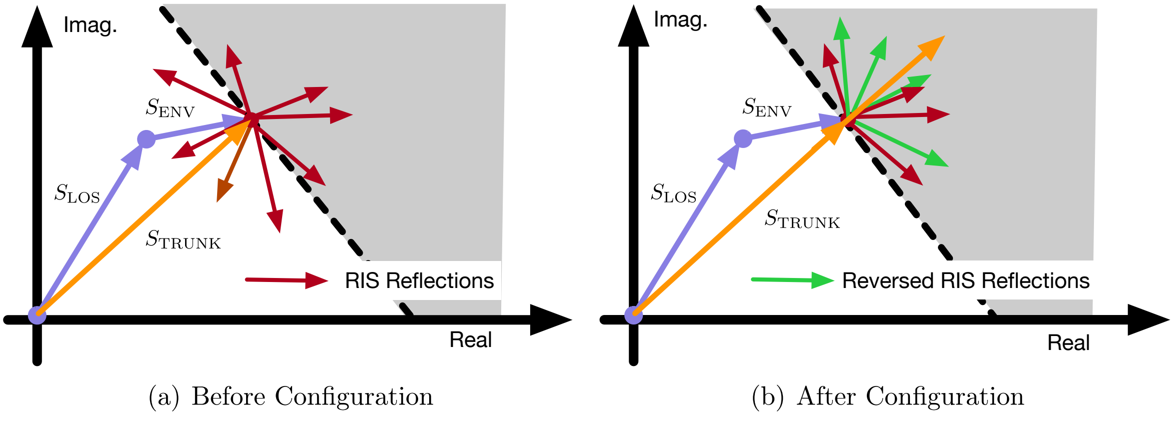
$$\begin{aligned} S(\chi, V_N) &= \left(\prod_{m=1}^{N-1} h(V_m)\right) \cdot S(\chi, V_N, -\omega) \\ &= \exp\left(-\sum_{m=1}^{N-1} \delta(V_m)\right) \cdot S(\chi, V_N, -\omega) \end{aligned} \tag{3.7}$$

• **Tracing from a Direction**. The process of signal tracing can be expanded from a single voxel to encompass the entire directional path by aggregating the RF signals transmitted from each voxel along that path. Consequently, the signal received from a specific direction ω can be reformulated as:

$$\begin{aligned} S(\chi, \omega) &= \sum_{n=1}^{N} S(\chi, V_N) \\ &= \sum_{n=1}^{N} \left(\exp\left(-\sum_{m=1}^{N-1} \delta(V_m)\right) \cdot S(\chi, V_n, -\omega)\right) \end{aligned} \tag{3.8}$$

where V_N at the furthest extent of the scene. Every voxel on the path functions simultaneously as a secondary source of RF radiation and as an obstacle that impedes the passage of RF signals traversing it.

• **Tracing from All Directions**. The AP is capable of capturing RF signals originating from all voxels within the scene. Nevertheless, computing the signal contributions



(a) Before Configuration

(b) After Configuration

Fig. 3.8: Configuration Strategy. The orientations of reflections from the unit cells of the RIS are maintained or reversed to ensure alignment with the primary trunk signal

from each voxel across the whole scene is computationally demanding. To manage this, we typically trace a discrete subset of potential directions to estimate the final RF signal received by the AP as follows:

$$\begin{aligned} S(\chi) &= \sum_{\omega\in\Omega} S(\chi,\omega) = \sum_{\omega\in\Omega}\sum_{n=1}^{N} S(\chi, V_n) \\ &= \sum_{\omega\in\Omega}\sum_{n=1}^{N}\left(\exp\left(-\sum_{m=1}^{n-1}\delta(V_m)\right)\cdot S(\chi, V_n, -\omega)\right) \end{aligned} \tag{3.9}$$

where Ω represents the set of all potential directions from which signals can be received. By summing over these directions, the AP effectively compiles a comprehensive picture of the RF landscape, capturing signals from every voxel along each path. As aforementioned, the Wi-Fi signals contain multiple orthogonal subcarriers. Thus, the ray tracing must be performed at these subcarriers respectively.

3.3.5 Training

RIS-NeRF^2 is developed to create a model of the scene. Once the dual MLPs are effectively trained, forecasting the RF signals becomes feasible. For this purpose, it is necessary to compile a set of CSI data, gathered by positioning the mobile device

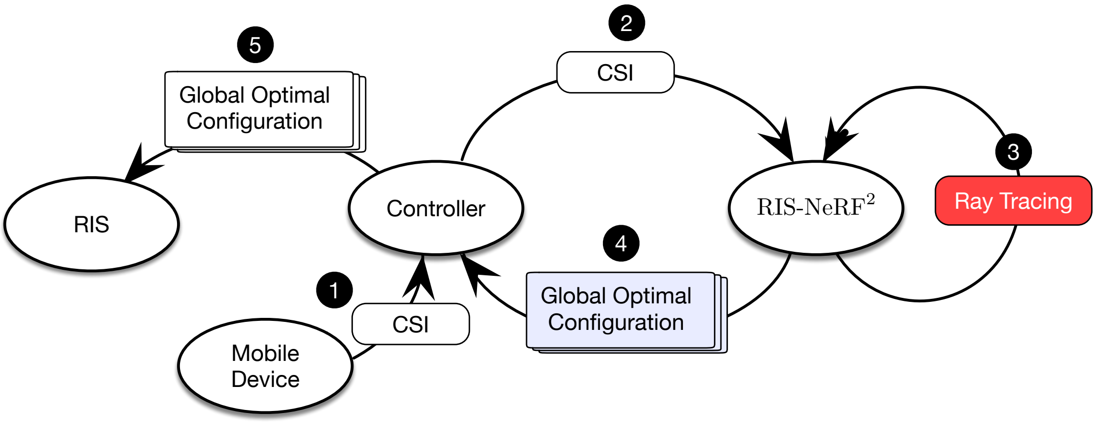


Fig. 3.9: Soft-landing based Configuration. The efficacy of a proposed configuration can be evaluated by the ray tracing algorithm with RIS-NeRF^2. This approach allows for the assessment of the configuration's impact without actually implementing it on the RIS or necessitating feedback from the TX.

at $\mathcal{N}$ distinct locations within the scene. The training objective is to minimize the following loss function:

$$\mathcal{L} = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \left\| S(\chi^i) - \chi^i \right\|^2 \tag{3.10}$$

The χ^i serves a dual purpose. Firstly, it is used as a location indicator of the mobile device for the signal prediction $S(\chi^i)$. Secondly, it acts as the true RF signal received by the AP. This training process is integral to aligning the model's predictions with the actual signal behaviors observed in the environment. As a result, the attenuation and radiance property of a voxel is desired to be well learned.

3.3.6 Soft-landing Reconfiguration Algorithm

Once RIS-NeRF^2 has been adequately trained, it becomes a powerful tool for identifying the ideal configurations of our MetaMosaic. In its operational phase, the AP initiates a CSI measurement χ from the device. With the capabilities of RIS-NeRF^2, we can predict the final RF signal, $S_{\text{AP}}(\chi)$, that the AP receives. This signal is composed of three distinct components:

$$S_{\text{AP}}(\chi) = S_{\text{LOS}}(\chi) + S_{\text{ENV}}(\chi) + S_{\text{RIS}}(\chi) = S_{\text{TRUNK}}(\chi) + S_{\text{RIS}}(\chi)$$

where S_{LOS} denotes the signal that traverses the LoS path, S_{ENV} represents signals from uncontrollable environmental reflections, S_{TRUNK} is the sum of the LoS signal, and the uncontrollable reflection signals, and S_{RIS} originates from all unit cells of the RIS. Given that the positions of the RIS unit cells are known, we can further trace S_{RIS} as:

$$S_{\text{RIS}}(\chi) = \sum_{V_i \in \mathcal{V}_{\text{RIS}}} S(\chi, V_i) \tag{3.11}$$

with $\mathcal{V}_{\text{RIS}}$ being the collection of all RIS unit cells and $S(\chi, V_i)$ calculated using Eqn. 3.7. Subsequently, we can deduce:

$$S_{\text{TRUNK}}(\chi) = S_{\text{LOS}}(\chi) + S_{\text{ENV}}(\chi) = S_{\text{AP}}(\chi) - S_{\text{RIS}}(\chi) \tag{3.12}$$

The unalterable trunk signal is assumed to be the primary contributor to the final signal. It is either strengthened or attenuated by the RIS unit cells. As shown in Fig. 3.8(a), RIS reflections have the potential to either boost or reduce the trunk signal, which depends on their phase relationship. Therefore, our configuration strategy is quite straightforward: *if the reflection from a unit cell $S(\chi, V_i)$ aligns in phase with the primary trunk signal, its reflective coefficient is maintained, i.e., $\Gamma(V_i) = 1$. If not, the coefficient is inverted, i.e., $\Gamma(V_i) = -1$.* The outcomes of applying this configuration strategy are illustrated in Fig. 3.8(b).

3.4 Implementation

The MetaMosaic architecture includes unit cells, surfaces, an RIS controller, and RIS-NeRF^2.

(1) Unit Cells: At its core is the EM4325 RFID chip [96], paired with a dual-frequency patch antenna. The reflection coefficient is managed using the BF1118 MOSFET [98]. Fig. 3.2(a) and (b) show the completed unit cells. The final PCB is produced using FR4 material with a thickness of 1.6mm, offering a cost-effective solution. Each cell is composed of an RFID chip (\$0.26), a MOSFET (\$0.19), two resistors (\$0.00051 each), two capacitors (\$0.0064 each), and two inductors (\$0.00019 each), totaling \$0.4759 per unit. With large-scale production, these costs are expected to decrease to a tenth of the current amounts. A total of 2,418 unit cells are prototyped for debugging and testing. We anticipate that the cost could be decreased to a tenth when the production volume exceeds 100k units. The compact dimensions of 8.6×4.6 cm^2 make these unit cells ideal for deployment as tiles on common surfaces such as walls and ceilings, due to their small footprint.

(2) Surfaces. To facilitate the evaluation in different scenes, we constructed six RISs by mounting unit cells in a grid pattern on movable planes, as illustrated in Fig. 3.2(c) and (d). Each surface comprises 13×31 unit cells and measures $1.2 \times 2\text{m}^2$. To visually indicate active unit cells, an ultra-low-power LED is integrated between the switch and the reflector. As shown in Fig. 3.2(e), the illuminated unit cells, distinguishable by their activated LEDs, represent those selected for engaging the phase-reversed reflections.

(3) Controller: The control mechanism in MetaMosaic essentially employs a standard ImpinJ R410 RFID reader [99] equipped with a 9dB directional antenna. The reader is connected through cables and manipulated through LLRP [100]. To activate the EM4325 chip's `select` state output through the AUX pin, it is necessary to modify two internal registers of the RFID chips, namely the I/O control word and the power management word, through the `Write` commands. Once these adjustments are made, all cells will consistently hook the `Select` events until they are powered down or unselected.

(4) RIS-NeRF^2. Our setup mirrors the configuration used in NeRF^2 [38], with

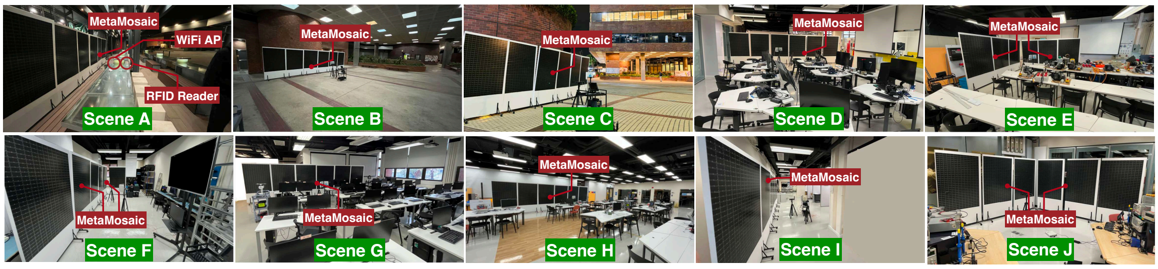


Fig. 3.10: Illustration of scenes.(a) an outdoor passage, characterized by its open-air setting and potential for varied environmental factors. (b) a semi-indoor hall, notable for its lack of enclosing walls, which influences signal propagation. (c) an open area, offering a broad and unobstructed space for testing. (d) a computer room, densely populated with computer devices and tables, presenting a scenario with significant electronic interference. (e) a laboratory room, equipped with a range of scientific apparatus, which could impact signal behavior. (f) a narrow corridor, a constrained space that may affect signal distribution differently compared to more open areas. (g) a crowded classroom, where the presence of many people could influence signal dynamics. (h) an activity room, likely with variable layouts and movement, presenting a dynamic testing environment. (i) a lobby, a typical transitional space in buildings that might offer unique challenges for signal transmission due to its layout and foot traffic.

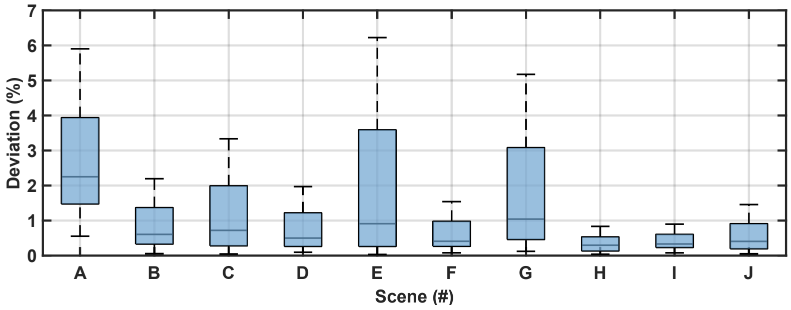


Fig. 3.11: Accuracy of RIS-NeRF^2

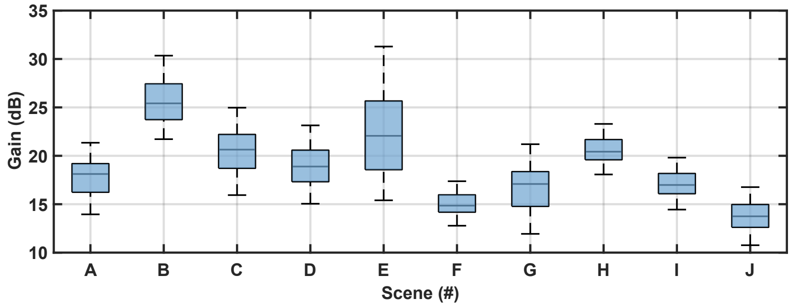


Fig. 3.12: Gain vs. scenario

specific adaptations. The voxel dimension is fixed at 1/8 of the wavelength. We set the batch size to 4096. The Adam optimizer [101] is employed for this process. The initial learning rate is established at $3e^{-4}$, which is then progressively reduced to $3e^{-5}$ through an exponential decay over the optimization period. We retain the default settings for other hyperparameters, such as $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-7}$. Typically, training the network for a single scene takes approximately 300 to $500k$ iterations to achieve convergence, using a single NVIDIA 3080Ti GPU, which roughly translates to a duration of about four hours. Nonetheless, the execution of interference is expedited, taking less than 10ms.

3.5 Experimental Results

We now evaluate the performance of MetaMosaic and verify its capability to improve WiFi communication.

3.5.1 Performance of RIS-NeRF^2

The principal innovation of MetaMosaic is the adoption of a neural radiance field, which greatly enhances the speed of evaluating reconfiguration strategies' effectiveness. Consequently, our evaluation centers on the accuracy of RIS-NeRF^2 in predicting signal strength. The precision of the radiance field model is deeply connected to the environment's specific setup, requiring tailored models for each scene. For this purpose, we collected a comprehensive dataset from 284,124 positions across ten varied scenes, as shown in Fig. 3.10. The six MetaMosaic surfaces were installed near the WiFi Access Point (AP), and mobile devices were relocated during the experiments.

(1) Accuracy: As standard procedures, we partitioned the dataset for each environment into two parts: allocating 80% for training purposes and reserving 20% for testing. To gauge the precision of our predictive model, we calculate the deviation, which quantifies the percentage discrepancy between predicted power and actual ground truth. A smaller deviation percentage reflects a prediction that more accu-

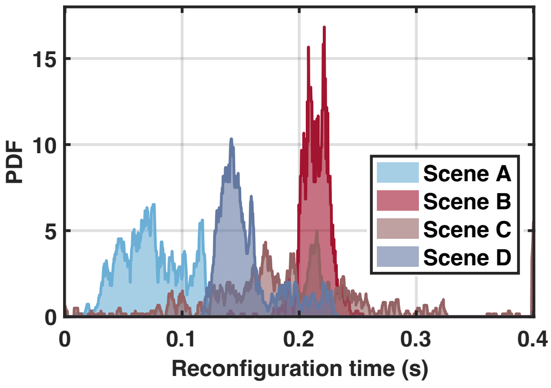


Fig. 3.13: Response Time

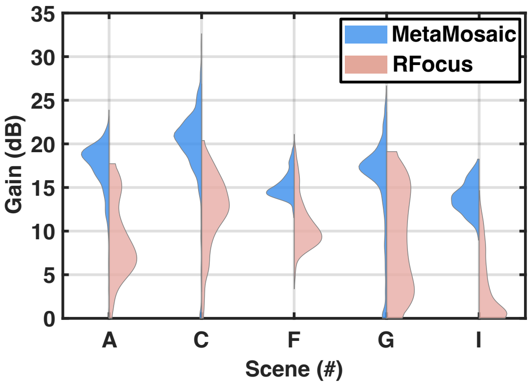


Fig. 3.14: RFocus vs. MetaMosaic

rately mirrors the ground truth. The results of this methodical testing are illustrated in Fig 3.11. Analysis of the data reveals that in line with the NeRF^2, the prediction precision of MetaMosaic is exceptionally high, exhibiting an average median deviation of 0.74%; the median deviation remains below 1% for 90% of the scenes, with Scene A being the sole exception; Scene A exhibits a deviation of 2.25%, which is notably higher due to its lower density of data points compared to other scenarios. This experiment confirms the efficacy of RIS-NeRF^2 in the domain of channel modeling and prediction, thereby laying the groundwork for an informed reconfiguration strategy.

(2) Response Time: We conducted time measurements for reconfigurations at 50 different positions within Scenes A, B, C, and D. The CDFs are depicted in Fig. 3.13. The durations recorded were 114ms, 97ms, 160ms, and 49ms for each scene, respectively. The primary factor contributing to these delays is the issuance of one or more `Select` commands, each taking about 0.825ms. The rapid response time of our system is a crucial aspect, particularly in dynamic environments where conditions change frequently. In contrast, the response times for the SOTA solutions (like RFocus [65]) are usually 1-10 seconds [64,65,72]. Their longer durations are a consequence of their hard-landing strategy, which requires substantial time to gather feedback from mobile devices. In comparison, our approach necessitates only one-tenth of this time, significantly enhancing response efficiency.

3.5.2 Overall Performance

Finally, we evaluate the overall performance of MetaMosaic from a complete system perspective.

(1) Gain across Scenes. To assess the overall enhancement provided by MetaMosaic, we deployed the six surfaces in the ten scenes and conducted evaluations at 50 random distinct positions within each scene to gauge the system's efficacy. The gain was calculated as the decibel (dB) ratio of signal strength with versus without reflection adjustments. The experimental results are shown in Fig. 3.12. From the figure, we have the following insights: MetaMosaic consistently enhances signal strength at WiFi terminals, with an average median gain of 18.82dB across all scenes; the highest median gain observed is 25dB (i.e., 316×) in Scene B (least uncontrollable reflections), whereas the lowest median gain recorded is 13.7dB in Scene J; the level of improvement is comparable in both semi-indoor and full-indoor settings, which validates the versatility of our methodology. These notable enhancements can be credited to two principal benefits of MetaMosaic. Firstly, the surfaces act as a fully reflective RIS regardless of switch states, thereby generating an increased number of reflections. This is especially beneficial in semi-indoor settings where wall reflections are absent. Secondly, our meticulously developed configuration algorithms enable precise alignment of these reflections, substantially mitigating mutual cancellation.

(2) Comparison to SOTA Solution. Our study primarily compares MetaMosaic with RFocus [65], a large-scale, one-bit amplitude-reconfigurable RIS. Limited by the unavailability of similar hardware, our analysis focuses on the performance gains achievable through the reconfiguration algorithm in our RIS setup (i.e., phase modulation). RFocus utilizes a hard-landing strategy involving a voting mechanism, requiring 165 randomized iterations and feedback to determine the optimal states of its unit cells for each assessment. We benchmarked MetaMosaic against RFocus in four different scenarios: Scenes A, B, C, and D. The outcomes of this comparison are

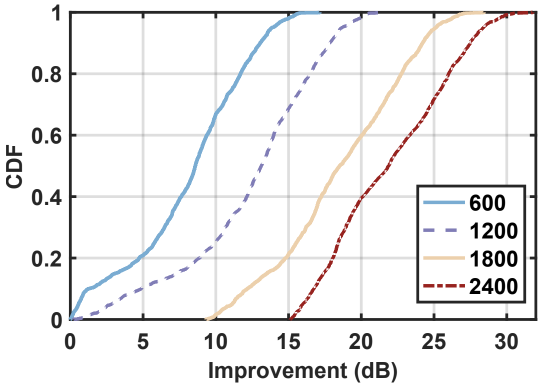


Fig. 3.15: Scalability

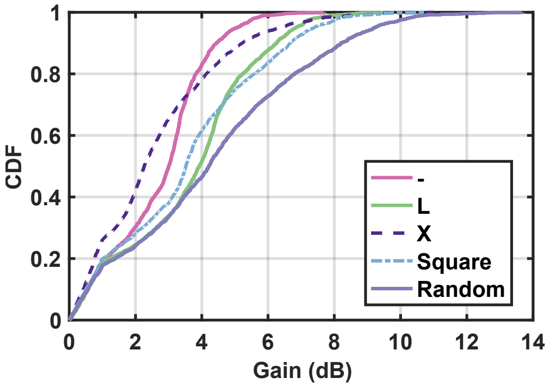


Fig. 3.16: Layout

illustrated in Fig. 3.14. Notably, MetaMosaic demonstrates superior performance, achieving 10.02dB, 7.74dB, 4.60dB, 8.43dB, and 11.74dB higher median gains than RFocus in each respective scene. This superior performance of MetaMosaic can be attributed to the advantages of its pre-learning model, RIS-NeRF^2, which ensures consistent and higher gains. Moreover, the results show that MetaMosaic typically follows Gaussian distributions, while RFocus displays more irregular patterns, resulting in a standard deviation that is 2.5dB higher than that of MetaMosaic. The inherent randomness in RFocus leads to relatively unstable gains in each iteration.

(3) Scalability. To assess the influence of varying the number of unit cells, we conducted experiments using different quantities of cells selected from three surfaces for reconfiguration, while keeping the rest unchanged. Specifically, we tested with subsets of 600, 1,200, 1,800, and 2,400 cells out of a total of 2,418. The results are depicted in Fig. 5.11. As a result, a progressive increase in median gain: from 8.59dB with 600 cells, to 13.12dB with 1200 cells, 18.51dB with 1800 cells, and reaching 21.88dB with 2400 cells. There is an approximate gain of around 4.4dB for every additional 600 cells. This trend is rational, as a greater number of unit cells leads to an increase in controllable reflections. Ideally, maximal gain is achievable when the entire scene is equipped with our unit cells, as discussed in our field study.

(4) Impact of Layout: The flexibility of MetaMosaic, enabled by discarding wired

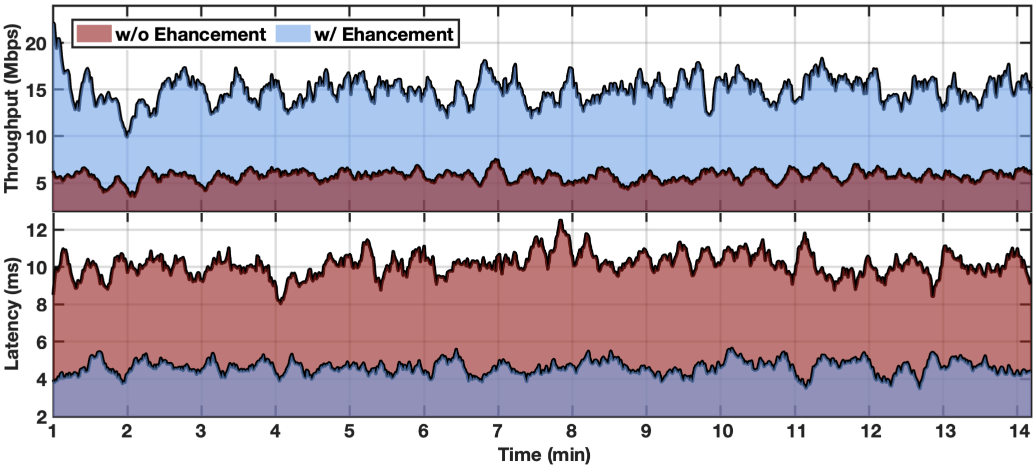


Fig. 3.17: Throughput and Latency in the Field Study

connections, allows for diverse configurations. In our tests, 186 unit cells were organized into linear, L-shaped, X-shaped, square, and random patterns to evaluate how different arrangements affect gain. According to results shown in Fig. 3.16, gain variations were minimal, within a 2dB range across layouts. The random pattern yielded the highest median gain of 4.29dB. This is because the random pattern may create a sparse array that provides a broader effective aperture. Irrespective of the unit cells' deployment pattern, an enhancement in performance was observed, showcasing MetaMosaic's versatile deployment capabilities.

