EXPLORING SPATIAL-TEMPORAL MULTI-DIMENSIONS IN OPTICAL WIRELESS COMMUNICATION AND SENSING

By

Xiao Zhang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science—Doctor of Philosophy

2023

ABSTRACT

Optical Wireless Communication (OWC) techniques are potential alternatives of the next generation wireless communication. These techniques, such as, VLC (visible light communication), OCC (optical camera communication), Li-Fi, FSOC (free space optical communication), and LiDAR, are increasingly deployed in our daily life. However, current OWC approaches are constrained by slow speeds and limited usage cases. The primary goal of this thesis is to boost the data rate of OWC with extended use scenarios and enable optical wireless sensing by exploiting the potentials on both the transmitter and receiver sides with designed effective strategies. We investigate the possibilities of various spatial-temporal dimensions (e.g., 1D, 2D, 3D, and 4D) as below.

1D Temporal Optical Wireless Communication. We found that compensation symbols, which are commonly used for fine-grained dimming, are not used for data transmission in OOK-based LiFi for indoor lighting and communication. We exploit compensation symbol in 1D temporal diversity to address the conflict of fine-grained dimming and transmission. We intend to demonstrate the LiFOD framework, which is installed on commercial off-the-shelf (COTS) LiFi systems, to increase the data rate of existing Li-Fi systems. We utilize compensation symbols, which were previously only used for dimming, to carry data bits (bit patterns) for enhanced throughput.

2D Spatial-Temporal Optical Wireless Communication. In our study of camera-based OWC (i.e., optical camera communication), we first investigate 2D rolling blocks in the camera imaging process rather than 1D rolling strips for improved optical symbol modulation and data rate. Our proposed RainbowRow overcomes the limitation of restricted frequency responses (i.e., tens of Hz) in traditional optical camera communication. We implement low-cost RainbowRow prototypes with adaptations for both indoor office and vehicular networks. The results demonstrate that RainbowRow achieves a 20× data rate improvement compared to existing LED-OCC systems.

3D Spatial Optical Wireless Communication. When compared to existing acoustic and RF-based approaches, underwater optical wireless communication appears promising due to its broad bandwidth and extended communication range. Existing optical tags (bar/QR codes) embed data in the plane with limited symbol distance and scanning angles. To address this limitation, we exploit

3D spatial diversity to design passive optical tags for simple and robust underwater navigation. We also develop underwater denoising algorithms with CycleGAN, CNN based relative positioning, and real-time data parsing. The experiments demonstrate that our U-star system can provide robust self-served underwater navigation guidance.

3D Spatial Optical Wireless Sensing. The vision approaches compatible with time-consuming image processing for hand gesture reconstructing adopt low 60 Hz location sampling rate (frame rate). To overcome this limitation, we propose RoFin, which first exploits 6 spatial-temporal 2D rolling fingertips for real-time 20-joint hand pose reconstructing. RoFin designs active optical labeling for massive fingers with fine-grained finger tracking. These features enable great potential for enhanced multi-user HCI and virtual writing for users, especially for Parkinson sufferers. We implement RoFin gloves attached with single-colored LED nodes and commercial cameras.

4D Spatial-Temporal Optical Wireless Integrated Sensing and Communication. Existing centralized radio frequency controlled from base stations face mutual interference and high latency, which causes localization errors. To avoid localization delay error, we explore optical camera communication for on-site pose parsing for drones. We exploit 4D spatial-temporal diversity (i.e., 3D spatial and 1D temporal diversities) for integrated sensing and communication. We propose PoseFly, an AI assisted OCC framework with integrated drone identification, on-site localization, quick-link communication, and lighting functions for swarming drones.

The variety of applications in many contexts demonstrates OWC's potential and usefulness as a foundation for next-generation wireless technology. By leveraging the multiple dimensions of spatial-temporal diversities, we were able to successfully overcome some aspects of current OWC systems, delivering critical insights and discoveries for the future of optical wireless communication.

ACKNOWLEDGEMENTS

First of all, I would like to say thanks to my advisor Prof. Li Xiao, who gave me numerous advice and strong support during my Ph.D. journey. She is a great mentor both mentally and technically. I learn a lot from her. Without Dr. Xiao's patience, support and guidance, I would not complete this dissertation. I would like to express my thanks to my guidance committee members Prof. Matt Mutka, Prof. Tianxing Li and Prof. Xiaobo Tan for their guidances as well.

As always, my parents are my strongest supporters both mentally and physically. I would like to express my greatest appreciation to my parents, Mr. Hongbao Zhang and Mrs. Liangping Zhu, who give me love unconditionally. I would like to thanks my brother, Mr. Xin Zhang, for his support as well. Without them, I can not finish my doctoral degree. I am also grateful to my dear friends Dr. Jie Huang, Mrs. Xuting Zou, Dr. Eakachai Kantawong, Mr. Baobing Lei, Mr. Yong Lei, who treat me as family member with sincere, care and love. I appreciate them for sharing my happiness and sadness.

I would like to thank members of ELANs and other mates, Prof. Yunhao Liu, Prof. Guanhua Tu, Prof. Qiben Yan, Prof. Zhichao Cao, Prof. Charles Ofria, Masoud, James, Yiwen, Hanqing, Griffin, Kanishka, Manni, Chenning, Li, Lingkun, Nick, Jianzhi, Yuanda, Juexing, Guangjing, Bocheng, Ce, Xinyu, Tian, Jingwen, Ao, Yang, Shenghong, Shuqi, Yuzhao, Ming, Wei, Yan. I also would like to thank our department chair Prof. Abdol Esfahanian, Prof. Sandeep Kulkarni, Prof. Colbry Katy, colleagues Brenda, Vincent, Amy. Also, I would like to thank my academic brothers Dr. Pei Huang, Dr. Chin-Jung Liu, Dr. Ruofeng Liu, Dr. Yan Pan, Dr. Yan Yan, for their selfless helps. I am also grateful to my master adviser Prof. Shining Li and other professors Prof. Zhe Yang, Prof. Yu Zhang, Prof. Zhigang Li, who inspired me to explore the wireless world. Finally, I would like to thank others who directly or indirectly offered helps to me.

The projects of this thesis are partially supported by the U.S. National Science Foundation under Grants CNS-2226888, CCF-2007159, and CNS-1617412. As for the remaining errors or deficiencies in this work, the responsibility rests entirely upon the author.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

OWC	Optical Wireless Communication
OCC	Optical Camera Communication
LiFi	Light Fidelity
VLC	Visible Light Communication
LiDAR	Light Detection and Ranging
FSOC	Free Space Optical Communication
RF	Radio Frequency
LED	Light Emission Diode
LD	Laser Diode
PD	Photo Diode
LCD	Liquid Crystal Display
CNN	Convolutional Neural Network
DNN	Deep Neural Network
PWM	Pulse Width Modulation
ESP	Effective Subcarrier Pairing
LiFOD	Lighting Extra Data via Fine-grained OWC Dimming
RBR	Rainbow Rows
U-Star	Underwater Stars
RoFin	Rolling Fingertips
PoseFly	Pose parsing of Flying drones
HotSys	Holographic Optical Tag based Systems
AR	Augmented Reality
VR	Virtual Reality
MR	Mixed Reality

XR	AR, VR, MR
CS	Compensation Symbols
FPS	Frame Per Second
FOV	Field of View
UAV	Unmanned Aerial Vehicle
V2X	Vehicle to Everything
CBS	Centralized Base Station
IMU	Inner Measurement Unit
LoS	Line-of-Sight
NLoS	None Line-of-Sight
OOK	On Off Keying
MPPM	Multiple-Pulse-Position Modulation
CSK	Color Shift Keying
VPPM	Variable Pulse Position Modulation
PRU	Programmable Real-time Unit
BBB	Beagle Bone Black
UOID	Underwater Optical Identification
HCI	Human Computer Interaction
AI	Artificial Intelligence
CV	Computer Vision

CHAPTER 1

INTRODUCTION AND MOTIVATION

Optical Wireless Communication (OWC) emerges as a compelling alternative to existing Radio Frequency wireless communication, thanks to its broad bandwidth. And OWC becomes a strong contender for the next generation of wireless communication. The high On/Off switching speed of LEDs enables them to serve as efficient OWC high-speed transmitters, allowing for both fast communication and effective lighting in our everyday scenarios. As for OWC receiver, there are two distinct types. The first type is a single-pixel device known as a photodiode (PD). The second type consists of cameras with millions of pixels.

However, current OWC systems mainly focus on point-to-point communication such as LiFi system and does not fully harness the potential of high-dimensional spatial-temporal diversities. This limitation hinders the data throughput of OWC, especially for camera-based OWC applications. To address these limitations, we investigate various spatial-temporal diversities in data embedding, such as 1D temporal dimming side-channel, 2D spatial-temporal rolling blocks, and 3D spatial diversity. Furthermore, it is challenge to uncover and define these spatial-temporal diversities. We must deal with technical challenges in system implementation when utilizing these diversities such as mutual interference among LEDs on both the transmitter and receiver ends, as well as denoising under a variety of ambient conditions.

To better motivate our work, I will present an overview of OWC and emphasize the similarities and differences between optical and traditional radio frequency mediums for wireless communication. Following that, I will showcase five fully developed projects where I served as the first author, focusing on harnessing innovative spatial-temporal diversities for data embedding in optical wireless communication and sensing to overcome the limitations in existing OWC systems.

1.1 OWC Background

1.1.1 OWC enabled Numerous Applications

There are various OWC technologies, as described in [15], such as VLC (Visible Light Communication), LiFi (Light Fidelity), OCC (Optical Camera Communication), FSOC (Free Space Optical Communication), and LiDAR (Light Detection and Ranging). These OWC approaches enable a wide range of applications [60, 82, 3, 116, 49]. For example, OWC techniques can be used in industry, transportation, workplaces, houses, malls, underwater, and space. Depending on the application type and the required data speed, communication type, and platform, different OWC techniques are employed. The traffic flow in optical wireless communication enabled applications is illustrated in Figure 1.1. The comparisons of different kinds of OWC scenarios are given below.

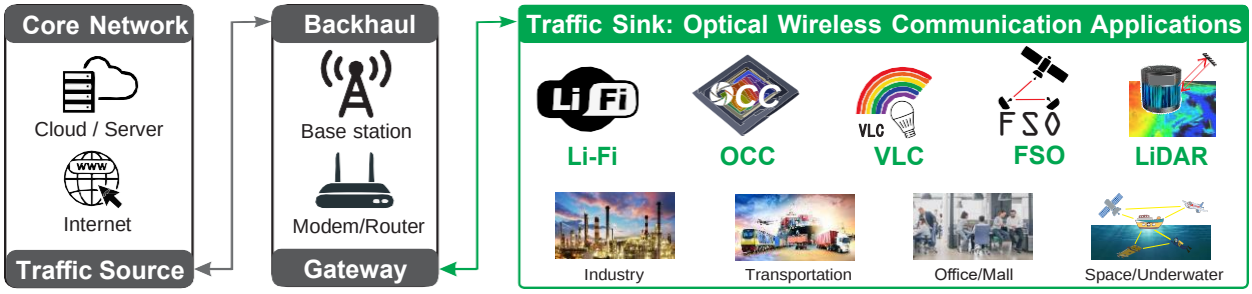


Figure 1.1 The network traffic flow in optical wireless communication and enabled numerous applications.

1.1.2 Modulated Optical Signals for Communication

Modulation is the technique that alters the amplitude, frequency, or phase of a carrier signal to convey information during signal transmission. We introduce some conventional OWC modulations below. (1) **OOK:** On–off keying (OOK) modulation is the simplest form of amplitude-shift keying (ASK) modulation [2]. OOK is applied to RF carrier waves as well as optical communication systems. OOK represents digital data by the presence or absence of a carrier wave. Bit '1' is represented by the light being turned on, whereas bit '0' is represented by the light being turned off. (2) **VPPM:** Variable pulse position modulation (VPPM) is a modulation technology that allows for simultaneous illumination, dimming control, and communication [2]. VPPM is intended for pulse-

width-based light dimming and protects against intraframe flicker. In VPPM, the pulse amplitude is always constant, and the dimming is controlled by pulse width rather than amplitude. (3) **CSK:** Color-shift keying (CSK) is a visible light communication intensity modulation described in the IEEE 802.15.7 standard that sends data invisibly by changing the color of red, green, and blue light emitting diodes[2]. The CSK symbol is produced by combining three color light sources from the seven color bands indicated in the standard. The center wave length of the three color bands on xy color coordinates determines the three vertices of the CSK constellation triangle.

1.2 Comparisons between Optical and RF Medium

1.2.1 Physical Feature Differences

Optical radiation is electromagnetic radiation that has wavelengths ranging from 100 nanometers to one millimeter. The wavelength range that the human eye can detect is referred to as visible radiation (VIS) and ranges between 400 nm and 800 nm [15]. UV light is optical radiation having wavelengths less than 400 nanometers. Infrared (IR) radiation has wavelengths greater than 800 nm. Microwave (1 mm - 1 m), VHF wave (1 - 10 m), LF wave (10-100m), MF wave (100 - 1000 m), LF wave (10 m - 1 km), and VLF wave are all examples of RF wavelengths (100 m - 10 km). The bandwidth of optical waves is around 30 PHz, which is 10,000 times greater than the bandwidth of radio waves (300 GHz). OWC necessitates a direct link between transmitter and receiver. Unlike RF transmissions, optical signals cannot flow through or around obstacles such as non-transparent objects. Light's LoS feature may provide a more secure physical layer than RF-based wireless communication. For RF signals, there are four propagation modes: (1) Free space propagation, (2) Direct modes (Line-of-Sight), (3) Surface modes (groundwave), and (4) Non-Line-of-Sight modes. Lower-frequency radio waves can pass through obstacles like buildings and plants, but this is still considered a Line-of-Sight approach. Surface modes are radio transmissions with lower frequencies ranging from 30 to 3,000 kHz that travel as surface waves following the curvature of the Earth. Non-Line-of-Sight propagation modes include ionospheric modes, meteor scattering, meteor scattering, auroral backscatter, sporadic-E propagation, tropospheric scattering, rain scattering, airplane scattering, and lighting scattering [41, 32].

1.2.2 Specific Advantage of Optical Signals

The performance of optical and radio frequency waves for underwater wireless communication differs as well. Two mechanisms impede light transmission in water: absorption and scattering. As a result of scattering, the quantity of photons captured by the receiver is reduced. Furthermore, in a murky underwater environment, numerous photons may arrive with delays, resulting in inter-symbol interference (ISI) [91]. RF results in extremely poor performance for long distance underwater communications, especially over long distances, due to heavily influenced elements such as multi-path propagation, channel time changes, and strong signal attenuation (particularly the electromagnetic shielding effect in sea water). As a result, the RF systems are constrained by the associated short link range [14]. When compared to an RF system, which necessitates energy-guzzling antennae and additional energy for cooling down, optical wireless communication uses energy-efficient LED bulbs and the consumed energy is not only for communication but also for simultaneous lighting [31]. Thus, OWC can provide considerable energy savings. Offloading traffic from RF networks to optical networks reduces overall power consumption [14].

1.2.3 Common Features of Optical and RF

Despite their distinct physical properties, optical waves and radio frequency waves have several similarities. (1) They both have the same propagation speed in the air that is faster than audio waves, (2) they have the same upper layers in the network architecture with the exception of differences in the Physical layer and the MAC layer, (3) they are both essentially electromagnetic waves, transverse waves rather than longitudinal waves like sound waves, (4) the mmWave in the RF spectrum propagates in a LoS way, similar to optical waves, and (5) except for the VL (visual light) optical spectrum, other optical spectrum are likewise invisible, similar to RF waves.

1.3 Problems in Existing OWC and Our Solutions

Despite the promising prospects of optical wireless communication, it is currently facing various challenges that limit its development and widespread application. For instance, in indoor optical wireless communication, a tradeoff needs to be considered between user illumination experience and the efficiency of optical data transmission, as shown at block of LiFOD in Figure 1.2. Another

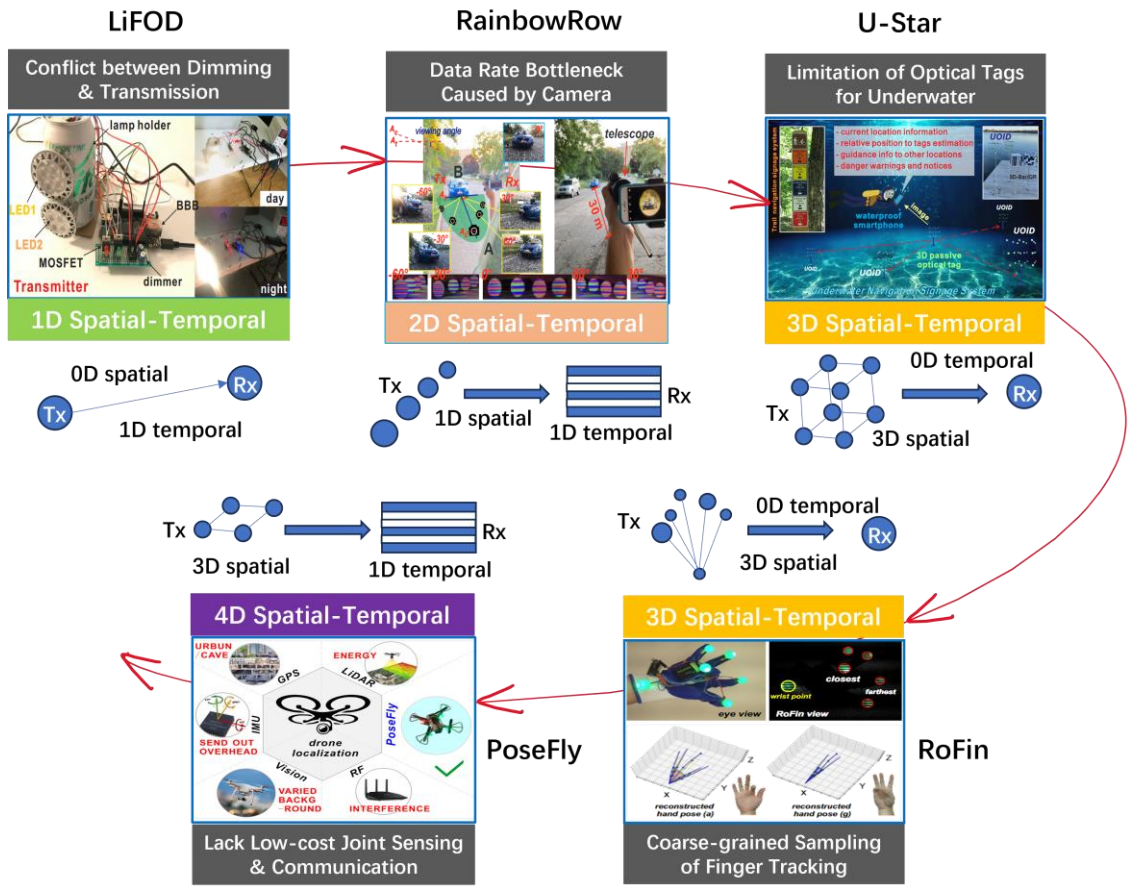


Figure 1.2 The problems (illustrated in gray blanks) in existing OWC systems and our solutions: an overview. To address these problems in different applications, we investigate multiple dimensions of spatial-temporal diversities in optical signals' propagation from the transmitter to the receiver.

example is in existing optical camera communication, where the limited camera response frequency restricts the achievable data rate to just a few Kbps, as shown at block of RainbowRow in Figure 1.2. Furthermore, existing optical tags are single-plane, lacking the capability to provide additional rich information in three-dimensional space such as underwater scenario, as shown at block of U-Star in Figure 1.2. Similarly, utilizing vision-based hand gesture recognition for finger tracking with only a few tens of Hz sampling rate hinders the provision of fine-grained finger tracking, as shown at block of RoFin in Figure 1.2. To achieve real-time, low-cost, on-site unmanned aerial vehicle (UAV) recognition, localization, and communication, it is challenge to meet all these requirements with one single solution, as shown at block of PoseFly in Figure 1.2. To address these problems, we specifically model spatial-temporal diversities and with different dimensions and leverage them

for specific OWC applications and scenarios, as described in Figure 1.2. We also briefly introduce each problem with our proposed solution below.

1.3.1 LiFOD to Address Conflicts between Dimming and Communication

Recent trends in lighting include replacing incandescent and fluorescent bulbs with high-intensity LEDs because of their high energy efficiency, low heat generation, and long lifespan[123, 109, 99]. LED lighting saves the average family approximately $225 in electricity bills each year[80]. Another benefit of LEDs is their capability to switch between different light intensities quickly and efficiently [151]. This feature creates opportunities for LEDs to be used as OWC transmitters for both high-speed communication and efficient lighting in everyday situations[11][132]. However, even with LED bulbs, lighting still accounts for around 15% of an ordinary home's electricity use[80]. Thus, for indoor LED bulbs, transmitting more data robustly with less retransmission while not sacrificing the user experience of lighting is another path to improve energy efficiency.

To transmit more data, we can design high-order modualtions in transmission. Recent research has focused on high-order modulation to improve throughput in OWC systems [151, 38, 124]. However, in poor optical channel conditions, such as indoor scenarios with complex artificial light sources or with sunny or underwater outdoor scenarios, the nonlinear effect of LEDs and the short symbol distance make decoding high-order modulation more complex and fragile, which leads to more error bits and, subsequently, more retransmissions that require energy consumption[131, 106, 46]. Thus most OWC systems, such as OpenVLC and LiFi [30, 84, 64, 151, 126, 69, 126, 34, 19], switch from high-order to low-order modulation such as simple OOK, which is defined as primary modulation in the OWC standard IEEE 802.15.7 [2]. As noted in [114], a tradeoff exists between the dimming performance and the achieved data rate due to the compensation symbols occupying the transmission bandwidth. To address the problems above, we propose **LiFOD** in Chapter 2 to achieve the fine-grained dimming and communication by utilizing the 1D temporal diversity of optical signals.

1.3.2 RainbowRow to Boost Restricted Data Rate in Optical Camera Communication

PDs are single-pixel light sensors and thus allow for fast light sensing that has the fast switching rate of LEDs at a couple of hundreds of KHz due to their simple and timely readout processing[30]. For example, OpenVLC[23] offers a data rate of about 150 Kbps at 3m for indoor use cases. However, they are not practical for outdoor and long-range scenarios due to varied optical environmental and strict directional requirements between the transmitter and the receiver.

Compared to single-pixel PD approaches, the image sensor (IS) in a **camera** has millions of pixels (each pixel element can be treated as a PD) and can easily separate the ambient light noise with the optical signals from the transmitter by reflecting them in different pixel zones[15]. Nonetheless, cameras require more processing and readout time for light sensations in contrast with single-pixel PDs[84, 30] and thus commercial cameras only offer *tens of Hz* frame rate and *several kHz* of rolling shutter rate. Given that LED-based transmitters offer ON/OFF switching rates of *several MHz*, this turns the camera-based receiver into the OCC systems' bottleneck and greatly restricts the data rate[151]. To overcome the bottleneck of optical camera communication, we introduce the **RainbowRow** protocol in Chapter 3. This protocol utilizes 2D spatial-temporal diversities of optical signals to significantly enhance the data rate.

1.3.3 U-Star to Address Limitations of Optical Codes in Underwater

Underwater Optical Wireless Communication (**UOWC**) has shown significant potential due to its longer propagation range, lower propagation delay, and lower power consumption compared with acoustic and RF-based techniques[91, 134, 147, 151, 117, 129, 141]. Moreover, UOWC systems based on passive optical tags, which utilize natural light sources, are more practical because they do not rely on finite battery power in underwater scenarios where it is not feasible to perform frequent battery replacement. Similar to terrestrial navigation procedures, underwater navigation systems need to be able to answer these two fundamental questions: (1) *Where am I now?* and (2) *How do I get to where I am going?* For GPS-based navigation, systems first determine the user's current location by GPS localization and then provide terrestrial navigation guidance based on a pre-established location database.

Another common method of terrestrial navigation guidance involves signage systems, such as visitor guidance boards in museums, campuses, or trails. These boards typically feature a tour map with notations (e.g., stars/dots) indicating the user's current location, allowing them to navigate to their desired destination based on the map's guidance [71]. In underwater environments, GPS is not viable, and other underwater acoustic/RF-based localization methods tend to be costly [89]. Consequently, divers traditionally rely on portable waterproof compasses and information provided by their guide before diving, which can be limiting in terms of intelligence, reliability, and flexibility [121, 45, 66]. Inspired by terrestrial navigation, we can adopt waterproof signage systems to show users rich location information for underwater navigation. This, however, has many challenges, as it is hard to find and read a finite-sized map image or messages underwater due to the harsh optical environment. Alternatively, we can use passive tags and a portable tag reader for more embedded and clear navigation information. In our daily life, passive optical tags such as barcodes and QR (Quick Response) codes are popular [81, 138], but their short communication range makes underwater navigation impossible because users cannot even find the tags to scan them. Increasing the size of the tag could indeed extend the communication range, but it comes with the trade-off of higher costs and a potentially greater disturbance to the original ecological environment. To circumvent the limitations of existing optical tags, we introduce the **U-Star** system in Chapter 4. This system is designed to offer a self-served navigation solution by leveraging the 3D spatial diversity of optical signals.