3.6 Field Study: XR Booster

As mentioned earlier, the true potential of RIS technology is unlocked when a space is entirely enveloped by RIS surfaces. This setup is particularly advantageous for extended reality (XR) applications that demand reliable, high-capacity, and swift wireless communication to ensure a compelling and interactive user experience, as shown in Fig. 3.2(e). In our setup, a PC in AP mode serves as the streaming source, sending data packets to a HoloLens 2 headset [102] continuously. The headset user moves to various locations every 30 seconds. We employ `iperf` and `ping` tools to measure throughput and latency, respectively. Fig. 3.17 compares the performance with and without MetaMosaic's enhancement over a 14-minute duration. The findings reveal average throughputs of 14.86 Mbps with RIS enhancement versus 5.66 Mbps without, marking a 2.6-fold increase. Additionally, latency is decreased from 10.07ms to 4.63ms, a reduction of 54%. According to a study by Ericsson, XR technologies necessitate network capabilities of tens of Mbps in bitrate and 10 - 20ms in latency for a high-quality user experience [103], benchmarks that our experiment successfully meets. Thus, we believe that MetaMosaic will emerge as a cost-efficient method to advance XR technologies without the need for protocol or hardware upgrades in communication systems.

3.7 Conclusion

This paper introduces MetaMosaic, a novel reconfigurable intelligent surface system utilizing cost-effective RFID tags for enhanced wireless communication. Our system, unique for its scalability and flexibility, significantly surpasses existing RIS solutions in signal enhancement and throughput improvement, thanks to its innovative use of battery-free, RFID-addressed unit cells. This work establishes a new benchmark in RIS technology, paving the way for future innovations in intelligent and responsive wireless network environments.

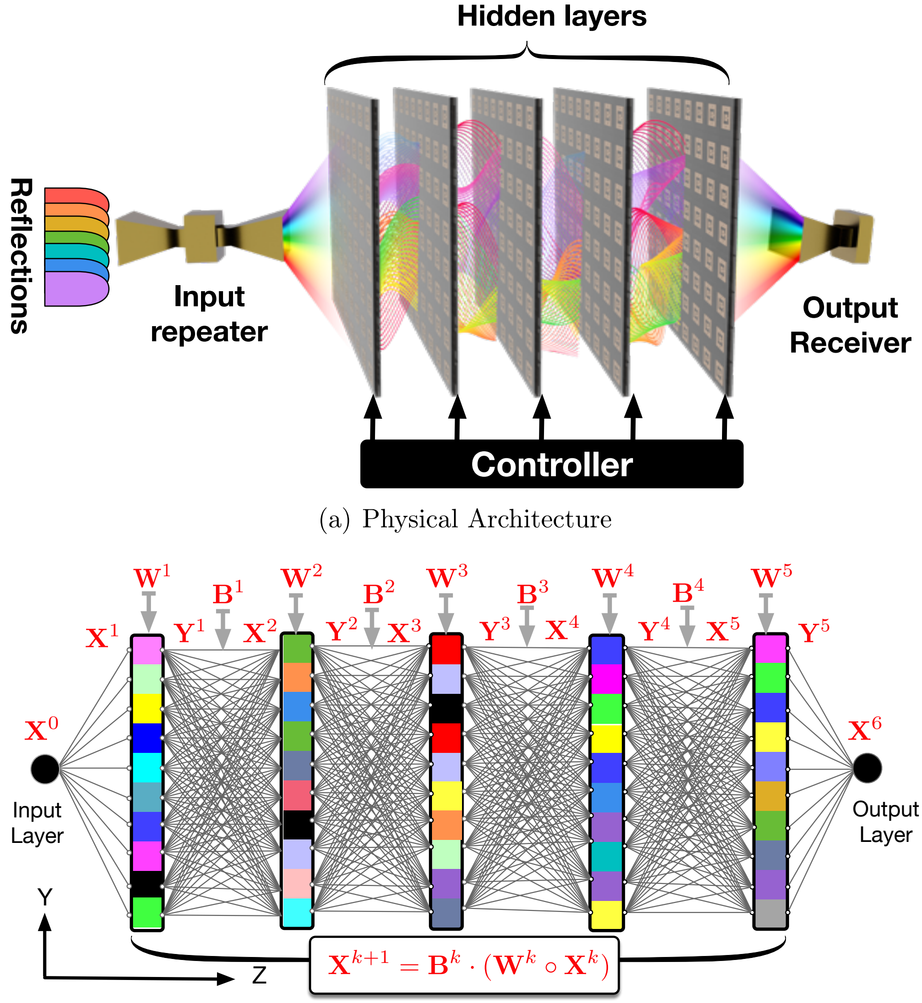
Chapter 4

Radio Frequency Neural Networks

4.1 Introduction

Wireless signals, traditionally associated with communication, are now being repurposed as sensing mediums by the community. Given that their propagation is influenced by physical locations and environmental factors, these signals inherently carry information that can depict a subject, object, or even the ambient setting. This novel utilization of wireless signals, termed “wireless sensing,” has emerged as a key role in the realm of the Artificial Intelligence of Things (AIoT). The applications of wireless sensing span a diverse spectrum, from intrusion detection, occupancy monitoring, and gesture recognition to vital sign tracking, identification, localization, and obstacle detection [104].

Wireless sensing is challenged by many factors, such as the multi-path effect, fast attenuation, ambient interference, frequency/phase offsets, hardware diversity, and so on. As a result, the accuracy of wireless sensing has been limited to a coarse-grained level for years. The goal of wireless sensing can eventually be reduced to the problems of pattern recognition, regression, or classification, which exactly fall into the domain of machine learning. With the rapid development of electronic neural networks (i.e., ENN – GPU and/or CPU-enabled logical neural networks), many works have demonstrated the successes of ENN-enhanced wireless sensing, which achieves incredible accuracy [23].



(a) Physical Architecture

(b) Logical Architecture

Fig. 4.1: Architecture of RFNN. (a) shows the physical structure of RFNN. An RFNN network is a physical neural network consisting of a K-layer metasurface. Each layer has N RF elements, which are regarded as neurons. They receive the signals from neurons on the previous layer and retransmit them with altered phase shifts to the next layer. (b) shows the logical structure of RFNN, which is abstracted as a fully connected neural network.

However, employing ENN to carry out AIoT remains challenging. Power-hungry and computation-sensitive ENNs are unaffordable by cost-effective and energy-saving sensory nodes. As a result, the recognition tasks must be offloaded into the AI cloud or edge servers. By 2032, the generated information from sensory nodes is equivalent to 10^{20} bit/s, much larger than the total collective human sensory throughput [105]. Despite the high capacity of 5G/6G networks, offloading the computations from billions of sensory nodes to the cloud remains challenging, especially for time-delay-sensitive applications (e.g., autonomous vehicles). This condition motivates us to develop in-sensor machine learning, that is, making learning operate close to or inside sensors.

To achieve the end of in-sensor machine learning, this work proposes radio-frequency neural networks (RFNNs), which take advantage of intelligent surfaces to construct physical neural networks for wireless sensing. Fig. 4.1 shows the architecture of the RFNN. Specifically, RFNN is deployed in front of a wireless sensory node (e.g., a BLE receiver, a WSN mote, etc.). The RFNN consists of multiple physical layers, each of which is made of a transmissive and reconfigurable intelligent surface (i.e., metasurface) that receives an RF signal from one side and transmits it to the other. As abstracted in Fig. 4.1(b), the entire arrangement mirrors the structure of a fully connected neural network as follows:

- Analogous to neurons, the RF elements on metasurfaces intake RF signals from prior-layer elements and retransmit modified signals to RF elements on the next layer.
- Analogous to weights, the adjustable transmission coefficient of an RF element can modulate passing-through RF signals by means of complex-valued multiplication.
- Analogous to sums, the RF signals produced by RF elements from one layer linearly combine when they reach the elements of the next layer.

An RFNN is fed with enough samples to learn the appropriate weights (i.e., voltages), making the IoT receiver detect different RSS levels, each representing a recognition result.

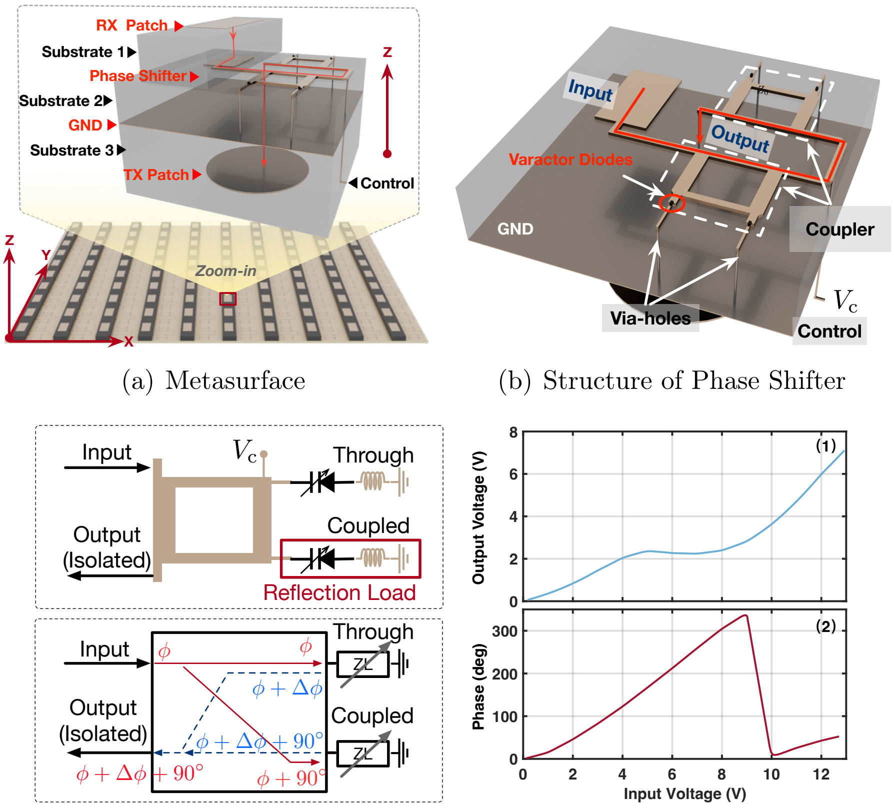
Unlike embedded machine learning approaches such as TinyML [106], which implement machine learning techniques on resource-limited embedded devices at the network's edge, RFNN introduces a novel paradigm. This innovative approach leverages the inherent properties of RF signals to execute inferences in-air, eliminating the need for traditional processing units such as CPU, GPU, or even MCU. Instead, RFNN capitalizes on the RF signals reflected by subjects or objects, allowing them to naturally perform a series of additions and multiplications as they traverse through physical layers. Consequently, RFNNs not only operate at an enhanced speed but also with greater efficiency. Furthermore, an RFNN can be seamlessly integrated into an existing sensory node, imposing negligible requirements on node configurations.

However, achieving RFNN faces many practical challenges:

• *How are gradients of an RFNN computed?* Traditional ENNs employ the backpropagation algorithm for error minimization. This method derives gradients by applying the chain rule to the differential of the network's recursive formula. Unfortunately, RFNN represents a unique form of moral computing system that cannot be accurately modeled using standard mathematical techniques. This limitation arises because its output is highly dependent on specific hardware characteristics [107], which results in a non-differentiable problem. To overcome this challenge, we introduce full-forward-propagation, which calculates loss, gradient, and error directly from the input to output layers, leveraging the RFNN's capacity for instant forward propagation. The technical details are presented in §4.4.

• *How are the samples labeled?* An RFNN uses the RSS of received signals as recognition labels. For example, "Apple"→-10 dBm, "Pear"→-20dBm, "Orange"→-30dBm. Without prior knowledge about the median RSS that the sensory node detects, the blind RSS labels might never be achieved, regardless of how much effort is made by the neural network. Instead of absolute and blind labels, we adopt contrastive learning to allow the machine to self-learn the labels by only feeding a pair of similar or dissimilar samples. In addition, we also introduce an algorithm to compress time-domain signals into the frequency domain for the recognition of time-dependent behavior sensing. The technical details are elaborated in § 4.5.

Contribution. Our contributions are summarized as follows. First, we design and implement the RFNN using cost-effective metasurfaces for wireless sensing. Second, we propose and achieve the new paradigm of "communication as inference" to meet the urgent demand for in-sensor machine learning. Third, we validate the feasibility and effectiveness of RFNN for wireless sensing applications with comprehensive experiments.



(a) Metasurface

(b) Structure of Phase Shifter

(c) Schematic of Coupler

(d) Transmissive coefficient

Fig. 4.2: Illustration of the phase-reconfigurable transmissive metasurface. (a) shows the entire metasurface; (b) zoomed-in image of the design of an RF element; (c) the schematic of a coupler; (d) transmissive coefficient as a function of the input voltage applied on a neuron.

4.2 Transmissive Metasurfaces

Metasurfaces, consisting of two-dimensional arrays of artificial structures, have the unique ability to manipulate electromagnetic waves. Reflective and transmissive metasurfaces are particularly noteworthy for their diverse applications. Reflective metasurfaces control the phase and amplitude of incoming electromagnetic waves to create a specific reflective field. In contrast, transmissive metasurfaces manipulate waves that pass through them by adjusting properties like phase, amplitude, and polarization, thereby achieving specialized transmission characteristics.

Phase-reconfigurable Metasurface. Our design chooses the phase-reconfigurable transmissive metasurfaces, as shown in Fig. 4.2(a). The design is inspired by the previous works [108, 109]. The metasurface features an array of N RF elements that function as antenna systems, receiving EM waves on one side and retransmitting them on the opposite side. A closer look at the RF element reveals a four-layer metallic structure printed on three distinct dielectric substrates: Taconic TLC-8, Rogers RT6010, and Rogers RO4350, arranged from top to bottom. Each RF element comprises four functional components: a receiving (RX) patch, a phase shifter, a ground layer, and a transmitting (TX) patch. These components are interconnected through via-holes. The RX patch captures incoming RF signals and converts them into guided-wave signals, which are then directed to the phase shifter via coupling effects. The phase shifter modulates the phase of the signal, which is then routed to the TX patch for retransmission into the air.

Phase Shifter. Fig. 4.2(b) zooms into the design of the phase shifter. The signal flows from the input port to the underlying TX patch. In the path, the phase of the signal can be altered between 0° and 360° by the two cascaded reflection-type couplers [110]. Fig. 4.2(c) depicts the blueprint of the outport directional coupler furnished with variable reflection loads. As the input signal enters, it is divided between the through-port and the coupled-port. Reflections from these ports converge to form the output at the isolated port. The phase alteration of this combined output signal is achieved by adjusting the dual reflection loads

$$\phi_{\text{out}} = \phi_{\text{in}} + \Delta\phi + \pi/2 \bmod 2\pi \tag{4.1}$$

where $\phi_{\text{in(out)}}$ denotes the phase of the input (output) signals, $\Delta\phi$ signifies the phase deviation introduced by the coupler. This phase shift is influenced by the impedance (Z_L) of the adjustable reflection loads and the intrinsic impedance (Z_0):

$$\Delta\phi = 2\tan^{-1}(\frac{Z_L}{Z_0}) \tag{4.2}$$

The reflection load, composed of a varactor diode and a resistor, enables adjustments to the impedance Z_L, thus allowing for variable phase shifts. In support of this, Fig. 4.2(d)(2) showcases simulated results, demonstrating how phase shifts correlate with the voltage applied to the phase shifter. Notably, the phase shift can be tuned over a full 360-degree range by varying the voltage between 0V and 12V.

4.3 System Overview

In this section, we introduce the design of RFNN at a high level from the physical and logical views.

4.3.1 Physical Architecture

Fig. 4.1(a) shows the physical architecture of RFNN. The entire network consists of three types of physical components:

(1) Input Repeater: All components, except the input repeater, are securely housed within an RF-shielded enclosure that is safeguarded against external interference through absorptive materials. During the running phase, the energy-efficient input repeater is a single-element transmissive surface – serving as a standardized interface that seizes reflected RF signals from the designated region of interest (ROI) and channels them to the subsequent hidden layers. During the training phase, the input repeater can be linked to a preprocessor that re-emits the collected RF signals from a dataset, thereby enhancing the pace of the training procedure.

(2) Hidden Layers: The hidden layers are constructed using phase-reconfigurable transmissive metasurfaces. Mirroring the function of neurons, RF elements on these metasurfaces receive RF signals and subsequently forward them to the subsequent layer. In a manner akin to neural weights, the outgoing RF signal can be modulated by the transmissive coefficient, primarily the phase shift. Mimicking the process of addition in neural networks, the emitted RF signals from all elements within a layer are coherently merged in the ambient medium, subsequently converging in the neurons of the following layer. This RF propagation process between two hidden layers bears a striking resemblance to the forward propagation seen in ENNs. A centralized controller connects all hidden layers, playing a pivotal role in imposing the weights onto the neurons.

(3) Output Receiver: RFNN aims to be effortlessly integrated with existing IoT devices to augment their sensing capabilities. Therefore, the output receiver could be any standard IoT device in deployment, such as ZigBee, BLE, or WiFi nodes. The final hidden layer functions as a planar antenna array, with adjustable weights facilitating the formation of a targeted RF beam. We propose two methods for conveying recognition results. The first involves deploying multiple IoT devices at varying angles, each corresponding to a different classification outcome. The device that receives the strongest signal power represents the final classification. Alternatively, a single IoT device can be placed behind the last hidden layer, and the strength of the received RF signal – measured as RSS – is categorized into different levels, each indicating a specific classification result. Here, we opt for the second approach.

4.3.2 Logical Architecture

Fig. 4.1(b) shows the logical architecture of the RFNN from the perspective of a fully connected neural network.

(1) Input Layer: RFNN only contains a single receiver (i.e., input repeater) in the input layer. We use the term x^0 to denote the RF signal outgoing from the input layer. It is a complex-valued number, i.e., $x^0 = a^0 e^{\mathbf{J}\phi^0}$ where a^0 and ϕ^0 are the initial amplitude and phase of the RF signal. Since WiFi uses OFDM modulation, x^0 is a broadband signal across dozens of MHz (e.g., 20 MHz).

(2) Neurons: Let $\mathbf{X}^k$ denote the RF signals received by the N neurons (i.e., RF element) on the k^{th} hidden layer, $k = 1, 2, \ldots, K-1$. The $\mathbf{X}^k$ is an N-dimensional vector as follows:

$$\mathbf{X}^k = [x_1^k, x_2^k, \cdots, x_N^k]^T \tag{4.3}$$

where x_i^k represents the RF signal received by the i^{th} neuron on the k^{th} layer, $i = 0, 1, \ldots, N$. The signal x_i^k is complex-valued, i.e., $x_i^k = a_i^k e^{\mathbf{J}\phi_i^k}$. When the x_i^k is retransmitted by the k^{th} neuron, it is altered by the transmissive coefficient w_i^k. Let $\mathbf{W}^k$ denote the coefficients across N neurons on the layer k. It is formally written as follows:

$$\mathbf{W}^k = [w_1^k, w_2^k, \cdots, w_N^k]^T \tag{4.4}$$

Similarly, w_i^k is also a complex-valued number, i.e., $w_i^k = \Delta a e^{\mathbf{J}\Delta\phi_i^k}$ where $\Delta\phi_i$ can vary across in the range of $[0°, 360°)$. The training procedure is to learn the appropriate transmissive coefficients. The RF signal outgoing from the i^{th} neuron becomes the result of x_i^k multiplied by w_i^k as follows:

$$y_i^k = x_i^k \cdot w_i^k \tag{4.5}$$

Let $\mathbf{Y}^k$ be the vector of RF signals outgoing from the k^{th} layer.

$$\mathbf{Y}^k = [y_1^k, y_2^k, \cdots, y_N^k]^T \tag{4.6}$$

Eventually, the outgoing RF signals from the k^{th} layer is summarized as follows:

$$\mathbf{Y}^k = \mathbf{W}^k \odot \mathbf{X}^k \tag{4.7}$$

where $\odot$ stands for the Hadamard product operation.

(3) Bias: According to the Huygens-Fresnel principle, each neuron is viewed as a secondary RF source transmitting y_i^k. Thus, the neurons on the $(k+1)^{\text{th}}$ layer receive a linear superposition of all RF signals transmitted from neurons on the k^{th} layer. Thus, the RF propagation between two adjacent layers is described as follows:

$$\mathbf{B}^k = \begin{bmatrix} b_{1\to 1}^k & b_{2\to 1}^k & \cdots & b_{N\to 1}^k \\ b_{1\to 2}^k & b_{2\to 2}^k & \cdots & b_{N\to 2}^k \\ \vdots & \vdots & \vdots & \vdots \\ b_{1\to N}^k & b_{2\to N}^k & \cdots & b_{N\to N}^k \end{bmatrix} \tag{4.8}$$

where $b_{i\to j}^k$ represents the channel parameter between the i^{th} neuron on the k^{th} layer to the j^{th} neuron on the $(k+1)^{\text{th}}$ layer. The $b_{i\to j}^k$ is also a complex-valued number that describes the amplitude attenuation and the phase rotation caused by the channel. In summary, the incoming RF signal at the $(k+1)^{\text{th}}$ layer is defined as follows:

$$\mathbf{X}^{k+1} = \mathbf{B}^k \cdot \mathbf{Y}^k = \mathbf{B}^k \cdot (\mathbf{W}^k \odot \mathbf{X}^k) \tag{4.9}$$

Equivalently, the RF signal received by the j^{th} neuron on the $(k+1)^{\text{th}}$ layer is given by the following recursive formula:

$$x_j^{k+1} = \sum_{i=1}^{N} b_{i\to j}^k w_i^k x_i^k \tag{4.10}$$

(4) Output Layer: The output layer also contains a single node representing the output IoT device. The amplitude of the RF signal received by the device indicates the classification result. Formally, the final result is written as follows:

$$x^K = \Big|\sum_{i=1}^{N} b_{i\to j}^{K-1}(w_i^{K-1}x_i^{K-1})\Big| \tag{4.11}$$

The ultimate goal is to train an RFNN to minimize the loss function of $||X^K - \hat{X}||_2$, i.e., $\hat{X}$ is the label.

(5) Nonlinearity: Unlike conventional linear classifiers such as SVM or LDA, neural networks derive their strength from nonlinear activation functions (e.g., Sigmoid, ReLU). The nonlinearity in RFNN is twofold: First, Fig. 4.2(d)(1) illustrates a unique nonlinear relationship between a neuron's output and input voltages, a characteristic not captured by traditional activation functions. The recursive formulation is updated as follows:

$$x_j^{k+1} = \sum_{i=1}^{N} b_{i\to j}^{k} g(w_i^k x_i^k) \quad \text{or} \quad \mathbf{X}^{k+1} = \mathbf{B}^k \cdot g(\mathbf{W}^k \odot \mathbf{X}^k) \tag{4.12}$$

where $g(\cdot)$ serves as a unique activation function influencing the amplitude of the output RF signal, which arises as a consequence of phase reconfiguration. Second, the final output at the output layer is determined by the strength of the received signal at the node. As Eqn. 4.11 indicates, this is akin to applying a modulo operation on the received complex-valued signal, resulting from the summation of RF signals from all neurons in the last hidden layer. Clearly, transforming a complex value into a real number introduces another form of nonlinearity.

4.3.3 RFNN v.s. ENN

The recursive formulas of RFNNs and ENNs are compared as follows:

$$\underbrace{\mathbf{X}^{k+1} = \mathbf{B}^k \cdot g(\mathbf{W}^k \odot \mathbf{X}^k)}_{\text{RFNN}} \quad \text{v.s.} \quad \underbrace{\mathbf{X}^{k+1} = g(\mathbf{W}^k\mathbf{X}^k) + \mathbf{B}^k}_{\text{ENN}}$$

Their recursive formulas are remarkably similar. The goal of both networks is to choose the optimal parameters (including the weights and the biases). However, there are still some slight differences. First, the weights of an ENN are assigned to the connections of two neurons, and thereby, a total of $(K-1)N^2$ weights are learnable in an ENN. On the contrary, only KN weights are learnable in an RFNN because its weights are assigned to neurons instead of connections. Second, in contrast to the real-valued ENNs, an RFNN is complex-valued because the RF signal is represented as a stream of discrete complex numbers. As a result, the basis of RFNN is multiplied instead of added to the outgoing RF signals. Third, the biases of an RFNN are dominated by the distance along the Z-axis between two adjacent layers. Thus, the biases are shared by neurons. Finally, the nonlinearity of RFNNs comes from the electronic components instead of the well-known activation functions (e.g., Relu, Sigmoid, etc.).

In contrast to traditional ENNs, which are power-intensive due to the high number of multiplications and additions required, RFNNs leverage the properties of electromagnetic waves to perform these calculations at the speed of light. Utilizing the Huygens-Fresnel principle, multiplications and additions occur inherently as RF signals traverse from preceding to succeeding neurons. As a result, RFNNs accomplish computations almost instantaneously, regardless of the number of neurons involved, while maintaining a consistently low power consumption.

4.4 Propagation Model

In this section, we introduce the propagation model for RFNNs, outlining the traditional forward-backward propagation methods and the challenges in RFNNs, followed by our proposed approach.

4.4.1 Forward-Backward Propagation

In neural networks, the propagation model refers to the process by which data and errors are transmitted through the layers of the network. Neural networks typically adopt a forward-backward propagation model: during the forward pass, input data is propagated through the network to compute outputs and loss values; during the backward pass, error gradients are propagated backward from the output layer to earlier layers to update the network's parameters.

• **Loss-Forward Propagation**. Forward propagation is the process of passing input data through the network and computing an output based on the network's weights and biases. In this process, the input data is multiplied by the weights and then passed through an activation function that determines the output of each neuron. The output of each neuron in one layer becomes the input for the next layer until the final output is generated. The forward propagation is mainly used to compute the loss, i.e., the difference between the output and the label. *In RFNN, the forward propagation occurs at the speed of light, yielding immediate results instantaneously once the input signal is relayed by the input repeater.* This propagation is driven not by computational processes but by the natural interplay between RF signals and metasurfaces, representing a key advantage of our system.