1.3.4 RoFin to Relieve Coarse Sampling in Vision Tracking

Human hands are not just crucial, vital organs for catching and grabbing; they have also long been used for communication, such as in greetings, sign language for the deaf, or hand signs in sports and wars. Hand poses have become direct, and cost-effective Human-Computer Interaction (HCI) across a wide variety of applications due to the fast development of computer technology and artificial intelligence (AI). For example, fingers and hands can be used in smart homes to control IoT devices for a variety of purposes (e.g., turning devices on/off), in interactive video games to provide a user-friendly and immersive gaming experience (e.g., accelerating race cars), and in XR

(AR, VR, and MR) enabled mobile applications to provide interactive operations that are close to reality (e.g., navigation) [59, 26, 137, 149, 142].

Vision-based hand gesture recognition systems have grown in popularity, simulating human vision to recognize hand shapes at a rate of roughly 60Hz [137]. Using deep learning, these algorithms attain an accuracy of more than 80%. They do, however, have limitations: (1) They struggle in poor light or at greater distances due to the camera's sensor receiving little light from the hand. (2) Cameras sample slowly (e.g., 60 Hz) when tracking fingers, mimicking human ocular limits and making it difficult to see detailed hand motions, such as tremors in Parkinson's patients [95, 24, 122]. (3) Complex hand form recognition with around 20 joints results in substantial processing costs and delays. (4) Privacy concerns arise when sensitive situations capture hand-related frames, thereby jeopardizing the privacy of persons [139]. To enhance finger tracking accuracy and reduce the overhead of hand pose reconstruction, we introduce the **RoFin** system in Chapter 5. This system is designed to offer fine-grained finger tracking and precise hand pose reconstruction by leveraging the 3D spatial-temporal diversity of optical signals for sensing.

1.3.5 PoseFly for Low-cost Joint Sensing and Communication

Currently, drones are primarily controlled by a centralized base station (CBS), such as a drone pilot on the ground or a satellite in orbit, utilizing the radio frequency (RF) spectrum [6, 36]. However, these centralized controlling techniques limit the potential use cases for drones since they lack mutual communication among drones. As a result, on-site data sharing directly among drones without the need for assistance from a centralized base becomes challenging. The requirement for each drone in the drone cluster to acquire commands from the CBS and transmit its status, including its surroundings and posture state measured by its inner sensors like IMU (Inner Measurement Unit), adds to the communication latency due to the centralized drone controlling mechanism. This can lead to significant localization errors, especially in high-motion scenarios, where the back-and-forth communication latency becomes a critical concern. As an example, consider two drones moving at a speed of 20m/s in opposing directions. The 0.25s required for location computation and communication between them would result in a 10m localization error ($0.25 \times 20 \times 2$). Furthermore,

as the number of drones in the cluster increases, the limited capacity of the RF spectrum becomes increasingly crowded. This congestion could lead to bit errors during retransmissions, exacerbating the localization error even further [144].

Optical camera communication (**OCC**) has garnered significant attention, particularly with the proliferation of commodity mobile devices equipped with built-in cameras. Compared to photodiode-based techniques like LiFi, OCC offers the advantage of low interference with ambient light. It also facilitates location-based services (LBS), enabling fine-grain AR navigation through the association of data from visible transmitters within a flexible communication range [148, 95, 24, 151, 124]. To enable low-cost localization and communication among swarming drones, we harness the 4D spatial-temporal diversities of optical signals and introduce the **PoseFly** system in Chapter 6.

1.4 Dissertation Organization

The rest of the dissertation is structured as follows. In **Chapter 2**, we provide a comprehensive exploration of the dimming side channel and illustrate how we leverage the 1D Spatial-Temporal diversity (i.e., 0D spatial with 1D temporal) to enhance the data rate of Li-Fi. Moving forward, **Chapter 3** delves into the details of our proposed RainbowRow protocol, which exploits 2D Spatial-Temporal diversities (i.e., 1D spatial with 1D temporal) through rolling strips to enhance optical camera communication. In **Chapter 4**, we introduce 3D hollowed-out optical tags (i.e., 3D spatial with 0D temporal) designed for underwater navigation, extending symbol distances in space. Shifting our focus to optical wireless sensing, **Chapter 5** presents the RoFin system, which leverages 3D spatial-temporal diversities (i.e., 3D spatial with 0D temporal) for fine-grained finger tracking and hand pose reconstruction. In **Chapter 6**, we delve into the use of 4D Spatial-Temporal diversities (i.e., 3D spatial with 1D temporal) for on-site pose parsing of swarming drones. Finally, we conclude this dissertation and discuss future research directions in **Chapter 7**.

CHAPTER 2

LIGHTING EXTRA DATA VIA 1D TEMPORAL DIVERSITY

Owing to the wide spectrum and rapid intensity switching capabilities of LEDs, optical wireless communication (OWC) holds tremendous promise for high-speed data transmission. In difficult conditions, many OWC systems switch from sophisticated, error-prone high-order modulation approaches to the more resilient On-Off Keying (OOK) modulation described in the IEEE OWC standard. In this chapter, we describe LiFOD, a new indoor OOK-based OWC system that can provide fine-grained dimming while maintaining robust communication at the same time with rates of up to 400 Kbps across a 6-meter distance.

LiFOD provides two crucial features. Firstly, LiFOD uses Compensation Symbols (CS) as a reliable side-channel to dynamically represent bit patterns for improved data rate. Secondly, LiFOD reconfigures optical data symbols (i.e., OOK symbols) and CS symbols placement algorithms in real time, optimizing them for fine-grained dimming and dependable decoding. Empirical tests using low-cost Beaglebone prototypes with commercial LED lights and photodiodes (PD) demonstrate LiFOD's superiority over state-of-the-art systems. LiFOD achieves 2.1× throughput boost based on the SIGCOMM17 data-trace.

2.1 Motivation

Considering the user experience of lighting, LED brightness may cause undesired flickers when transmitting data via the optical spectrum[2, 38, 124]. Meanwhile, dimming is essential to adjust light intensity for a variety of purposes and activities, such as office or hallway lighting, sleeping, reading, or other activities, with benefits that include reduced eye strain, mood setting, and LED life extension. Therefore, within the OWC standard [2], compensation symbols (CS) are employed in OOK modulation for smooth lighting and dimming control, while not affecting wireless communication. The entire PHY frame in OOK-based OWC is split into multiple subframes. In each subframe, a continuous number of CS symbols proportional to the length of the subframe are inserted in front of the OOK symbols (P, H, RF, DS fields) to adjust (i.e., increase, keep or decrease) average brightness (AB) smoothly.

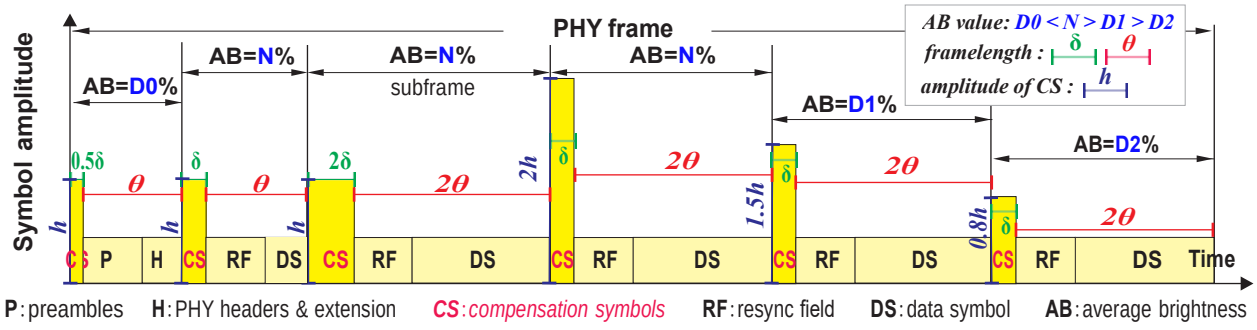


Figure 2.1 Illustration of OOK dimming control with compensation symbols (CS), redesigned from IEEE OWC standard [2]. A higher ratio of CS symbols in a subframe and higher CS symbol amplitude can both achieve higher average brightness (AB).

A tradeoff is observed when more control is needed to achieve fine-grained dimming, there is less of an opportunity for wireless communication transmission, which results in lower throughput[114, 2]. Moreover, CS symbols are solely used for dimming[147]. This consumes transmission resources in the time domain and limits the data rate of OOK, which already has a limited number of bits.

There are two key observations that motivate our approach. **(1) Bit patterns** [39, 40, 73] occur in transmitted bit-streams. A bit pattern is a bit sequence (i.e., multiple continuous bits), that frequently occur in traffic during a historical period. **(2) Compensation symbols** have not been used for data transmission in OOK-based OWC networks, as shown in Figure 2.1. In related dimming research[108, 128, 127, 133], approaches focus only on dimming itself without considering the potential for data transmission. However, we can use CS as a reliable **side-channel** to denote bit patterns for improved throughput considering the significant symbol distances between CS and OOK symbols.

To achieve these goals, we present LiFOD, which uses compensation symbols (CS) to not only assist dimming, as has been done in the past, but also to encode data bits primarily for better throughput in OOK-based OWC networks. In our method, CSs perform dual functions in dimming control and data transfer. A repositioned CS symbol inside the PHY subframe can signify a specific bit pattern within a transmitted sequence. Along with modulation, the transmitter performs a lightweight bit pattern discovery procedure on a regular basis and transmits the most recent bit pattern information to the receiver via preambles.

2.2 Background and Related Work

Single-color LED lamps are the most popular trend as a cost-effective choice for eco- and user-friendly residential lighting fixtures in our daily lives. Lighting and dimming are the **primary** functions of these LED lamps. Besides, photodiode (PD)-based OWC systems, such as OpenVLC and LiFi[30, 84, 64, 151, 126, 69, 126, 34, 19] with low-order modulations such as OOK, MPPM, and their varieties, treat wireless communication as **secondary** functions of these commercial LED lamps. We provide a primer of OWC dimming functions and modulation below to better define our research problem.

2.2.1 Dimming in OWC

Light dimming is defined as controlling a light source's perceived brightness based on a users' requirements. We classify the primary OWC dimming methods in the IEEE OWC standard[2] into two types, **coupled dimming with transmissions** and **decoupled dimming with transmissions**, as shown in Figure 2.2.

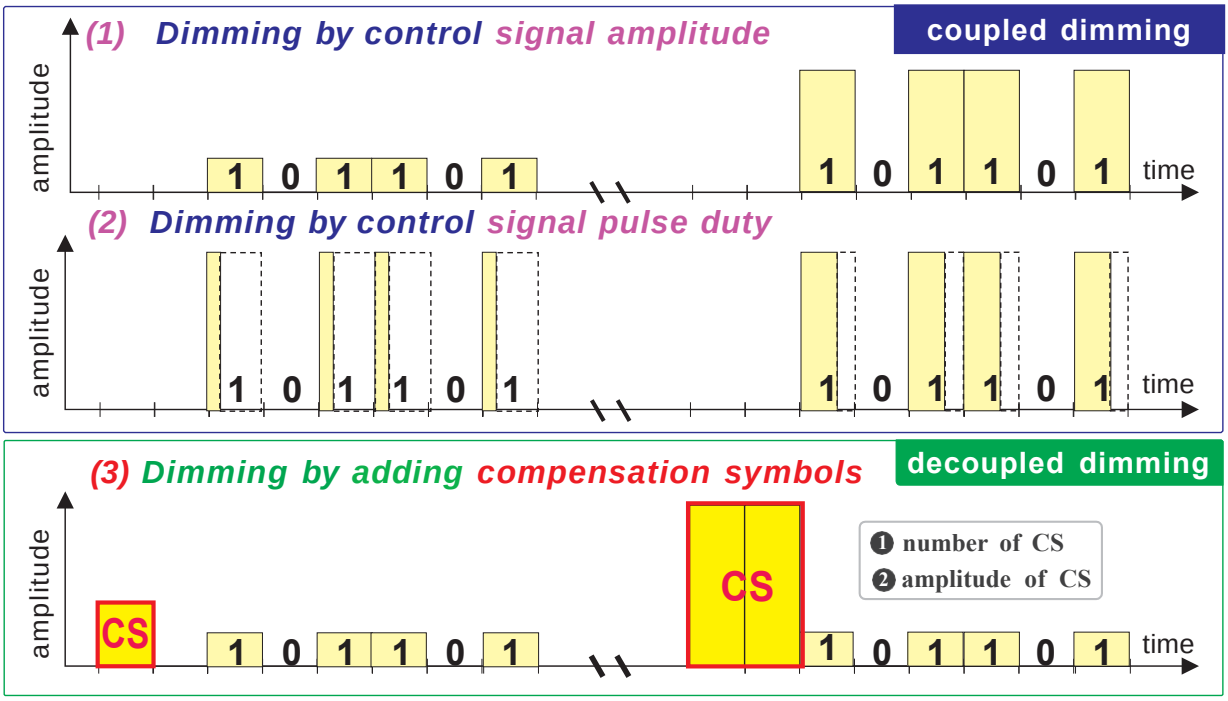


Figure 2.2 Couple/decoupled OWC dimming with transmission. Core idea of LiFOD: utilizing CS as robust side-channel to denote more bits.

For **coupled dimming with transmission**, the control signals' amplitude has no impact on the time slots/carrier bandwidth of transmission while the control signals' pulse width influences the carrier's bandwidth. As observed in SmartVLC[115], a drawback of fine-grain coupled dimming control is the lower throughput that can be achieved because complex modulations that allows fine-grained dimming control wastes transmission bandwidth and adds more error bits. The researchers proposed Adaptive Multiple Pulse Position Modulation (AMPPM), which designs super symbols to generate more pulse width combinations for fine-grained dimming. However, AMPPM is still discrete step dimming with more modulation cost than the same-order OOK.

Decoupled dimming with transmission inserts compensation symbols (CS) into the data frame and sends the constant brightness symbols of OOK modulation to adjust the average brightness of the light source. This treats data transmission and light dimming as two relatively individual modules with limited interaction. It has more robust communication and fine-grained dimming control while also providing the potential of using CS symbols to transmit extra data in comparison to coupled dimming methods. However, the CS symbols take up the time slots for data symbols compared with coupled dimming.

2.2.2 Communication in OWC

Besides lighting, it is also crucial to provide users with high-speed communication. Based on the receiver type and modulation, we classify OWC into two types:

(1) Camera-based OWC with high-order modulation. Image Sensors in commercial cameras can be treated as millions of single-pixel photodiodes (PD) and require more processing time than one PD [109]. The limited frequency response of the camera makes it hard to achieve a sufficiently high data rate as the switch speed of the transmitters is too fast for the frequency response of the receiver [124, 51]. Rolling shutter cameras on smartphones offer a frequency response only up to a couple of **tens of kHz**, which is well below the needed value for high speed communication of **hundreds of kHz**.

To overcome the bottleneck of camera-based OWC systems, many researchers[72, 123, 38, 124] focus on designing high-order modulation schemes to improve throughput. In [38], authors

proposed ColorBars to utilize Color Shift Keying (CSK) modulation to improve the data rate via Tri-LEDs. They achieved a data rate of up to **5.2 Kbps** on smartphones. Similarly, Yanbing et al. proposed Composite Amplitude-Shift Keying (CASK)[124] to improve the throughput of the Camera-based OWC system. CASK modulates data in a high-order way without a complex CSK constellation design. CASK achieves a data rate of up to **7 Kbps** by digitally controlling the On-Off states of several groups of LED chips.

These existing high-order modulations are high cost due to the necessity of specific devices and therefore cannot scale easily. For example, CSK modulation requires Tri-color LEDs as transmitters, which costs more than single color LEDs used in OOK and are quite unlikely to be deployed in real life[124]. CSK also needs a complicated and expensive receiver to precisely detect intensities of three colors: Red, Green, and Blue in the CIE color space chromaticity diagram[16].

(2) Photodiode-based OWC with primary modulation. Photodiodes (PD) are semiconductor P-N junction devices that convert the analog light signal into digital electrical current[57, 136]. PDs are single-pixel with a small surface area, which allows PDs to have a fast response time of sensing processing. This means the receiver can achieve a fast and robust symbol detection for high-speed communication. Most OWC systems, such as LiFi [30, 84] and OpenVLC[23, 115, 19, 69] adopt PDs as receivers for high-speed transmission and achieve a frequency response of a couple **hundreds of kHz**.

To suit a high-speed transmission frequency, PD-based OWC adopts primary and low-order modulations such as **OOK**. This occurs because it is non-trivial to demodulate higher-order optical symbols (e.g., 8-CASK, 32-CSK) at the PD-based clock speed of hundreds of kHz, due to reduced symbol distances compared to OOK symbols. Moreover, in poor optical channel conditions such as sunshine/underwater scenarios, the nonlinear effect of LED and short symbol distances makes them more complex and fragile with more error bits[106, 46, 67, 21, 131]. Higher-order modulations will bring more error bits and need more retransmission for the required BER. Thus most popular OWC systems such as LiFi [30, 84] switch from high-order modulations to low-order modulation such as OOK for robust transmission with a low BER in changing environments with poor channel

conditions. The latest version of OpenVLC[23] can achieve, on average, about **150 Kbps** at 4m under optical interference.

Our scope: We focus on the indoor OWC systems equipped with low-cost **PD** sensors and single-color commercial **LED** lamps, which are resilient lighting infrastructures. Our goal is to boost throughput and fine-grained dimming simultaneously without additional cost.

2.3 Our Approach: LiFOD

LiFOD consists of commercial LED lamp based transmitter and PD-based receiver. The architecture diagram and workflows of LiFOD are shown in Figure 2.3.

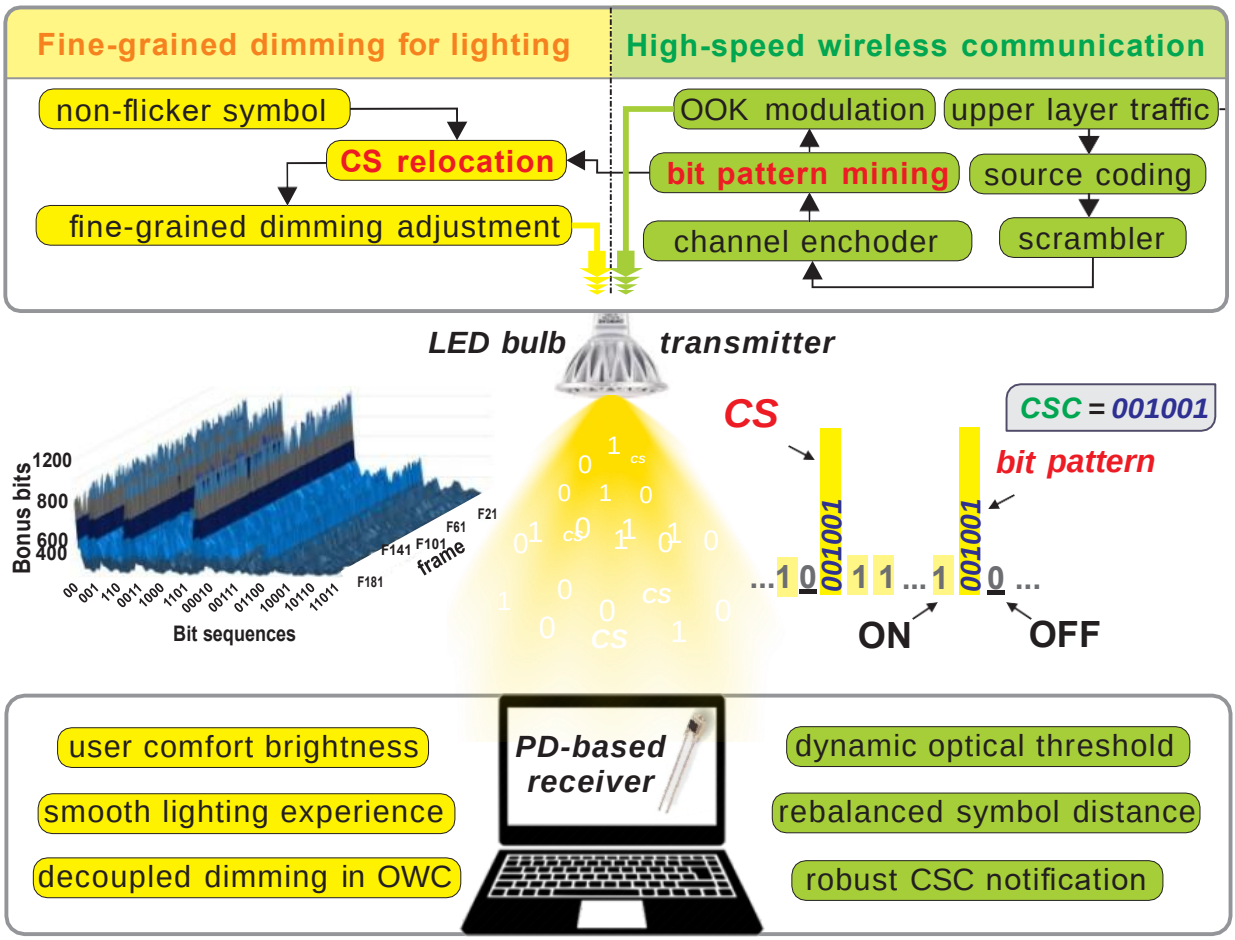


Figure 2.3 System architecture and workflow of LiFOD.

(1) **Dimming workflow**: After a user turns on an LED lamp, they may start OWC. They can smoothly control the dimming level by adjusting the knob (actual physical knob or virtual knob on

the IoT user interface). Manual adjustment is the most accessible and most fine-grained manner of dimming control as opposed to various communication-coupled dimming methods that can only provide digital and discrete step dimming. OOK symbols are constant brightness for data communication. In contrast, CS symbols are brightness-adjustable for fine-grained light dimming. Instead of the original continuous CS symbol insertion, LiFOD uses discrete CS symbol relocation to denote bit patterns without impacting CS-based smooth dimming and the detection of OOK symbols for robust communication.

(2) **OWC workflow**: Modulation occurs when Internet data from upper layers is encoded as optical data symbols. There are three essential network modules before OWC modulations defined in the standard [2]: Source coding, Scrambler, and Channel coding. Our introduced module in LiFOD is a lightweight ***bit pattern mining*** module added after these three network modules, but before modulation. Although scrambling and channel coding has already occurred, there are still some frequently appearing bit sequences (e.g., "001001" in the illustration). These are bit pattern candidates. In a real-world trace, SIGCOMM 2017[101], as shown in the middle left in Figure 2.3, multiple bit sequences appear in high frequency and introduce bonus bits (i.e., we can add CS symbols to assist transmission and achieve a higher data rate than current standards).

(3) **Overview.** We encode p-length bit patterns into a Compensation Symbol Code (**CSC**) as shown in the middle right in Figure 2.3. Each instance of a CSC code increases transmission speed because more bits are transmitted if $p > 1$. When allocating bits, we first check whether the next p bits match the predefined CSCs from our bit pattern discovery. If false, one bit is allocated to an OOK symbol as usual. On the contrary, we define it as a **hit** if the bits match the predefined CSCs. Instead of mapping only one bit to an OOK symbol, p bits are transmitted through a CS symbol. Once the receiver detects a CS symbol's existence, it inserts a p-bit CSC into the data stream. The receiver now can detect only one CS symbol that denotes p bits, instead of needing to detect p OOK symbols. Because (p-*1*) more bits (i.e., **bonus bits**) are transmitted when there is a hit and all symbol types/(ON/OFF/**CS**) are used for transmission, it is clear that the data rate of our system will increase.

2.3.1 Challenges and Solutions

There are two technical challenges that LiFOD should deal with. When a larger degree of control is necessary to accomplish exact dimming, the capability for wireless communication transmission is lowered, resulting in poorer throughput [114, 2]. Furthermore, using only CS symbols for dimming costs transmission resources in the temporal domain, limiting the data rate of OOK, which has a limited bit capacity by design.

In our design, CSs are used in both dimming controls and data transmission. A bit pattern in a transmitted bitstream can be represented by one relocated CS symbol in the PHY subframe. The transmitter periodically conducts lightweight bit pattern discovery in parallel with modulation and notifies the receiver of the latest bit patterns via preambles.

Network throughput improves remarkably due to improved data rate and decoding performance. **(1)** Data rate: CS symbols become data symbols without consuming transmission resources in the time domain. Moreover, each CS symbol carries more bits than an OOK symbol. **(2)** Decoding: CS symbols have a lower detection error rate than OOK symbols. Furthermore, the receiver decodes the CS symbol to its corresponding bit pattern directly instead of decoding multiple OOK symbols for that bit pattern, which reduces decoding error possibilities.