• **Error-Backward Propagation**. By contrast, backward propagation entails the transmission of the error signal from the output layer towards the input layer. Utilizing the chain rule of calculus, it calculates the gradient of the loss function concerning each layer's weight. Subsequently, optimization algorithms, like stochastic gradient descent, employ this gradient to refine the network's weights.

In short, forward propagation enables the network to transform input signals into output predictions, whereas backward propagation drives the learning from training datasets. These two processes are alternatively repeated over many iterations until the network can make accurate predictions on new, unseen data.

4.4.2 Backward-Propagation Challenge

Next, we delve deeper into the error-backward propagation in RFNN. The weights of the network are the transmissive coefficients of neurons. A desired transmissive coefficient (i.e., phase shift) is achieved by applying a specific voltage to the phase shifter of the corresponding neuron, as shown in Fig. 4.2(d). Thus, the weight of a neuron is a function of the voltage, i.e., $w_i^k = f(V_i^k)$. By substituting it into Eqn. 4.12, the final recursive formulas are precisely modeled as follows:

$$x_j^{k+1} = \sum_{i=1}^{N} b_{i \to j}^k g\big(f(V_i^k) x_i^k\big) \text{ or } \mathbf{X}^{k+1} = \mathbf{B}^k \cdot g\big(f(\mathbf{V}^k) \odot \mathbf{X}^k\big) \tag{4.13}$$

where $\mathbf{V}^k = [V_1^k, V_2^k, \cdots, V_N^k]$ and V_i^k is the voltage applied onto the i^{th} neuron on the k^{th} layer. *Therefore, the voltages applied on neurons are the true but hidden parameters that an RFNN network really needs to learn.*

The error-backward propagation has achieved great success in the last decades. It has become the de facto method for almost all ENNs. Taking RFNN as an example, the gradients of neurons on the last layer are computed by taking the differential on the chain (Eqn. 4.13) as follows:

$$\nabla V_j^K(S) = \frac{\partial \mathcal{L}(V_j^K, S)}{\partial V_j^K} = \frac{\partial \mathcal{L}(V_j^K, S)}{\partial g(f(V_j^K))} \cdot \frac{\partial g(f(V_j^K))}{\partial f(V_j^K)} \cdot \frac{\partial f(V_j^K)}{\partial V_j^K} \tag{4.14}$$

where $\mathcal{L}(V_j^K, S)$ is the loss when the current voltage is V_j^K and the sample S is fed, and $\nabla V_j^K(S)$ is the corresponding gradient for the j^{th} neuron. With the chain rule, the error gradients of neurons on the previous k^{th} layer can be computed using that of the subsequent neurons that have been computed as follows:

$$\nabla V_j^k(S) = \sum_i^N \nabla V_i^K(S) \times \sum_i^N \nabla V_i^{K-1}(S) \times \cdots \times \sum_i^N \nabla V_i^{k+1}(S) \tag{4.15}$$

Eventually, the parameter V_j^k of the neuron is updated using the computed gradient as follows:

$$V_j^k = V_j^k - \alpha \nabla V_j^k(S) \tag{4.16}$$

where α is the learning rate, a user-defined hyperparameter.

Unfortunately, the aforementioned approach is unavailable for the training of RFNN. The precondition for differentiability is an accurate mathematical model of the objective function. Eqn. 4.14 involves two nonlinear responses from analog components, namely $f(\cdot)$ and $g(\cdot)$. Hardware imperfections result in considerable variability among analog components, even those produced on the same assembly line. Accurately modeling the characteristics of diodes, particularly when dealing with hundreds of them, is nearly unfeasible. Moreover, the RF signals within our system's enclosure may experience reflections from adjacent layers or shells, adding another layer of complexity practically unmodelable to the channel matrix $\mathbf{B}^k$.

Geoff Hinton termed this feature as *mortal computation* [107]. That is, when the computing results highly rely on some analog hardware's unique properties, their computation is mortal: it dies with the hardware. In short, the outputs of the same software cannot be repeated on different machines because the hardware-caused dependency is difficult to model. We are suffering from the same issue here. The hardware dependency makes us unable to compute the differential of Eqn. 4.13. Consequently, RFNNs cannot use error-backward propagation to learn the network parameters.

4.4.3 Full-Forward Propagation

Fundamental calculus principles indicate that even in the absence of an explicit function form, the differential can still be computed using a rudimentary limit approach as below:

$$\nabla x = \frac{\partial h(x)}{\partial x} = \lim_{\Delta x \to 0} \frac{h(x + \Delta x) - h(x)}{\Delta x} \tag{4.17}$$

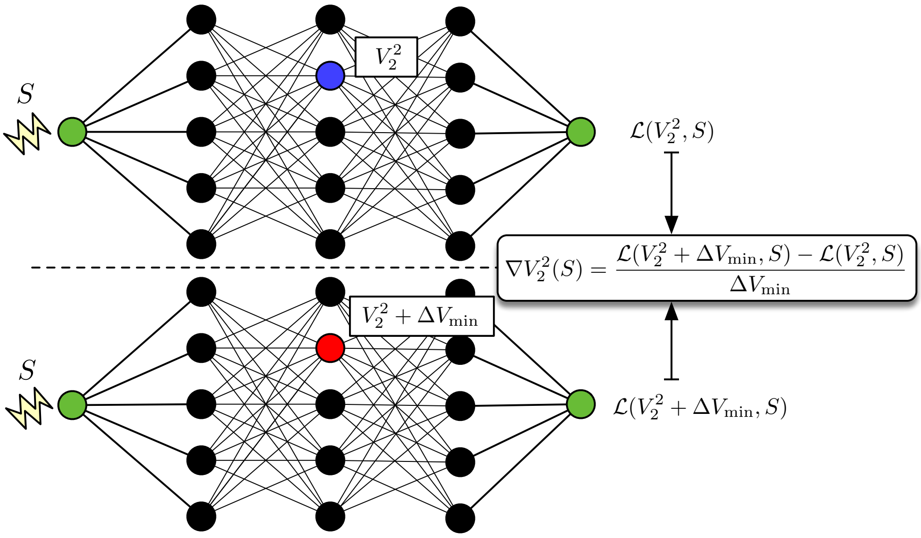

S
V_2^2
$\mathcal{L}(V_2^2, S)$
$\nabla V_2^2(S) = \frac{\mathcal{L}(V_2^2 + \Delta V_{\min}, S) - \mathcal{L}(V_2^2, S)}{\Delta V_{\min}}$
S
$V_2^2 + \Delta V_{\min}$
$\mathcal{L}(V_2^2 + \Delta V_{\min}, S)$

Fig. 4.3: Illustration of Gradient Measurement in RFNN. Setting the neuron's voltage to V_2^2, we evaluate the loss $\mathcal{L}(V_2^2, S)$ associated with the sample S. Subsequently, the neuron's voltage is adjusted to $V_2^2 + \Delta V_{\min}$, and the updated loss $\mathcal{L}(V_2^2 + \Delta V_{\min}, S)$ is recorded. The gradient derived from this sample is denoted as $\nabla V_2^2(S)$.

In a similar vein, the gradient for a specific voltage can be determined as:

$$\begin{aligned}\nabla V_j^k(S) &= \frac{\partial \mathcal{L}(V_j^k, S)}{\partial V_j^k} = \lim_{\Delta V \to 0} \frac{\mathcal{L}(V_j^k + \Delta V, S) - \mathcal{L}(V_j^k, S)}{\Delta V} \\ &\approx \frac{\mathcal{L}(V_j^k + \Delta V_{\min}, S) - \mathcal{L}(V_j^k, S)}{\Delta V_{\min}}\end{aligned} \tag{4.18}$$

This equation implies that, for a given input sample S, we evaluate the output losses twice: once when the voltage of the neuron is V_j^k and again when it is changed to $V_j^k + \Delta V_{\min}$. Here, $\Delta V_{\min}$ represents the smallest adjustable voltage increment. We can directly measure these two losses and calculate an approximate gradient using the above equation. Fig. 4.3 outlines this process.

The propagation process can be succinctly conceptualized as follows: while maintaining the voltage weights of other neurons constant, we seek to increment the voltage of a particular neuron by $\Delta V_{\min}$. If this results in a reduced loss, the adjustment is deemed in the correct direction, and the new voltage is subsequently fine-tuned accordingly. Conversely, if the loss increases, the adjustment is made in the opposite direction. The specific magnitude of the increment or decrement is determined by $-\alpha\nabla V_j^k(S)$ (see Eqn. 4.16). Notably, the gradient measurement for each neuron is independent of preceding neurons, obviating the need for a gradient chain and its associated backward propagation approach. This independence allows us to update network parameters in any sequence. Typically, earlier layers in a network focus on extracting coarse-grained features, while later layers refine these into fine-grained features. Therefore, a more intuitive approach is to update parameters starting from the input layer and progressing to the output layer, thereby gradually minimizing errors. In contrast to the error-backward propagation used in ENNs, we employ this error-forward propagation strategy.

4.5 Learning Model

In this section, we discuss the learning model that enables RFNN to recognize objects and subjects.

4.5.1 Pre-Processing

RFNN is designed to handle two sensing tasks: material recognition and behavior recognition. While material recognition is geared towards identifying object categories (e.g., apples, pears, alcohol, etc.), behavior recognition focuses on discerning actions performed by subjects (e.g., running, jumping, etc.). Upon receiving a reflected RF signal from a target, the system instantly outputs a recognition result. This immediate response is highly effective for material recognition tasks, given the time-invariant nature of material features. In contrast, recognizing behaviors often requires a few seconds due to the time-dependent context of actions.

To tackle this limitation, we propose a simple pre-processing strategy that transforms time-domain features into the frequency domain. Specifically, the input repeater's pre-processor uses a sliding window to select the top P principal frequency components out of 234. These components from successive $234/P$ windows are then aggregated to form a new time-domain RF signal for input, thereby simplifying temporal information processing. The P is a user-defined parameter.

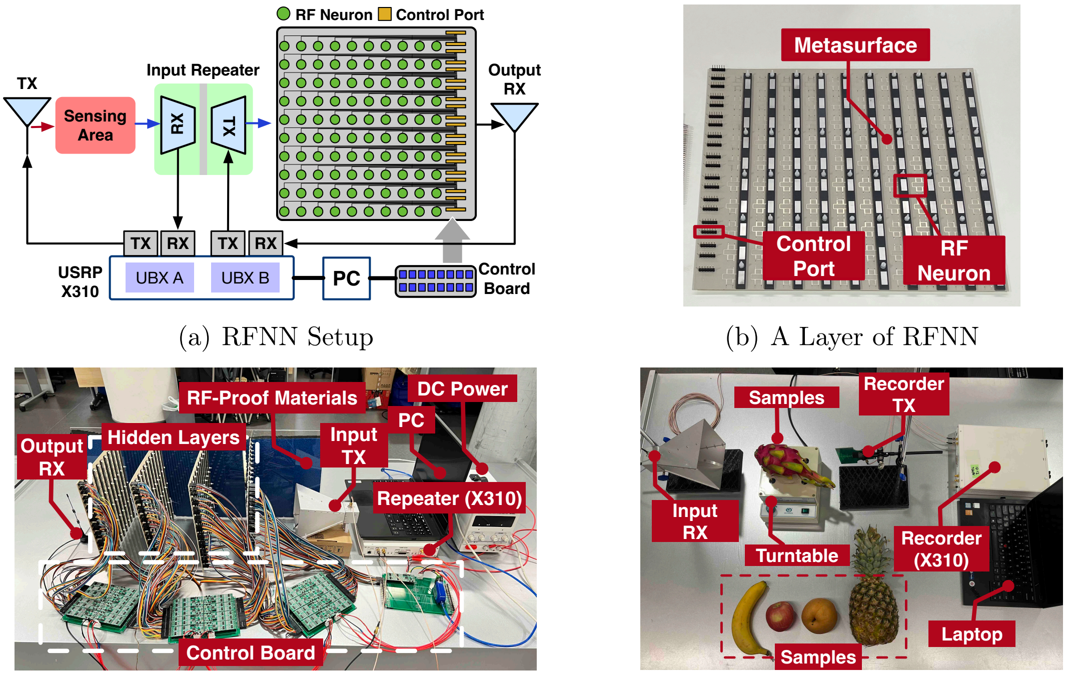
4.5.2 Contrastive Learning

In RFNN, the RSS at the IoT device serves as a labeling mechanism for recognition outcomes. For instance, RSS values like -10 dBm, -20 dBm, and -30 dBm could correspond to "Apple," "Pear," and "Orange," respectively. While users may lack prior knowledge about the median RSS of the RF signal (e.g., -60 dBm), arbitrarily labeled RSS values (like -20 dBm) that deviate significantly from this median may never be achieved. Given that the main aim of labeling is to distinguish samples, absolute RSS labels lose their significance. A more effective approach is to allow RFNN to self-generate labels, employing a self-supervised learning paradigm known as contrastive learning.

In contrastive learning, the machine is trained to differentiate between similar and dissimilar samples by pulling similar samples closer together and pushing dissimilar samples farther apart in the latent space. Formally, given a pair of samples (I_i, I_j) with a label Y, $Y = 0$ indicates similarity, while $Y = 1$ indicates dissimilarity. When the system processes I_i and I_j, it outputs two RSS values, $\text{RSS}_i = \mathcal{F}(I_i)$ and $\text{RSS}_j = \mathcal{F}(I_j)$, where $\mathcal{F}(\cdot)$ represents the neural network. The contrastive loss can then be defined based on these outputs as follows:

$$\mathcal{L} = (1 - Y)||\text{RSS}_i - \text{RSS}_j||^2 + Y \cdot \max(0, d_{\max} - ||\text{RSS}_i - \text{RSS}_j||^2) \tag{4.19}$$

where $d_{\max}$ is a hyperparameter that defines the fluctuation scope caused by the reflector, i.e., the maximal distance between dissimilar samples. Our optimization goal is to find the network parameters to minimize the loss. Clearly, we have two



(a) RFNN Setup

(b) A Layer of RFNN

(c) Prototype without the shell

(d) Dataset Collection

Fig. 4.4: Illustration of prototype and data collection.

different cases:

- If the samples are similar ($Y = 0$), then we minimize the term $||\mathrm{RSS}_i - \mathrm{RSS}_j||^2$ to zero.
- If the samples are dissimilar ($Y = 1$), then we minimize the term $\max(0, d_{\max} - ||\mathrm{RSS}_i - \mathrm{RSS}_j||^2)$ that is equivalent to maximizing their Euclidean distance until some limit $d_{\max}$.

Consequently, the neural network finds the RSS values corresponding to a recognition result by itself. The process is equivalent to automatic clustering.

4.6 Implementation

In this section, we introduce the implementation of our prototype and the data collection.

4.6.1 RF Neural Network Hardware

Fig. 4.4 shows the prototype of an RFNN, which consists of an input repeater, an output, and four hidden layers.

• **Schematic Design**. Fig. 4.4(a) shows the electrical schematic of the entire system. A USRP X310 with two individual UBX daughterboards is shared for the recorder, input repeater, and output receiver. Specifically, we first use a TX channel to emulate a WiFi AP, which broadcasts WiFi packets at 5.805 GHz. For the input repeater, we use one RX channel on daughterboard A to collect the sensing signal and then forward it to another TX channel on daughterboard B. The horn antenna is used for high-gain signal transceiving. We use the second RX channel with one omnidirectional antenna as a low-end IoT receiver. The TX power is set to 20 dBm. The gain of RX is 30 dB. The sampling rate is 20 MS/s. Notably, the USRP-based repeater is not necessary during the inference phase. Hence, two Intel AX210 NICs are utilized as the transmitter and receiver during the inference stage.

• **Hidden Layers**. Fig. 4.4(b) shows the front view of a hidden layer, i.e., a phase-reconfigurable metasurface, with 100 functional RF elements. It has a size of 27.68×29.28 cm^2 and a thickness of 4.4 mm. The inner structure is shown in Fig. 4.2(a). The varactor diode is MACOM MA46H120 [111], which has a large capacitance range from 0.1 pF to 1 pF. For each unit, we use four diodes to enable a 0-360° phase shift. The working bandwidth is from 5.725 GHz to 5.875 GHz. Each RF element has a 3 dB loss and 102° beamwidth. The minimal adjustable voltage $\Delta V_{\min}$ is set to 1 V.

• **Control Board**. The voltage on a neuron must vary between $0 - 12$ V to achieve the $0 - 360^\circ$ phase shift. We use a group of MCUs connected in series to provide the needed voltages for the 400 neurons. Usually, each MCU has two individual DAC channels to produce two analog signals with dynamic voltages. In this manner, we require 200 MCUs for the 400 neurons. Notably, this method is unscalable and cost-prohibitive. Instead, we use a pulse-width modulation (PWM) device followed by a low-pass filter to emulate a DC voltage [112]. An analog signal emulated by a PWM signal is filtered to a DC with the desired voltage. The DC signal is then amplified by TP2274 [113]. We only need to manipulate the duty cycle of the PWM device. The model of our PWM device is STM32F072 [114], which has 16 PWM channels. Therefore, we simply use 25 PWM devices to provide $25 \times 16 = 400$ DCs for the elements. They are connected in series to a PC via an RS-485 cable. It takes about 68 ms to apply the desired voltages for all elements.

4.6.2 Dataset Collection

We employ two distinct approaches to collect datasets, each tailored to specific types of tasks:

- **Behavior Sensing**: We utilize two public datasets to evaluate behavior sensing tasks, specifically localization and gesturing. For localization, we employ the DLoc dataset [23], which comprises CSI samples across 234 subcarriers at 5.626 GHz. The objective is to accurately locate the subject within one of six predefined grid zones. For gesturing, we use the Widar 3.0 dataset [115], which collects CSI samples across 30 subcarriers at 5.825 GHz.

- **Material Sensing**: We place various materials (e.g., fruits, liquids, etc.) on a rotating table and capture their RF reflections from different angles using the recorder, as shown in Fig. 4.4(d). This setup allows us to gather data on material properties, complementing the behavior-sensing dataset. Our evaluation covers three material types: oil, wine, and fruit. The first task focuses on detecting the authenticity and quality of olive oil to identify whether it is adulterated or has expired. The second task aims to detect fraudulent wine activities by classifying wine brands, production years, or grape origins. The final task is to classify categories of fruits.

In the target environment, we broadcast RF datasets at random positions. Consequently, the input repeater captures a composite of signals from the dataset and environmental reflections. This setup ensures the trained model is contextually relevant to the current scene. Should the environment undergo significant changes, or if RFNN is deployed in a new setting, retraining the model requires only a few minutes.

4.7 Evaluation

Table 4.1: Dataset Setup

	#	Task	Goal	Samples Setting	Samples	Precision	Recall
Material	1	Oil Safety	Identify Adulterated Oil	Authentic: Colavita extra-virgin olive oil; Adulerated: Authentic+10% canola oil	200	99.2%	98.9%
	2		Identify Expired Oil	Unexpired: Canola oil; Expired: Canola oil expired over 6 months	200	99.4%	99.1%
	3	Wine Fraud	Identify Adulteration	Authentic: Red Label Shiraz Grenache; Tainted: Authentic+10% ethanol	200	99.1%	99.2%
	4		Classify brands	Two brands: Max Cabernet Sauignon, Chatrau La Lagune	400	93.3%	94.1%
	5		Classify Production Year	Chatrau La Lagune wine made in: 2012, 2016, 2017	300	78.0%	79.8%
	6		Classify Grape Origins	Mancura Etnia with 2 grape origins: Merlot and Cabernet	300	96.3%	96.6%
	7	Fruit	Identify fruit types	5 types of fruits: Pineapple, Apple, Pear, Pitaya, Banana	500	84.3%	84.5%
Behavior	8	Localization	Locate people	DLoc [2]: A office with 1500 sq.ft.	600	90.0%	90.1%
	9	Gesture	Identify gesture	Widar 3.0 [16] with 6 gestures: push, sweep, clap, slide, circle, and zigzag	240	84.3%	84.2%

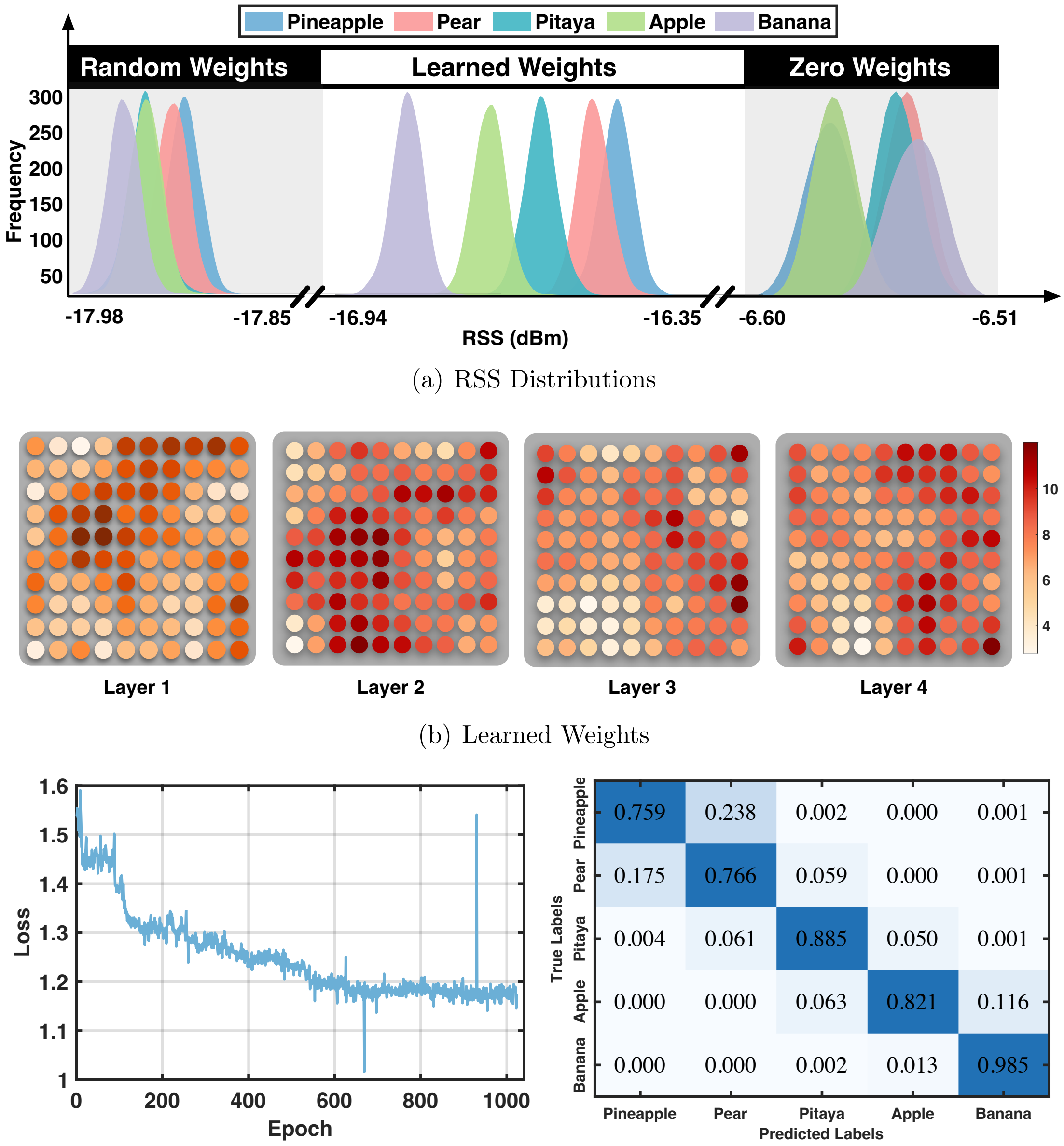
In this section, we comprehensively evaluate RFNN across a diverse set of nine inference tasks, which span both material and behavior sensing. Detailed information about these tasks is provided in Table 4.1. The evaluation dataset consists of 2,940 samples, distributed across the tasks, with each task containing between 200 and 600 samples. For the purposes of our experiments, 80% of these samples are utilized for training, while the remaining 20% are set aside for testing.

4.7.1 Feasibility

We first investigate the feasibility of RFNN with respect to the fruit classification task.

(1) Output Demo: Fig. 4.5 (a) illustrates the distributions of RSS values collected at the output receiver in three cases. Case I: Zero Weights. This case serves as a baseline, capturing RSS values in the absence of RFNN. In this case, the RSSI hinders direct fruit classification due to overlapping signal strength distributions. Case II: Random Weights. This case explores the impact of RFNN operating with randomly initialized weights. We observe a notable decline in the RSS to approximately -17 dBm, attributed to the attenuation effects of the hidden layers. Despite the random nature of the weights, the data begins to show minor separation among the four fruit categories. Case III: Learned Weights. This case demonstrates the system's capability after undergoing training via contrastive learning. The RSS distributions become significantly more distinct, enabling accurate fruit classification through the mean values of multiple measurements. Concurrently, the RSSI experiences a slight increase to around -16.94 dBm, a benefit derived from the beamforming capabilities of the final hidden layer.

(2) Training Analysis: For training, we utilize a dataset comprising $5 \times 80 = 400$ fruit samples, each of the five categories contributing 80 samples. To form a training batch, we randomly select 8 samples from each category, yielding 40 samples. These are then combined into $\binom{40}{2} = 780$ unique contrastive pairs. Consequently, each training epoch is composed of $400/40 = 10$ batches. The dataset is shuffled at the beginning of each epoch to improve generalization. The network parameters are updated in each batch. Training continues until the loss metric stabilizes, as



(a) RSS Distributions

(b) Learned Weights

(c) Training Loss

(d) Confusion Matrix

Fig. 4.5: Feasibility validation of RFNN in fruit classification.

illustrated in Fig. 4.5(c). Fig. 4.5(b) reveals the optimized hidden layer weights. These weights manifest as intricate patterns across the layers, serving as deep, despite typically inexplicable, features.

(3) Inference Accuracy: For evaluating the inference accuracy of RFNN, we set aside 100 test samples, with each of the five categories contributing 20 samples. The corresponding confusion matrix is presented in Fig. 4.5(d). Our system achieves a commendable mean accuracy of 84.5%. The underlying principle of material sensing lies in the differential absorption of RF signals across varying materials. Consequently, materials with similar absorption profiles may result in less accurate classifications; for instance, 17.5% of the samples categorized as pears were misclassified as pineapples. Overall, these experimental outcomes robustly validate the capability of RFNN to perform machine learning effectively.

4.7.2 Overall Performance

We next scrutinize the classification efficacy of the RFNN prototype over the nine distinct datasets. The evaluation metrics employed are precision and recall. As detailed in Table 4.1, we summarize our key observations as follows:

- Across nine datasets, RFNN achieves a mean precision of 91.5% and a mean recall of 91.8%. These metrics are commendably congruent with prior works such as RFIQ [116] (Precision: 97.3%; Recall: 97.3%) and Widar3.0 [115] (Precision: 92.7%; Recall: 92.6%). Both of these studies leveraged wireless signals for specialized detection tasks using ENNs.
- In the realm of material sensing, RFNN excels in identifying adulterated and expired oil, as well as fraudulent wine, with precisions of 99.2%, 99.4%, and 99.1%, respectively. It performs less robustly in discerning the year of wines, with a precision rate of 78%. This can be attributed to the inherent chemical stability of the wine, particularly for adjacent years, such as 2016 and 2017.
- For behavior sensing tasks, RFNN registers a mean precision of 84.3% in recognizing time-series human gestures. This is approximately 6% lower than its performance in localization tasks. This discrepancy arises from the frequency-domain compression conducted by the pre-processor.

In summary, the empirical results robustly affirm that RFNN possesses a learning capacity comparable to conventional ENNs, particularly in classification tasks.

4.7.3 Compared with ENN

In a subsequent evaluation, we compare RFNN with state-of-the-art ENN architectures, specifically focusing on the eighth task – localization within a 1,500-square-foot office space segmented into six distinct regions. For comparison, we employ two conventional ENN frameworks: a Multi-Layer Perceptron (MLP) and a Convolutional Neural Network (CNN). The MLP model consists of a six-layer fully connected neural network with four hidden layers, each containing 100 neurons, mirroring the architecture of RFNN. The CNN model is designed around the ResNet18 architecture [117], augmented with a 1D convolutional layer. These models are deployed on two platforms: a high-performance server with an Nvidia 3080Ti GPU and an edge computing device equipped with an Nvidia Jetson TX2 board. Consequently, we establish four baseline configurations for our comparisons: MLP/Cloud, CNN/Cloud, MLP/Edge, and CNN/Edge. The results are tabulated in Table 4.2, from which we glean several noteworthy insights.

(1) Power: Initially, we evaluate the minimum operational power required for each of the five systems under consideration. Power consumption for RFNN is ascertained through external power metering [118], while internal system monitors provide this data for the ENNs. In the case of RFNN, the PIN diodes and the RF transceiver are the primary power-consuming components. Given the maximum current of PIN diodes draw of 0.1 μA at a reverse-biased voltage of 14 V [111], an RFNN with 400

Table 4.2: Comparison with ENN

Model	Power	Accuracy	Speed	Energy
MLP/Cloud	89 W	99%	198 μs	17.6 mJ
CNN/Cloud	145.8 W	99%	3620 μs	530 mJ
MLP/Edge	1.45 W	99%	6750 μs	9.78 mJ
CNN/Edge	2.3 W	99%	119400 μs	274.62 mJ
RFNN	4.2W	90%	16 μs	67.2 μJ

neurons necessitates a maximum power of $400 \times 100\,\mathrm{nA} \times 14\,\mathrm{V} = 560\,\mu\mathrm{W}$. For the RF transceivers, the power consumption is approximately 4.2 W for the Intel AX210 NIC [119].

(2) Energy Consumption: Considering a 16µs WiFi symbol, RFNN requires approximately $(4.2\mathrm{W} + 560\mu\mathrm{W}) \times (16\mu\mathrm{s} + 0.33\mathrm{ns}) = 67.2\mu\mathrm{J}$ of energy per inference. This is roughly 146× more energy-efficient than the most frugal conventional ENN, specifically the MLP/Edge. The superior energy efficiency of RFNN arises from its ability to harness the physical properties of RF signals to perform a large number of additions and multiplications. Conversely, conventional ENNs operating on cloud or edge devices still depend on power-hungry arithmetic units, such as GPUs and CPUs, to perform these computations.

(3) Speed. In RFNN, RF signals propagate at the speed of light, making the inference time solely dependent on the path length between the repeater and the output receiver. As a result, RFNN achieves a time-saving advantage of approximately 91.9% when compared to the fastest conventional ENN, which is MLP running on a GPU (i.e., GPU-MLP).

(4) Accuracy. The average accuracy achieved by conventional ENN models is approximately 99%, a figure that surpasses RFNN's performance by about 9%. This discrepancy is mainly because of the different architecture of RFNN and ENNs. Specifically, in RFNN, the weights are applied directly to the neurons rather than the connections, resulting in a reduced learning capacity compared to conventional ENNs with the same neuron count. Besides, the current prototype of RFNN lacks anti-interference and error-correction mechanisms.

4.7.4 Impact Analysis

We test four impacts on RFNN performance regarding the eighth task (i.e., localization) as follows:

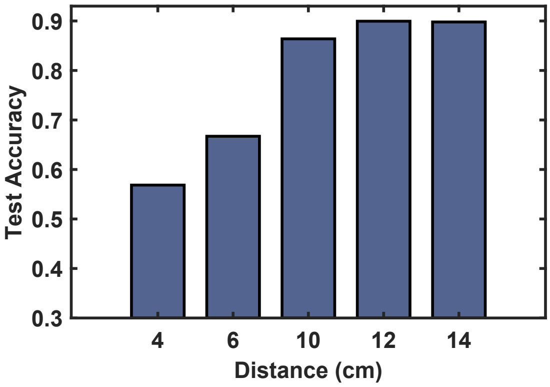


Fig. 4.6: Impact of Distance

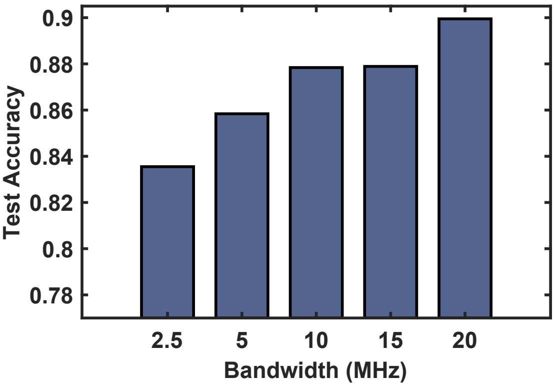


Fig. 4.7: Impact of Bandwidth

(1) Impact of distance. As discussed, the distance between hidden layers affects the RF propagation and network bias. Fig. 4.6 shows the accuracy as a function of distance. It can be seen that the accuracy increases rapidly from 66.7% to 86.4% when the layer gap increases from 4 cm to 10 cm. This is because the TX patch of a neuron is a directional antenna, whose beamwidth is around 102°. If two layers are placed too close, the main lobe of the neurons in the previous layer can only cover a few neurons in the next layer and lose many connections. Our experiments suggest a minimum layer spacing of 12 cm.

(2) Impact of frequency band. We test the classification accuracy with five bandwidths ranging from 2.5 to 20 MHz. The results are illustrated in Fig. 4.7. With an increasing bandwidth, accuracy grows gradually because a larger bandwidth can carry more information.

(3) Impact of neural number. The number of neurons is one of the key factors for any kind of neural network. We test five types of networks, where each hidden layer contains 20, 40, 60, 80, and 100 neurons. The results are shown in Fig. 4.8. Clearly, a network with more neurons shows a higher learning ability.

(4) Impact of the number of hidden layers. Fig. 4.9 shows the accuracy as a function of the neuron number in a hidden layer. Accuracy keeps increasing with additional hidden layers. The result remains consistent with deep neural networks, where

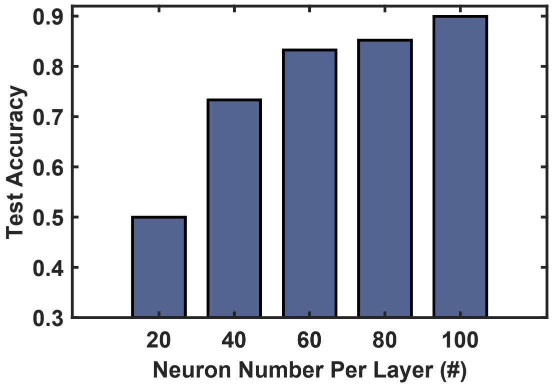


Fig. 4.8: Impact of Neuron Number.

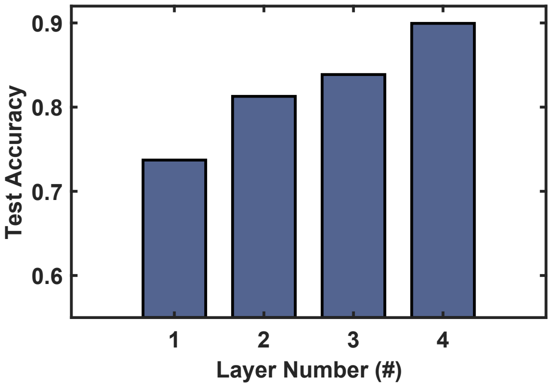


Fig. 4.9: Impact of Layer Number

deeper networks can learn more complex representations. Thus, a deeper RFNN is recommended to realize more complex tasks.

4.8 Discussion and Future Works

In this section, we discuss the limitation of RFNN and the practical issues that may happen during the usage.

Generalizability. It is a significant challenge to transfer a neural network across different scenarios, not unique to RFNN. However, there are several methods that can mitigate such an issue. To address this limitation, we can implement a test-time adaptation approach. This strategy allows the network to adjust dynamically to new environments at deployment by fine-tuning the model based on a small set of data from the new environment. Besides, researchers have explored more advanced neural network models with domain discriminators to achieve environment-independent wireless sensing. These techniques aim to train the models to be robust to environmental variations, reducing the need for frequent retraining. In future work, we consider exploring advanced techniques with RFNN to achieve environment-independent wireless sensing on physical neural networks.

Scalability. The physical dimensions of RFNN need to be adequately sized to house a sufficient number of neurons. However, there are several methods that can be employed to enhance the scalability and address size constraints. First, by increasing the operational frequency, we can significantly reduce the size of individual neurons and increase density. For instance, shifting the carrier frequency from 5GHz to 60GHz would allow a tenfold reduction in the size of physical neurons, enabling each layer to accommodate up to 10,000 neurons, an increase from the current 100. Second, using substrate materials with higher permittivity for the metasurfaces could reduce the physical dimensions of the layers without affecting functionality. Third, expanding the depth of the RFNN can further scale up the neuron count, providing additional pathways to enhance network capacity and functionality.

Comparison with Optical Neural Networks. While Optical Neural Networks (ONNs) offer high-speed and low-latency inference through light-based computing, they are not readily compatible with existing wireless sensing systems. To apply ONNs in RF sensing, the analog RF signals must first be captured through front-end components such as antennas, amplifiers, and ADCs, and then converted into optical signals before being processed optically. This significantly increases system complexity and power consumption. In contrast, RFNNs can directly operate on native RF signals, performing inference in the analog domain without digitization or conversion. Therefore, RFNNs represent a more practical and scalable solution for extending intelligent computation into existing wireless infrastructure.

Chapter 5

Enhancing Temporal-Spatial Traceability Through AC Flickering

5.1 Introduction

In the realm of optical wireless communications, ensuring robust security and traceability is indispensable, especially when handling sensitive data transfers such as those in mobile payment systems. Traditional methods, which often rely on static optical codes like QR codes, are fraught with significant security challenges. These static codes are vulnerable to a range of threats, primarily because they lack spatial-temporal traceability. This flaw allows the codes to be scanned and reused without any contextual verification, thereby exposing them to significant risks of unauthorized access and fraudulent transactions.

To overcome these vulnerabilities, there is a compelling need for the implementation of dynamic coding systems. Such systems would enhance security by incorporating temporal information and facilitating real-time tracking and authentication of transactions. Moreover, the immutable nature of printed optical codes, while convenient, poses substantial security risks in environments such as mobile payments or digital ordering systems. In such settings, the permanence of the code can be exploited for malicious purposes. Here are two real-world examples that illustrate these risks:

- **Risk of Missing Temporal Context**: In China, a consumer unintentionally posted a selfie with a static QR code for restaurant orders on social media. This code was later misused by others to place unauthorized orders, resulting in fraudulent charges amounting to approximately $60,000 for the subsequent guest [120]. This highlights the critical need for optical codes that have an expiration period, thereby avoiding transactions via outdated codes.
- **Risk of Missing Spatial Context**: In the Netherlands, a significant incident occurred where attackers replaced legitimate QR codes at cashier desks with fraudulent ones. When unsuspecting customers scanned these codes, the payments were redirected to the attackers' accounts. If systems could verify the locations of transactions, service providers could prevent payments at locations that do not match the pre-registered sites.

To mitigate the above risks, the People's Bank of China has imposed regulations that cap daily transaction amounts to 500 RMB (about 70 USD) at merchant locations when *static QR codes* are used [121]. This move is intended to reduce the risks of unauthorized transactions.

The susceptibility of static optical codes to fraud stems from their lack of embedded spatial and temporal contexts, which means it is unclear where and when a scan took place. Several potential solutions to mitigate the issue come with their own challenges. **(1) IP Detection**: This method is not fully reliable, especially in environments where users are likely to connect through mobile networks like 5G, which do not reliably tie an IP address to a specific location. **(2) GPS Detection**: This method can be effective but often fails indoors, such as in restaurants, and also raises concerns about user privacy. **(3) Transient Codes**: More restaurants are adopting transient static codes printed for new customers to place orders. These codes, linked

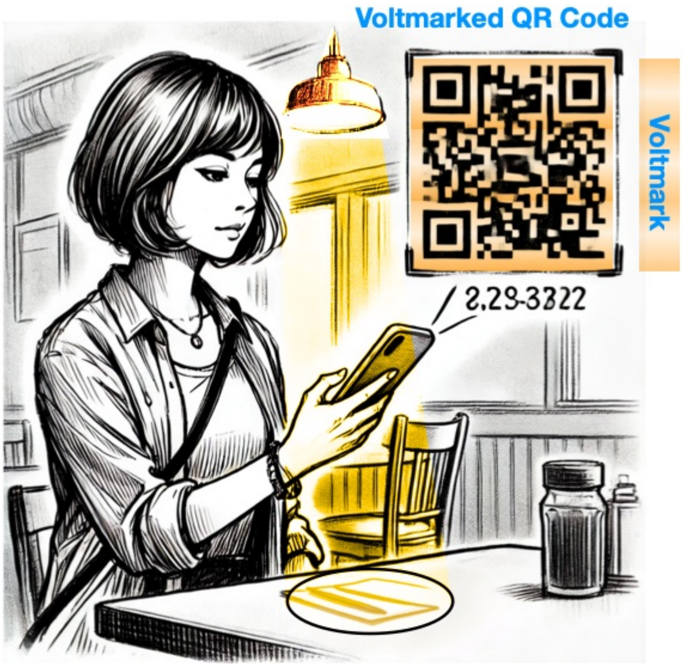
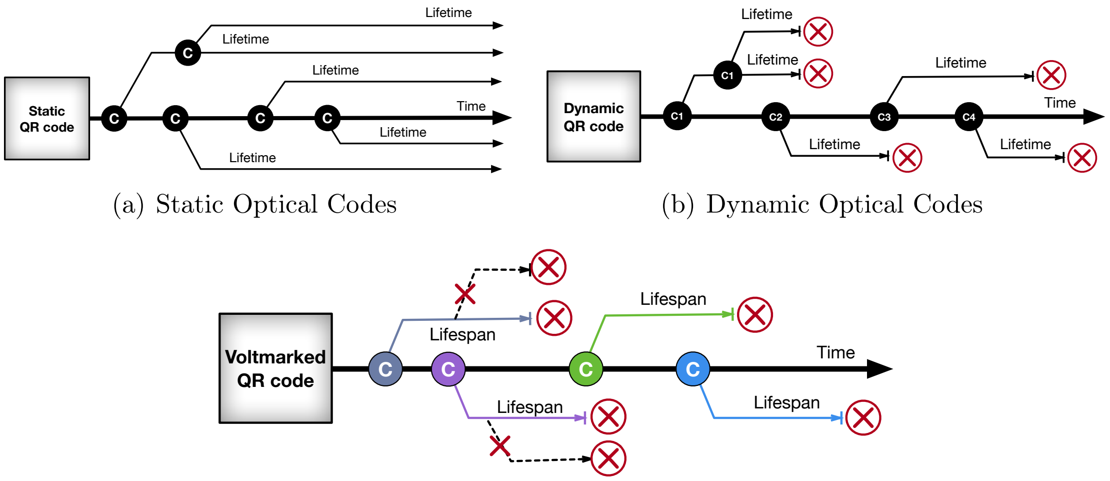


Fig. 5.1: Usage Scenario. The Voltmark is naturally embedded in photographs of optical codes that are illuminated by indoor bulbs. This allows us to extract subtle and latent Voltmarks, enabling us to trace the spatial and temporal contexts of the scan.



(a) Static Optical Codes

(b) Dynamic Optical Codes

(c) Voltmarked Optical Codes

Fig. 5.2: Traceability Comparison across Three Optical Codes.(a) Static optical codes have an unlimited lifespan and cannot differentiate between originals and copies due to their fixed nature. (b) Dynamic optical codes have a defined lifespan, rendering linked transactions invalid post-expiration, but lack the detailed temporal context to distinguish scans from originals or copies. (c) Voltmarked optical codes combine the static codes' ease and flexibility with the dynamic codes' traceability, invalidating copies by exhibiting variable contexts.

to the ordering system, expire at the end of the meal. While this mitigates some security risks, the codes remain vulnerable to attacks during the meal and necessitate upgrades to the ordering system. **(4) Dynamic Codes**: These address the security concerns more effectively by using one-time codes that become invalid as soon as a transaction is initiated. Dynamic codes also enable backend systems to accurately track both the location and timing of each transaction. However, they generally require additional hardware, such as screens or displays, for their generation. As a result, the majority of small and medium-sized merchants continue to prefer static optical codes to facilitate their business operations despite the potential risks. Thus, finding effective and privacy-preserved solutions to secure transactions facilitated by static optical codes remains a significant challenge.

In this work, we introduce Voltmark, which leverages the subtle AC flickers as a form of "Voltmarks" to determine the spatial-temporal context of photographed optical codes. Specifically, the intensity of light bulbs is known to vary with fluctuations in AC voltage, resulting in rapid and subtle flickers in images captured by rolling-shutter cameras [122–126]. We take advantage of this latent Voltmarks to enhance temporal and spatial traceability of optical codes as follows:

- **Spatial Traceability**: Each bulb (whether an incandescent or LED) displays a unique spectral response function (BSRF) due to distinct physical properties such as material composition or manufacturing techniques. These characteristics produce specific Voltmarks associated with one or more bulbs, linking them to a particular location and thus enabling spatial traceability.
- **Temporal Traceability**: The AC voltage naturally fluctuates over time, influenced by both inherent voltage and frequency variations. The Voltmarks captured in images of optical codes are thus time-specific. This property allows us to confirm whether the Voltmarks correspond to a specific claimed time, enabling temporal traceability.

Voltmarks offer a bundle of advantages compared to potential alternative solutions. Firstly, unlike dynamic codes that require the installation of numerous screens across various locations, Voltmarks preserve the cost-effectiveness of static optical codes with minimal additional expense. We provide a detailed comparison of the traceability features of the three codes in Fig. 5.2. Secondly, Voltmarks are naturally derived from AC voltage fluctuations, allowing existing infrastructure (such as lighting fixtures) to remain unchanged. This contrasts with previous work that employed tailored smart bulbs to inject flickers [127]. Thirdly, Voltmarks enhance consumer privacy by eliminating the need for activating intrusive smartphone sensors like GPS or 5G connectivity.

However, implementing the spatial-temporal traceability through Voltmark introduces several significant challenges.

• First, it is crucial to determine how latent Voltmarks are consistently captured by rolling-shutter cameras, standard in modern smartphones. These cameras sequentially expose and read the CMOS pixel arrays, allowing each line to capture AC-caused brightness fluctuations. We have thoroughly modeled the flicker generation process and confirmed Voltmarks' presence under typical camera settings.

• Second, AC flickering is typically minimized by adjusting the camera's exposure time to match the AC cycle in default settings. Additionally, regular patterns or stripes in images (e.g., barcodes) can exhibit spatial frequencies similar to Voltmarks. Both factors increase the difficulty of Voltmark extraction. To overcome these issues, we employ a Vision Transformer (ViT) to accurately detect Voltmark within augmented datasets.

• Thirdly, measuring BSRFs typically requires specialized equipment, such as integrating spheres or high-speed spectrometers, which can add to the cost and complexity of deploying Voltmarks. To mitigate this, we leverage the Kolmogorov-Arnold Network (KAN) to model the BSRFs of bulbs using only a few voltage samples taken directly from the bulbs. This approach reduces the need for expensive equipment and simplifies the process of capturing BSRFs.

Contribution. While AC flickers themselves are not a new phenomenon, extracting both temporal and spatial contexts from these flickers has not been previously accomplished. To the best of our knowledge, it is the first to use AC flickers as watermarks to enhance the spatial-temporal traceability of optical codes. This design strikes an effective balance between security and cost. On one hand, it does not impose additional burdens on the merchant while maintaining the accessibility and convenience of static optical codes. On the other hand, it enhances traceability and security to a level comparable to dynamic optical codes without privacy compromise. In short, our main contribution lies in the design of the Voltmark and the associated verification protocol.

5.2 Preliminary

In this section, we present the background knowledge of AC flickering.

5.2.1 From AC Electricity to Light

The AC in the electrical grid is characterized by a zero-mean sinusoidal voltage $V(t)$, oscillating at f_{AC}. The AC voltage at any time t is given by:

$$V(t) = V_{\max} \sin(2\pi f_{\mathrm{AC}} t + \phi_0) \tag{5.1}$$

where $V_{\max}$ represents the maximum peak voltage (e.g., $220 \times \sqrt{2}$ volts or $110 \times \sqrt{2}$ volts); f_{AC} is typically 50 Hz or 60Hz, depending on the region; ϕ_0 denotes the initial phase. The initial phase can be one of three values: 0°, 120°, and 240°, arising from the utilization of a three-phase power system. Given that the wavelength of AC can extend up to 5,995 km, the phase shift caused by the distance from an electric substation to a residential home within a city is typically negligible.

A bulb is a device that converts electrical voltage, $V(t)$, into spectral flux, $L(t)$. In the process of converting electricity to light, the spectral flux is proportional to the square of the voltage, i.e., $L(t) \propto V(t)^2$, which results in the flux varying at twice the frequency of the alternating current (AC),i.e., $f_{\text{LUX}} = 2f_{\text{AC}}$ where f_{LUX} is the frequency of flux. Practically, this conversion often exhibits nonlinear characteristics due to various mediating factors. For incandescent bulbs, mediators such as heat, gas, and phosphorescence facilitate the conversion from electricity to light, whereas in non-incandescent bulbs, components like diodes and inductors serve a similar mediating role. These mediators introduce nonlinearities that lead to distortions in the conversion process. To accurately describe such nonlinearities, a unit-less bulb BSRF, denoted as $h(\delta, V(t))$, is utilized. This function characterizes the flux response in a specific spectral band δ (e.g., red, green and blue channels) to an input voltage, expressed as $L(\delta, t) = h(\delta, V(t))$. The BSRF can be measured by using specialized equipment such as integrating spheres and high-speed spectrometers.