Our **contributions** are summarized as follows:

- We creatively exploit compensation symbols (CS symbols) to improve throughput. CS symbols were traditionally used only for dimming in OOK-based OWC systems. We explore bit pattern possibilities and propose a greedy mining algorithm to identify multiple bit patterns to maximize the overall throughput.

- We redesign non-flicker optical symbols (OOK and CS symbols) for smooth lighting and communication. This ensures the robust identification of symbol types in a changing environment. Initially, CSs are inserted continuously and proportionally into subframes for constant lighting. In our approach, CSs are relocated to discrete locations to denote bit patterns, which may introduce undesired flickers, however, we also design CS relocation schemes for stable

lighting.

- We implement a LiFOD prototype on commercial devices and validate its lighting and communication performance in different transmission settings. Our comprehensive evaluation results demonstrate that **LiFOD** can achieve up to **400 Kbps** up to **6m** with fine-grained dimming, effectively **doubling** throughput at a longer range compared with SmartVLC on the SIGCOMM17 datatrace.

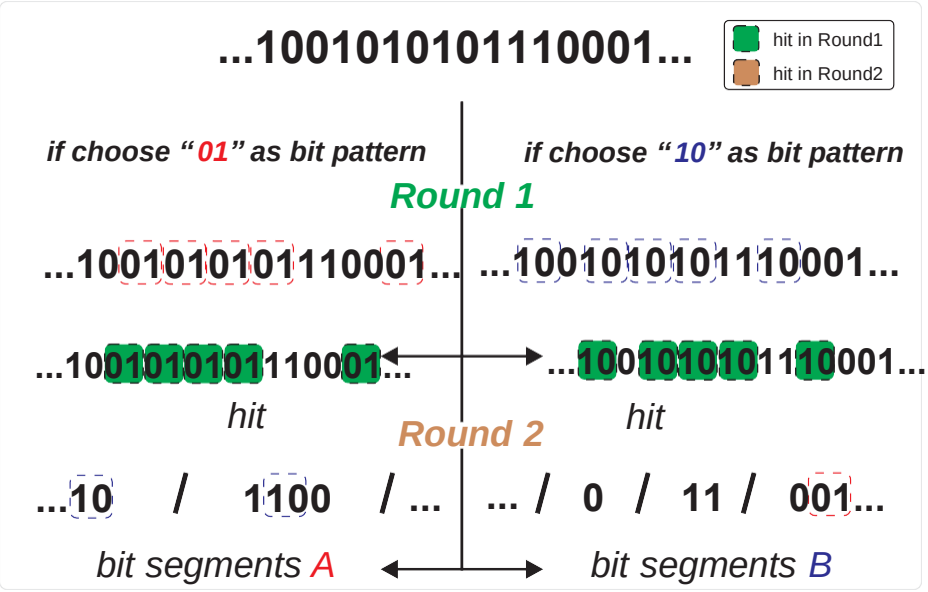
2.4 Bit Pattern Discovery

2.4.1 Mining Challenges.

Throughput improvement depends on the length of p and the hit rate in a given data frame. For example, as the length of a bit sequence increases, the probability of a hit decreases, and vice versa. There is a clear tradeoff between bit sequence length and hit probability. Moreover, not only one bit sequence is likely to be a bit pattern. When one bit sequence is selected as a bit pattern, the bitstream will be split by this bit pattern. After one bit pattern is assigned, depending on which pattern is chosen, the resulting allocation of the data bits is wholly changed. The next challenge, is to decide which pattern will be selected as the next bit pattern. All options need to be explored based on the choice of the previous bit patterns.

An example is illustrated in Figure 2.4. Suppose the bit sequence "01" appears most often when allocating the bitstream "...1001010101110001...". Also, it offers the maximal bonus bits when compared with other potential bit sequences. In this case it is $(2-1)\times 5 = 5$ bonus bits. We may encode bit sequence "01" as one type of CSC. However, other bit sequences may also exist, such as "10", which often appears and brings the same level of bonus bits as "01", $(2-1)\times 5 = 5$. A challenge of LiFOD is deciding which bit sequence, in this case "01" or "10", should be selected as the bit pattern. **(1)** If we choose "01" as the bit pattern, the bit stream will be split into three bit segments: "...10", "...1100..." and "...". **(2)** If choosing "10", the bit stream will be split into four bit segments: "...", "0", "11", and "001...".

Additional bit sequences also frequently appear in the split bit segments produced after the first



sequence	hit in *R1*
00	2
01	5
10	5
11	1

sequence	hit in *R2-A*
00	1
01	0
10	*2*
11	1

sequence	hit in *R2-B*
00	*1*
01	*1*
10	0
11	*1*

Figure 2.4 Bit pattern candidates change in next round.

round of bit pattern selection. These sequences can be chosen as another bit pattern to further speed up the data rate. However, the bit pattern selected for a specific round impacts the bit pattern choice for the next round, and previously discovered bit pattern candidates in earlier rounds may not be candidates anymore. When choosing bit patterns, we need to consider the total bonus bit performance of all chosen bit patterns of all rounds.

2.4.2 Identify Patterns Greedily.

To address the problem above, we execute bit pattern mining in multiple rounds shown in Figure 2.5. The bit pattern for each round will be selected as different types of CSCs. After several rounds of mining, there will be less opportunity to find bit patterns because bitstreams have already been split into short-length segments. Consequently, any obtained bonus bits will decrease as the number of rounds increases. Furthermore, if there are too many types of CSCs, the compensation symbol design for modulation will be more complicated and therefore increase the error rate of demodulation. Therefore, the choice to continue bit pattern mining is a tradeoff between increased data rate and error rate. The number of rounds we run for bit pattern mining depends on the bonus ratio for each round. The bonus ratio is defined as the ratio of bonus bits introduced by CSC for a specific round to bit numbers of the entire data frame. When the bonus ratio is less than **10%**, bit

pattern mining stops at that round, and any previously mined bit patterns are chosen as CSCs.

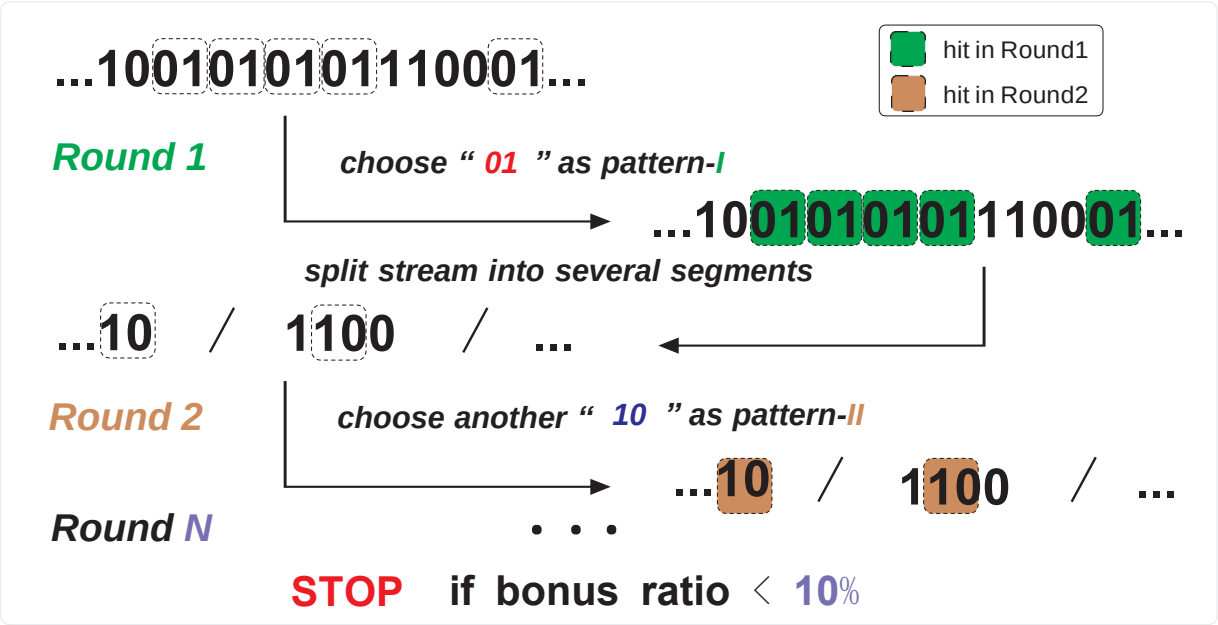


Figure 2.5 The illustration of multiple rounds mining.

According to analysis above, we design a lightweight greedy algorithm to explore bit patterns and summarize multiple rounds algorithm. The goal is to bring the maximum number of bonus bits possible in each mining round locally and obtain the maximum bonus bits of all mining rounds globally.

Based on our experimental results, we've determined that with a bit sequence length larger than six bits the total number of bonus bits we gain starts to fall, and therefore we search for bit sequences whose length is up to 6 as bits long. The number of bit sequences possible is $\sum_{i=2}^{6} 2^i = 124$. We scan each of them in the frame, count hit number, and calculate bonus bits. We then choose the bit sequence with the most bonus bits as the bit pattern at that mining round. We calculate the bonus ratio of the bit pattern for each round and compare it with the 10% threshold. If the bonus bit ratio less than the threshold, mining will stop at that round.

2.4.3 Ablation Study of Bit Pattern

Real-world Daily Data-trace. The OWC backhaul is connected with the Internet[30]. We conduct CSC code abstraction based on two sets of real-world wireless traffic data from the (1)

SIGCOMM 2017 trace [101], which is the recorded wireless network activities at the SIGCOMM 2017. (2) Another trace is from CAIDA 2019 [102], which collects the daily network traffic of a city in the US. These data packets are scrambled and encoded with the convolutional encoder specified in the IEEE 802.11 standards.

Bonus Bits Distribution and Potentials. Figure 2.6 shows heat maps of our bit pattern mining results in Round 1 and 2 among different frames from our two traces. There are more bit pattern candidates in Round 1 (i.e., six strongly highlighted columns). In Round 2, there are fewer bit pattern candidates (i.e., two significant highlighted columns) and the bonus bits in Round 1 are much more significant than Round 2. It implies that there are abundant known bits in the first round of mining used because of the high probability of having a hit on the CSCs. In high-order rounds, opportunities to use CSCs are few.

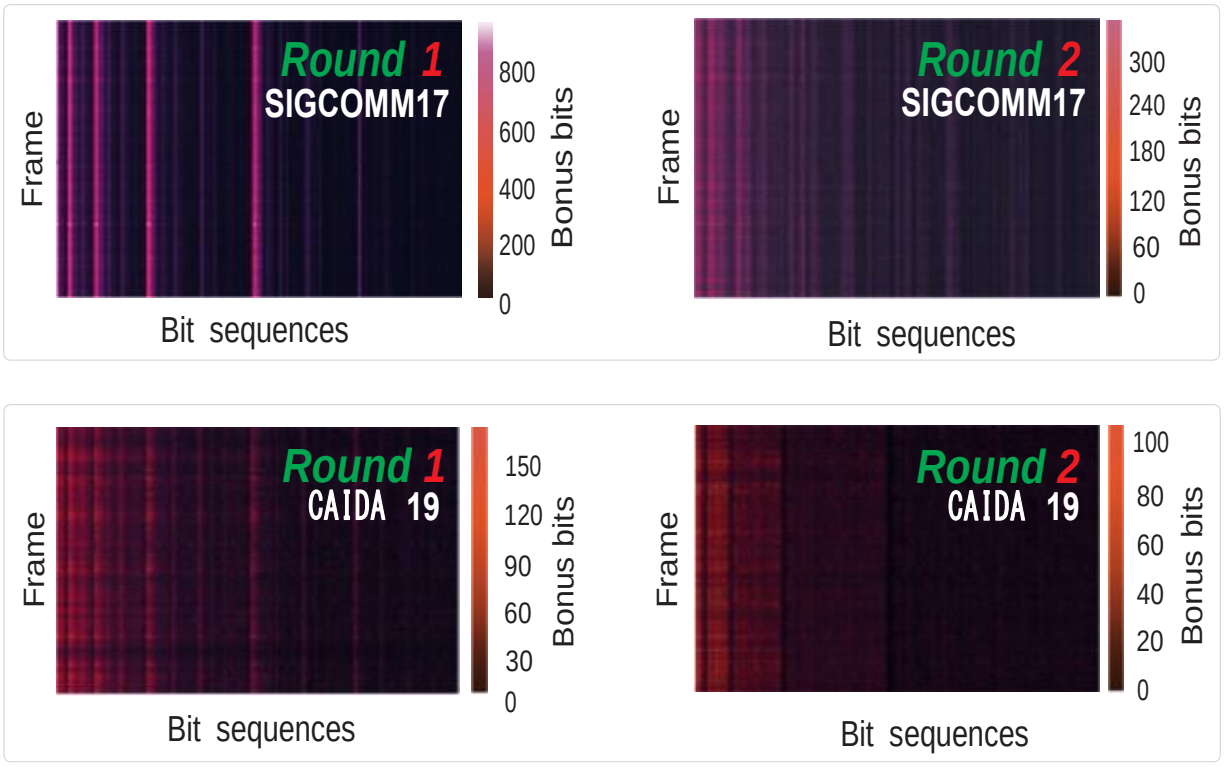


Figure 2.6 Bonus bit heat maps for two rounds mining on two daily traffic: SIGCOMM17[101] and CAIDA19[102].

Tricks of CSC Decision in a Round. In general, the decision to choose a particular bit pattern

candidate as a CSC code for each round depends on their bonus bits. However, if two bit pattern candidates have identical bonus bits, as occurs in Round 1 of the SIGCOMM17 trace shown at the top in Figure 2.7, we choose the longer bit pattern candidate “000000” as the bit pattern even if other bit pattern candidates have the same bonus ratio performance for that round. The reason is that when two or more bit pattern candidates have identical bonus bits the longer one will make the bit segments shorter after splitting the longer bit pattern. Thus, there will be less hits in the next round which means there will be more CSC-I and less CSC-II.

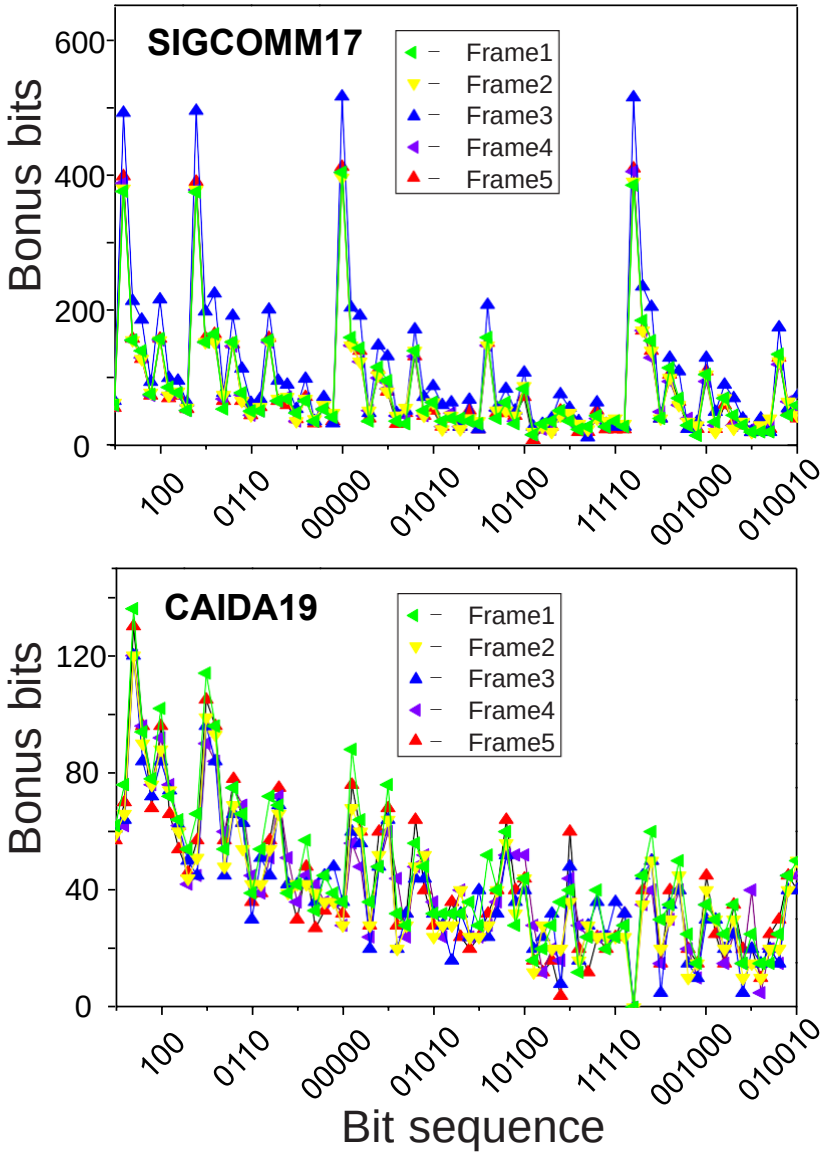


Figure 2.7 CSC decision tricks in a mining round.

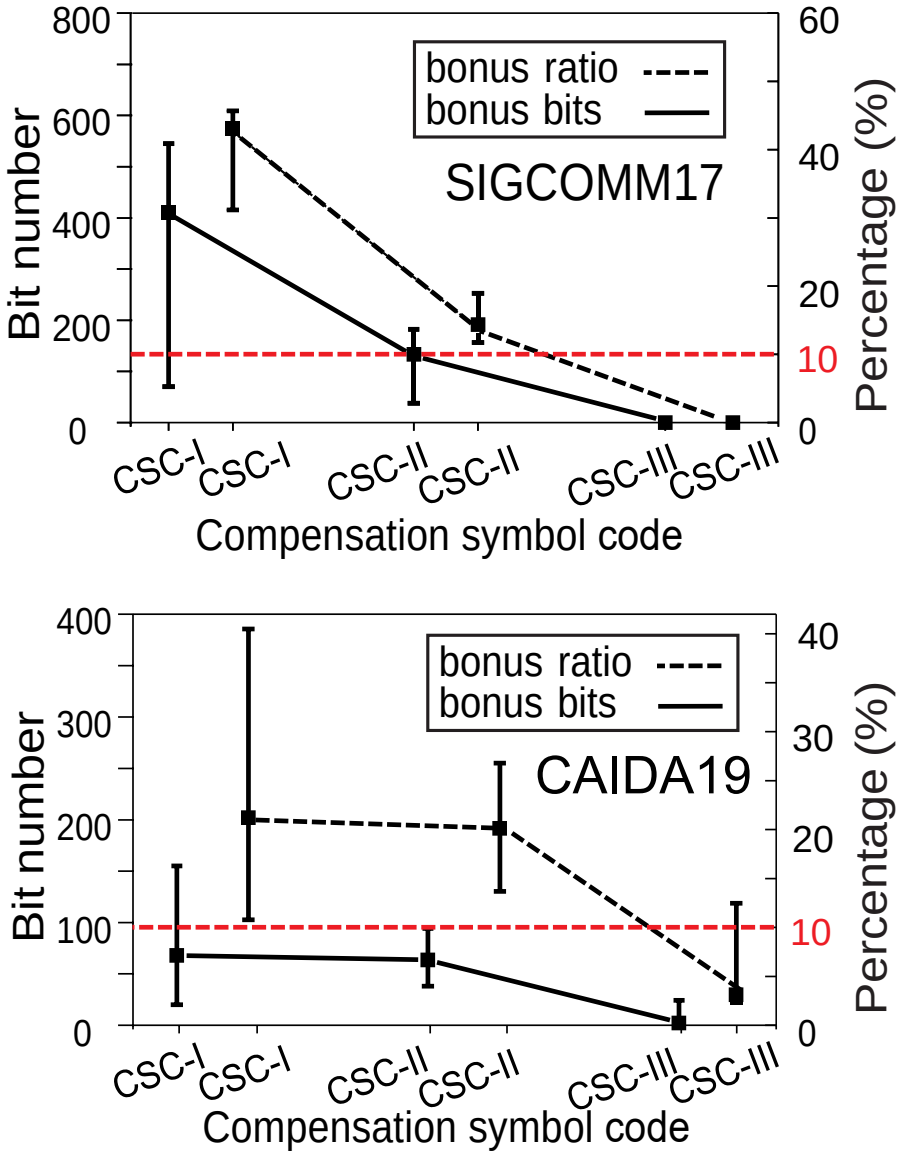


Figure 2.8 Two CSC can embed considerable extra data.

Two CSC with Considerable Exta Data. Figure 2.8 shows that in Round 1 of mining, more than 40% of all bits are transmitted as bonus bits through CSC-I of the SIGCOMM17 trace. The CAIDA19 trace, also achieves a bonus ratio of more than 20% for CSC-I. As the number of mining rounds increases, a lower percent of bonus bits can be used, however, the bonus ratio is still above 10% Round 2 in the SIGCOMM17 trace. The bonus ratio in Round 2 for the CAIDA19 trace remains near 20%, showing almost no decline from Round 1. In Round 3 of mining for both traces, the bonus ratio falls below the threshold of **10%**, and subsequently, the mining stops after Round 3.

Finally, we choose **two CSCs** (CSC-I and CSC-II) that will be used for transmission. The total bonus ratio of the two rounds of mining on two real-world traces is, combined, more than **40%**. Although the transmission rate benefits less directly from bonus bits when utilizing CSC-II, it still provides decoding benefits from the known bits represented by CSC-II. Overall, the more bits represented by CS symbols, the fewer opportunities for the false detection for OOK symbols.

Delay and Overhead Measurement. We analysis and measure the overhead of bit pattern mining based on real-world data traces. The results of execution time and memory overhead of our greedy bit pattern mining are shown in Table 2.1 and Table 2.2. The bit pattern mining process for SIGCOMM 17 and CAIDA 19 consumes 0.78 s and 0.37 s in average, which is short enough as normal delay time before transmission. The computation cost of our pattern mining for SIGCOMM 17 and CAIDA 19 data-traces are both 144 MiB of memory in average, which is pretty low even compared with the computation abilities of MCU devices such as BeagleBone Black device (512MB RAM). The results show bit pattern mining of LiFOD is lightweight, real-time, and thus suitable for usage in the real world.

Two real-world data trace	Execution Time (s)								
	Round 1			Round 2			Total		
	min	max	ave	min	max	ave	min	max	ave
SIGCOMM 17	0.44	0.87	0.61	0.12	0.24	0.17	0.56	1.67	**0.78**
CAIDA 19	0.11	0.38	0.22	0.07	0.25	0.15	0.18	0.63	**0.37**

Table 2.1 Delay measurement of bit pattern mining on two real-world data traces.

Two real-world data trace	Memory Overhead (MiB)								
	Round 1			Round 2			Total		
	min	max	ave	min	max	ave	min	max	ave
SIGCOMM 17	72	72	72	72	72	72	144	144	**144**
CAIDA 19	72	72	72	72	72	72	144	144	**144**

Table 2.2 Overhead measurement of bit pattern mining on two real-world data traces.

2.5 Fine-grained Dimming via CS

2.5.1 Non-flicker Symbol Design

Flicker is the temporal modulation of lighting perceivable by the human eye, which can negatively affect a user's lighting experience. The maximum flickering time period (MFTP) is the maximum time period over which the light intensity can be changed and not sensed by human eyes. Thus any brightness changes over periods longer than MFTP must be avoided (i.e., significant low frequency brightness changes cause flickers and should be mitigated)[2].

In the current standard, OFF/ON and CS symbols have different amplitudes, and as shown in Figure 2.9, CS-I and CS-II also have different amplitudes. The random distribution of CSCs encoded by LiFOD that appear in PHY frames at low frequencies causes significant flickering. To address this, our flicker-mitigation solution is inspired by Manchester coding [2], where each symbol is extended to include itself and its complementary symbol. This guarantees that any significant brightness changes will appear too fast to be sensed by human eyes.

There are three amplitude scales in the new symbol design: B0, B1, and B2 (brightness: B0 < B1 < B2) for OFF, ON, CS-I, and CS-II symbols instead of four brightness amplitudes in the original symbol design. Symbol **OFF** is designed as **B0+B1**. In the first half of a symbol's duration, it has an amplitude of B0. In the second half of a symbol's duration, it has an amplitude of B1. Similarly, symbol **ON** is designed as **B1+B0**. And we design **CS-I** as **B2+B0**, while **CS-II** is **B0+B2**. Our newly designed symbols only need two thresholds rather than three for demodulation, decreasing the complexity and load of symbol detection. This increases the symbol distance and decoding robustness further. Additionally, CS-I and CS-II have the same brightness in our non-flicker symbol design, which further reduces the flickering possibility compared to the standard symbol design.