5.2.2 From Light to Image

An image comprises $R \times C$ pixels, where r and c represent the row and column indices, respectively. The intensity of the pixel at the r^{th} row and c^{th} column, $I_{r,c}(t)$, when illuminated by a bulb, is described by the equation:

$$I_{r,c}(\delta, t_s) = \int_{t_s}^{t_s+\Delta T_{\text{exp}}} G_{r,c}(\delta) h(\delta, V(t)) dt \tag{5.2}$$

where t_s is the time point at which the pixel begins exposure and ΔT_{exp} is the duration of this exposure. The channel coefficient $G_{r,c}(\delta)$ represents the light propagation channel coefficient, which is influenced by various factors, including the scene, the distance of the object from the bulb, scene, albedo, lens aperture, bidirectional re-

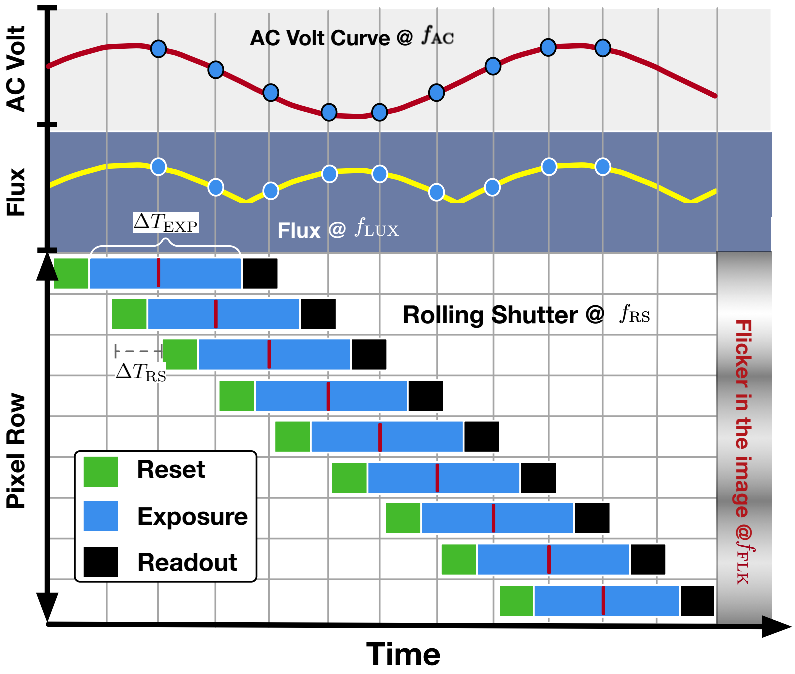
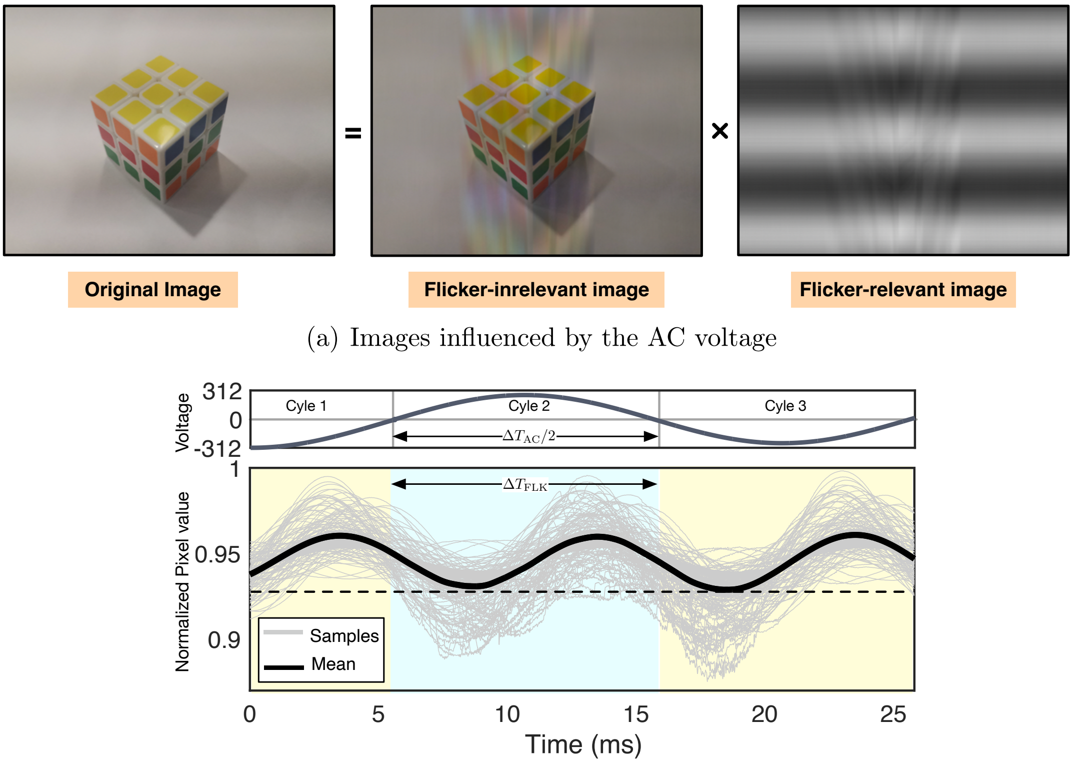

AC Volt
AC Volt Curve @ f_{AC}
Flux
Flux @ f_{LUX}
ΔT_{EXP}
Rolling Shutter @ f_{RS}
ΔT_{RS}
Pixel Row
Reset
Exposure
Readout
Flicker in the image @f_{FLK}
Time

Fig. 5.3: Rolling Shutter Model for AC Flickering. A bulb converts sinusoidal AC voltage at a frequency of f_{AC} into light flux with a frequency of $f_{\mathrm{LUX}} = 2f_{\mathrm{AC}}$, which is captured by the rolling shutter at a frequency of f_{RS}. Each row of pixels is exposed for a fixed duration, capturing a segment of the flux. This process can be approximated as sampling the AC voltage at the midpoint of the exposure window, resulting in a rapid and repeated flicker appearing at a spatial frequency of $f_{\mathrm{FLK}} = f_{\mathrm{RS}}/f_{\mathrm{FLX}}$ on the photograph.

flectance, and inter-reflections. This formulation is applicable to cameras equipped with a global shutter, where all pixels commence and conclude light accumulation simultaneously during the exposure period.



(a) Images influenced by the AC voltage

(b) Flicker signals

Fig. 5.4: Flicker signals extracted from an image captured in a bulb-illuminated scene. (a) shows the original image, differentiating between flicker-irrelevant and flicker-relevant components. (b) illustrates the reference AC voltage alongside the flicker signals (gray) extracted from 100 columns and the average flicker signal (black) across all columns.

In the consumer electronics market, CMOS cameras utilized in smartphones and digital cameras predominantly employ rolling shutters. Unlike global shutters, rolling shutters record the image sequentially, capturing one row of pixels at a time. This process progresses vertically across the image sensor. As illustrated in Fig. 5.3, the intensity captured by each pixel in the rolling shutter model is rewritten as follows:

$$I_{r,c}(\delta) = \int_{t_s + r\Delta T_{\mathrm{RS}}}^{t_s + r\Delta T_{\mathrm{RS}} + \Delta T_{\mathrm{exp}}} G_{r,c}(\delta) h(\delta, V(t) dt \tag{5.3}$$

where ΔT_{RS} denotes the exposure delay between two adjacent rows, t_s represents the time point at which the camera begins its exposure, and $r = 0, 1, 2, \ldots, R-1$. The parameter ΔT_{RS} is a constant, varying across different camera models, while ΔT_{exp} is a user-configurable parameter that can be adjusted according to specific photographic needs.

Strictly speaking, the pixel value does not represent a sampling of the AC voltage at a specific instant but rather over an exposure window, even though this window may be very small. There is some temporal overlap between adjacent rows. To address this issue, we use the mean value of the pixel intensities within the exposure window to approximate the sampling at the midpoint of this window.

$$I_{r,c}(\delta) \approx G_{r,c}(\delta)\bar{h}\Big(\delta, V(t_r)\Big) \tag{5.4}$$

where $t_r = t_s + r\Delta T_{\text{RS}}$ and

$$\bar{h}\Big(\delta, V(t_r)\Big) = \frac{1}{\Delta T_{\text{EXP}}}\int_{t_r}^{t_r+\Delta T_{\text{exp}}} h(\delta, V(t))dt \tag{5.5}$$

Consequently, the pixel value provides an approximate representation of the sampling of the AC voltage at the frequency of $f_{\text{RS}} = 1/\Delta T_{\text{RS}}$.

5.2.3 From Image to Flicker

Due to the rolling shutter effect, each row records the voltage-induced flux sequentially, resulting in AC flicker. This phenomenon manifests as rapid, faint, and recurring stripes of varying light intensity within the photograph, often imperceptible to the human eye. Notably, the spatial frequency of the flicker along the column, represented by f_{FLK}, is calculated as the ratio of the sampling frequency to the flux frequency, specifically $f_{\text{FLK}} = f_{\text{RS}}/f_{\text{LUX}}$. Leveraging this prior knowledge, we can isolate the flicker from other image components. Referring back to Eqn. 5.4, we implement a logarithmic transformation as below:

$$\ln I_{r,c}(\delta) \approx \underbrace{\ln G_{r,c}(\delta)}_{\text{flicker-irrelevant}} + \underbrace{\ln \bar{h}(\delta, V(t_r)))}_{\text{flicker-relevant at } f_{\text{FLK}}} \tag{5.6}$$

where $I_{r,c}(\delta)$ denotes the pixel value at the r^{th} row and c^{th} column within the color band δ. This logarithmic transformation converts the product of two components into a linear sum while preserving the frequencies of the flicker-relevant component. This allows us to effectively separate the two components using a filtering technique. Specifically, due to the non-linear transformation introduced by the logarithmic operation, the flicker-relevant term contains harmonics at frequencies $\pm f_{\text{FLK}}$, $\pm 2f_{\text{FLK}}$, $\pm 3f_{\text{FLK}}, \cdots$. Thus, we can employ a linear filter $F(f_{\text{FLK}})$ designed to attenuate these frequencies. This vertical homomorphic spatially invariant filtering technique can be formally expressed as follows:

$$\begin{aligned}
&\exp\left(\ln\Big(I_{r,c}(\delta)\Big) * F\big(f_{\text{FLK}}\big)\right) \approx \exp\left(\ln\Big(G_{r,c}(\delta)\bar{h}(\delta, t_r)\Big) * F\big(f_{\text{FLK}}\big)\right) \\
&= \exp\left(\ln G_{r,c}(\delta) * F\big(f_{\text{FLK}}\big) + \ln \bar{h}\big(\delta, V(t_r)\big) * F\big(f_{\text{FLK}}\big)\right) \\
&= \exp\Big(\ln G_{r,c}(\delta) * F\big(f_{\text{FLK}}\big)\Big) \approx G_{r,c}(\delta)
\end{aligned} \tag{5.7}$$

where $*$ denotes convolution. The last approximation assumes that filtering has a negligible influence on $G_{r,c}(\delta)$. This assumption is valid when the scene does not contain any spatial frequency at f_{FLK}. In practice, we begin by applying a logarithmic transformation to the pixel values and then perform a Fast Fourier Transform (FFT) on the image. Subsequently, we apply a series of digital filters, represented as $F(f_{\text{FLK}}) = F_1(f_{\text{FLK}}) * F_2(f_{\text{FLK}}) * \ldots * F_K(f_{\text{FLK}})$. Each filter $F_k(f_{\text{FLK}})$ specifically targets and attenuates the k^{th}-order flicker harmonic using a 4th order bandstop Bessel filter. These filters operate within a critical bandstop range $[kf_{\text{FLK}} - \varepsilon, kf_{\text{FLK}} + \varepsilon]$, where $K = 8$ and $\varepsilon = 8$ Hz, to effectively isolate and eliminate undesired flicker frequencies from the image. Then, we apply an exponential transformation to the filtered components to obtain $G_{r,c}(\delta)$.

Finally, the flicker signal denoted as $\widetilde{\omega}(\delta, r)$ is estimated as follows:

$$\widetilde{\omega}(\delta, r) = \frac{1}{C} \sum_{c=0}^{C-1} \frac{I_{r,c}(\delta)}{\widetilde{G}_{r,c}(\delta)} \tag{5.8}$$

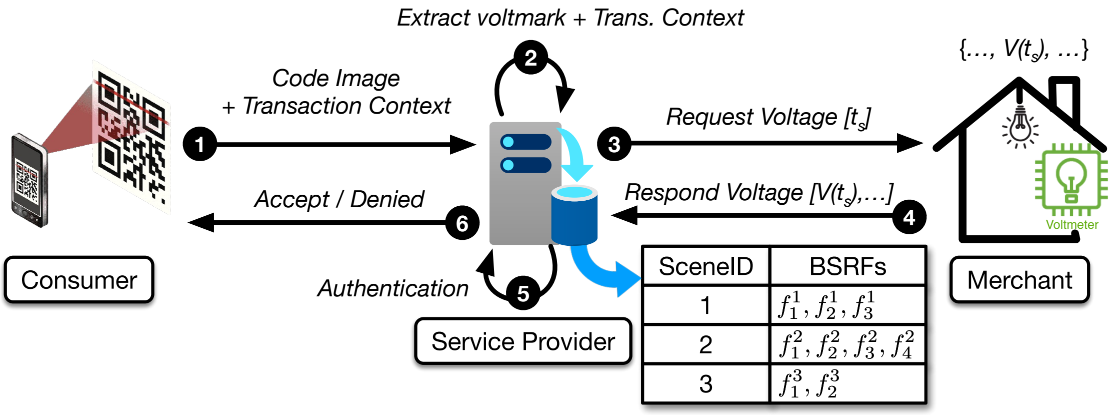
This signal is smoothed by averaging the values across the corresponding row, taking into account the characteristics of the rolling shutter. The complete flicker signal is given by:

$$\widetilde{\omega}(\delta) = \Big[\widetilde{\omega}(\delta, 0), \widetilde{\omega}(\delta, 1), \ldots, \widetilde{\omega}(\delta, R-1)\Big] \tag{5.9}$$

where $\widetilde{\omega}(\delta)$ denotes the observed flicker signal extracted across the entire image.

5.2.4 Verification

To validate our analysis, we conducted an experiment using an iPhone 8 to capture an image under lighting provided by a bulb operating at a 50Hz AC supply. The results are displayed in Fig. 5.4. Specifically, Fig. 5.4(a) showcases the original image along with images that have been processed to accentuate both flicker-irrelevant and flicker-relevant components after filtering. To enhance the visibility of the flicker, the intensity values of the flicker-relevant image were normalized, thereby making the flicker more noticeable. The approximation and filtering processes partially influence the flicker-irrelevant image. Fig. 5.4(b) illustrates the full flicker signal extracted from the image, including a plot of the smoothed signal and 50 individual flickers sampled every 100 columns in the red channel. This experiment effectively confirms our mathematical model and demonstrates the practicality of extracting AC flicker from images.



SceneID	BSRFs
1	f_1^1, f_2^1, f_3^1
2	$f_1^2, f_2^2, f_3^2, f_4^2$
3	f_1^3, f_2^3

Fig. 5.5: Verification Workflow. The service provider confirms the authenticity of the optical code scans in the specified environment at the claimed time in collaboration with the merchants.

5.3 Design of Voltmark

In this section, we elaborate on the design of Voltmark and the corresponding verification protocol.

5.3.1 Insight

An AC flicker ω is shaped by seven critical parameters beyond the color band. These parameters encapsulate AC information, spatial-temporal context, and camera settings as follows:

$$\omega \sim (\underbrace{f_{\text{AC}}, V_{\max}, \phi_0,}_{\text{AC Information}} \overbrace{h(\cdot), t_s}^{\text{Spatial-Temporal Context}}, \underbrace{\Delta T_{\text{RS}}, \Delta T_{\text{EXP}}}_{\text{Camera Settings}})$$

With known AC information and camera settings, the spatial-temporal context can be inferred from the flicker signal. Particularly, ω is intricately linked to the bulb's BSRF, $h(\cdot)$, which acts as a distinctive physical-layer identifier for different bulb types. Utilizing a comprehensive database correlating BSRF profiles with specific environments allows us to ascertain whether a scan of optical code was taken in

Table 5.1: Details of Tested Bulbs Available on the Market

#	**Type**	**Brand**	**Power**	#	**Type**	**Brand**	**Power**
1	CFL	OSRAM DULUX S	7W	16	INC	OSRAM	60W
2		OSRAM	11W	17		TUNGSRAM L5 LM	100W
3		ELECTRA MINI STAR 6500K	23W	18		ONOUR	100W
4		PHILIPS PL E-T	23W	19		LUXLIGHT 17/6	100W
5		LEELITE 6500K	11W	20		LUXLIGHT 17/6	100W
6		NEPTON Daylight 1022lm	20W	21		HATCHI	100W
7		HYUNDAI ECO T2 2700K	15W	22	HPS	ORSAM VIALOX	70W
8		NEPTON Daylight 1022lm	20W	23	LED	LEELITE 3000K	6W
9		LEELITE 6400K	11W	24		SAYNET 6500K	3W
10		PHILIPS MASTER PL-C 1	18W	25		SEMICON	17W
11	FL	LUXLIGHT L18 W/765	18W	26		ELECTRA LED 530	12W
12		LUXLIGHT L18 W/765	18W	27		SEMICON	18W
13		LUXLIGHT L18 W/765	18W	28		FSL	11W
14		LUXLIGHT L18 W/765	18W	29		EUROLUX 6000K	30W
15	HAL	SYLVANIA E06D	42W				

a specific location by examining the BSRF's unique signature. Furthermore, ω is influenced by t_s, the precise moment the image was captured, adding a temporal layer to our analysis.

Inspired by this insight, we define the AC flicker as a spatial-temporal watermark, which we refer to as *Voltmark*. This *Voltmark* not only confirms that an image was taken in a particular environment under specific bulb illumination but also pinpoints the exact moment of the capture. *Our method involves authenticating the spatial-temporal context by comparing the extracted Voltmarks with the genuine version, which is calculated using known BSRFs and AC voltage values that can be measured by an IoT device present at the location of interest.*

5.3.2 Verification Protocol

We detail the protocol to verify the spatial-temporal context of a QR code within a business ecosystem that involves three roles, which are interconnected via a secure Internet:

- **Merchant**: Equipped with IoT devices, merchants monitor and record the AC voltage in their establishments, where multiple bulbs with known BSRFs are installed. Meanwhile, merchants must register their BSRFs in the provider's database beforehand. Merchants deploy static optical codes within their premises to facilitate business transactions like ordering or payments. Additionally, they respond to queries from service providers about the measured AC voltage values.

- **Consumer**: Consumers participate by scanning static QR codes, which are illuminated by bulbs within the merchant's scene. This action initiates a transaction request that is transmitted to the service provider for further validation and processing.

- **Service Provider**: The service provider (such as PayPal or Alipay) assesses each transaction request. Before approving transactions, the service provider checks the spatial-temporal context of each request. Transactions that occur within the merchant's pre-registered scene and within the specified timeframe are processed and approved. Those that do not meet these criteria are rejected.

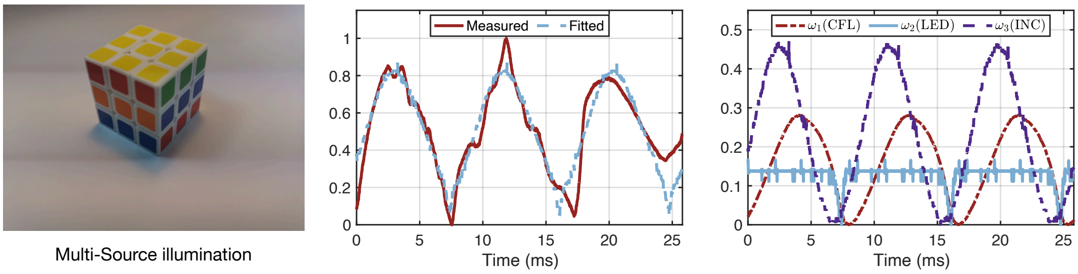
The interactions among the three roles are depicted in Fig. 5.5 and involve the following six steps:

❶ **Consumer Interaction**: Consumers scan QR codes that include transaction details such as scene ID, merchant ID, and service URL. The consumer's device captures the original image of the QR code and forwards it, along with the transaction request, to the service provider.

❷ **Data Extraction:** Upon receiving the transaction request, the service provider extracts the Voltmark ($\tilde{\omega}$) from the image, the claimed transaction timestamp (t_s), camera settings (ΔT_{EXP}, ΔT_{RS}, R), and the BSRFs relevant to the scene ID from its database.

❸ **Voltage Request** and ❹ **Response**: The service provider requests voltage data from the merchant for the duration from t_s to $t_s + (R-1)\Delta T_{\text{RS}}$. The merchant responds with the corresponding voltage values, namely $\{V(t_s), V(t_s + \Delta T_{\text{RS}}), V(t_s + 2\Delta T_{\text{RS}}), \ldots, V(t_s + (R-1)\Delta T_{\text{RS}})\}$.

❺ **Voltmark Prediction:** In a simplified scenario with a single bulb, the service



(a) Original image (b) Voltmarks (c) Decomposed Voltmarks

Fig. 5.6: Illustration of Multiple Lighting Sources. (a) shows the original scene image illuminated by three different types of bulbs—CFL, LED, and INC—positioned on the left, top, and right sides, respectively. (b) compares the measured Voltmark with the optimized combination of Voltmarks fitted from the BSRFs of the three bulbs. (c) shows the individual Voltmarks decomposed from each of the three bulbs.

provider retrieves the BSRF $h(\delta, V(t))$ associated with the scene ID from the database to predict the authentic Voltmark as follows:

$$\omega(\delta, r) = \frac{1}{\Delta T_{\mathrm{EXP}}} \int_{t_s + r\Delta T_{\mathrm{RS}}}^{t_s + r\Delta T_{\mathrm{RS}} + \Delta T_{\mathrm{EXP}}} h(\delta, V(t))dt$$

where r progresses from 0 to $R-1$, using the voltage values $V(t)$ provided by the merchant.

❻ Authentication and Approval: The distance between the measured and predicted Voltmarks is calculated:

$$d = \|\omega(\delta) - \widetilde{\omega}(\delta)\|_2 \tag{5.10}$$

If the distance falls below a set threshold d_{auth}, this confirms the code image was taken within the designated scene at the claimed time t_s, leading to transaction approval; if not, the transaction is denied.

Verification Correctness. This verification process ensures that only legitimate and timely requests are processed, supported by two critical aspects of verification correctness. First, the BSRFs are unique for each bulb, converting input AC voltage

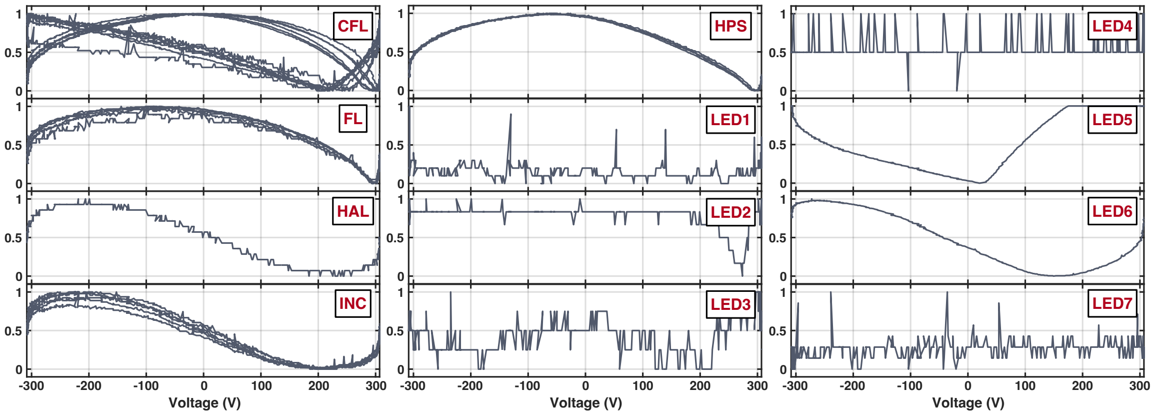


Fig. 5.7: Normalized BSRFs of 29 bulbs. Different bulbs show different spectra responses due to the hardware discrepancy.

into Voltmarks. This uniqueness prevents attackers from replicating Voltmarks using their own bulbs, thereby ensuring the accuracy of the spatial context. Second, a Voltmark reflects a segment of the actual voltage series indirectly. Any misalignment in timing results in significant discrepancies in the Voltmarks, which secures the temporal context. Consequently, both the location and the time claimed can be verified against the Voltmark, enhancing the spatial-temporal traceability of transactions initiated via QR codes. Next, we will elaborate on the protocol details.

5.3.3 Account for BSRF Uniqueness

Previous work [122] has demonstrated that BSRFs could act as distinct hardware fingerprints, varying significantly across different light bulbs. This distinctiveness is primarily due to variations in hardware, such as the different materials used within incandescent bulbs—from gases to phosphorescent substances—which can vary not only between models but also among individual bulbs. To verify the uniqueness, we measured the BSRFs of 29 different bulbs from the market. The details of these bulbs are in Table 5.1. These bulbs were selected for their variety of fabrication techniques, brands, and power ratings. The measurement results are shown in Fig. 5.7. From the figures, we have three key findings.

First, as anticipated, these bulbs show varied waveforms in response to standard voltage ranges between -312 and 312 volts, affirming their uniqueness. Such variability in BSRFs leads to distinctive Voltmarks in images, aiding in the identification of various light sources (or equivalent locations). Second, the BSRF characterizes the conversion of voltage into light flux, which ideally peaks when the input voltage reaches its highest absolute amplitude. However, for non-LED bulbs, peak brightness often occurs at voltages below 300 volts due to the time required to heat their filaments to optimal luminosity. Third, the BSRFs of LED bulbs are generally smoother compared to those of other types because LEDs convert AC to DC before emitting light. Despite this smoother characteristic, noticeable spikes are still present in the BSRF data. These anomalies are likely due to physical variations in the diodes that influence electron flow through the LEDs. This observation underlines the feasibility of applying Voltmarks not just to traditional incandescent bulbs but also to LED bulbs.

5.3.4 Account for Voltage Uncertainty

Fig. 5.7 also suggests that the BSRFs of bulbs produced in the same batch display slight differences due to their similar material compositions. This resemblance marginally increases the risk of cloning attacks, where an attacker could attempt to replicate Voltmarks using bulbs of the same model. Nonetheless, it is crucial to understand that the security of Voltmark does not rely solely on the uniqueness of each bulb's model but also on the actual voltage values used. Even if an attacker knows the bulb model, accurately duplicating Voltmarks is challenging without direct access to the specific voltages at the scene, as the resultant Voltmarks are dependent on these input voltages. It is presumed that proactive security measures are in place to prevent attackers from physically measuring or altering the voltage at the installation site.

One might wonder if an attacker could predict the AC voltage after knowing its value at a specific time, especially considering it behaves as an approximately perfect sinusoidal function (see Eqn. 5.1). Unfortunately, such prediction remains infeasible due to two primary sources of uncertainty [128]:

- **Voltage Uncertainty**. The AC voltage typically fluctuates within a 10% range. In environments like restaurants, which utilize numerous high-power electrical appliances, these fluctuations can be even more pronounced. The irregular use of such appliances adds complexity to the prediction of voltage levels, making it particularly challenging.
- **Frequency Uncertainty**. The mechanical structure of electric generators introduces slight imprecision in rotor rotation, leading to frequency fluctuations within 1%. This variability accumulates over time, significantly compounding errors in voltage prediction.

Thus, the inherent complexities in forecasting electrical behaviors from static measurements reinforce the security model against potential attacks by unauthorized entities. It is worth noting that our focus is on general settings such as restaurants, hotels, and shopping malls, where voltage stability is less demanded. Certainly, proactive measures can be implemented to intentionally vary the voltage of bulbs locally and randomly if necessary [129].

5.3.5 Account for Camera Settings

When verifying Voltmarks, it is essential to obtain two critical camera parameters: the exposure time (ΔT_{EXP}) and the rolling delay (ΔT_{RS}).

• **Exposure Time**. This parameter can be readily found in the image's metadata, which is usually stored in the EXIF data. This metadata includes various details such as the timestamp, smartphone model, exposure time, aperture, ISO setting, and more. Exposure time, specifically, refers to how long the camera's sensor is exposed to light, typically ranging from 1/10 to 1/4000 seconds [130]. Commonly, the default value is set to a multiple of $1/2f_{\text{AC}}$ to balance the light accumulation across each row, minimizing visible flickering. Nevertheless, given the potential variability in AC frequency, Voltmarks are still detectable under these conditions. Surely, given that the verification protocol requires collaboration with scanning apps (i.e., transmitting the image to a service provider), it is reasonable to request that they set the exposure time to a value that is not a multiple of $1/2f_{\text{AC}}$ to enhance both detectability and visual discretion.

- **Shutter Delay**. This parameter, absent from the metadata, represents the consistent time lag in exposure across consecutive rows and is strongly dependent on the smartphone model. By referencing the model number found in the metadata, one can consult a well-documented database to determine the specific delay value of that model, as detailed in sources like [131].

5.3.6 Account for Multiple Lighting Sources

In many scenarios, scenes are lit by multiple bulbs, each imparting its distinct Voltmarks onto the image. Consequently, the final Voltmark emerges as a linear weighted combination of these individual Voltmarks [123]. Recall that each bulb has its unique BSRF, resulting in different waveform characteristics for the Voltmarks generated by multiple bulbs. We can utilize the distinctive properties of each BSRF to decompose these Voltmarks. Suppose there are K bulbs in the scene, with BSRFs denoted by $h_1(\delta, V), h_2(\delta, V), \ldots, h_K(\delta, V)$. The problem then reduces to solving the following optimization:

$$(w_1, w_2, \ldots, w_K) = \underset{w_k \in [0,1]}{\operatorname{argmin}} \parallel \left(\sum_{k=1}^{K} w_k \omega_k(\delta) \right) - \widetilde{\omega}(\delta) \parallel_2 \tag{5.11}$$

where

$$\omega_k(\delta, r) = \frac{1}{\Delta T_{\mathrm{EXP}}} \int_{t_s + r\Delta T_{\mathrm{RS}}}^{t_s + r\Delta T_{\mathrm{RS}} + \Delta T_{\mathrm{EXP}}} h_k(\delta, V(t))dt \tag{5.12}$$

for $r = 0, 1, \ldots, R - 1$.

To illustrate our method, we analyzed an image of a Rubik's cube illuminated by three distinct light sources located to the right, left, and top, as depicted in Fig.5.6(a). Using our approach, we effectively decomposed the Voltmark into a linear combination of individual contributions from each of the three bulbs, as shown in Fig.5.6(b) and (c). The uniqueness of each bulb's BSRF enables this accurate decomposition. It is worth noting that multi-source illumination also boosts security. This approach increases the difficulty of executing clone attacks since it is more challenging for an attacker to source multiple bulbs with similar BSRFs that mimic the exact lighting conditions.

5.4 Implementation of Voltmark

This section delves into the practical implementation details of Voltmark, focusing on its capabilities for spatial-temporal traceability of optical codes.

5.4.1 Augmented Extraction

Our previous method relies on the known operating band of AC to filter Voltmarks from images, but this approach encountered several limitations. Firstly, the presence of regular patterns or stripes in an image (e.g., barcodes) might be mistakenly identified as components of the Voltmark if they match the spatial frequency f_{FLK}. Secondly, smartphones often default to exposure time that approximate integral multiples of f_{FLK} to minimize flickers, making them less noticeable to the human eye. This subtlety in flickers complicates the process of accurately extracting Voltmarks.

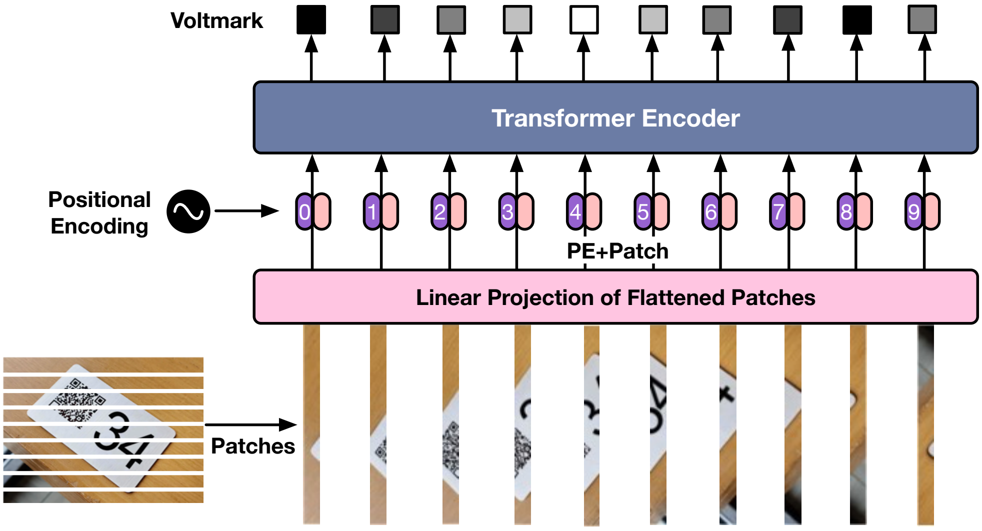


Fig. 5.8: Vision in Transformer. We adopt the ViT for Voltmark extraction from images.

ViT. To overcome these challenges, we propose utilizing a deep learning-based approach, leveraging its proven capabilities in image processing and pattern recognition. Specifically, we employ the Vision Transformer (ViT) architecture to enhance the precision of Voltmark extraction, as illustrated in Fig. 5.8. In our implementation, we adopt the $R \times C$ pixel image of QR code by default. Each image is first divided into N patches (e.g., fixed-sized sub-region of the image). Each patch encompasses R/N rows of pixels and spans the entire column width, thereby including $R/N \times C$ pixels. The divided N patches are treated as individual tokens and further processed by the ViT model. We adopt the variant model called ViT-Base [132]. Specifically, these patches are flattened into 768-dimensional vectors and then undergo positional encoding to retain index information. They are further fed into the model, accommodating 12 blocks. Within each block, self-attention mechanisms with 12 heads compute attention among tokens or patches to detect underlying patterns. Since ViT maintains the dimensionality of the input, the model outputs N points, each of which is treated as the point of the Voltmark sampling at $t_s + N\Delta T_{\mathrm{RS}}$. Here, we adopt $R = 1920$, $C = 1080$, and $N = 240$.

Training. To effectively train our model, it is crucial to obtain a substantial number

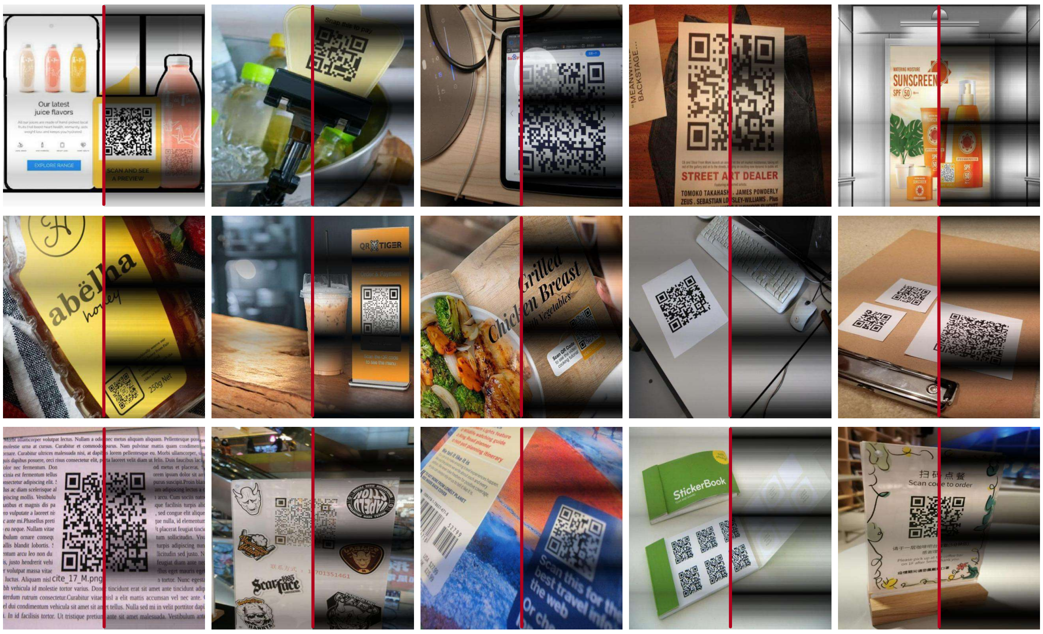


Fig. 5.9: Augmented Datasets. This extensive dataset was created by overlaying known flickers onto publicly available barcode datasets. For illustrative purposes, each image above is divided by a red vertical line; the synthesized image appears on the left, and the extracted Voltmark is shown on the right.

of training samples. We start by extracting typical BSRFs from 29 popular bulbs in the market (see Table 5.1). We then randomly choose K BSRFs (e.g., $K = 1 \sim 4$) and segments of AC voltage values to create synthesized Voltmarks. These are integrated into images from publicly available optical code datasets [133]. Some examples are illustrated in Fig. 5.9. Our dataset comprises 100,000 synthesized images derived from an original collection of 2,438 optical code images. We allocate 80% of these images for training and 20% for testing. The primary goal of this training process is to minimize the discrepancies between the Voltmarks generated and those extracted by ViT. The Adam optimizer [101] is employed for this process. The initial learning rate is set at $5e^{-4}$, which is progressively reduced to $3e^{-5}$ through exponential decay over the optimization period. Typically, training the network for a single scene requires approximately 200 to 300 epochs to achieve convergence using a single NVIDIA 4090 GPU, which translates to a duration of about two hours.

5.4.2 Augmented BSRF Modeling

To verify the spatial-temporal context of a Voltmark, it is essential to pre-measure the BSRFs $h(\delta, V(t))$ for each bulb. Traditionally, this requires specialized equipment like integrating spheres or high-speed spectrometers, complicating deployment. To simplify this, we adopt the Kolmogorov-Arnold Network (KAN) – a neural network architecture inspired by the Kolmogorov-Arnold representation theorem – to approximate a BSRF function [134]. Unlike MLPs with static activation functions at nodes, KAN features learnable activation functions on the connections between nodes, allowing for greater adaptability and flexibility, which is particularly suited for modeling nuanced functions or curves like BSRF.

The tailored KAN architecture is shown in Fig. 5.10. This three-layer KAN structure reformulates the original $h(\delta, V(t))$ to include two variables, namely V and t. For a given voltage V at time t, the KAN outputs the corresponding light intensity from the bulb. The hidden layer comprises five nodes, with each edge utilizing a B-Spline function as the activation function $\phi(x)$. The network operation can be described by the equation:

$$h(V(t)) = (\Phi_2 \circ \Phi_2)(V, t) \tag{5.13}$$

For further details on KAN, please refer to [134].

We choose KAN instead of MLP for modeling BSRF functions because of KAN's unique benefits. MLPs generally need more complex network structures to manage complex functions such as exponential or trigonometric equations, whereas KAN can tailor its activation functions to fit these specific mathematical forms directly. This adaptability allows for more accurate modeling of BSRFs using simpler network structures and fewer neurons, greatly improving efficiency and reducing the computational load that is often associated with the elaborate network configurations needed for precise MLP models.

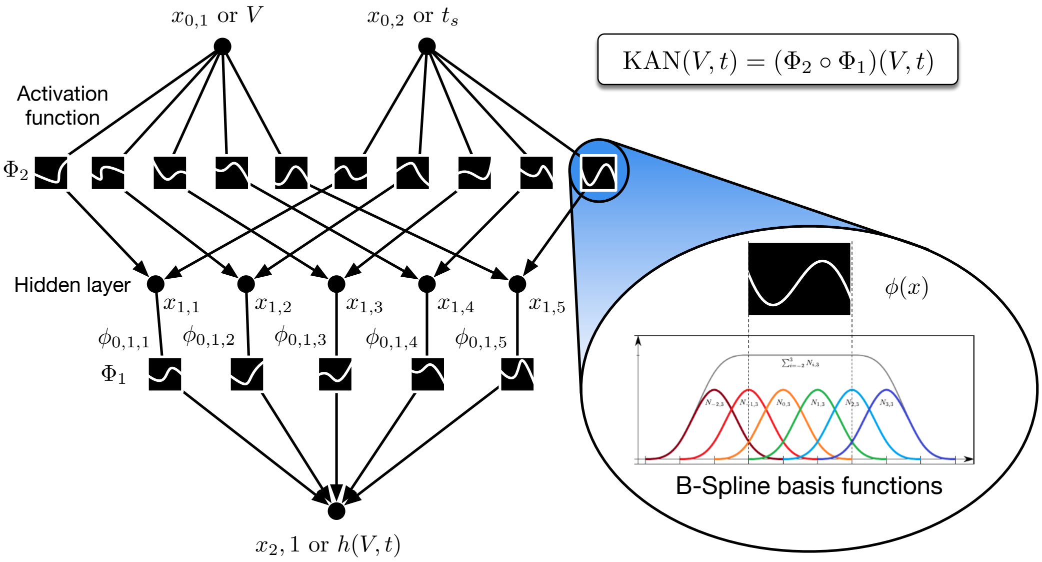


Fig. 5.10: Architecture of the Kolmogorov–Arnold Network. A KAN consists of two main layers. The first layer applies a non-linear transformation to each input variable independently, while the second layer combines these transformed outputs through a sum to approximate the target function.

Training. Employing the methodology described in [123], we collect BSRF samples using a standard smartphone. To isolate the bulb's light intensity from background interference, we positioned a ground glass over the camera lens of the smartphone, which was aimed at the bulb. This setup diffused the light from the bulb, effectively removing scene-dependent spatial variations and ensuring that the captured image primarily represented the BSRF curve with high fidelity. We then calculated the average light intensity for each row along with the corresponding voltage measurements taken by an IoT device, creating a dataset focused on these values. The goal of the training is to reduce the discrepancy between the predicted and actual light intensity for each pixel row. For this, the KAN grid is configured with 40 nodes, each layer having a width of 5, and the interpolation uses a cubic spline. Training employs the Adam optimizer [101], with a learning rate set at $1e^{-2}$. Typically, it takes around 4000 steps or about two minutes on a single NVIDIA 4090 GPU to train the KAN to convergence for a single BSRF.

Verification. When performing the verification, we first find the optimal combi-

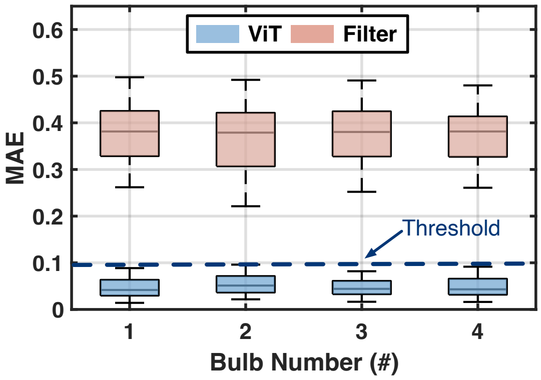


Fig. 5.11: Extraction Accuracy

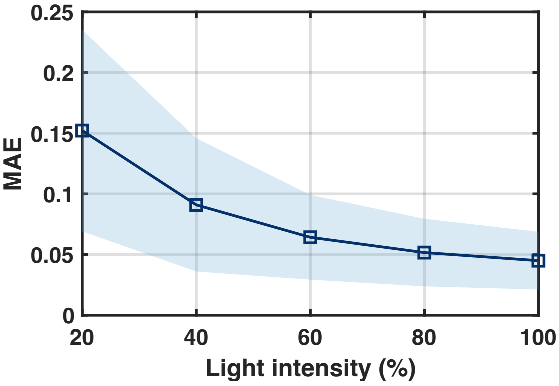


Fig. 5.12: Impact of Light

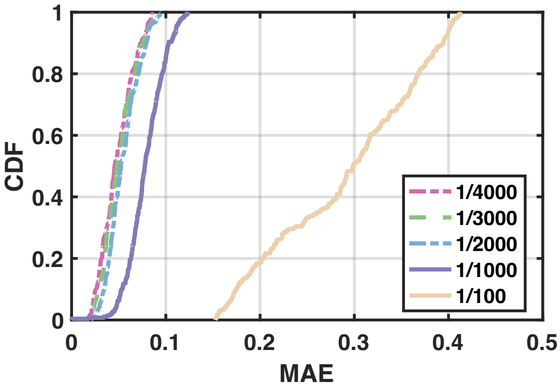


Fig. 5.13: Impact of Exposure

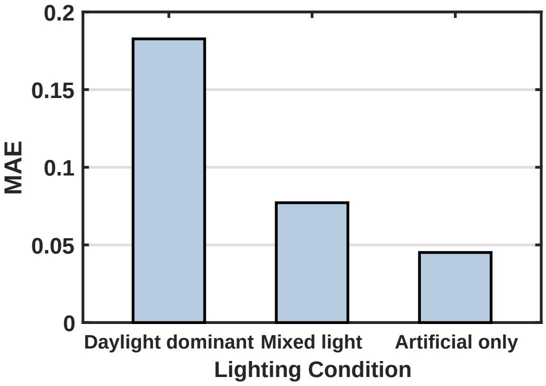


Fig. 5.14: Impact of Daylight

nation of the K Voltmarks generated by the K BSRF functions to approximate the claimed Voltmark $\widetilde{\omega}$. The search is formally expressed as follows:

$$\underset{w_k \in [0,1]}{\text{argmin}} \parallel \left(\sum_{k=1}^{K} w_k \omega_k(t_s) \right) - \widetilde{\omega}(t_s) \parallel \tag{5.14}$$

Since the optimization is a simple linear combination, we use the gradient descent algorithm directly to search for the optimal solution without using neural networks.

5.5 Evaluation

We assess the effectiveness of Voltmark, focusing on its capability to authenticate legitimate entities and its robustness against a range of security threats. Our experiments utilize two lamps fitted with 29 distinct types of bulbs (as listed in Table 5.1) to create Voltmarks in a lab environment. To capture AC signals, we employ a current transformer to monitor power line currents, connected to a DAQ system [135, 136] for voltage measurement. Additionally, we use six different smartphone models to record static codes embedded with Voltmarks, including three Android models (Google Pixel 7 Pro, HUAWEI Mate 20, and Honor 8X Max), and three iPhone models (iPhone 8, iPhone 12, and iPhone 13). By default, the iPhone 13 is utilized with $\Delta T_{\text{EXP}} = 1/4000$, maintaining a distance of 50cm from the optical codes.

5.5.1 Performance of Voltmark Extraction

Our primary objective is to effectively extract Voltmark from images containing optical codes. Initially, we evaluate the ViT-based model designed for this purpose.

(1) Accuracy: To gauge the precision of our extraction model, we calculate the Mean Absolute Error (MAE), which quantifies the average discrepancies between the Voltmarks extracted by the model and the actual ground truth, denoted as $\frac{1}{n}\sum_{i=1}^{n}||\omega_i(\delta) - \tilde{\omega}_i(\delta)||$. This assessment is performed on 20% of the synthesized optical codes previously mentioned. We compare these results against a traditional filtering algorithm (detailed in §5.2) as a baseline. The computed MAEs for varying bulb counts are depicted in Fig. 5.11.

From the figure, we observe several key findings: Firstly, the ViT model records mean absolute errors (MAEs) of 0.049, 0.060, 0.054, and 0.055 across four different cases, with the deviations in the extracted Voltmarks being less than 1%. Based on these results, we recommend an authentication threshold of $d_{\text{auth}} = 0.1$ (see Eqn. 5.10). Sec-

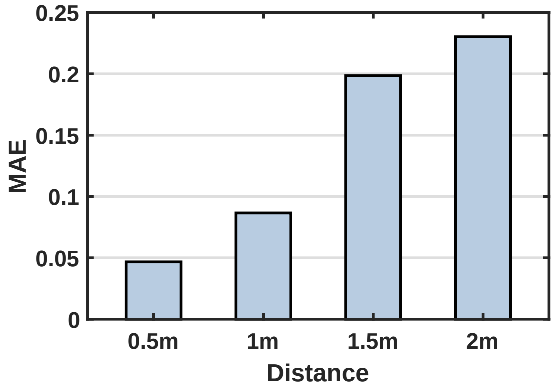


Fig. 5.15: Impact of Distance

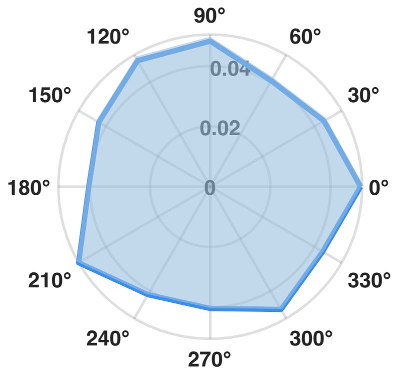


Fig. 5.16: Impact of Directionality

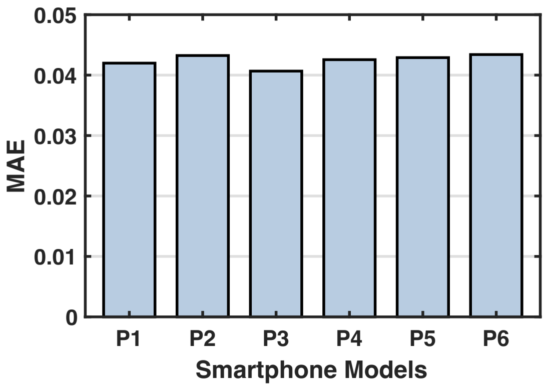
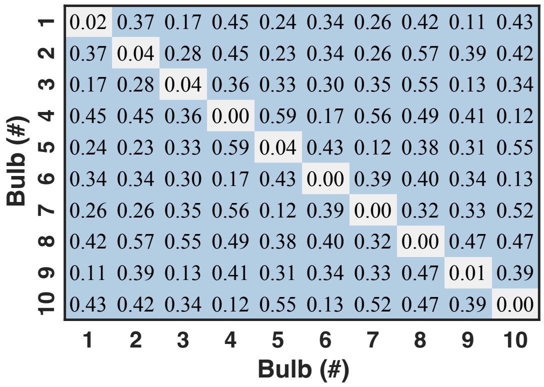


Fig. 5.17: Impact of smartphones



Bulb (#)	1	2	3	4	5	6	7	8	9	10
1	0.02	0.37	0.17	0.45	0.24	0.34	0.26	0.42	0.11	0.43
2	0.37	0.04	0.28	0.45	0.23	0.34	0.26	0.57	0.39	0.42
3	0.17	0.28	0.04	0.36	0.33	0.30	0.35	0.55	0.13	0.34
4	0.45	0.45	0.36	0.00	0.59	0.17	0.56	0.49	0.41	0.12
5	0.24	0.23	0.33	0.59	0.04	0.43	0.12	0.38	0.31	0.55
6	0.34	0.34	0.30	0.17	0.43	0.00	0.39	0.40	0.34	0.13
7	0.26	0.26	0.35	0.56	0.12	0.39	0.00	0.32	0.33	0.52
8	0.42	0.57	0.55	0.49	0.38	0.40	0.32	0.00	0.47	0.47
9	0.11	0.39	0.13	0.41	0.31	0.34	0.33	0.47	0.01	0.39
10	0.43	0.42	0.34	0.12	0.55	0.13	0.52	0.47	0.39	0.00

Fig. 5.18: BSRF Uniqueness

ondly, the ViT model consistently outperforms the baseline filtering method, showing improvements of 32.76%, 29.88%, 31.89%, and 31.49% in MAE across different tests. This enhancement is attributed to the filtering algorithm's vulnerability to disturbances from patterns and background stripes in optical codes, which may mimic the frequency components of Voltmark. In contrast, the ViT model's self-attention mechanism adeptly reduces such disturbances. Lastly, scenarios involving optical codes illuminated by multiple bulbs result in Voltmarks that are linear combinations of contributions from each bulb. This complexity is addressed as an optimization challenge (refer to Eqn. 5.14). Notably, both evaluated methods capably manage this complexity, regardless of the number of bulbs involved.

(2) Light Intensity: The variability in light intensity, influenced by the distance or angle between the camera and optical codes, impacts the MAE. To examine this effect, we selected five different bulbs (#4, #12, #20, #25, and #29) to illuminate a static QR code in our laboratory setting. We used a voltage regulator to adjust the input voltage to these bulbs, with the light intensity measured by a photodiode. We defined the standard light intensity-achieved with a 220V input-as 100% intensity and scaled it down to 80%, 60%, 40%, and 20% in our tests. During these experiments, the distance between the smartphones and bulbs was maintained at 50cm. Fig. 5.12 displays the results. As expected, the average MAE across the bulbs increased from 0.045 to 0.152 as light intensity decreased from 100% to 20% due to the diminished light flux reaching the smartphone's camera. Concurrently, the standard deviation increased from 0.023 to 0.083. Notably, at the lowest light level of 20%, the QR code recognition rate also drastically dropped to 10%, reflecting the suboptimal lighting conditions. These experimental results suggest the potential need for an adaptive authentication threshold based on the overall image brightness in practical applications.