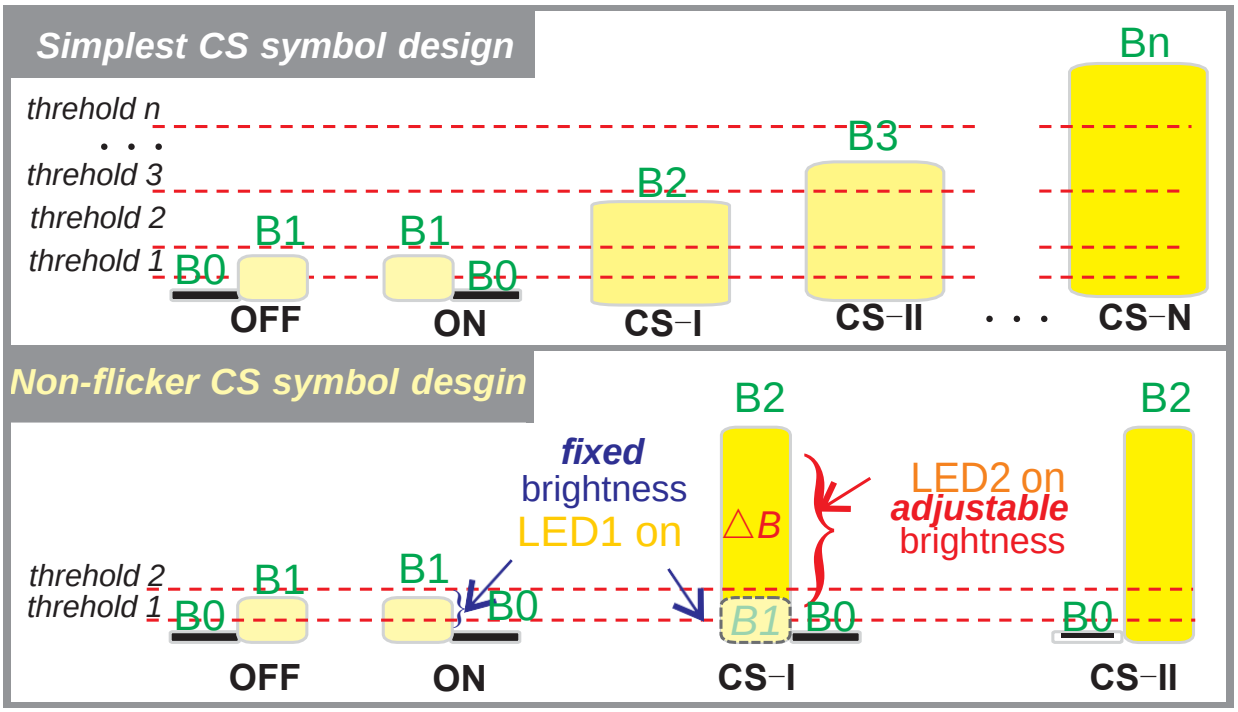


Figure 2.9 Non-flicker optical symbol design in LiFOD.

Note that there exists more CS-I symbols than CS-II symbols. It is easier for the receiver to distinguish the amplitude difference between B2 and B0 than between B1 and B0. Suppose a symbol has an amplitude of B2 in the first half of symbol duration. In this case, the symbol will be decoded as one CS-I symbol directly without estimating the amplitude of the second half symbol duration. That is why we design the CS-I symbol as B2+B0 instead of B0+B2. This design decreases the detection error rate (DER) of the CS-I symbol, which carries more data than the CS-II symbol. Finally, this benefits total throughput and BER performance.

2.5.2 Compensation Symbols Relocation

Fine-grained dimming control. LiFOD consists of two commercial LED lamps that are controlled synchronously shown in Figure 2.10. The transmitter sends out OOK symbols via LED1 and sends out compensation symbols via LED1 and LED2 together. LED1's brightness is set by the user and fixed before OWC begins. Users can continuously adjust LED2 by the dimmer knob to provide the additional brightness of (B2-B1, i.e., ΔB) to increase or decrease the average brightness (**AB**) without impacting optical symbol detection. This saves transmission bandwidth and does

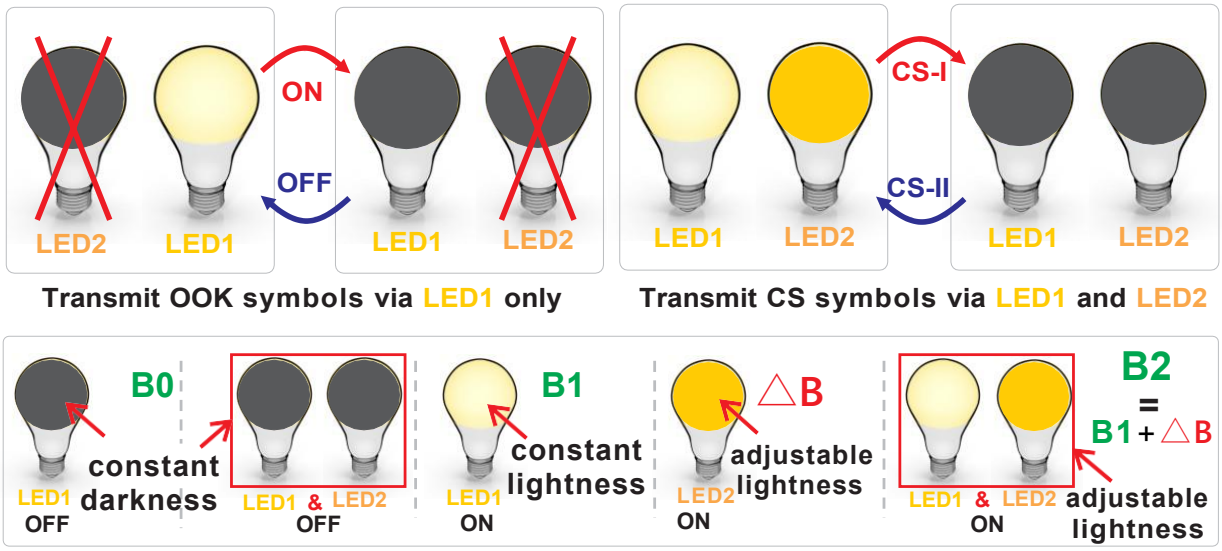


Figure 2.10 Two commercial LED bulbs (<$10) in LiFOD.

not affect symbol decoding. The number of CS symbols is proportional to each frame's length to guarantee the same AB between frames. This mitigates **inter-frame** flickers and keeps constant brightness, even after an updated dimming is set.

Random CSC Locations and Numbers. There are subframes in each frame. Currently, compensation symbols are continuously inserted into subframes for dimming control in the IEEE OWC standard[2]. However, these are incapable of denoting the bit patterns that may appear discretely in the bitstream of one frame for transmission. Moreover, the hit numbers of CSC-I and CSC-II are not always the same in subframes, even though different subframes should have the same brightness to reduce **intra-frame** flickers. This means each subframe should have an equal proportion of CS-I and CS-II symbols.

CS Relocation. In Figure 2.11, there are 40 OOK and CS symbols in each subframe. We set $\frac{1}{5}$ of the symbols (i.e., 8 CS symbols) for dimming to keep a constant AB of the subframe. There are 8 CS symbols at the beginning of each subframe initially. If there is a CSC-I/II in the subframe, we put one CS-I/II symbol in that location. These picked CS-I/II symbols are used both for dimming and assisting transmission. The left redundant CS-I/CS-II symbols at the front part of the subframe are only used for dimming. The CS symbols only used for dimming are separated by the resync field (RF) with symbols used for transmission (OOK and picked CS symbols). We only

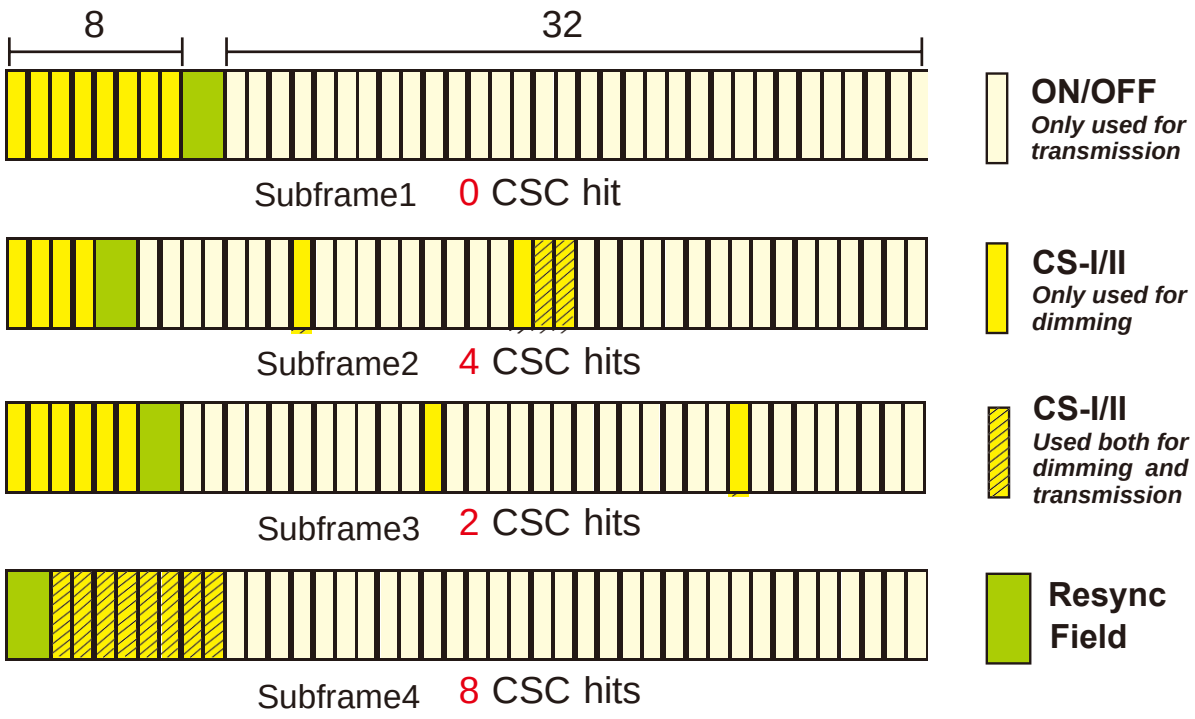


Figure 2.11 CS symbol relocation scheme.

decode the symbols after the RF field. Compared with the original, continuous CS symbols, CS relocation provides the potential to create robust side-channels for data transmission and mitigates the flickering possibility further as an unintentional benefit while keep constant brightness.

2.6 Robust Decoding of CS

2.6.1 Dynamic Optical Threshold

As shown in Figure 2.9, the receiver checks grayscales of two parts in one received symbol to identify its symbol type by its grayscale threshold. In LiFOD's non-flicker design, there are three brightness levels B0, B1, and B2. The receiver distinguishes them based on grayscale thresholds informed by a preamble from the transmitter.

However, as shown in Figure 2.12, a received grayscale is not identical to the one transmitted by the transmitter under four different dimming levels (i.e., B2's incremental brightness). The received grayscale of different brightness may overlap with others, and B2 in different dimming settings can influence the perceived brightness of B0, B1 due to their continuous distribution in the PHY frame. To identify an optical symbol's type with varying brightness, the receiver should

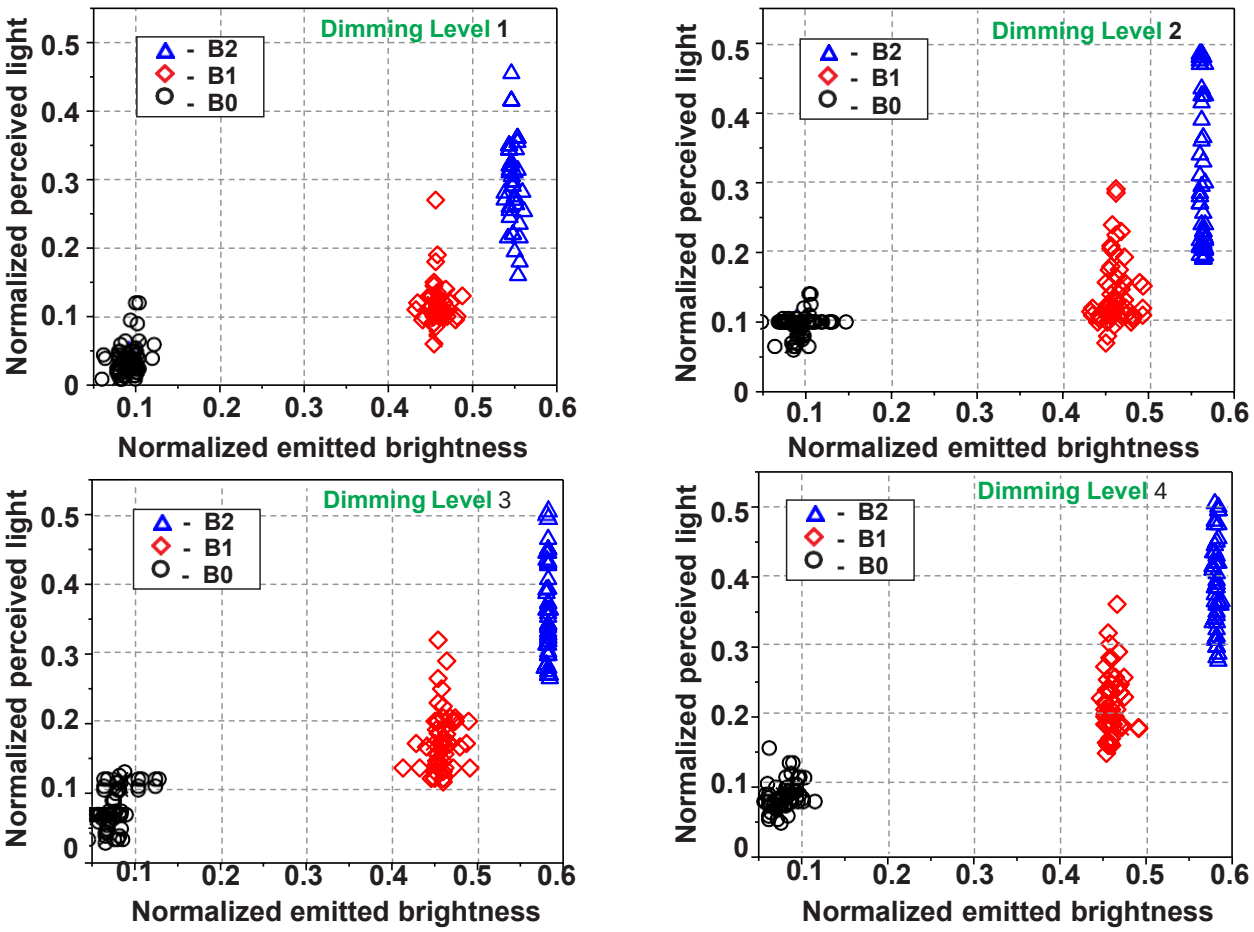


Figure 2.12 Grayscale diagram of B0, B1, B2 on four incremental dimming levels.

be informed of dynamic thresholds among B0, B1 and B2 via a **preambles** from the transmitter. Grayscale thresholds are measured and calculated based on short training symbols in the preamble field. The threshold values are dynamically adjusted based on the measurement informed by the preamble.

2.6.2 Rebalanced Magnitude Distance

In addition to our dynamic threshold measurement with preambles for different dimming settings in varying environments, we also need to combat any environmental influences. When an optical signal radiates away from its transmitting light source, the signal spreads out in different directions. Parts of spreading light beams reflect off objects and arrive at receiving light sensors from different paths. Consequently, different ambient light brightness will impact detection of original optical symbols.

If the ambient light is weak, the brightness of B1 or B2 will dominate the receiver's sensed intensity. When ambient light gets stronger, the ambient light will dominate the received brightness and the brightness of B0, B1, and B2 will have a similar high grayscale level, as shown in the left of Figure 2.13. The same case happens when the transmission distance increases. When the transmission distance between transmitter and receiver becomes larger, ambient light will dominate the receiver's brightness as well, as shown in the right of Figure 2.13. The intensity of B0, B1, and B2 will have a similar low grayscale level.

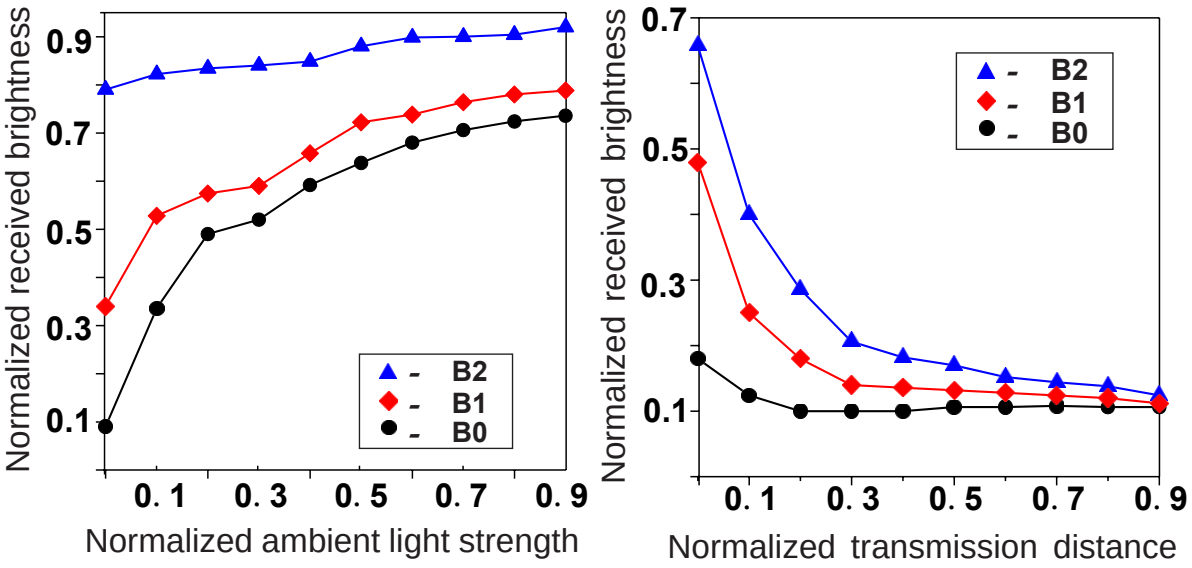


Figure 2.13 Influence of ambient light and distance.

These two factors significantly cause the perceived magnitude of brightness transmitted to be harder to distinguish from one another, and therefore, the received symbol is not identical to the transmitted optical symbol. We need to estimate the optical channel response using the **preamble** to further conduct equalization to eliminate the influence of ambient light and transmission distance.

Suppose optical channel response O is $H(O)$ and the transmitted brightness is $\boldsymbol{b}$. The received brightness is

$$b' = H(O)b \tag{2.1}$$

A sequence of known brightness values in the preamble, $\boldsymbol{S}$, are transmitted to help estimate channel

response. *H(O)* is estimated as

$$\hat{H}(O) = \frac{S'}{S} \tag{2.2}$$

where received brightness ***S'*** includes an ambient light and transmission distance factor. The $\hat{H}(O)$ is not equal to *H(O)* due to other noises such as the temperature variation and noise figures at receiver, but it is still well estimated because ***S*** is known at receiver.

The subsequent brightness magnitudes, ***x***, B0, B1 or B2, are finally estimated by multiplying received brightness ***x'*** with the multiplicative inverse of the estimated optical response of channel $\hat{H}(O)$:

$$\hat{x} = \frac{x'}{\hat{H}(O)} \tag{2.3}$$

2.6.3 Robust CSC Notification

Preambles are used in LiFOD to notify the receiver of the CSC codes used in our system. The IEEE 802.15.7 standard [2] defines the format of the Physical Protocol Data Unit (PPDU). The PHY frame consists of a synchronization header (SHR), a PHY header (PHR), and Physical Service Data Unit (PSDU). The SHR contains the preamble field. CSC-I and CSC-II are prepended to the data packet in the preamble field to inform the receiver of the bit patterns being used. The receiver stores CSC codes and understands that they are specified for CS-I and CS-II symbols separately.

When the receiver estimates the transmitted brightness magnitude by dividing the estimated optical response of channel $\hat{H}(O)$, the absolute magnitude change on a symbol with a lower magnitude is lower than that on a higher magnitude symbol, as shown in Figure 2.14. For example, if the estimation is that a received symbol should be magnified by 20%. The absolute magnitude changes of symbols are different. Low magnitude symbols have a minor error margin, while magnitude errors of high magnitude symbols are scattered in a broader range than that of low magnitude symbols. Because LiFOD only adopts three brightness magnitudes (B0, B1, B2) in symbol design, the equalization can successfully eliminate the influence of the varying environment.

When using compensation symbols for transmission, the dimming will not impact the ON/OFF symbol identification due to the smaller magnitude estimation error margin of B0 and B1 in OOK

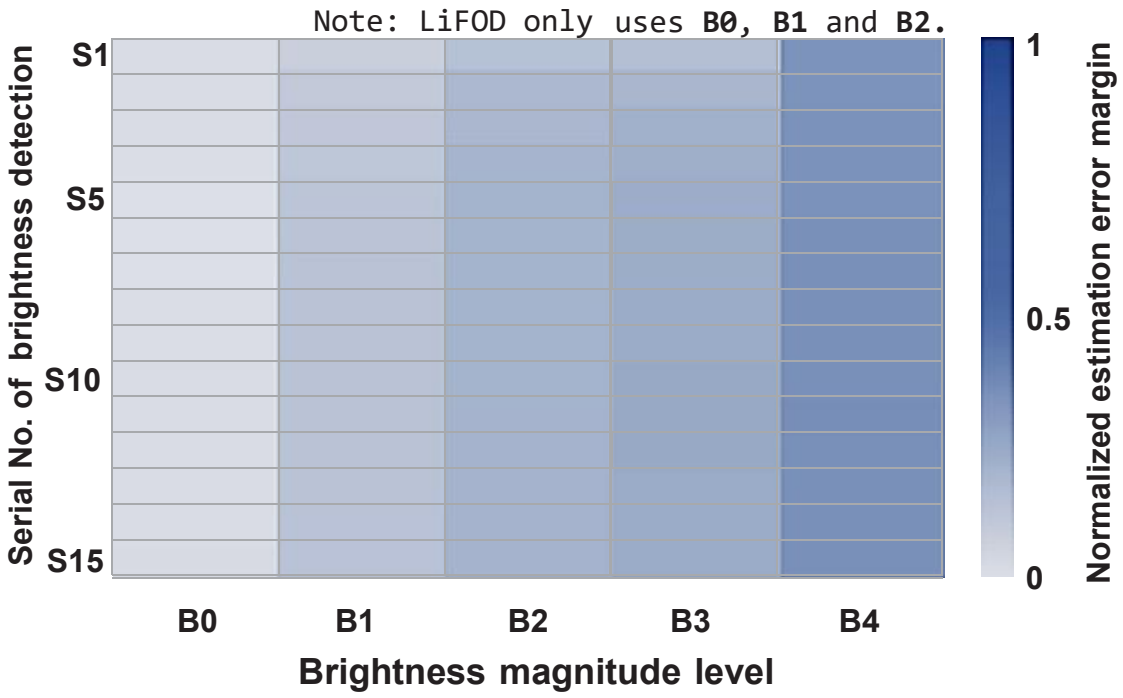


Figure 2.14 The normalized magnitude estimation error margin of 15 detections in varying environment.

symbols than B0 and B2 in CS symbols. Nevertheless, suppose there are too many types of CS symbols. In that case, the decoding performance of CS symbols with a higher magnitude will get worse due to the broader estimation error margin. LiFOD uses two CS symbols with B0 and B2 brightness magnitudes, ensuring robust CSC notification.

2.7 Implementation and Evaluation

2.7.1 Hardware

Transmitter. Our LiFOD transmitter consists of several commercial components: two regular LED lamps (LED1, LED2), and MOSFET and BeagleBone Black (BBB) boards, as shown in Figure 2.15. LED1 is used to generate constant-brightness OOK symbols, LED1 and LED2 are used to generate variable brightness CS symbols. They are controlled uniformly by the BBB board. Because BBB can only provide 3.3V control signals, which can not drive high-power LEDs, we use a MOSFET transistor as a fast switch to drive the LEDs. To provide variable and fine-grained dimming, we wired a potentiometer as a dimmer knob between the DC power with the LED positive lead. We removed the AC-DC converter in our daily LED lamp, which affects the ON-OFF

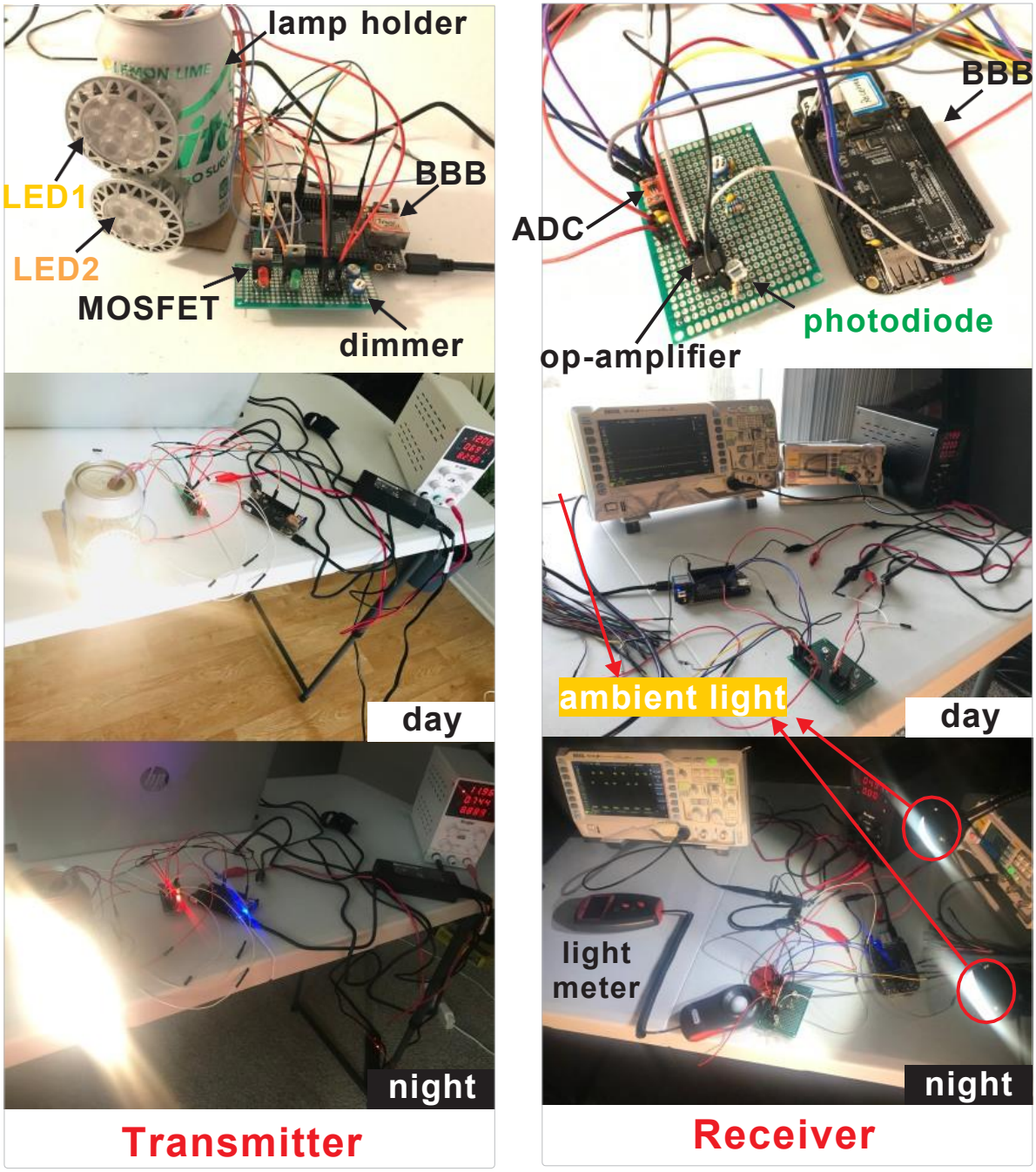


Figure 2.15 LiFOD prototype: transmitter, receiver and experiment scenarios in day and night.

switching speed significantly.

Receiver. The LiFOD receiver prototype has three main components: analog-digital converter (ADC), operational amplifier (OPA), and the photodiode (PD), as shown in Figure 2.15. The light is sensed by the PD to convert the light signal to a small current and amplified by the OPA. Finally, analog values are converted to digital values in the SPI data format. SPI data is then processed to estimate analog light intensities for symbol decoding. The driving circuit can be fully powered and controlled by the BBB.

System cost. The system cost of LiFOD is shown in Table 2.3. The Beaglebone Black board ($80) in our prototype can be fully replaced with Beaglebone pocket($37), which is cheaper. Thus, totally including transmitter and receiver, the LiFOD system costs less than $100.

Component	Brand/Model/Type	Unit Price (USD)
LED Bulb	BAOMING-5W-MR16	**4.2**
MOSFET	BOJACK-30N06LE	**0.7**
Photodiode	OSRAM SFH206K	**1.4**
Op-amplifier	Todiys-TLC272	**2.4**
ADC	TI-ADS7883	**3.2**
potentiometer	HUAREW-PTM15	**0.1**
BBB board	Beaglebone-Black or Pocket	**80** or **37**

Table 2.3 Price table and system cost of LiFOD.

2.7.2 Software

There are two main tasks on the software side: (1) send out optical symbols at high speed from the transmitter; (2) demodulate received optical symbols at high speed with reliability on the receiver. We use low-cost BBB platforms. Ideally, the PRU of BBB can achieve high-frequency modulation and demodulation at the *200MHz* level. But due to significant distortion of light signals generated by commercial LED lamps at such high transmission frequencies, and we set the transmission frequency at *hundreds KHz* level, which is the same as the state-of-art SmartVLC or OpenVLC. Other software modules, such as our lightweight bit pattern mining and CS relocation,

as shown in Figure 2.3 are run on the BBB as firmware to provide services among the PHY layers and upper layers.

2.7.3 Setup

(1) Dataset. We choose two real-world datasets SigCOMM17 and CADIDA19, to simulate user's daily Internet traffic. **(2) Transmission frequency.** We set the transmission frequency to be lower than 200KHz. **(3) Sampling rate.** To better detect the optical symbol shape, we set the ADC sampling rate to 1.2MHz, six times of transmission frequency. **(4) Ambient light setting.** Based on real-world scenarios, we conduct experiments in a 4 x 8 m^2 living room in the day and night scenarios. **(5) Dimming setting.** We set the dimming level by adjusting the dimmer knob neatly and using a light meter to measure its granularity.

2.7.4 Lighting Performance

Fine-grained dimming: The brightness of LiFOD can be manually adjusted to any continuous setting. We evaluate ten incremental dimming levels at different distances, as shown in Figure 2.16. The dimming range is from 0 lux to 450 lux, which meets the office lighting requirement from U.S. General Services Administration [27]. In the different dimming setting index, the brightness sensed by the user increases depending on the day or night scenarios. The experiment results prove that the dimming function works well.

Non-flicker performance: We measure the non-flicker performance with the light meter based on the photometric quality, which measures the foot candle (FC) value range from its maximum to minimum values. The more extensive range of FC values, the more flickering possibility. When the transmission frequency increases, the flicker possibilities reduce for the two optical symbol designs. Figure 2.16 shows that users sense no flickers since the transmission frequency for LiFOD's non-flicker symbols are lower than the original optical symbols. Due to the unexpected low frequency of CS symbols, LiFOD's non-flicker symbols will provide more smooth lighting without flickering than the original symbol design, even at a very high transmission frequency such as 200KHz. Results show that our flicker-mitigation solution addresses the flicker well.

We also investigate users' perception of flickering and comfortableness of lighting, as shown

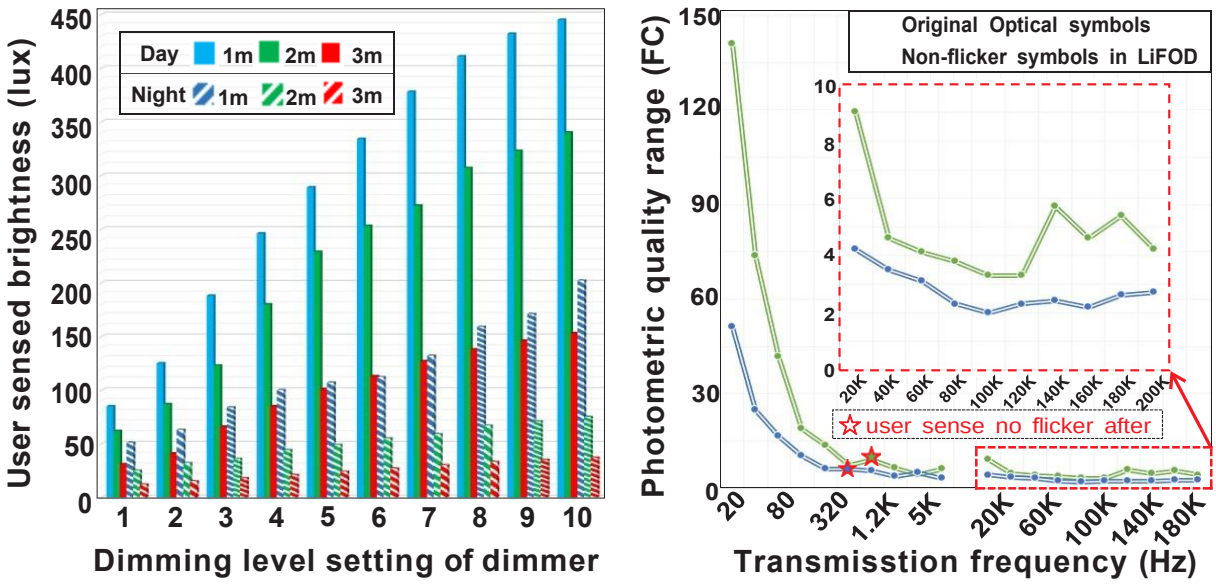


Figure 2.16 Dimming and non-flicker evaluation.

in Table 2.4. Three volunteers are invited to experience the lighting function of LiFOD. Each user scores their user experience for at 10 dimming settings in different conditions such as facing directly or indirectly, at different distances to LED lamp. The results show all users have good experience with comfortable and stable lighting perception.

Total Scores at 10 DimmingSettings		User A		User B		User C		Average	
		FLK	LIT	FLK	LIT	FLK	LIT	FLK	LIT
View	direct view	9	10	10	10	10	10	9.7	10
	side view	10	8	10	10	10	9	10	9
Distance	1 m	8	9	10	9	9	9	9	9
	3 m	10	8	10	10	10	9	10	9
	5 m	10	8	10	10	10	9	10	9

Table 2.4 Users' perception scores of flickering (FLK) and lighting (LIT) for 10 dimming setting at 100 KHz transmission frequency. If one senses no flickers or has comfortable lighting at specific setting, the score is 1, otherwise 0. The score value in each cell is the sum of 10 settings.

2.7.5 Communication Performance

In this section, we evaluate the throughput performance of LiFOD in three aspects: (1) through-put vs. transmission frequency and distance; (2) throughput vs. incident angle and position; (3)

throughput comparison with the state-of-art OWC schemes considering fine-grained dimming and high-speed communication simultaneously.

(1) Impact of transmission frequency and distance.

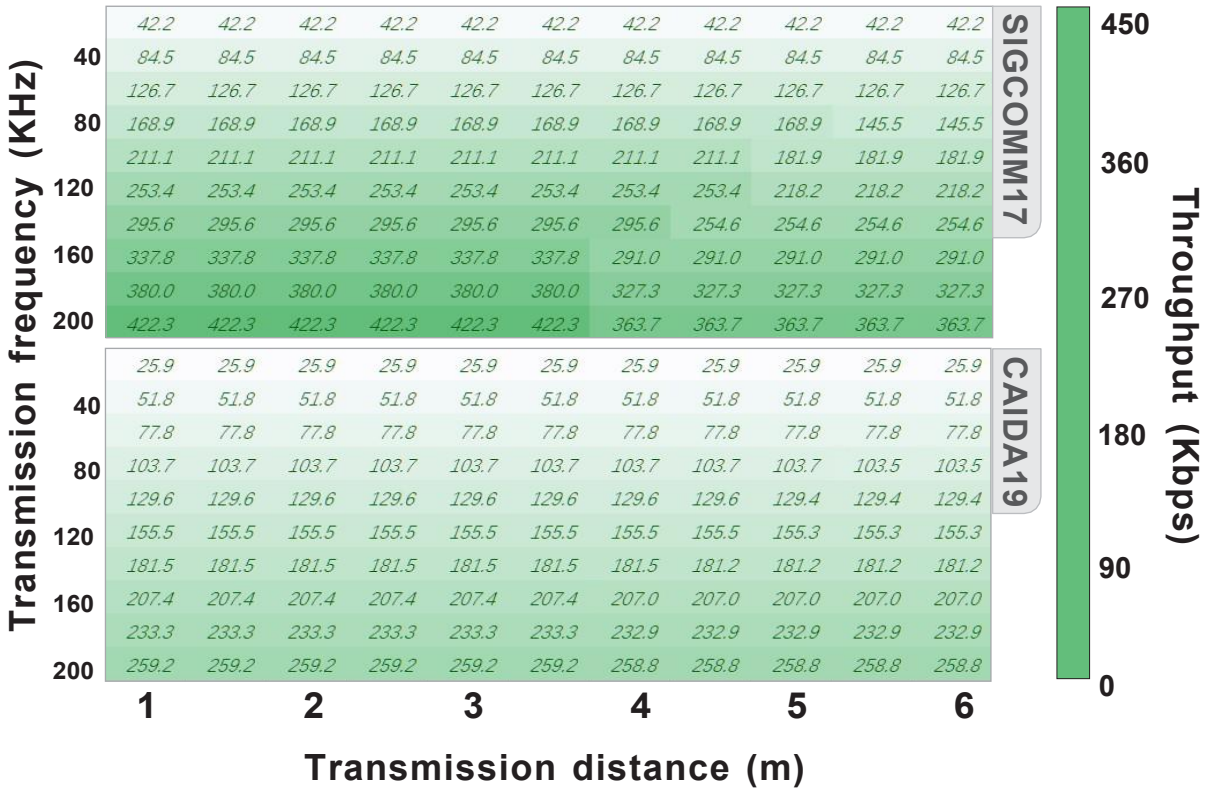


Figure 2.17 Throughput vs. distance and frequency.

We first evaluate LiFOD's throughput performance at different transmission frequencies and distances based on two real-world data traces. As shown in Figure 2.17, the throughputs increase significantly as transmission frequency increases at the same distance setting. Although increasing distance will cause the throughput decline, it decreases less noticeably due to the reliable OOK modulation and our robust symbol detection. Due to the higher bonus bits introduced by CSC, LiFOD achieves up to **400 Kbps** in data rate at a range of up to **6m** in SIGCOMM17 traffic. It is about **2.7** times better for throughput and **1.5** times better for communication range compared with the latest OpenVLC (average **150 Kbps** at 4m under optical interferences).

(2) Impact of incidence angle and position.

Because light beams emit and spread in the line-of-sight (LOS) manner, the pointing and direction setting is essential in high-speed OWC systems. We evaluate the influence of different facing angles and the receiver's relative locations as shown in the experimental schematic Figure 2.18. The transmitter is fixed while the receiver's location and its facing angle are changed incrementally at 5and 2cm from its base location L_0 and direction. We set the transmission distance from L_0 of the receiver to the transmitter at **3.5m** and the transmission frequency to **125KHz** for our two data traces.

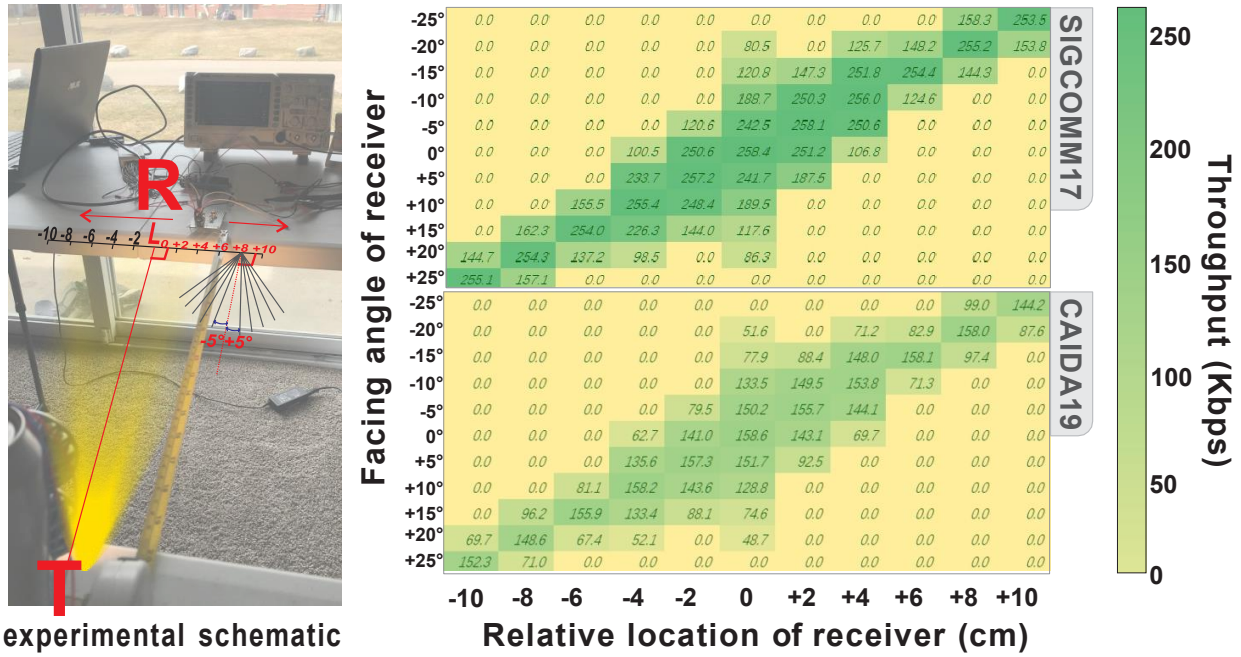


Figure 2.18 Throughput vs. Incidence angle and position.

As shown in Figure 2.18, when the receiver is set at L_0, LiFOD can tolerate more unaligned angles. When the receiver is moved left or right in small ranges such as 2 or 4 cm, it is the same. For long-range location movement, the throughput can drop dramatically unless the proper angle is set. The performance trend is consistent for the two data traces. Thus, it is important for real-world usage of LiFOD to make sure the transmitter's light directly points to the receiver. However, this is consistent with normal usage habits of using lamps for our daily lighting.

(3) Throughput comparison with the state-of-art.

Finally, we make comparisons among LiFOD with state-of-art methods: OOK-CT, MPPM,

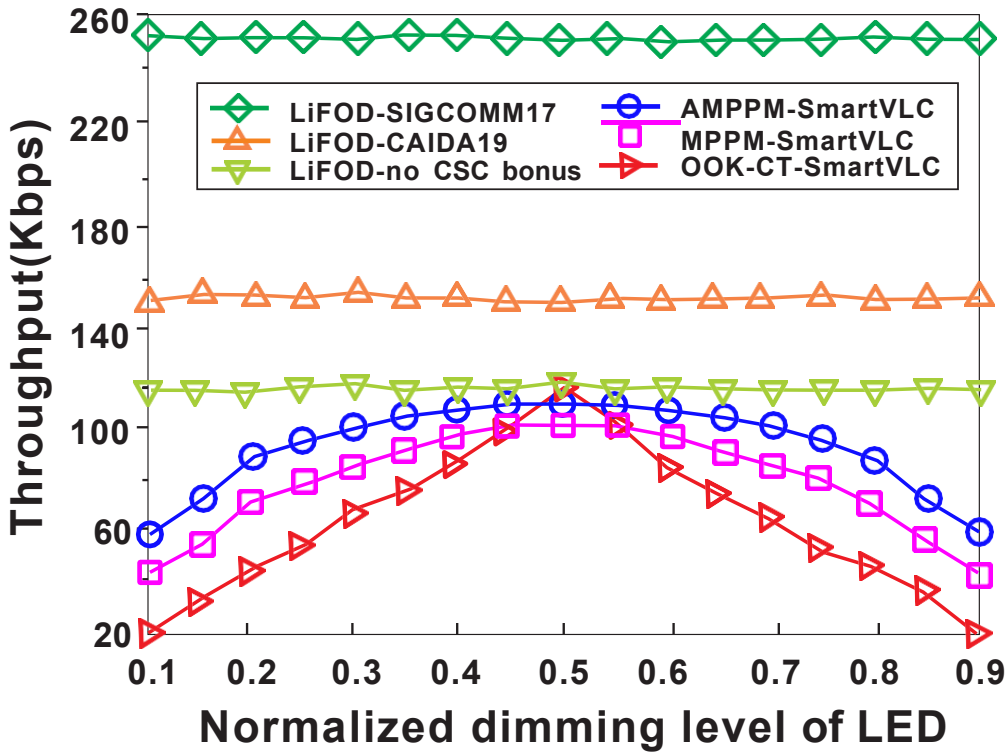


Figure 2.19 Comparison with state-of-art[115].

and AMPPM discussed in SmartVLC[115]. We set the same transmission frequency to **125KHz** and distance to **3.5m**, as described in SmartVLC. OOK-CT is OOK with Compensation Time, it keeps the CS symbols' amplitude constant and only changes the inserted number of CS symbols for dimming. Thus, OOK-CT, MPPM, and AMPPM are coupled-dimming-based OWC. We evaluate LiFOD's performance with the SIGCOMM17 and CAIDA19 data traces. We transmit OOK symbols without CSC bonus in LiFOD as a comparison.

First of all, our LiFOD throughput performances are better than **coupled-dimming**-based OWC methods in all scenarios. The reason is that LiFOD decouples the dimming with transmission and releases most times slots for standard data symbol transmission. Based on different CSC bonus ratios in various data traces, LiFOD for SIGCOMM17 traffic performs best and achieves **250 Kbps** in all dimming settings, which is an improvement of at least **110%** compared to AMPPM. Although lower than SIGCOMM17, LiFOD for CAIDA19 traffic which collects the daily network traffic of a city in the US still achieves **155 Kbps** in all dimming settings, which corresponds to at least a **34%** improvement over AMPPM in SmartVLC (the best throughput performance is **120 Kbps**).

2.8 Discussion and Summary

Generalizability. The throughput improvement ratio in LiFOD is based on the bonus ratio of traffic. Other OWC platforms, such as the LiFi system, can apply LiFOD approach to imporve their performance. Suppose the common OWC platform is improved in engineering or products such as robust symbol transmission and decoding at the **MHz/GHz** level. In that case, LiFOD can also be adopted to achieve the throughput improvement at the same boost ratio and may achieve the data rate at **hundreds of Mbps/Gbps** with fine-grained dimming support.

The LiFOD exploits opportunities of expanding dimming methods for its use in data transmission: using compensation symbols as a side-channel to carry data bits to improve the throughput in OOK-based OWC networks. First, we design a lightweight greedy algorithm to identify bit patterns to maximize the total bonus bit performance in real-world traces. Then we utilize the preamble to notify CSC codes, dynamic thresholds, and estimate channel conditions for robust demodulation in the changing optical environment. Most importantly, we design non-flicker optical symbols and compensation symbol relocation scheme to support smooth lighting and communication with improved throughput. LiFOD can achieve up to **400 Kbps** throughput in the communication range up to **6m** with fine-grained dimming. Compared with SmartVLC at the same transmission parameters, LiFOD improves more than **34%** and **110%** throughput for two real-world data traces respectively in all dimming levels.

CHAPTER 3

BOOSTING OCC VIA 2D SPATIAL-TEMPORAL DIVERSITIES

Optical camera communication (**OCC**) has garnered increasing attention, driven by the widespread availability of affordable mobile devices equipped with built-in cameras. Additionally, OCC stands out for its low interference with ambient light, distinguishing it from other optical wireless communication (OWC) techniques. Notably, OCC offers location-based services (LBS), enabling fine-grained AR navigation through the association of data from visible transmitters within a flexible communication range [95, 24]. Despite these advantages, developing a high-speed and practical OCC system remains an open challenge, particularly for LED-based OCC.

In this project, our main objective is to design a practical data embedding protocol that capitalizes on the 2D spatial diversities of optical signals. By doing so, we aim to overcome the limitations of existing optical camera communication systems and break through the current bottleneck caused by the low frequency response at the receiver side.

3.1 Motivation

Currently, the Radio Frequency (RF) spectrum below 10 GHz is widely utilized for our everyday wireless communication. However, with the increasing demand for massive high-speed wireless services in the future, even higher RF bandwidths like mmWave and nanometer waves may soon become inadequate [95, 24, 83].

In contrast to the strictly regulated RF band, which covers frequencies between 3 kHz and 300 GHz on the electromagnetic spectrum, the optical spectrum boasts a bandwidth over 10,000 times broader than RF spectrum [15]. The growing adoption of light-emitting diode (LED) lamps for indoor and outdoor lighting, as well as information display, is due to their energy efficiency, cost-effectiveness, and extended lifespan. These widespread LED infrastructures, including home lighting fixtures, street lamps, traffic lights, and car headlights [13, 85], possess superior ON/OFF switching rates. This characteristic facilitates optical wireless communication (OWC) in various aspects of our daily lives [15, 151].