(3) Exposure Time: Exposure time (i.e., shutter speed) is another factor affecting the extraction of Voltmarks. To assess its influence, we adjusted the shutter speed of the smartphone camera to 1/4000, 1/3000, 1/2000, 1/1000, and 1/100 seconds. The outcomes are illustrated in Fig. 5.13. As the exposure time increases from 1/4000 to 1/1000 seconds, the median MAE marginally increases from 0.045 to 0.077. This slight rise can be attributed to Eqn. 5.5, which uses average pixel intensity to approximate the light sampling at the midpoint of the exposure window. A longer exposure time typically captures a smaller proportion of light from the non-overlapping windows, resulting in less distinction in pixel intensity between adjacent rows. In addition, an exception is observed when the exposure is extended to 1/100 seconds. There is a significant spike in MAE to 0.2987. This spike occurs because the flicker frequency is about 100 Hz, matching our lab's 50Hz AC setting, causing nearly all rows to capture a complete exposure cycle and leading to minimal fluctuation between adjacent rows.

(4) Natural Light: We evaluated Voltmark under three lighting conditions: *Daylight-*

dominant, where all curtains were fully open under direct sunlight; *Mixed light*, with curtains half-open allowing partial daylight; and *Artificial-only*, with all curtains closed. As shown in Fig. 5.14, the extraction error increases with stronger natural light, from 0.0451 in artificial-only to 0.183 under full daylight. This is because daylight can overwhelm the AC-induced modulation from indoor lights, reducing the SNR of Voltmark. Voltmark is therefore best suited for indoor or semi-outdoor environments where artificial lighting is present.

(5) Spatial Resolution: To assess the spatial resolution of Voltmark, we conducted an experiment where the distance between two cellphones was set from 0.5m to 2m. As shown in Fig. 5.15, the MAE increases significantly with distance, indicating that Voltmark exhibits measurable variation across space. Notably, when the difference between 0.5m and 1m, the MAE remains less than 0.1, suggesting that an attacker positioned very close to the target QR code may generate a similar Voltmark. Based on this analysis, we recommend a minimum spatial separation of 1m to ensure reliable differentiation. This requirement aligns well with typical restaurant seating layouts, where adjacent tables are often spaced at least 1.0 to 1.2 meters apart.

(6) Directionality: We assessed the effect of camera orientation on Voltmark extraction by positioning the smartphone at 12 different angles, from 0° to 330° in 30° increments. The findings, depicted in Fig. 5.16, show a minimal variation in median MAE, with a low of 0.04 and a high of 0.05. This consistency suggests that Voltmark robustly captures AC flickers, regardless of the camera's orientation, highlighting its reliability for diverse application settings.

(7) Smartphones: Finally, we tested the extraction accuracy using six different smartphone models, maintaining a 50 cm distance between the smartphones and the bulbs. The results are shown in Fig. 5.17. The MAE of 0.0425 ± 0.0027 is observed across the devices. This demonstrates Voltmark's versatility across various hardware platforms.

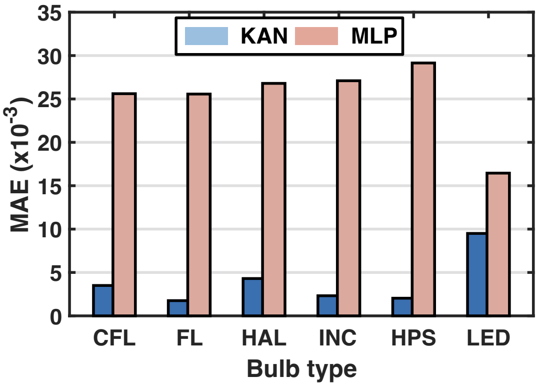


Fig. 5.19: Model Accuracy

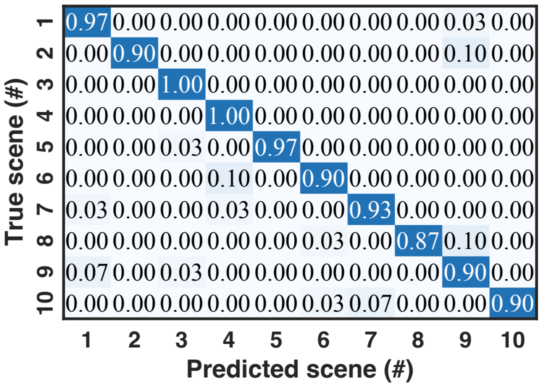


Fig. 5.20: Spatial Traceability

5.5.2 Performance of Voltmark Verification

Next, we evaluate the verification accuracy via Voltmarks from a complete system perspective.

(1) BSRF Uniqueness: The distinctiveness of bulb BSRFs significantly impacts the accuracy of the verification process. To evaluate this, we selected 1 ∼ 4 bulbs from the pool of 29 and established 10 different scenarios within the same environment. In each scenario, we measured the Voltmarks illuminated by these bulbs while varying the input voltage from -312 volts to 312 volts. We then assessed the similarity of the measured Voltmarks to gauge BSRF uniqueness. The results, displayed in a confusion matrix in Fig. 5.18, show a minimum MAE of 0.1115 and an average MAE of 0.3550, which is three times higher than the previously set threshold of 0.1. These results suggest that Voltmarks derived from different BSRF groups maintain unique characteristics, even under identical input voltages.

(2) BSRF Modeling: To model the BSRF, we implemented the KAN and assessed its modeling accuracy. We randomly selected six bulbs from each category for this purpose. Both a KAN and an MLP were used to model each BSRF. The MLP configuration included five layers with two hidden layers consisting of 128 and 256 nodes. These models were trained on 256 samples and then tested on another 300 samples. The resulting MAEs are illustrated in Fig. 5.19. The KAN and the MLP

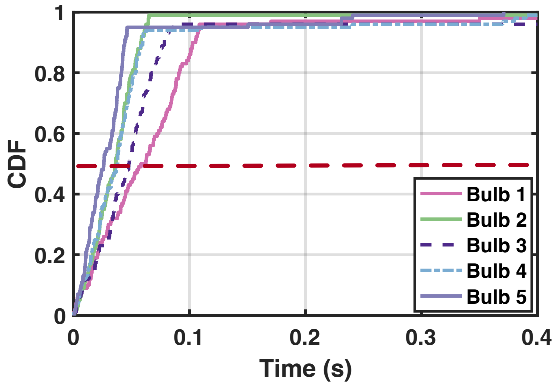


Fig. 5.21: Temporal Traceability

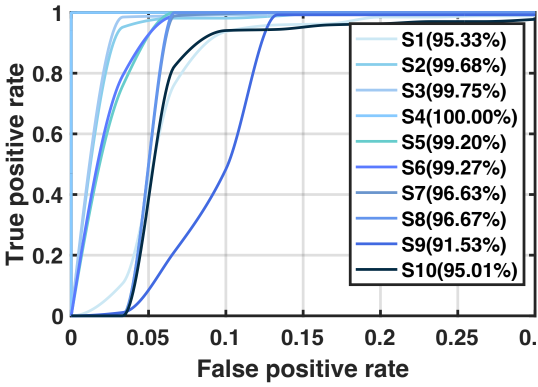


Fig. 5.22: Overall Traceability

achieved average MAEs of 0.004 and 0.0208, respectively. The KAN outperforms the MLP by a factor of five. This experiment confirms the KAN's superior capability in function fitting as claimed in [134], attributable to its inherent design principles.

(3) Spatial Traceability: To assess the spatial traceability of Voltmark, we deployed 1 to 4 bulbs with known BSRFs across ten real-life scenes, including three office rooms, two restaurants, two shopping malls, two classrooms, and one MTR station. In each scene, the QR code was scanned 100 times, resulting in a total of 1,000 Voltmarks extracted. Each Voltmark instance was used to classify the scan location among the ten scenes. To eliminate the impact of time, the measured input voltage during the scanning period was provided directly to compute the true Voltmarks. Fig. 5.20 shows a confusion matrix that details the classification accuracy for each scene. The average classification accuracy was 93.3%, with the lowest accuracy recorded at 87% in scene #8, where only one bulb was utilized. This supports prior observations that the distinctiveness of Voltmark is enhanced with the use of multiple bulbs. This experiment effectively illustrates the precision of using BSRFs for accurate location tracing.

(4) Temporal Traceability: To evaluate temporal traceability, we set up a single bulb with a known BSRF in a restaurant and conducted 100 random QR code scans between the operating hours (i.e., 8:00 AM and 9:00 PM). For each scan, we adjusted the time window to identify the moment that minimized the distance between the extracted Voltmark and the predicted one. The CDF of time discrepancies between the pinpointed and actual times is shown in Fig. 5.21. The findings indicate an average time drift of approximately 0.0417 seconds, enabling us to differentiate between two consecutive scans separated by 42 milliseconds, equivalent to a video frame rate of 24 frames per second. This capability of Voltmark to track scanning times capitalizes on the inherent fluctuations in the voltage and frequency of AC power, which, as noted earlier, impart a unique and subtle temporal marker to the Voltmarks over different cycles.

(5) Overall Accuracy: Lastly, we assessed whether Voltmark can accurately verify the spatial-temporal context embedded within the transaction context. For this purpose, we recaptured images of the QR code across the previously mentioned ten scenes (S1-S10). In each scene, 100 images were taken—half were accurately labeled with the correct location and scanning time, while the other half were wrongly labeled with location or time. We applied our verification protocol to ascertain the accuracy. The ROC curves and the corresponding Area Under the Curve (AUC) for different scenes are displayed in Fig. 5.22. Notably, Scene 4 achieved an AUC of 100%, indicating flawless classification and underscoring the system's superior performance in that setting. Scene 10 also performed well, achieving an AUC of 95.01%, whereas Scene 9 had the lowest AUC at 91.53%. The ROC curves demonstrate Voltmark's robust performance in distinguishing between accurate and inaccurate claims across various scenes. With most AUC values exceeding 95%, the system shows a high degree of reliability and effectiveness. This consistent performance across different scenes and lighting conditions underscores Voltmark's adaptability. The high AUC values reflect the system's ability to effectively reduce false positives and negatives, thereby ensuring secure and reliable user verification.

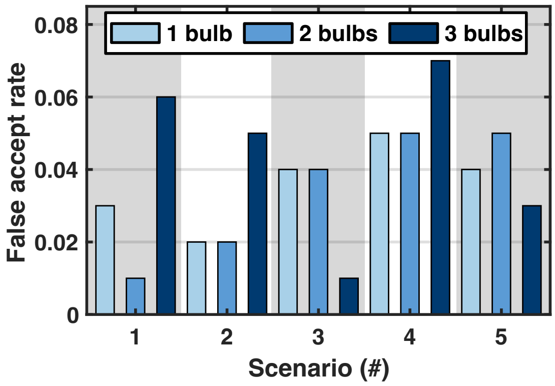


Fig. 5.23: Clone Attack

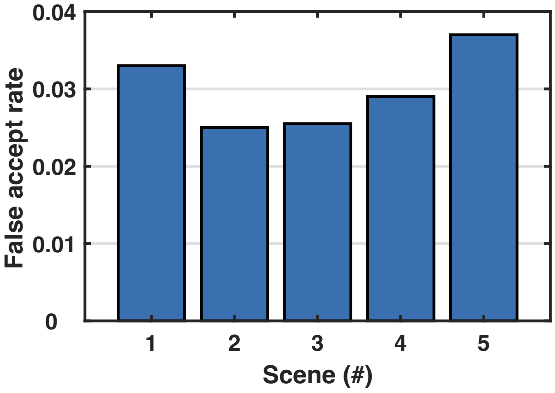


Fig. 5.24: Replay Attack

5.5.3 Resilience Against Attacks

Lastly, we now analyze the security of Voltmarks under various kinds of attacks.

(1) Clone Attack: An attacker might replicate the lighting settings by using bulbs of the same model to create similar Voltmarks alongside an identical QR code to that of the victim. We simulated this attack across five different scenes, varying the number of bulbs used. We calculated the false acceptance rate, which quantifies the frequency at which the clone attack goes undetected. Our system demonstrated a mean false acceptance rate of 3.8%, indicating that approximately 96.2% clone attacks will be successfully prevented.

(2) Replay Attack: An attacker might obtain a code image from a customer and later resubmit it to the service provider in an attempt to forge another transaction. We simulated this type of attack across five scenes. Since our system is designed to process each transaction with a unique timestamp only once, any subsequent transactions with the same timestamp should be automatically denied. The effectiveness of this approach is demonstrated by the false acceptance rate for these replay attacks, detailed in Fig. 5.23. Our system reported a low false acceptance rate of 5.98%.

(3) Replacement Attacks: In the event of a replacement attack, where an attacker redirects the acquisition of voltage data to their own server, the Voltmarks generated using these compromised values will differ from those calculated using the authentic, registered BSRFs. As a result, the discrepancy between the expected and actual Voltmarks leads to the automatic rejection of the transaction request.

(4) Cost Analysis: Implementing voltmarks is economically feasible, requiring only a single, low-cost IoT voltage meter (approximately 5 USD) connected to the merchant's electrical network to monitor voltage changes. This setup is significantly cheaper than dynamic codes, which require screens at each table.

5.6 Disscusion and Limitations

In this section, we discuss the limitations of Voltmark and the practical issues that may happen during the usage.

Global Shutter Cameras. Voltmark was originally designed for rolling shutter cameras, where each row of the sensor is exposed at a slightly different time. This row-wise exposure delay introduces a natural temporal sampling effect within a single frame, enabling the reconstruction of illumination flicker patterns. In contrast, global shutter cameras expose all pixels simultaneously, thereby eliminating the per-row temporal offset that Voltmark relies upon. As a result, the standard extraction method used by Voltmark is not directly applicable to a single global shutter image. Nevertheless, alternative approaches may partially compensate for this limitation. For instance, if a global shutter camera records a continuous video stream, the temporal variations in ambient light can still be captured across frames. In this setting, each frame acts as a uniform temporal sample of the light source's waveform, allowing waveform reconstruction over time.

Programmable Spoofing Attacks. While Voltmark is designed to resist cloning through the physical uniqueness of BSRF-modulated light flickers, an adversary might attempt to spoof Voltmarks using programmable light sources or image manipulation.

In theory, an attacker could employ finely controlled smart lighting to simulate the original flicker waveform at a spoofed location. However, such an attack requires reproducing not only the flicker frequency, but also its exact phase, waveform shape, and relative amplitude ratios, all of which are influenced by the specific geometry and reflectance properties of the original environment. Alternatively, an adversary might digitally manipulate images by synthesizing plausible flicker patterns. Yet, the temporal and spatial coherence enforced by Voltmark imposes strong constraints that make such fabrications difficult to execute without introducing detectable inconsistencies. In practice, we argue that replicating Voltmark without physical access to the original site remains a highly non-trivial challenge.

Mobile Inference and Deployment. Although Voltmark employs a transformer-based backbone, recent work demonstrates that real-time inference is feasible on mobile hardware. Models such as MobileViT [137] and EfficientFormer [138] have shown that quantized and pruned vision transformers can run efficiently on mobile devices, achieving inference latencies as low as 10–20 ms. These findings suggest that Voltmark can be deployed on mobile CPUs or NPUs, which can be found in smartphones, without introducing significant latency.

5.7 Conclusion

We present Voltmark, an innovative watermarking approach that capitalizes on the inherent characteristics of AC flickering to authenticate and secure static optical codes. By harnessing the unique spatial-temporal patterns of AC flicker signals, Voltmark enhances the traceability and security of optical codes without the need for additional hardware.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This dissertation has made contributions to the field of smart reconfigurable environments by focusing on the development and application of both radio frequency (RF) and optical signals. The research is structured into two primary segments: the evolution of smart reconfigurable environment technology and its practical applications.

Firstly, the dissertation introduces a scalable and flexible Reconfigurable Intelligent Surface (RIS). By utilizing an array of RFID tags to create a large-scale RIS, each RFID tag functions as an independent unit cell wirelessly controlled by a remote RFID reader. This design significantly improves upon traditional RIS systems, which are often constrained by wired connections that limit scalability and flexibility. Additionally, a specialized neural radiance field for RIS has been developed to swiftly and accurately estimate channel variations, thereby enhancing the overall efficacy and responsiveness of the wireless system.

Secondly, we proposed the concept of Radio Frequency Neural Networks (RFNNs), which are designed to execute neural network computations directly on RF signals.

This approach addresses the substantial power and computational demands of Electronic Neural Networks (ENNs), which render them impractical for use in cost-effective and energy-efficient sensor nodes. By employing full-forward propagation and contrastive learning techniques, RFNNs are capable of autonomously learning from data, thereby enhancing their adaptability and accuracy in recognizing unknown RSS labels.

Furthermore, the dissertation explores the enhancement of temporal-spatial traceability in optical wireless communication systems, particularly for applications involving sensitive data transfers such as mobile payments. By utilizing environmental AC flickering signals reconfigured by the electric network as unique fingerprints, this research successfully improves the security and reliability of optical wireless systems. These "voltmarks" inherently carry distinctive temporal and spatial characteristics, enabling precise verification of the time and location where optical signals are captured.

6.2 Future Work

6.2.1 Wireless Digital Twin for Smart Reconfigurable Environments

The concept of a digital twin has emerged as a groundbreaking approach in simulation and real-time analytics, providing a dynamic digital representation of physical systems. In the context of smart reconfigurable environments, the development of a wireless digital twin is crucial. Such a model offers a precise, continuously updated representation of the physical wireless environment, enabling simulation, analysis, and optimization of wireless networks under various environmental conditions. This capability is essential for the rapid changes in modern wireless communications.

With digital twin techniques, real-time simulation techniques can be employed to test and validate different configurations and scenarios, providing valuable insights into system responses and interactions without the risk of disrupting the actual environment. This approach is particularly beneficial for evaluating new signal optimization strategies or deployment configurations prior to physical implementation. Moreover, iterative refinement—where the digital twin is continuously updated and improved based on feedback loops from the physical environment—ensures that the twin evolves and adapts over time, maintaining its accuracy and relevance. To build a wireless digital twin system, there are two possible methods:

- **Integrating Visual Priors:** In smart reconfigurable environments, integrating visual priors into the digital twin framework can significantly enhance system accuracy and responsiveness. This integration leverages visual data from the environment to enrich the digital twin model with precise, real-time geometric and material information about the surroundings. By fusing visual data with RF measurements, the digital twin not only reflects the physical state more accurately but also predicts electromagnetic behavior in dynamic environments with higher fidelity. This capability is crucial for modeling complex interactions between radio signals and the physical environment, which is essential for applications such as advanced wireless communication systems and precision localization services. The fusion of visual priors facilitates rapid adaptation of the digital twin to changes, reducing the need for extensive recalibrations and enabling more agile responses to environmental fluctuations.
- **Differential Ray Tracing:** Differential ray tracing represents a significant advancement in the simulation and optimization of wireless networks by extending traditional ray-tracing capabilities. This method enables the computation of gradients for various radio propagation metrics, such as channel impulse responses, with respect to a broad array of system and environmental parameters. These parameters include material properties, antenna configurations, and the spatial orientations and positions of transmitters and receivers. The ability to differenti-

ate these outcomes facilitates gradient-based optimization of system designs and configurations in complex scenarios, such as urban settings or dynamically changing environments typical in modern wireless communications. Integrating differential ray tracing with machine learning techniques opens new avenues for predictive modeling and automatic tuning of network parameters. This capability is crucial for developing adaptive communication systems that optimize performance in real-time based on immediate environmental feedback, enhancing both the accuracy and efficiency of simulations and leading to more robust and adaptable communication infrastructures.

6.2.2 Exploring Alternative Architectures for Physical Neural Networks

Currently, physical neural networks (PNNs), particularly in the radio frequency (RF) domain, are constrained by the relatively small number of parameters they can implement, which limits their computational capabilities. In contrast, high-performance machine learning systems like ChatGPT require millions or even billions of parameters to achieve their advanced functionalities. Therefore, it is essential to explore methods for developing PNNs with significantly increased parameter counts.

Furthermore, the current design of PNNs predominantly relies on the traditional Multilayer Perceptron (MLP) architecture. However, other neural network architectures have demonstrated superior performance in various machine learning applications due to their ability to model intricate patterns and dependencies within data. Incorporating more complex neural network models—such as Convolutional Neural Networks (CNNs), Residual Networks (ResNets), and Long Short-Term Memory (LSTM) networks—into PNNs could significantly enhance their computational power and enable them to handle more complex tasks. To address these limitations, adopting alternative architectures for PNNs offers the potential for larger-scale implementations and enhanced computational capabilities.

- **Optical Neural Networks:** Optical Neural Networks (ONNs) use light to perform computations and are particularly advantageous for their high speed and low energy consumption. Since the wavelengths of optical signals are significantly smaller than those of RF signals, the size of unit cells required to modify the propagation coefficient is much smaller. This miniaturization makes it possible to build ONNs with millions or even billions of parameters, vastly increasing their computational capacity.
- **Implementing Complex Physical Neural Networks:** Advanced architectures like CNNs, ResNets, and LSTMs have achieved significant success in traditional machine learning systems. Integrating these advanced machine learning architectures into physical neural networks can enhance their capabilities. For example, CNNs could utilize optical diffractive elements or wave-based interference to perform spatial convolutions directly. Residual Networks (ResNets) can be implemented using parallel pathways or recursive physical processes to facilitate efficient signal flow and prevent degradation of information. Similarly, LSTMs may leverage dynamic systems like optical delays or tunable resonators to process temporal sequences. By incorporating these architectures into physical neural systems, it may be possible to enhance computational capabilities while benefiting from the inherent speed and energy efficiency of physical implementations.

References

[1] F. Liu, L. Zhou, C. Masouros, A. Li, W. Luo, and A. Petropulu, “Toward dual-functional radar-communication systems: Optimal waveform design,” *IEEE Transactions on Signal Processing*, vol. 66, no. 16, pp. 4264–4279, 2018.

[2] F. Liu, Y.-F. Liu, A. Li, C. Masouros, and Y. C. Eldar, “Cramér-rao bound optimization for joint radar-communication beamforming,” *IEEE Transactions on Signal Processing*, vol. 70, pp. 240–253, 2021.

[3] F. Liu, Y. Cui, C. Masouros, J. Xu, T. X. Han, Y. C. Eldar, and S. Buzzi, “Integrated sensing and communications: Toward dual-functional wireless networks for 6g and beyond,” *IEEE journal on selected areas in communications*, vol. 40, no. 6, pp. 1728–1767, 2022.

[4] J. Zou, S. Sun, C. Masouros, Y. Cui, Y.-F. Liu, and D. W. K. Ng, “Energy-efficient beamforming design for integrated sensing and communications systems,” *IEEE Transactions on Communications*, 2024.

[5] Z. Ren, X. Song, Y. Fang, L. Qiu, and J. Xu, “Fundamental crb-rate tradeoff in multi-antenna multicast channel with isac,” in *2022 IEEE Globecom Workshops (GC Wkshps).* IEEE, 2022, pp. 1261–1266.

[6] X. Liu, T. Huang, N. Shlezinger, Y. Liu, J. Zhou, and Y. C. Eldar, “Joint transmit beamforming for multiuser mimo communications and mimo radar,” *IEEE Transactions on Signal Processing*, vol. 68, pp. 3929–3944, 2020.

[7] X. Tong, Z. Zhang, J. Wang, C. Huang, and M. Debbah, “Joint multi-user communication and sensing exploiting both signal and environment sparsity,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 6, pp. 1409–1422, 2021.

[8] C. Wen, Y. Huang, L. Zheng, W. Liu, and T. N. Davidson, “Transmit waveform design for dual-function radar-communication systems via hybrid linear-nonlinear precoding,” *IEEE Transactions on Signal Processing*, vol. 71, pp. 2130–2145, 2023.

[9] R. Liu, M. Li, Q. Liu, and A. L. Swindlehurst, “Joint waveform and filter designs for stap-slp-based mimo-dfrc systems,” *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 6, pp. 1918–1931, 2022.

[10] Q. Wu and R. Zhang, “Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network,” *IEEE communications magazine*, vol. 58, no. 1, pp. 106–112, 2019.

[11] M. Dajer, Z. Ma, L. Piazzi, N. Prasad, X.-F. Qi, B. Sheen, J. Yang, and G. Yue, “Reconfigurable intelligent surface: Design the channel–a new opportunity for future wireless networks,” *Digital Communications and Networks*, vol. 8, no. 2, pp. 87–104, 2022.

[12] Y. Liu, X. Liu, X. Mu, T. Hou, J. Xu, M. Di Renzo, and N. Al-Dhahir, “Reconfigurable intelligent surfaces: Principles and opportunities,” *IEEE communications surveys & tutorials*, vol. 23, no. 3, pp. 1546–1577, 2021.

[13] J.-B. Gros, V. Popov, M. A. Odit, V. Lenets, and G. Lerosey, “A reconfigurable intelligent surface at mmwave based on a binary phase tunable metasurface,” *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1055–1064, 2021.

[14] Y.-X. Ren, R.-D. Lu, and L. Gong, “Tailoring light with a digital micromirror device,” *Annalen der physik*, vol. 527, no. 7-8, pp. 447–470, 2015.

[15] K. J. Mitchell, S. Turtaev, M. J. Padgett, T. Čižmár, and D. B. Phillips, "High-speed spatial control of the intensity, phase and polarisation of vector beams using a digital micro-mirror device," *Optics express*, vol. 24, no. 25, pp. 29 269–29 282, 2016.

[16] R. Alwahishi, M. M. M. Ali, G. H. Elzwawi, and T. A. Denidni, "Beam-switching antenna using reconfigurable intelligent frequency selective surfaces for internet of things applications," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 4152–4162, 2023.

[17] H. Li, Q. Cao, L. Liu, and Y. Wang, "An improved multifunctional active frequency selective surface," *IEEE Transactions on Antennas and Propagation*, vol. 66, no. 4, pp. 1854–1862, 2018.

[18] L. Liu, X. Zhang, M. Kenney, X. Su, N. Xu, C. Ouyang, Y. Shi, J. Han, W. Zhang, and S. Zhang, "Broadband metasurfaces with simultaneous control of phase and amplitude," *Advanced materials*, vol. 26, no. 29, pp. 5031–5036, 2014.

[19] R. I. Zelaya, W. Sussman, J. Gummeson, K. Jamieson, and W. Hu, "Lava: fine-grained 3d indoor wireless coverage for small iot devices," in *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, 2021, pp. 123–136.

[20] R. Ma, R. I. Zelaya, and W. Hu, "Softly, deftly, scrolls unfurl their splendor: Rolling flexible surfaces for wideband wireless," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–15.

[21] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Zero-effort cross-domain gesture recognition with wi-fi," in *Proceedings of the 17th annual international conference on mobile systems, applications, and services*, 2019, pp. 313–325.

[22] Y. Liu, L. Jiang, L. Kong, Q. Xiang, X. Liu, and G. Chen, “Wi-fruit: See through fruits with smart devices,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, no. 4, pp. 1–29, 2021.

[23] R. Ayyalasomayajula, A. Arun, C. Wu, S. Sharma, A. R. Sethi, D. Vasisht, and D. Bharadia, “Deep learning based wireless localization for indoor navigation,” in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–14.

[24] R. Zhang, Y. Shao, M. Li, L. Lu, and Y. C. Eldar, “Optical integrated sensing and communication with light-emitting diode,” in *2024 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2024, pp. 2059–2064.

[25] L. Shi, Z. Liu, B. Béchadergue, H. Guan, L. Chassagne, and X. Zhang, “Experimental demonstration of integrated optical wireless sensing and communication,” *Journal of Lightwave Technology*, 2024.

[26] Y. Xia, J. Zhang, T. Guo, H. Wang, C. Geng, Y. Zhu, R. Han, Y. Yang, G. Song, X. Wan *et al.*, “High-speed flexible near-infrared organic photodetectors for self-powered optical integrated sensing and communications,” *Advanced Functional Materials*, p. 2412813, 2024.

[27] X. Zhao, Z. An, Q. Pan, and L. Yang, “Nerf2: Neural radio-frequency radiance fields,” in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–15.

[28] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, “Spotfi: Decimeter level localization using wifi,” in *Proc. of ACM SIGCOMM*, 2015.

[29] A. Eid, J. Zhu, L. Xu, J. G. Hester, and M. M. Tentzeris, “Holography-based target localization and health monitoring technique using uhf tags array,” *IEEE Internet of Things Journal*, vol. 8, no. 19, pp. 14 719–14 730, 2021.

[30] E. Soltanaghaei, A. Prabhakara, A. Balanuta, M. Anderson, J. M. Rabaey, S. Kumar, and A. Rowe, “Millimetro: mmwave retro-reflective tags for accurate, long range localization,” in *Proc. of ACM MobiCom*, 2021.

[31] B. Liang, P. Wang, R. Zhao, H. Guo, P. Zhang, J. Guo, S. Zhu, H. H. Liu, X. Zhang, and C. Xu, “Rf-chord: Towards deployable rfid localization system for logistic networks,” in *Proc. of USENIX NSDI*, 2023, pp. 1783–1799.

[32] W. Jiang, H. Xue, C. Miao, S. Wang, S. Lin, C. Tian, S. Murali, H. Hu, Z. Sun, and L. Su, “Towards 3d human pose construction using wifi,” in *Proc. of ACM MobiCom*, 2020.

[33] K. Qian, C. Wu, Z. Yang, Y. Liu, and K. Jamieson, “Widar: Decimeter-level passive tracking via velocity monitoring with commodity wi-fi,” in *Proc. of ACM MobiHoc*, 2017, pp. 1–10.

[34] H. Shanbhag, S. Madani, A. Isanaka, D. Nair, S. Gupta, and H. Hassanieh, “Contactless material identification with millimeter wave vibrometry,” in *Proc. of ACM MobiSys*, 2023.

[35] Y. Ren, S. Tan, L. Zhang, Z. Wang, Z. Wang, and J. Yang, “Liquid level sensing using commodity wifi in a smart home environment,” *Proc. of ACM IMWUT*, vol. 4, no. 1, pp. 1–30, 2020.

[36] C. R. Karanam and Y. Mostofi, “3d through-wall imaging with unmanned aerial vehicles using wifi,” in *Proc. of ACM/IEEE IPSN*, 2017.

[37] C. X. Lu, S. Rosa, P. Zhao, B. Wang, C. Chen, J. A. Stankovic, N. Trigoni, and A. Markham, “See through smoke: robust indoor mapping with low-cost mmwave radar,” in *Proc. of ACM MobiSys*, 2020.

[38] X. Zhao, Z. An, Q. Pan, and L. Yang, “Nerf ^{2}: Neural radio-frequency radiance fields,” *arXiv preprint arXiv:2305.06118*, 2023.

[39] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “Nerf: Representing scenes as neural radiance fields for view synthesis,” in *European conference on computer vision*. Springer, 2020, pp. 405–421.

[40] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, “D-nerf: Neural radiance fields for dynamic scenes,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 318–10 327.

[41] K. Deng, A. Liu, J.-Y. Zhu, and D. Ramanan, “Depth-supervised nerf: Fewer views and faster training for free,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 882–12 891.

[42] T. Orekondy, P. Kumar, S. Kadambi, H. Ye, J. Soriaga, and A. Behboodi, “Winert: Towards neural ray tracing for wireless channel modelling and differentiable simulations,” in *The Eleventh International Conference on Learning Representations*, 2022.

[43] S. Wu, H. Wang, and C.-H. Youn, “Visible light communications for 5g wireless networking systems: from fixed to mobile communications,” *Ieee Network*, vol. 28, no. 6, pp. 41–45, 2014.

[44] P. H. Pathak, X. Feng, P. Hu, and P. Mohapatra, “Visible light communication, networking, and sensing: A survey, potential and challenges,” *IEEE communications surveys & tutorials*, vol. 17, no. 4, pp. 2047–2077, 2015.

[45] Z. Ghassemlooy, P. Luo, and S. Zvanovec, “Optical camera communications,” *Optical Wireless Communications: An Emerging Technology*, pp. 547–568, 2016.

[46] A. Sevincer, A. Bhattarai, M. Bilgi, M. Yuksel, and N. Pala, “Lightnets: Smart lighting and mobile optical wireless networks—a survey,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 1620–1641, 2013.

[47] D. Karunatilaka, F. Zafar, V. Kalavally, and R. Parthiban, “Led based indoor visible light communications: State of the art,” *IEEE communications surveys & tutorials*, vol. 17, no. 3, pp. 1649–1678, 2015.

[48] R. T. Solberg and A. R. Jensenius, “Optical or inertial? evaluation of two motion capture systems for studies of dancing to electronic dance music,” 2016.

[49] S. Teli, W. A. Cahyadi, and Y. H. Chung, “Optical camera communication: Motion over camera,” *IEEE Communications Magazine*, vol. 55, no. 8, pp. 156–162, 2017.

[50] K. Krombholz, P. Frühwirt, P. Kieseberg, I. Kapsalis, M. Huber, and E. Weippl, “Qr code security: A survey of attacks and challenges for usable security,” in *Human Aspects of Information Security, Privacy, and Trust: Second International Conference, HAS 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014. Proceedings 2.* Springer, 2014, pp. 79–90.

[51] T. Vidas, E. Owusu, S. Wang, C. Zeng, L. F. Cranor, and N. Christin, “Qrishing: The susceptibility of smartphone users to qr code phishing attacks,” in *Financial Cryptography and Data Security: FC 2013 Workshops, USEC and WAHC 2013, Okinawa, Japan, April 1, 2013, Revised Selected Papers 17.* Springer, 2013, pp. 52–69.

[52] A. Dabrowski, K. Krombholz, J. Ullrich, and E. R. Weippl, “Qr inception: Barcode-in-barcode attacks,” in *Proceedings of the 4th ACM workshop on security and privacy in smartphones & mobile devices*, 2014, pp. 3–10.

[53] Y. Li, Y.-C. Chen, X. Ji, H. Pan, L. Yang, G. Xue, and J. Yu, “Screenid: Enhancing qrcode security by fingerprinting screens,” in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications.* IEEE, 2021, pp. 1–10.

[54] S. K. Thamer and B. N. Ameen, "A new method for ciphering a message using qr code," *Comput. Sci. Eng*, vol. 6, no. 2, pp. 19–24, 2016.

[55] J. Lu, Z. Yang, L. Li, W. Yuan, L. Li, and C.-C. Chang, "Multiple schemes for mobile payment authentication using qr code and visual cryptography," *Mobile Information Systems*, vol. 2017, no. 1, p. 4356038, 2017.

[56] H. Pan, Y.-C. Chen, L. Yang, G. Xue, C.-W. You, and X. Ji, "mqrcode: Secure qr code using nonlinearity of spatial frequency in light," in *The 25th Annual International Conference on Mobile Computing and Networking*, 2019, pp. 1–18.

[57] W. Tang, J. Y. Dai, M. Chen, X. Li, Q. Cheng, S. Jin, K.-K. Wong, and T. J. Cui, "Programmable metasurface-based rf chain-free 8psk wireless transmitter," *Electronics letters*, vol. 55, no. 7, pp. 417–420, 2019.

[58] Z. Li, Y. Xie, L. Shangguan, R. I. Zelaya, J. Gummeson, W. Hu, and K. Jamieson, "Towards programming the radio environment with large arrays of inexpensive antennas," in *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*, 2019, pp. 285–300.

[59] M. Dunna, C. Zhang, D. Sievenpiper, and D. Bharadia, "Scattermimo: Enabling virtual mimo with smart surfaces," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–14.

[60] Z. Wu, Y. Ra'di, and A. Grbic, "Tunable metasurfaces: A polarization rotator design," *Physical Review X*, vol. 9, no. 1, p. 011036, 2019.

[61] H. Pan, L. Qiu, B. Ouyang, S. Zheng, Y. Zhang, Y.-C. Chen, and G. Xue, "Pmsat: Optimizing passive metasurface for low earth orbit satellite communication," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–15.

[62] A. Pallaprolu, W. Hurst, S. Paul, and Y. Mostofi, “I beg to diffract: Rf field programming with edges,” in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–15.

[63] K. Qian, L. Yao, K. Zheng, X. Zhang, and T. N. Ng, “Uniscatter: a metamaterial backscatter tag for wideband joint communication and radar sensing,” in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–16.

[64] C. Feng, X. Li, Y. Zhang, X. Wang, L. Chang, F. Wang, X. Zhang, and X. Chen, “Rflens: metasurface-enabled beamforming for iot communication and sensing,” in *Proc. of ACM MobiCom*, 2021.

[65] V. Arun and H. Balakrishnan, “{RFocus}: Beamforming using thousands of passive antennas,” in *17th USENIX symposium on networked systems design and implementation (NSDI 20)*, 2020, pp. 1047–1061.

[66] K. W. Cho, M. H. Mazaheri, J. Gummeson, O. Abari, and K. Jamieson, “{mmWall}: A steerable, transflective metamaterial surface for {NextG}{mmWave} networks,” in *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*, 2023, pp. 1647–1665.

[67] F. Lestini, G. Marrocco, and C. Occhiuzzi, “Feasibility of rfid-based control of reconfigurable intelligent surfaces (riss) for wireless communication systems,” in *2023 IEEE 13th International Conference on RFID Technology and Applications (RFID-TA)*. IEEE, 2023, pp. 241–244.

[68] D. Xie, X. Wang, and A. Tang, “Metasight: localizing blocked rfid objects by modulating nlos signals via metasurfaces,” in *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*, 2022, pp. 504–516.

[69] Z. X. Wang, C. K. Shen, J. W. Wu, H. Xu, Q. Cheng, T. T. Ye, and T. J. Cui, “A long-range and nearly passive rfid-controlled information metasurface,” *Advanced Optical Materials*, p. 2203114, 2023.

[70] X. Li, C. Feng, F. Song, C. Jiang, Y. Zhang, K. Li, X. Zhang, and X. Chen, “Protego: securing wireless communication via programmable metasurface,” in *Proc. of ACM MobiCom*, 2022.

[71] P. Staat, S. Mulzer, S. Roth, V. Moonsamy, M. Heinrichs, R. Kronberger, A. Sezgin, and C. Paar, “Irshield: A countermeasure against adversarial physical-layer wireless sensing,” in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 1705–1721.

[72] L. Chen, W. Hu, K. Jamieson, X. Chen, D. Fang, and J. Gummeson, “Pushing the physical limits of iot devices with programmable metasurfaces.” in *Proc. of USENIX NSDI*, 2021, pp. 425–438.

[73] K. W. Cho, M. H. Mazaheri, J. Gummeson, O. Abari, and K. Jamieson, “mmwall: A reconfigurable metamaterial surface for mmwave networks,” in *Proc. of HotMobile*, 2021.

[74] H. Chen, H. Saeidi, S. Venkatesh, K. Sengupta, and Y. Ghasempour, “Wavefront manipulation attack via programmable mmwave metasurfaces: from theory to experiments,” in *Proc. of ACM WiSec*, 2023.

[75] J. Hu, H. Zhang, K. Bian, M. Di Renzo, Z. Han, and L. Song, “Metasensing: Intelligent metasurface assisted rf 3d sensing by deep reinforcement learning,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 7, pp. 2182–2197, 2021.

[76] X. Fan, H. Ding, S. Li, M. Sanzari, Y. Zhang, W. Trappe, Z. Han, and R. E. Howard, “Energy-ball: Wireless power transfer for batteryless internet of things

through distributed beamforming," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 2, pp. 1–22, 2018.

[77] X. Fan, L. Shangguan, R. Howard, Y. Zhang, Y. Peng, J. Xiong, Y. Ma, and X.-Y. Li, "Towards flexible wireless charging for medical implants using distributed antenna system," in *Proc. of ACM MobiCom*, 2020, pp. 1–15.

[78] P. Zhang and D. Ganesan, "Enabling {Bit-by-Bit} backscatter communication in severe energy harvesting environments," in *Proc. of USENIX NSDI*, 2014, pp. 345–357.

[79] L. Dodds, I. Perper, A. Eid, and F. Adib, "A handheld fine-grained rfid localization system with complex-controlled polarization," *arXiv preprint arXiv:2302.13501*, 2023.

[80] T. Boroushaki, M. Lam, W. Chen, L. Dodds, A. Eid, and F. Adib, "Exploiting synergies between augmented reality and rfids for item localization and retrieval," in *2023 IEEE International Conference on RFID (RFID)*. IEEE, 2023, pp. 30–35.

[81] Y. Ma, N. Selby, and F. Adib, "Minding the billions: Ultra-wideband localization for deployed rfid tags," in *Proceedings of the 23rd annual international conference on mobile computing and networking*, 2017, pp. 248–260.

[82] E. Sie and D. Vasisht, "Rf-annotate: Automatic rf-supervised image annotation of common objects in context," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2590–2596.

[83] J. Wang, J. Li, M. H. Mazaheri, K. Katsuragawa, D. Vogel, and O. Abari, "Sensing finger input using an rfid transmission line," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 531–543.

[84] J. Han, H. Ding, C. Qian, D. Ma, W. Xi, Z. Wang, Z. Jiang, and L. Shangguan, “Cbid: A customer behavior identification system using passive tags,” in *Proc. of IEEE ICNP*, 2014.

[85] M. I. Ahmed, A. Bansal, K. Yuan, S. Kumar, and P. Steenkiste, “Battery-free wideband spectrum mapping using commodity rfid tags,” in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, 2023, pp. 1–16.

[86] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, “All-optical machine learning using diffractive deep neural networks,” *Science*, vol. 361, no. 6406, pp. 1004–1008, 2018.

[87] L. G. Wright, T. Onodera, M. M. Stein, T. Wang, D. T. Schachter, Z. Hu, and P. L. McMahon, “Deep physical neural networks trained with backpropagation,” *Nature*, vol. 601, no. 7894, pp. 549–555, 2022.

[88] T. Wang, S.-Y. Ma, L. G. Wright, T. Onodera, B. C. Richard, and P. L. McMahon, “An optical neural network using less than 1 photon per multiplication,” *Nature Communications*, vol. 13, no. 1, p. 123, 2022.

[89] C. Liu, Q. Ma, Z. J. Luo, Q. R. Hong, Q. Xiao, H. C. Zhang, L. Miao, W. M. Yu, Q. Cheng, L. Li *et al.*, “A programmable diffractive deep neural network based on a digital-coding metasurface array,” *Nature Electronics*, vol. 5, no. 2, pp. 113–122, 2022.

[90] Y. Zuo, B. Li, Y. Zhao, Y. Jiang, Y.-C. Chen, P. Chen, G.-B. Jo, J. Liu, and S. Du, “All-optical neural network with nonlinear activation functions,” *Optica*, vol. 6, no. 9, pp. 1132–1137, 2019.

[91] S. G. Sanchez, G. Reus-Muns, C. Bocanegra, Y. Li, U. Muncuk, Y. Naderi, Y. Wang, S. Ioannidis, and K. R. Chowdhury, “Airnn: Over-the-air compu-

tation for neural networks via reconfigurable intelligent surfaces," *IEEE/ACM Transactions on Networking*, 2022.

[92] S. Gong, X. Lu, D. T. Hoang, D. Niyato, L. Shu, D. I. Kim, and Y.-C. Liang, "Toward smart wireless communications via intelligent reflecting surfaces: A contemporary survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2283–2314, 2020.

[93] C. Liu, F. Yang, S. Xu, and M. Li, "Reconfigurable metasurface: A systematic categorization and recent advances," *arXiv preprint arXiv:2301.00593*, 2023.

[94] "RFID Forecasts, Players and Opportunities 2023-2033," https://www.idtechex.com/en/research-report/rfid-forecasts-players-and-opportunities-2023-2033/927, 2023.

[95] "EPC Gen2, EPCglobal." www.gs1.org/epcglobal.

[96] EM Microelectronic, "EPC and UHF ICs - EM4325," https://www.emmicroelectronic.com/product/epc-and-uhf-ics/em4325, 2023.

[97] A. Sayanskiy, A. Belov, R. Yafasov, A. Lyulyakin, A. Sherstobitov, S. Glybovski, and V. Lyashev, "A 2d-programmable and scalable reconfigurable intelligent surface remotely controlled via digital infrared code," *IEEE Transactions on Antennas and Propagation*, vol. 71, no. 1, pp. 570–580, 2022.

[98] NXP, "BF1118," https://www.nxp.com/part/BF1118R#/, 2023, accessed: 2024-01-01.

[99] "Impinj, Inc," http://www.impinj.com/.

[100] EPCglobal, "Low level reader protocol (llrp)," 2010.

[101] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[102] "Hololens," https://www.microsoft.com/en-us/hololens, 2024.

[103] Ericsson, "Future network requirements for xr apps," n.d.

[104] J. Liu, H. Liu, Y. Chen, Y. Wang, and C. Wang, "Wireless sensing for human activity: A survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1629–1645, 2019.

[105] F. Zhou and Y. Chai, "Near-sensor and in-sensor computing," *Nature Electronics*, vol. 3, no. 11, pp. 664–671, 2020.

[106] J. Lin, W.-M. Chen, Y. Lin, C. Gan, S. Han *et al.*, "Mcunet: Tiny deep learning on iot devices," *Advances in Neural Information Processing Systems*, vol. 33, pp. 11 711–11 722, 2020.

[107] G. Hinton, "The forward-forward algorithm: Some preliminary investigations," *arXiv preprint arXiv:2212.13345*, 2022.

[108] C. Huang, W. Pan, X. Ma, B. Zhao, J. Cui, and X. Luo, "Using reconfigurable transmitarray to achieve beam-steering and polarization manipulation applications," *IEEE Transactions on Antennas and Propagation*, vol. 63, no. 11, pp. 4801–4810, 2015.

[109] J. Y. Lau and S. V. Hum, "Reconfigurable transmitarray design approaches for beamforming applications," *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 12, pp. 5679–5689, 2012.

[110] C.-S. Lin, S.-F. Chang, C.-C. Chang, and Y.-H. Shu, "Design of a reflection-type phase shifter with wide relative phase shift and constant insertion loss," *IEEE Transactions on microwave theory and techniques*, vol. 55, no. 9, pp. 1862–1868, 2007.

[111] MACOM, "MA46H120," https://www.macom.com/products/product-detail/MA46H120, 2023, accessed: 2023-02-24.

[112] P. S. Wright and J. R. Pickering, “An ac voltage standard based on a pwm dac,” *IEEE transactions on instrumentation and measurement*, vol. 48, no. 2, pp. 457–461, 1999.

[113] 3PEAK, “TP2274,” https://www.lcsc.com/product-detail/Instrumentation-OpAmps_TP2274-SR_C95870.html, 2023, accessed: 2023-02-24.

[114] STMicroelectronics, “STM32F072C8,” https://www.st.com/en/microcontrollers-microprocessors/stm32f072c8.html, 2023, accessed: 2023-02-24.

[115] Y. Zhang, Y. Zheng, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, “Widar3. 0: Zero-effort cross-domain gesture recognition with wi-fi,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 8671–8688, 2021.

[116] U. Ha, Y. Ma, Z. Zhong, T.-M. Hsu, and F. Adib, “Learning food quality and safety from wireless stickers,” in *Proc. of ACM HotNets*, 2018.

[117] Y. Zhou, linda Qiao, J. Shang, M. Wu, Q. Wang, H. Li, and J. Xie, “K-margin-based residual-convolution-recurrent neural network for atrial fibrillation detection,” in *Proc. of IJCAI*, 2019.

[118] KEYSIGHT, “34465A,” https://www.keysight.com/us/en/product/34465A/digital-multimeter-6-5-digit-truevolt-dmm.html, 2023, accessed: 2023-02-24.

[119] R. Liu and N. Choi, “A first look at wi-fi 6 in action: Throughput, latency, energy efficiency, and security,” *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 7, no. 1, pp. 1–25, 2023.

[120] F. Lu, “Food faux pas: China diner shocked by us$60,000 bill for meal after accidentally posting photo of dish ordering qr code online,” *South China Morning Post*, December 2023, accessed: 2024-05-30.

[Online]. Available: https://www.scmp.com/news/people-culture/trending-china/article/3243392/food-faux-pas-china-diner-shocked-us60000-bill-meal-after-accidentally-posting-photo-dish-ordering

[121] CGTN, “China to regulate qr code payments amid fraud concerns,” *CGTN News*, September 2020, accessed: 2024-05-30. [Online]. Available: https://news.cgtn.com/news/2020-09-15/China-to-regulate-QR-code-payments-amid-fraud-concerns-TJGifQ8LeQ/index.html

[122] M. Sheinin, Y. Y. Schechner, and K. N. Kutulakos, “Computational imaging on the electric grid,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6437–6446.

[123] ——, “Rolling shutter imaging on the electric grid,” in *2018 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2018, pp. 1–12.

[124] L. Blakely and M. J. Reno, “Phase identification using co-association matrix ensemble clustering,” *IET Smart Grid*, vol. 3, no. 4, pp. 490–499, 2020.

[125] L. Xu, G. Hua, H. Zhang, L. Yu, and N. Qiao, “" seeing" electric network frequency from events,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 18 022–18 031.

[126] Z. Shah, A. Yen, A. Pandey, and J. Taneja, “Gridinsight: Monitoring electricity using visible lights,” in *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 2019, pp. 243–252.

[127] S. Zhu, C. Zhang, and X. Zhang, “Automating visual privacy protection using a smart led,” in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, 2017, pp. 329–342.

[128] *Electric Power Systems and Equipment - Voltage Ratings (60 Hertz)*, American National Standards Institute, 2020, specifies the nominal voltage ratings and

operating tolerances for 60 Hz electric power systems. [Online]. Available: https://webstore.ansi.org/Standards/NEMA/ANSINEMAC8412020

[129] C. Lee, S. Hui, and H. Chung, "A randomized voltage vector switching scheme for 3-level power inverters," in *2000 IEEE 31st Annual Power Electronics Specialists Conference. Conference Proceedings (Cat. No. 00CH37018)*, vol. 1. IEEE, 2000, pp. 27–32.

[130] Cambridge in Colour, "Camera exposure: Understanding the essentials," n.d., accessed: 2024-08-16. [Online]. Available: https://www.cambridgeincolour.com/tutorials/camera-exposure.htm

[131] D. Cardinal and Horshack, "The rolling shutter effect," 2023, accessed: 2024-07-18. [Online]. Available: https://horshack-dpreview.github.io/RollingShutter/

[132] A. Dosovitskiy, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[133] B. Souchet, "Ai-generated barcode images/masks dataset," https://github.com/BenSouchet/barcode-datasets, 2023, accessed: [insert date of access here].

[134] Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljačić, T. Y. Hou, and M. Tegmark, "Kan: Kolmogorov-arnold networks," *arXiv preprint arXiv:2404.19756*, 2024.

[135] ZishuTech, "USB3214," https://gitee.com/zishutech/DAQ-Product/tree/master/DAQ-USB3213A_USB3214, 2024, accessed: 2024-08-28.

[136] National Instruments, "Data acquisition (daq)," https://www.ni.com/en/shop/data-acquisition.html#:~:text=Data%20acquisition%20(DAQ)%20is%20the,temperature%2C%20pressure%2C%20or%20sound, 2024, accessed: [insert date here].

[137] S. Mehta and M. Rastegari, “Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer,” *arXiv preprint arXiv:2110.02178*, 2021.

[138] Y. Li, G. Yuan, Y. Wen, J. Hu, G. Evangelidis, S. Tulyakov, Y. Wang, and J. Ren, “Efficientformer: Vision transformers at mobilenet speed,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 12 934–12 949, 2022.