OWC offers reliable connections through line-of-sight (LOS) spread, ensuring secure commu-

nication and high-capacity networks with broad spectrum bandwidth, low power consumption, and high speed compared to RF-based communication [42].

In contrast to RF approaches, Optical Wireless Communication (OWC) offers several advantages, including reliable connections through Line of Sight (LoS) for secure communication and spatial multiplexing. High-capacity networks are made possible by leveraging spatial multiplexing and broad spectrum bandwidth, while still maintaining low power consumption for high-speed services [141, 145]. There are primarily two types of OWC based on the receiver types: (1) PD (photo diode) based OWC, exemplified by technologies like LiFi [84], and (2) Camera based OWC, commonly referred to as Optical Camera Communication (OCC) [2, 15]. OCC can be further classified based on transmitter types into **(1) LCD-OCC**: liquid crystal display based OCC such as the screen-camera communication [82, 138, 58]; and **(2) LED-OCC**: LED based OCC such as ColorBar, CASK[124, 38]. We discuss their differences below.

The **LCD-OCC** approach captures each frame and subsequently decodes the embedded data, such as a QR code, in that frame. Despite the fact that the spatial diversity provided by millions of pixels at both screen and camera sides are exploited for dense data embedding in each frame and achieves hundreds of *Kbps* with the constraint of LC's low response frequency at *tens of Hz*[82, 138], the expensive screen, complicated decoding and limited range (i.e., within 0.9m) hinder it from having an enormous market like LED-OCC.

The **LED-OCC** utilizes LED's faster On/Off switching rate rather than low-speed liquid crystal and thus record data with a faster shutter rate than the frame-rate at the camera side contrasted with LCD-OCC. Researchers have made many attempts to further improve its data rate, including Yanbing[124, 122] who investigated a high-order modulation, CASK (composite amplitude shift keying), which encodes data into different brightness levels, and Pengfei [38][37] who proposed ColorBar, that uses CSK (color shift keying) in OCC, which encodes data into different colors. They achieved up to *8 Kbps* data rate for commercial smartphone-based OCC. However, these approaches only consider the grayscale difference (amplitude diversity) and color difference (spectrum diversity) recorded in **1D rolling strips** for improved data rate and do not consider the 2D

spatial diversity in optical imaging at both transmitter and receiver sides.

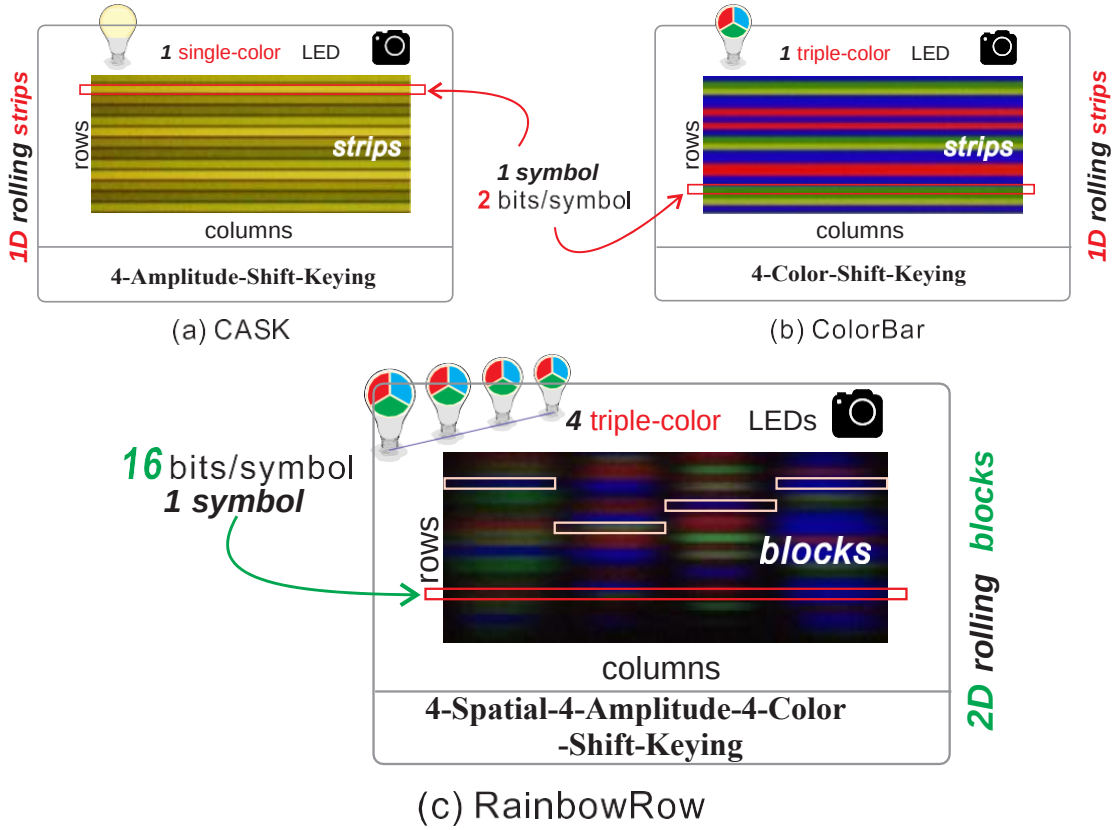


Figure 3.1 The illustration of 2D rolling blocks spatial diversity in our proposed (c) RainbowRow and its comparison with 1D rolling strips spatial diversity in state of the arts in OCC: (a) CASK[124] and (b) ColorBar[38].

As shown in Figure 3.1, in the process of camera imaging, existing LED-OCC systems do not consider spatial diversity, and treat the whole row (**1D rolling strips**) from the rolling shutter as one value by taking the overall average. However, the camera can capture transmitter units at different horizontal locations in each row with different amplitudes and colors and generate **2D rolling blocks** to embed more data and therefore boost the data rate of OCC.

Researchers have made many attempts to improve the data rate of LED-based OCC, including Yanbing[124][122] who investigated a high-order modulation, CASK (composite amplitude shift keying), which encodes data into different luminance levels, and Pengfei [38][37] who proposed ColorBar, that uses CSK (color shift keying) in OCC, which encodes data into different colors. They achieved less than *8 Kbps* data rate for commercial smartphone-based OCC. However, these

approaches only consider the grayscale difference (amplitude diversity) and color difference (spectrum diversity) combined with **1D rolling strips** in modulation for improved data rate and do not consider the **2D rolling blocks** spatial diversity of camera imaging.

Motivation: (1) RF techniques are insufficient for future numerous high-speed and high-density services due to congested spectrum and **severe interference**. (2) PD based OWC such as LiFi senses light with single-pixel and thus requires rigorous direction pointing and is vulnerable to ambient light. (3) Although LCD-based OCC uses spatial diversity, its narrow market potential is hindered by its slow LC response frequency, expensive screen cost, and limited range. (4) Existing LED-OCC approaches do not share drawbacks in (1)-(3), however, they only consider amplitude, spectrum diversities in 1D rolling strips and achieve limited data rate. (5) Despite using the spatial multiple LED sources and camera pixels to achieve spatial redundancy forward error correction (FEC) in transmission, UFSOOK (undersampled frequency shift on-off keying) encodes data with On/Off blinking at frame rate level (tens of Hz) and does not exploit rolling effect and 2D rolling blocks in transmission[2].

To address the problems above, we design **RainbowRow**, an OCC framework with 1D spatial diversity in the design of the transmitter and 1D temporal diversity enabled by rolling shutter effect, as illustrated in Figure 3.1. RainbowRow is made up of an LED bar with four transmission units and a standard camera. Our RainbowRow protocol includes the following 5 key features: (1) Low cost: It only requires basic LEDs and cameras. (2) High-speed: It significantly enhances data transfer, exceeding conventional LED-OCC by a factor of 20. Because of the camera's pixel count and simple modulation, it remains unaffected by motion and ambient light, with customizable distance and a wide vision. (4) Energy-saving: LEDs conserve energy while acting as data transmitters and lighting sources. (5) Practical: RainbowRow is suitable for a variety of applications, including indoor communication and vehicle networks, while also providing illumination benefits.

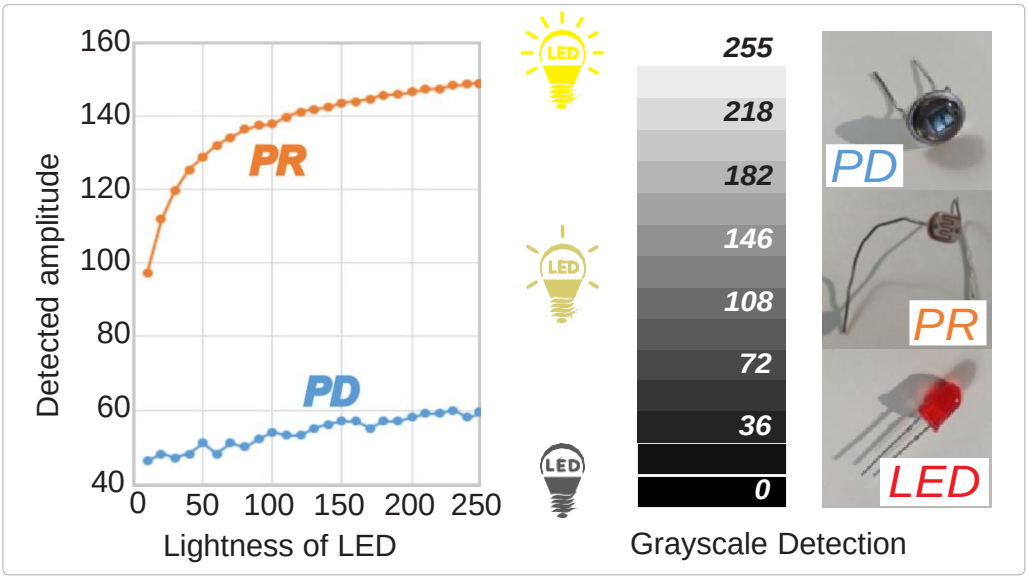


Figure 3.2 Amplitude diversity: generation at Tx and detection at Rx.

3.2 Background and Related Work

3.2.1 Amplitude Diversity

Amplitude diversity is generated by different brightness of the light source and measured by the light sensor (i.e., PhotoDiode, PhotoResistance, and the camera) as grayscale, as depicted in Figure 3.2. Due to the photoelectric effect, these semiconductor devices transform optical signals into electrical signals, and thus the different brightness can be encoded as data bits. Suppose the detected grayscale range is normalized from 0 to 255. Ideally, we can design 256-ASK (amplitude shift keying) modulation mapping 256 grayscale levels into 8 bits. However, because to the narrow range of illumination and varied optical environment, the majority of OWC systems could only map 8 grayscale levels into 3 bits.

Additionally, as seen in Figure 3.3, the data rate changes nonlinearly while the amplitude diversity changes linearly. When the amplitude diversity increases from 16 to 64, the denoted bits in each symbol improves from 4 to 6, but the symbol distance reduces from 16 to 4 sharply. The shorter symbol distance that comes with higher-order ASK results in minor performance improvements but significant detection errors because of the smaller margin for correct detection between symbols. *RainbowRow* adopts 4 amplitude diversity, which is of a relatively low order,

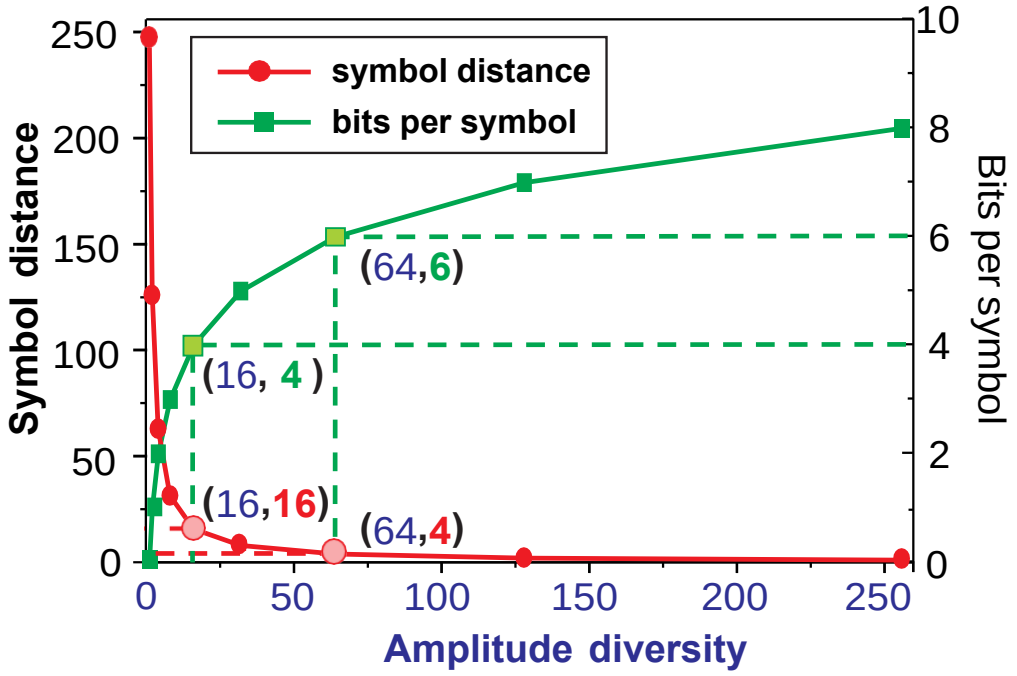


Figure 3.3 Symbol distance/bits per symbol vs. amplitude diversity.

to boost the transmission's robustness. However, this is supplemented with spectrum and spatial variety to increase the data throughput while preserving robustness.

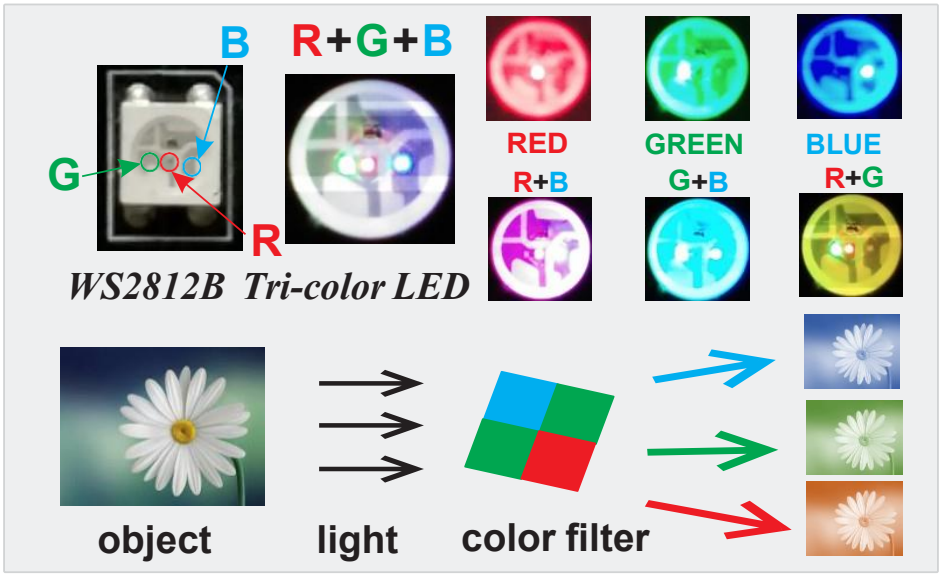


Figure 3.4 Spectrum diversity: generation at Tx and detection at Rx.

3.2.2 Spectrum Diversity

Commercial RGB Tri-LEDs can generate a variety of colors by combining different amounts of **R**ed (700nm), **G**reen (546.1nm), and **B**lue (435.8nm) colors based on the RGB model. For example, as shown in Figure 3.4, the mixture of pure red and green light emits the yellow light. A set of RGB values will eventually be applied to the LED's voltage to generate colored (i.e., different light wavelength/frequency) optical symbols. For color detection, three filters with R, G, B wavelength sensitivities are used to measure the wavelengths of red, green, and blue color components, respectively. The sensor responds using the light-to-voltage converter by producing a voltage corresponding to the detected color.

IEEE OWC standard[2] defines color shift keying (CSK) modulation. In CSK, the optical symbols are generated based on the points on constellation triangles based on the RGB model. The CSK constellation is decided by combining the selected three color bands to form a triangle on the xy color coordinates of CIE 1931[16]. It increases the symbol distance when compared to the same order ASK modulations. However, different devices generate different optical signals even with the same RGB parameter input. Furthermore, even detecting the same optical signal from the same device, the varying optical environment could bring challenges of accurate symbol recognition for high-order CSK (e.g., 32-CSK[38]). As a result of the hue's one-to-one relationship with the color, we employ the HSV (Hue, Saturation, Value) model to reliably identify colors instead of the RGB model.

For color detection, three filters with R, G, B wavelength sensitivities are used to measure the wavelengths of red, green, and blue color components, respectively. Based on the activation of these filters, the color of the optical signal is categorized. A light-to-voltage converter is also present in the sensor. The sensor responds to color by generating a voltage proportional to the detected color at the receiver.

To utilize spectrum diversity for transmission, in the IEEE OWC standard[2], it defines color shift keying (CSK) modulation. The optical symbols are generated based on the points on the CSK constellation triangles based on the RGB model, as shown in Figure 3.5. The CSK constellation

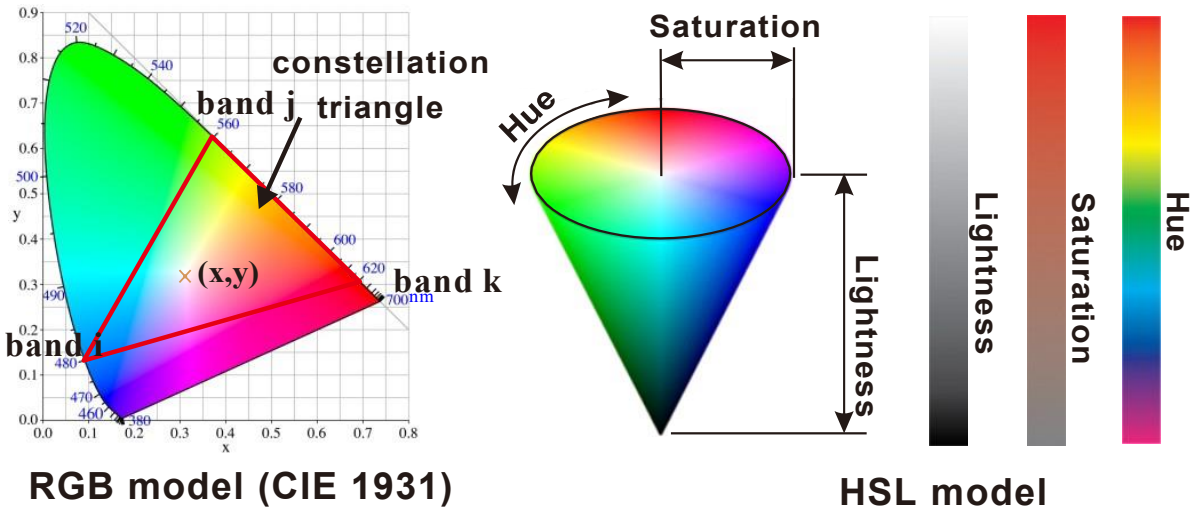


Figure 3.5 Comparison of RGB and HSL model[16].

is decided by combining the selected three color bands, which can form a triangle on the xy color coordinates of CIE 1931[16]. It increases the symbol space and distance than the same order ASK modulations. However, CSK modulation has a complicated and high requirement for control at the transmitter with additional overhead and cost. Moreover, different devices generate different optical signals even when they have the same input RGB parameters. Furthermore, even detecting the same optical signal of the same device in the varying optical environment can also bring the challenge of accurate symbol recognition at the receiver for high-order CSK such as 16-CSK, 32-CSK[38].

Compared with the RGB model used for color generation, the HSL model is more natural to describe colors and more popular for color recognition, as shown in Figure 3.5. H stands for Hue, which corresponding to the red, orange, yellow, green, cyan, blue, violet and so on. Hue reflects the changes and differences of colors more directly, which is the spectrum diversity of the optical wavelength. The more kinds of wavelength, the higher S (Saturation) value. L stands for Lightness or Luminance, and it reflects the grayscale of the light. The HSL model separates the lightness and color of the light, which are the amplitude and spectrum diversity separately.

3.2.3 Spatial Diversity

(1) Camera shutter and spatial diversity in camera.

The shutter is an essential camera mechanism that controls a photographic film's effective exposure time. There are two shutter types: global shutter and rolling shutter, as shown in Figure 3.6. **(1) Global shutter** exposes the whole scene at the same time. Light sensors at each pixel collect light synchronously and are exposed at the same time. At the beginning of the exposure, all light sensors begin to collect the light, and cut off light sensing and collection at the end of the exposure. **(2)** Unlike a global shutter, the **rolling shutter** is implemented by exposing one row of pixels simultaneously and row by row generates an entire image.

Spatial diversity is generated by millions of pixels in 2D camera image sensors with multiple light sources shown in camera's FOV. Each pixel or each cluster of pixels can record the optical features such as amplitude and spectrum diversities of each light source shown in FOV of camera. Based on camera shutter types and the transmission frequency of LED sources, the spatial diversity can be classified into two categories: (1) with **frame-level** update speed, and (2) with **faster row-level** update, as depicted in Figure 3.7 (a) and (b) separately and illustrated below.

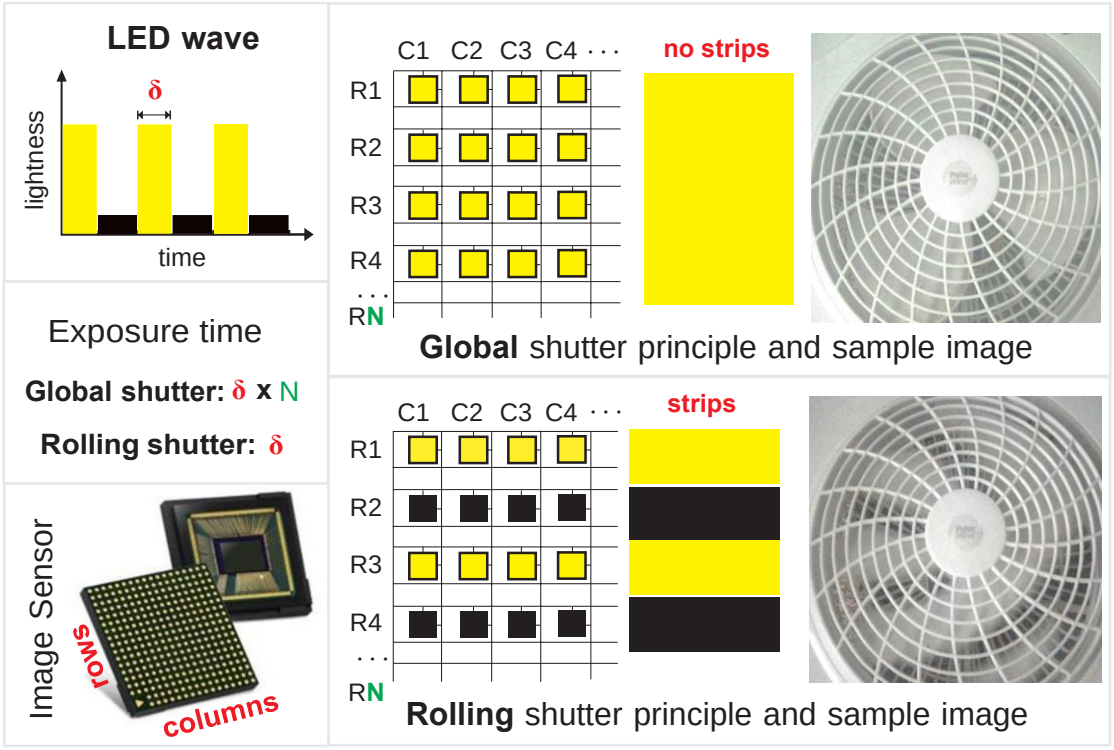
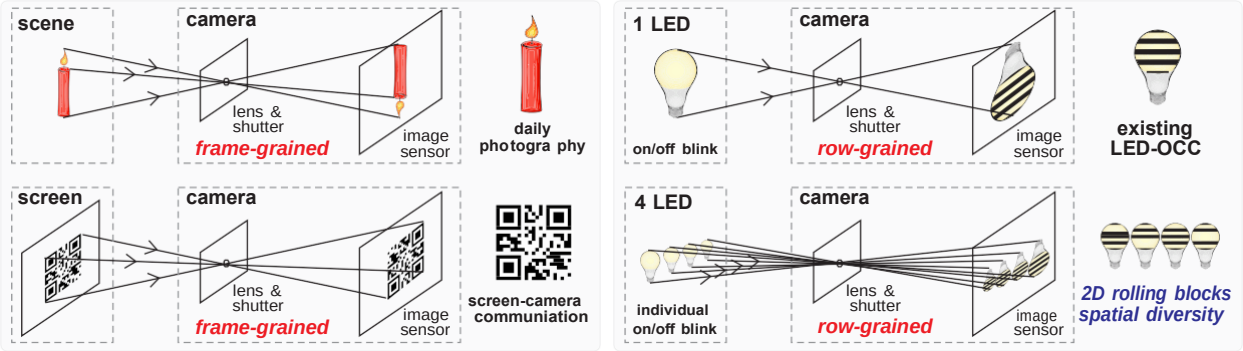


Figure 3.6 Rolling shutter effect.



(a) spatial diversity at frame level update (b) spatial diversity at row level update

Figure 3.7 Spatial diversity in camera imaging with frame / row level updates.

(2) Update with frame-level vs. row-level.

Frame-level updated spatial diversity. When one period of transmitted data from all light sources in FOV is emitted (synchronously or asynchronously) during the frame period and captured by cameras whatever the global shutter or rolling shutter, the captured frame will have no rolling strips and the transmitted data will be decoded at the frame level. For example, the existing screen-camera communication approaches[82, 58] captures each frame as a full unit and subsequently decodes the embedded data, such as a QR code, in that frame. The UFSOOK [2] is also updated at the frame level even though it repeats the data over several LEDs to provide spatial redundancy FEC.

Row-level updated spatial diversity. When one period of transmitted data from all light sources in FOV is emitted synchronously during the rolling shutter period and captured by rolling shutter camera, the captured frame will have rolling strips and the transmitted data will be decoded at the faster rolling-shutter level than the frame level. Compared with existing screen-camera communication and UFSOOK, which utilize the low frame-level spatial diversity, the approaches that adopted rolling-shutter-level update speed are supposed to have higher data rate due to its faster update rate. Nonetheless, these approaches (e.g., ColorBar, CASK[124, 38]) do not consider the spatial diversity and only exploit the **1D rolling strips** in communication instead of the**2D rolling blocks** in our proposed RainbowRow.

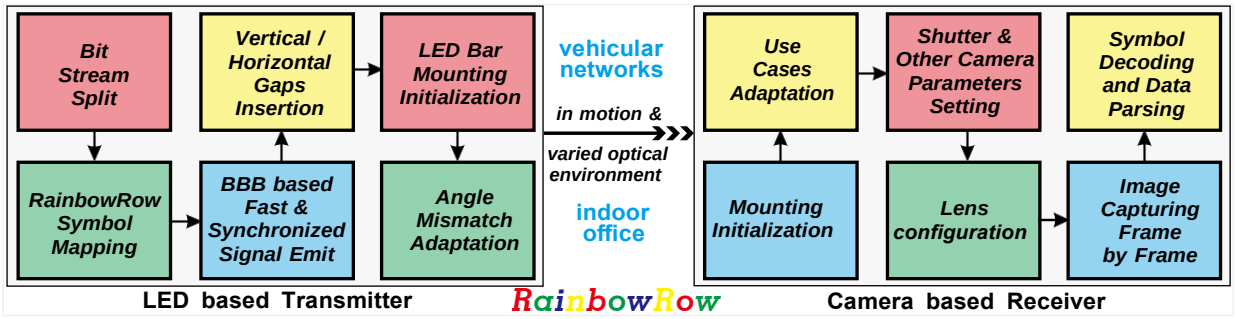


Figure 3.8 RainbowRow system overview and technical challenges at both the transmitter and the camera sides.

3.3 Our Approach: RainbowRow

System Overview: Our proposed RainbowRow consists of two parts, as shown in Figure 3.17: (1) Tri-color LED bar based RainbowRow transmitter, and (2) Commercial camera based mobile RainbowRow receiver. LED Transmitter: LED bar consists of 4 spatial transmission units and each include 3 LED bulbs (i.e, red, green and blue). Camera Receiver: The receiver is a commercial camera such as COTS smartphones.

The transmission workflow is: (1) bit stream split, (2) RainbowRow symbol mapping, (3) BeagleBone Black (BBB) based fast and synchronized signal emission , (4) vertical/horizontal gaps insertion, (5) mounting initialization, and (6) angle mismatch adaptation.

The decoding workflow at the receiver side is: (1) mounting initialization, (2) use case adaptations, (3) shutter and other camera parameters setting, (4) lens configuration, (5) image capturing frame by frame, (6) symbol decoding and data parsing.

Technical Challenges. (1) Modeling of spatial diversity in 2D rolling blocks: The spatial diversity in 2D rolling blocks has never been considered and exploited before. It is a challenge to investigate deeply and model 2D rolling blocks clearly because this spatial diversity is dependent on: LED transmitter, optical propagation, and rolling shutter camera. **(2) Optical imaging management at both Tx and Rx sides**: In contrast to 1D rolling strips in existing work, it is a challenge to control multiple spatial located LED transmission units to emit optical signals synchronously in high frequency. The optical signals from various transmission units would also destroy the decoding owing to their mutual interference and overlapping despite the fact that the inner fusion

of optical signals in each transmission unit is the basis of amplitude and spectrum diversities. **(3) Practical adaptations for real use cases:** The misaligned rotation angle between the LED bar and the horizontal axis of the camera will result in a data rate drop in an indoor office setting. Additionally, in vehicular scenarios, RainbowRow encounters a long distance caused by weak optical signals and a variety of horizontal gaps at various viewing angles.

Our main **contributions** can be summarized as follows:

- RainbowRow is the **first** work to employ 2D rolling blocks for LED based optical camera communication. We model 2D spatial diversity in optical imaging and use it to break the throughput bottleneck of LED-OCC systems.
- We propose the *RainbowRow* protocol, which exploits the spatial diversity in **2D rolling blocks** instead of **1D rolling strips** and combine it with amplitude and spectrum diversities to boost LED-OCC's data rate.
- We implement a *RainbowRow* prototype based on commercial devices and address technical challenges including optical imaging management in transmission and adaptations for indoor/vehicular cases.
- We evaluate *RainbowRow* on our testbed and conduct a case study for two real-world applications for its practicality. Our RainbowRow achieves up to **170 Kbps**, over **20** times of existing LED-OCC approaches.

3.4 2D Rolling Blocks Modeling

3.4.1 Why a LED bar instead of a LED matrix?

Each row comprises multiple pixels, which can denote multiple colors or grayscales in different parts of pixels in that row. This spatial diversity on each row provides a great potential to boost the throughput without an additional cost by allowing more data to be embedded in multiple light sources. We name this spatial diversity as **2D rolling blocks spatial diversity** to differentiate it from the **1D rolling strips spatial diversity** in the state-of-the-art. To take advantage of the

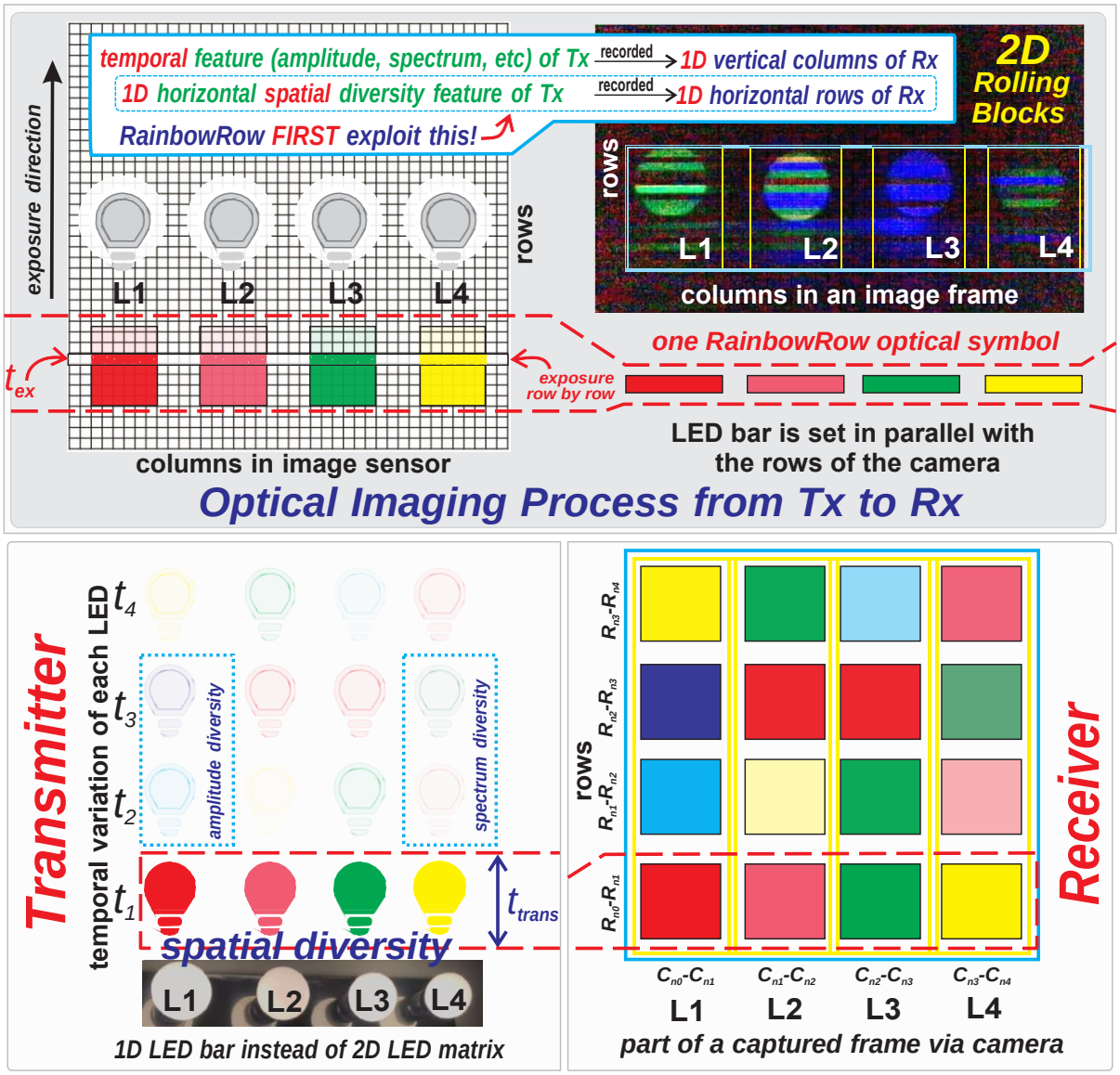


Figure 3.9 The illustration of 2D rolling blocks with diversity combination of amplitude and spectrum.

spatial diversity at the receiving end, spatially related coding and modulation are required at the transmitting end.

As shown in Figure 3.9, the transmitter is designed as a LED bar with multiple transmission units horizontally instead of a LED matrix. Each transmission unit generates temporal varied optical signals with different brightness and colors that are recorded as strips vertically while other transmission units horizontally located in camera's FOV conduct emission synchronously. Our RainbowRow creatively combines the spatial diversity with fast shutter-rate-level temporal diversities (i.e, amplitude and spectrum) in LED-OCC's modulation via 2D rolling blocks instead of fully spatial diversity with low frame-level update speed in a LED matrix with more severe vertical interference.

3.4.2 Is it possible to boost OCC via 2D Rolling Blocks?

We propose to combine amplitude diversity, spectrum diversity, and spatial diversity of 2D rolling blocks to improve the data rate of OCC systems, as shown in Figure 3.9. The benefit of this combination is that we can eliminate the short symbol distance limitations for each diversity. We can employ the robust and proper range in each diversity to encode and decode the data separately. Let A denote the amplitude diversity, S_1 and S_2 denote the spectrum and spatial diversity of 2D rolling blocks respectively. The bits encoded in each symbol can be represented as: $\log_2(A \times S_1) \times S_2$.

$$\log_2(A \times S_1) \times S_2 \tag{3.1}$$

For instance, we adopt 4 brightness and 4 colors, the same order level of 4-CASK and 4-CSK separately. The modulations and decoding in each diversity of 4 individual spatially located transmission units are very simple and reliable compared with high order modulations such as 8/16-CASK, 32/64-ColorBar and so on [38, 124]. This diversity combination can output a total of $\log_2(4 \times 4) \times 4 = 16$ bits per symbol period without the limitation of short symbol distance in each diversity and is faster and more robust.

3.4.3 RainbowRow Modulation

(1) Modulation Exploration.

To design a robust and fast OCC system, we explore 9 modulation methods on our testbed for

spatial, spectrum, and amplitude diversities, as shown in Figure 3.10. For each diversity, we set up to 4 levels for illustration.

OOK: On-Off-Keying is the primary amplitude-based modulation, and it is 2-Amplitude-Shift-Keying. It only has amplitude diversity.

4-ASK: 4-Amplitude-Shift-Keying utilizes four amplitude statuses to denote 2 bits in each symbol. It only has amplitude diversity.

4-SOOK: 4-Spatial-On-Off-Keying adopts basic OOK at four different spatial locations, making each symbol denote 4 bits, 4 times that of OOK. It has amplitude and spatial diversities.

4-S-4-ASK: 4-Spatial-4-Amplitude-Shift-Keying adopts 4-ASK at four different spatial locations, making each symbol denote 8 bits, 4 times that of 4-ASK. It has amplitude and spatial diversities.

4-SC-4-ASK: 4-Spatial-Colored-4-Amplitude-Shift-Keying adopts 4-ASK at four different spatial locations. The only difference with 4-S-4-ASK is that each ASK has a different color instead of the same color. It still only has amplitude and spatial diversities without spectrum diversity.

4-CSK: 4-Color-Shift-Keying utilizes four colors to denote 2 bits in each symbol. It only has spectrum diversity.

4-A-4-CSK: 4-Amplitude-4-Color-Shift-Keying utilizes four colors combining with four amplitudes to denote 4 bits in each symbol. It has amplitude and spectrum diversities.

C-4-SOOK: Colored-4-Spatial-On-Off-Keying is similar to 4-SOOK with the same denoted bits. The only difference is that OOK has a different color at each location instead of the same color. It still only has spatial and amplitude diversities without spectrum diversity.

4-S-4-CSK: 4-Spatial-4-Color-Shift-Keying adopts 4-CSK at four different spatial locations, making each symbol denote 8 bits, 4 times of 4-CSK. It has spatial and spectrum diversities without amplitude diversity.

(2) 4-order RainbowRow.

As shown in Figure 3.10 and 3.11, RainbowRow adopts 4-Spatial-4-Amplitude-4-Color-Shift-

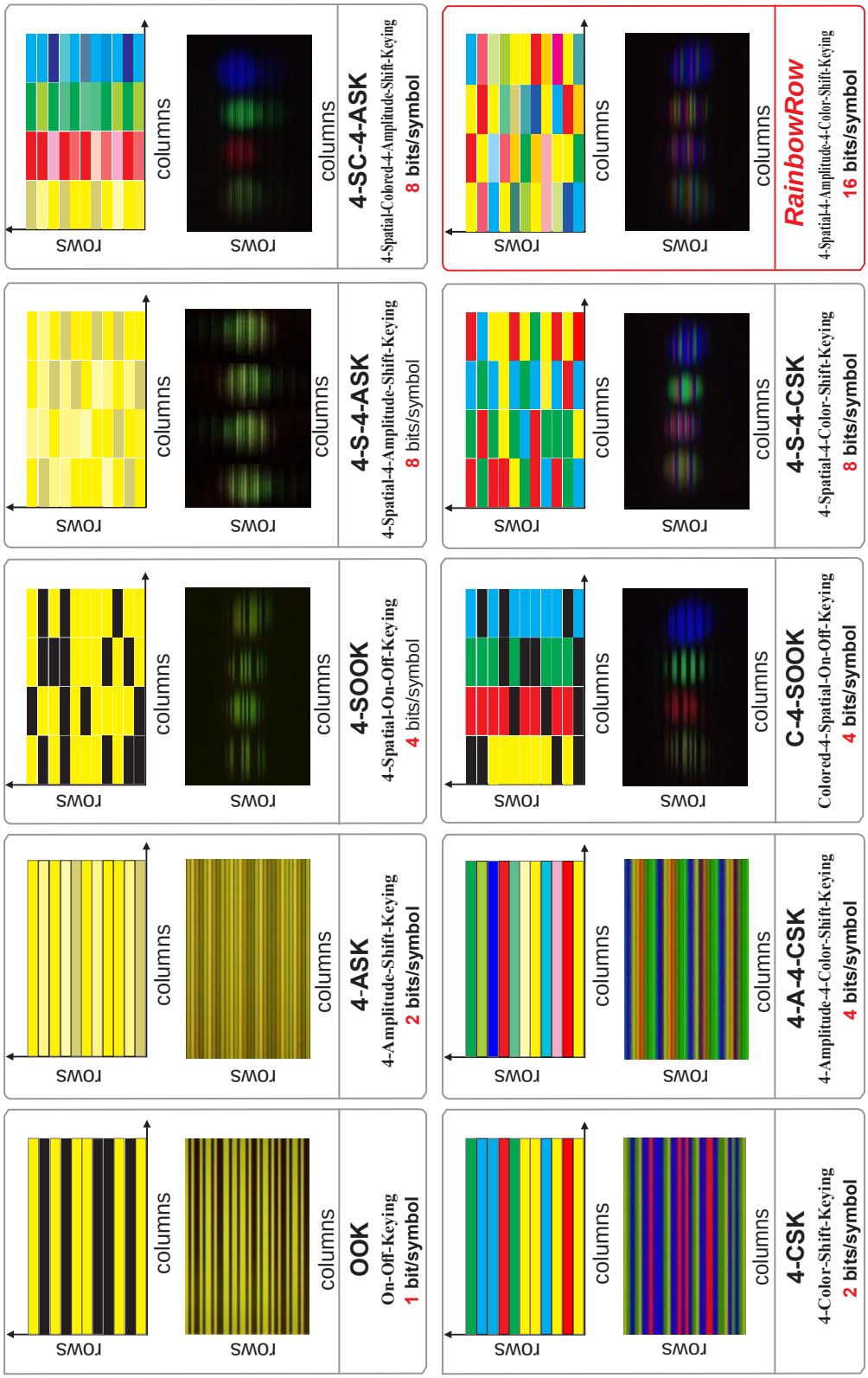


Figure 3.10 The illustration and captured images of 9 explored modulations and RainbowRow balanced coding table.

Keying, which uses 4-CSK combined with 4-ASK at four different locations, making each symbol denote 16 bits. This is a significant improvement on existing work [38, 124]. We named it **RainbowRow** due to the generated strip patterns with random colors and lightness at different locations on a specific row. Ideally, the RainbowRow protocol can extend to N-order and transmit the $\log_2(N \times N) \times N$ bits per RainboRow symbol. Moreover, RainbowRow can fully utilize amplitude and spectrum diversities to present random bit sequences at each location, guaranteeing the random appearance of different colors and lightness for non-flickering during data transmission.

RainbowRow Coding Table					
Color	Amplitude	Location			
		#1	#2	#3	#4
Red	level-*1*	0000	0000	0000	0000
	level-*2*	0001	0001	0001	0001
	level-*3*	0010	0010	0010	0010
	level-*4*	0011	0011	0011	0011
Green	level-*1*	0100	0100	0100	0100
	level-*2*	0101	0101	0101	0101
	level-*3*	0110	0110	0110	0110
	level-*4*	0111	0111	0111	0111
Blue	level-*1*	1000	1000	1000	1000
	level-*2*	1001	1001	1001	1001
	level-*3*	1010	1010	1010	1010
	level-*4*	1011	1011	1011	1011
Yellow	level-*1*	1100	1100	1100	1100
	level-*2*	1101	1101	1101	1101
	level-*3*	1110	1110	1110	1110
	level-*4*	1111	1111	1111	1111

Figure 3.11 RainbowRow balanced coding table.

Undesired Flicker Mitigation. Although we want cameras to clearly record multiple colors and levels of brightness for robust communication, we do not expect human eyes to sense the flickers in its concurrent lighting function. We avoid undesired flickers in two aspects. **(1)** Fast

transmission frequency. RainbowRow adopts transmission frequency at several to tens of KHz, which is faster than the response frequency of human eyes (i.e., 60Hz). **(2)** Color/Amplitude Balanced Coding. As presented in Figure 3.11, each transmission unit has 16 combination of color and amplitude (i.e, R1,R2,R3,R4,G1,G2,G3,G4,B1,B2,B3,B4,Y1, Y2,Y3,Y4) that are mapped to 16 different 4-bits segments (e.g, '0010') with equal appearance possibility, preventing some color or amplitude appearances at low frequencies that would have resulted in unwanted flickers.

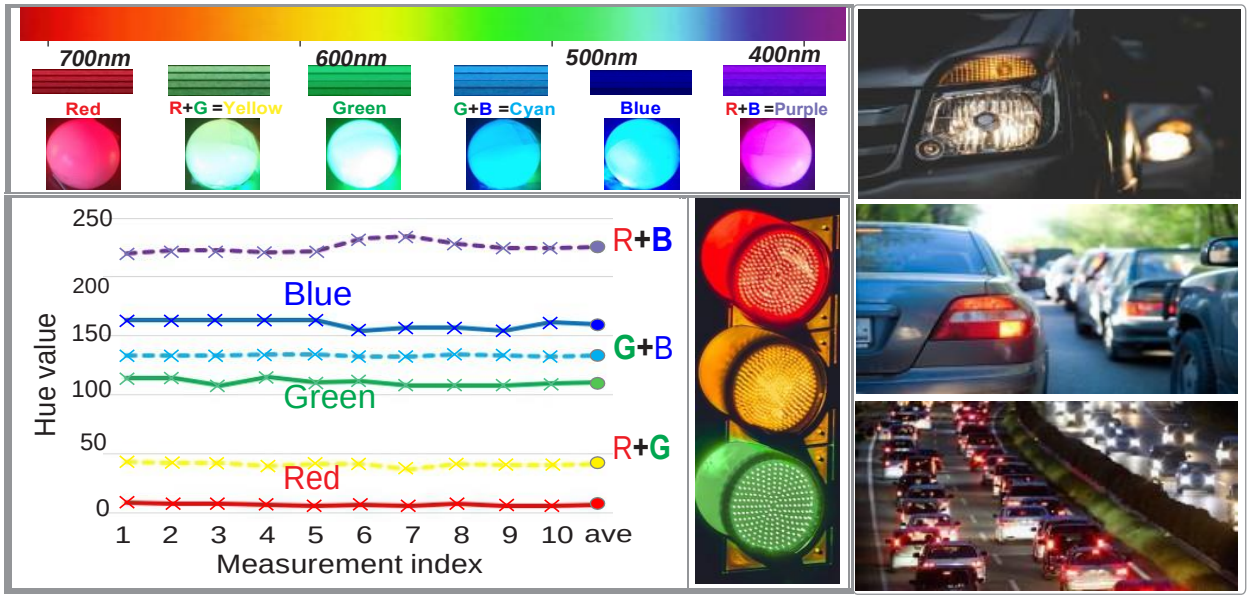


Figure 3.12 Color choice of RGBY in spectrum diversity.

Color Choice. The top of Figure 3.12 shows that R+G generates Yellow, G+B generates Cyan and R+B generates Purple. The bottom-left of Figure 3.12 shows the measured hue values on our testbed. Cyan is too close to blue and green. Purple has the shortest wavelength out of these six colors, although having a wider hue gap than yellow. Thus we chose yellow as the 4^{th} color in addition to red, green and blue. Furthermore, yellow, red and green have longer wavelengths than cyan and purple, which makes them suitable for long distance propagation, the same as traffic lights and headlights.

3.5 Optical Imaging Management

Different from traditional wireless systems such as RF-based approaches with **severe interference** at the receiver side, the Line-of-Sight (LoS) propagation makes optical signals easier to manage their paths. In the camera imaging process, the optical signals from the transmitter are reflected on the millions of pixels at the image sensor via the principle of pinhole imaging. Thus, the main interference is at the transmitter side as well as the ambient noise in its propagation when the camera's parameters are set properly. In this section, we address technical challenges in optical imaging management **from 1D to 2D** at both transmitter and the receiver sides to guarantee the final robust decoding.

3.5.1 At Transmitter Side

(1) Fast and Synchronized Transmission.

LED selection. As shown in Figure 3.13, the low-power and single color LED elements only propagate optical signals for a short distance. High-power Tri-LED strips and Tri-LED panels are suitable to achieve spatial diversity and long communication range. However, the LED control manner of strips and panels is serial control, which will cause the wrong emission of RainbowRow optical symbols. Finally, we adopt 12V T10-194 car interior LED bulbs. Each bulb has 5 single-color 5050 SMD LED elements. We combine 1 red, 1 green, and 1 blue bulb together in each transmission unit and totally 12 LED bulbs for fast and synchronized transmission.

Beaglebone Black. In our proposed RainbowRow, the transmitter should control the color and lightness of 12 LED bulbs synchronously and achieve the transmission frequency at several kHz to match rolling shutter frequency of commercial smartphones. We adopt low-cost Beaglebone Black ($80) for fast and synchronized transmission. When using Pulse Width Modulation (PWM) for amplitude control, the Beaglebone's 12MHz GPIO speed is insufficient as well as Arduino boards with the similar 16MHz GPIO speed. Besides, all these GPIO mentioned above are read/write in serial manner.

PRU. However, BBB has the Programmable Real-time Unit (**PRU**) which can speed up LED control speed up to 200MHz and synchronously control 12 LED via register. Thus we can exploit

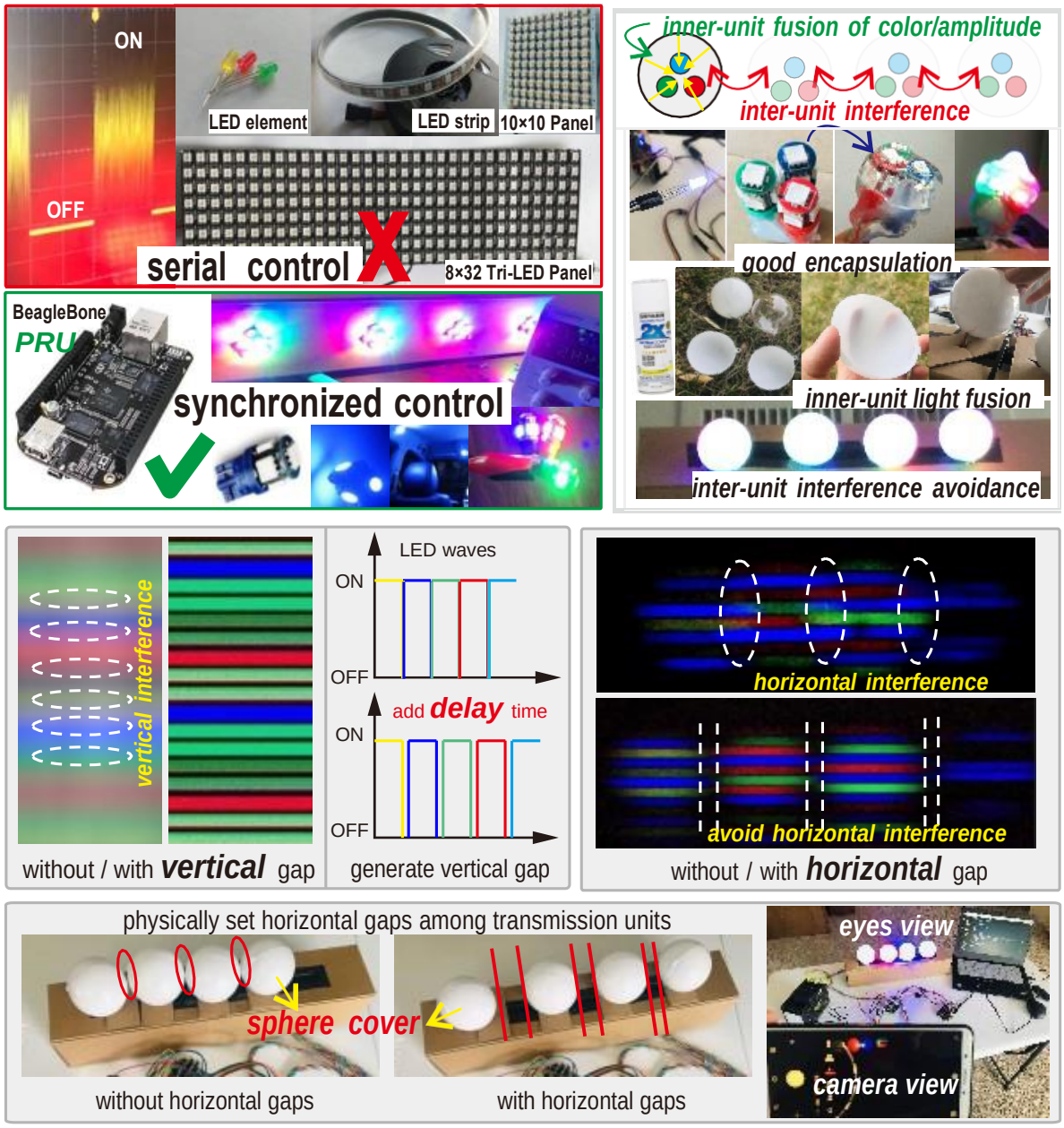


Figure 3.13 Optical imaging management at the transmitter side in RainbowRow design.

BBB's PRU and achieve fine-grained amplitude control of each of 12 LED bulbs with [0, 100] step range at the same time at several to tens of kHz, suitable for the fast and synchronized transmission in our RainbowRow.

(2) Inner-unit Fusion vs. Inter-unit Interference.

From 1D to 2D. In 1D rolling strips based approaches, we care about the inner amplitude or color fusion inside of only one transmission unit. To generate the expected amplitude level or specific color, the transmitter should emit different amounts of brightness or R,G,B color components properly during the symbol duration. These components overlap and fuse among optical signals from one transmission unit to provide the base of the amplitude and spectrum diversity. However, by increasing the transmission units from 1 to multiple (e.g., 4 in RainbowRow), the optical signals from different transmission units will overlap as well. In contrast to inner unit color fusion, there is mutual interference among different transmission units that generate undesired brightness and colors for each transmission unit, which cause the wrong amplitude and color detection at the receiver side (e.g., camera in RainbowRow). The challenge here is to minimize this mutual interference among different transmission units while enhancing the fusion within each transmission unit.

Inner-unit Light Fusion. Each of our self-made transmission units consists of 3 separate R,G,B LED bulbs. They are well-encapsulated tiny Tri-LED elements emitting expected colors by using great color fusion. However, they may cause incorrect symbol detection (e.g, one transmission unit wants to emit yellow by lighting up its red and green bulbs, but the detected color is red or green). We address this issue by encapsulating R,G,B bulbs with hot melt adhesive and covered with a sphere cover shown in Figure 3.13.

Vertical Interference and Temporal Avoidance. The vertical optical signals with amplitude and spectrum features are varied with time and thus we can add the proper delay time between two optical signals switching to generate gaps vertically, as shown in Figure 3.13. However, a longer delay time sacrifices more of the transmission bandwidth with lower throughput. We set the delay time as 0.05 times of the symbol duration to guarantee a significant vertical gap for detection without transmission bandwidth sacrifice.

Horizontal Interference and Spatial Avoidance. (1) Sphere cover. The captured RainbowRow symbols in a frame without any cover have strong overlapping and aliasing horizontally shown in Figure 3.13. We should constrain the optical signals from a specific transmission unit in its expected spatial area. Inspired by our daily light bulbs, we use a transparent plastic ball as the light cover for each transmission unit, the outside of the ball is smooth without spraying, and the inside surface is sprayed with thin and uniform white paint. **(2)** Physical horizontal gaps. In addition, we assign 4 transmission units horizontally with proper mutual physical distance to mitigate horizontal interference further.

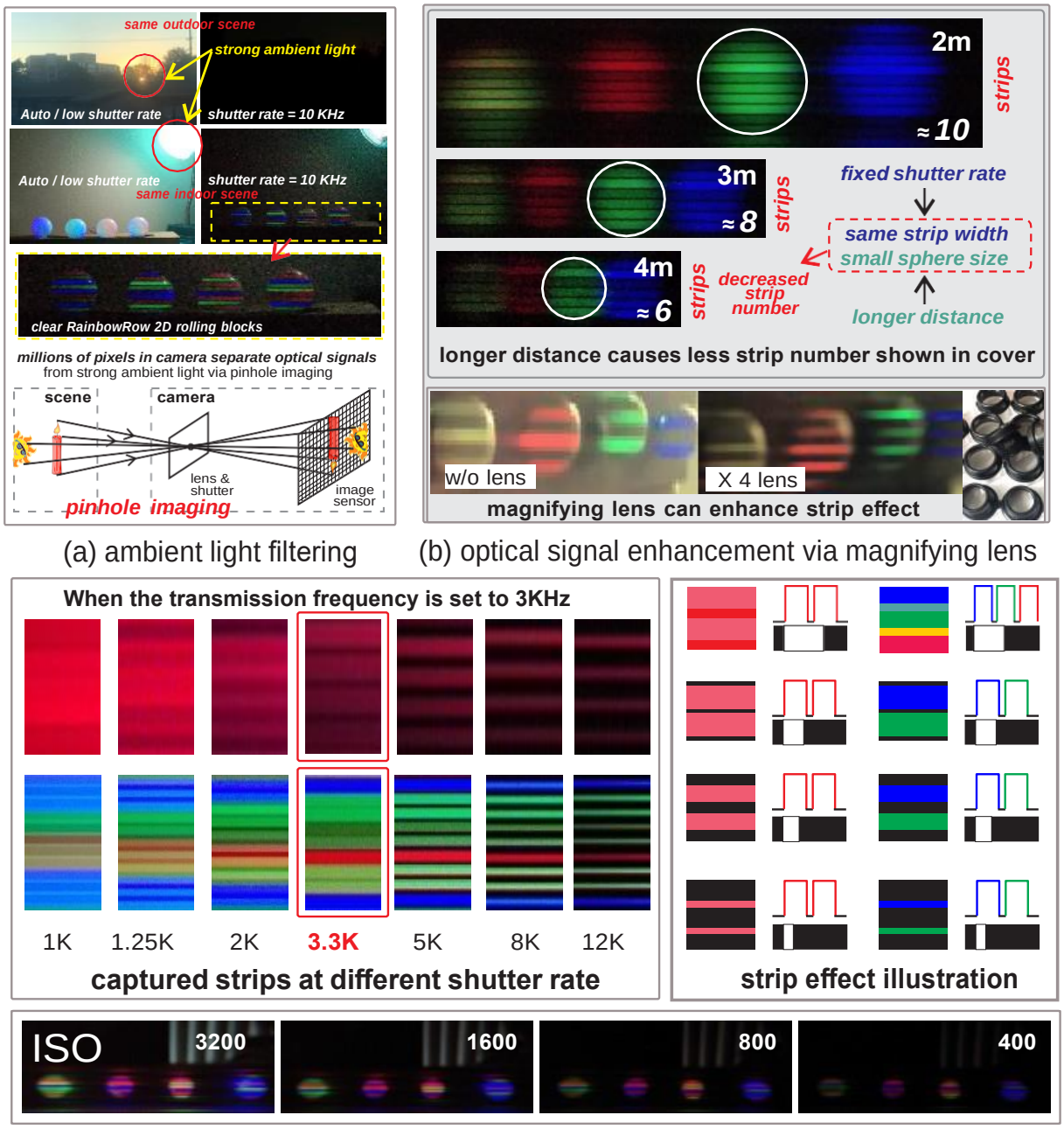
3.5.2 At Receiver Side

(1) Ambient Light Filtering.

There are two aspects in our proposed RainbowRow to filter out the ambient light from both natural world and artificial light sources. **(1)** High shutter rate. To record clear rolling blocks, the rolling shutter rate in RainbowRow is set from several to tens of KHz. The faster shutter rate leads to a decrease in amount of light coming in. In contrast with the active lights from high-power RainbowRow transmitter, most of the weak ambient light can be filtered out and not recorded in the captured image frames. **(2)** Millions of pixels. Even very strong ambient light such as direct incident sunlight, thanks to the millions of pixels in camera, the ambient light source is projected in different pixel zones from our RainbowRow rolling blocks based on the pinhole imaging principle, as shown in Figure 3.14 (a).

(2) Optical Signal Enhancement.

When optical signals from the RainbowRow transmitter propagate to the camera via increased communication range, there exist two main problems. **(1)** Decreased vertical strip number present on sphere cover. While the rolling strip's width is constant because of the fixed shutter rate, the increased communication distance will result in a smaller captured sphere size. As a result, there are fewer rolling strips shown on the cover of transmission unit. **(2)** Optical signal attenuation. The non-trivial attenuation of optical signals caused by a longer propagation distance will also result in weaker captured RainbowRow symbols.



(a) ambient light filtering (b) optical signal enhancement via magnifying lens

(c) camera parameters influence of the quality of captured strips

Figure 3.14 Optical imaging management at the camera side in RainbowRow design.

By placing an appropriate magnifying lens in front of the camera, we solve these issues. The lens can assist the camera in capturing the larger sphere sizes of each transmission unit, and therefore (1) increasing the number of shown rolling strips on the light cover, (2) enhancing the strength of optical signals by presenting more pixels.

(3) Capture clear strips via proper camera parameters.

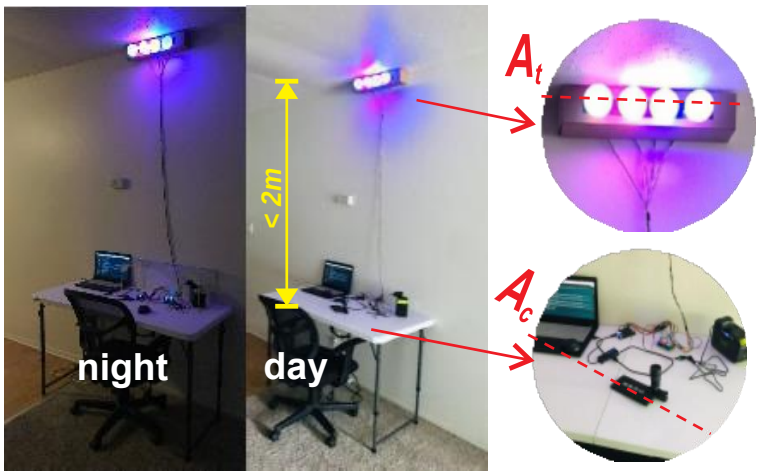
The camera parameter setting is crucial for capturing the correct and clear RainbowRow strips (i.e., each RainbowRow strip is made up of four rolling blocks, as illustrated in Figures 3.1 and Figure 3.9), so that they can be decoded.

Rolling shutter rate. The strip width S_w are related to only two factors: (1) the transmission frequency F_t, and (2) the rolling shutter frequency F_r . When $F_r < F_t$, the captured strips are then mixed together and overlapped into the wrong optical symbols shown in Figure 3.14 (c). When $F_r \geq F_t$, the S_w decreases with the F_r increases from their maximum strip width when $F_r = F_t$. Thus we should set $F_r \approx \mathrm{F}_t$.

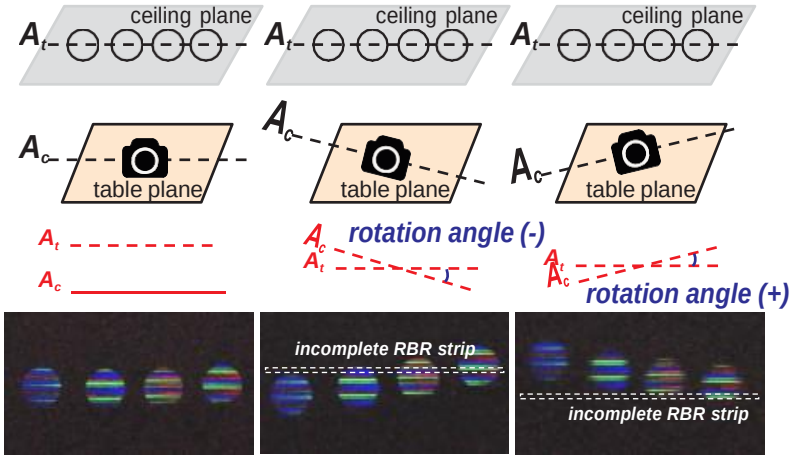
Other parameters. Two key camera parameters: (1) ISO, and (2) resolution may also affect the quality of captured RainbowRow strips. ISO refers to camera's sensitivity to the light. Thanks to the high shutter rate setting filtering out the ambient light already, the higher ISO setting will not cause increased noise points. Thus, to enhance the captured RainbowRow strips, the camera should set a high ISO. Resolution is defined as the pixel numbers of the captured image frame. A higher resolution may improve the clarity of the recorded strips. Therefore, we ought to set high enough resolution such as 1080P instead of 480P.

3.6 Use Case Adaptations

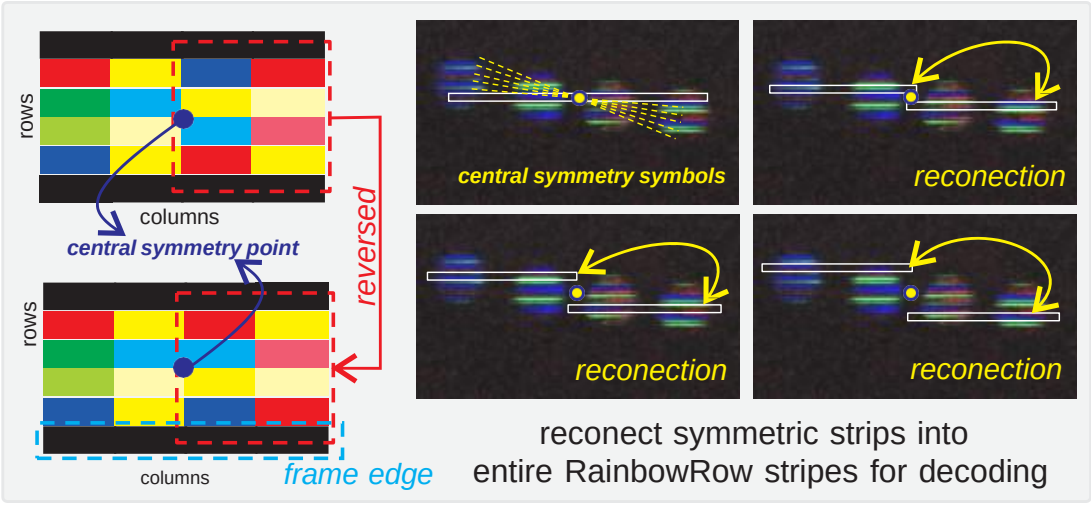
Our proposed RainbowRow protocol has great potential because of its expansibility (i.e., increase the order of spatial/amplitude/spectrum diversities) and flexibility (i.e., numerous applications including mobile/static, day/night, indoor/outdoor, terrestrial/aerial). In this section, we deploy the 4-order RainbowRow design to two real-world use cases: (1) indoor office, and (2) vehicular networks by applying some adaptations for specific requirements.



(a) indoor office use case



(b) rotation angle mismatch



(c) centrosymmetric intra-frame embedding

Figure 3.15 RainbowRow adaptation for indoor office: rotation angle mismatch avoidance.

3.6.1 Adaptations for Indoor Office

(1) Rotation angle mismatch.

As shown in Figure 3.15 (a), the RainbowRow transmitter is mounted on the ceiling and the camera is set on the table to access the Internet (e.g., data downloading for multimedia services as supplement of WiFi to improve user experience with higher data rate). In this case, both the transmitter and receiver remain relatively fixed. It is normal if the camera's horizontal axis A_c is not parallel to the transmitter bar A_t. However, it will cause a decreased number of RainbowRow strips that can be correctly decoded. We define the angle between A_c and A_t as **rotation angle** due to they are in two parallel planes (i.e., ceiling and table).

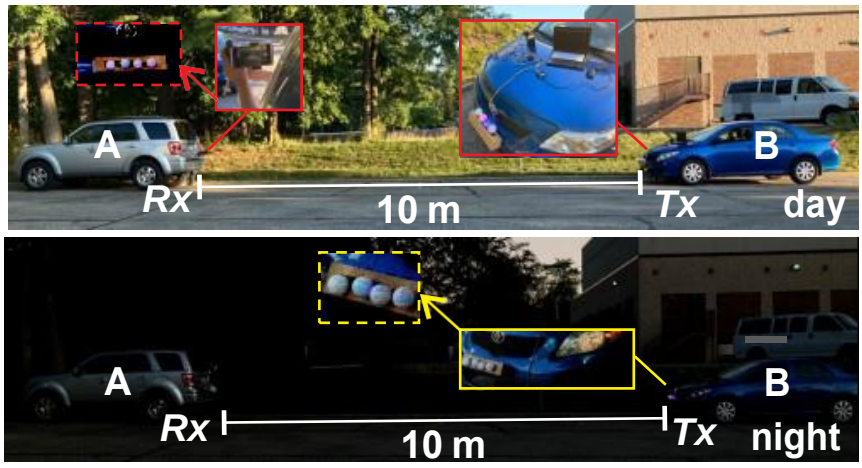
(2) Centrosymmetric Intra-Frame Embedding.

To address the issues above, we simply adjust an original RainbowRow symbol mapping in each frame into cenrosymmetric symbol mapping, as shown in Figure 3.15 (c). For instance, each frame contains 10 RainbowRow strips S_1 - S_{10}. When the transmitter embeds data of one frame, the half of data (from L_1 and L_2) in S_1 and the half of data (from L_3 and L_4) in S_{10} are emitted at the same time, while similar to $S_2 \leftrightarrow S_8$, $S_3 \leftrightarrow S_7$. Therefore, even with rotation angle mismatch, we can reconstruct most RainbowRow symbols in each frame to avoid the data rate drop caused by the decreased number of entire RainbowRow strips. We also set frame borders before the first strip and the last strip.

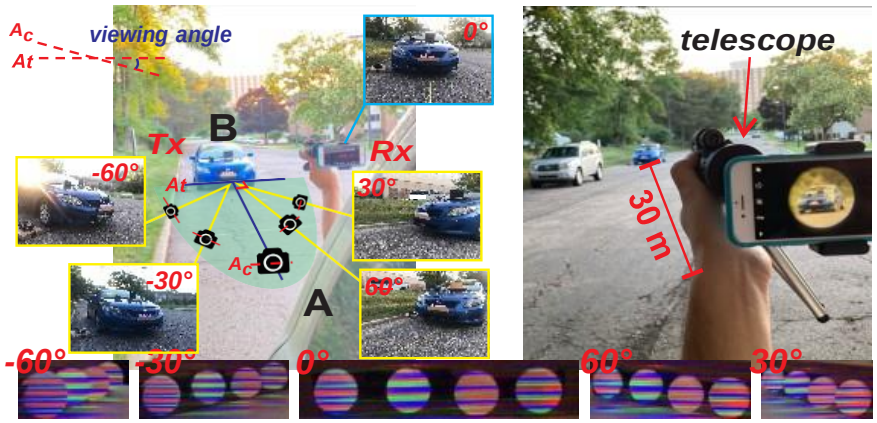
3.6.2 Adaptations for Vehicular Networks

(1) Varied viewing angle & long distance.

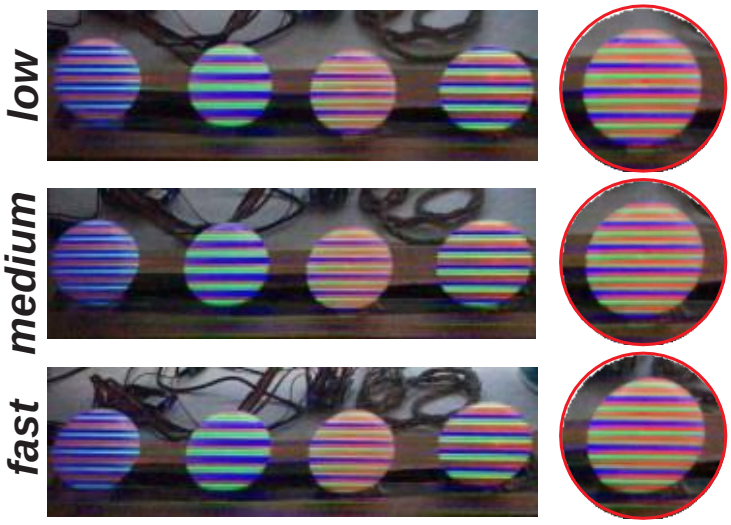
The RainbowRow transmitters and receivers can be mounted to cars and traffic infrastructures for both uplink and downlink services. Given one example of uplink from car B to car A, as shown in Figure 3.16 (a), the camera is installed on the back of A, while the RainbowRow transmitter is mounted on the front of B. In this case, both the transmitter and receiver are in a mobile scenario. The camera's horizontal axis A_c and the LED bar A_t are coplanar. However, these two lines are not in parallel when car A and B are in different or curved lanes. We define the angle between A_c and A_t as the **viewing angle**. Despite setting the physical horizontal gaps among nearby transmission



(a) vehicle to vehicle use case



(b) viewing angle mismatch and long distance



(c) impact of varied speed motion

Figure 3.16 RainbowRow adaptation for vehicular network: avoid impact of viewing angle, long distance and motion.

units, the different ***viewing angles*** will result in different physical horizontal gaps being captured, which will make decoding difficult. Furthermore, these gaps decrease significantly in the ***long distance*** between the transmitter and the receiver because of the perspective principle, as shown in Figure 3.16 (b).

(2) Use telescope instead of magnify lens.

Instead of magnifying lens, we switch to telescope lens to shrink the distance from the transmitter to the camera to eliminate varied and decreased horizontal gaps caused by varied viewing angles and long distance.

(3) Impact of high speed motion.

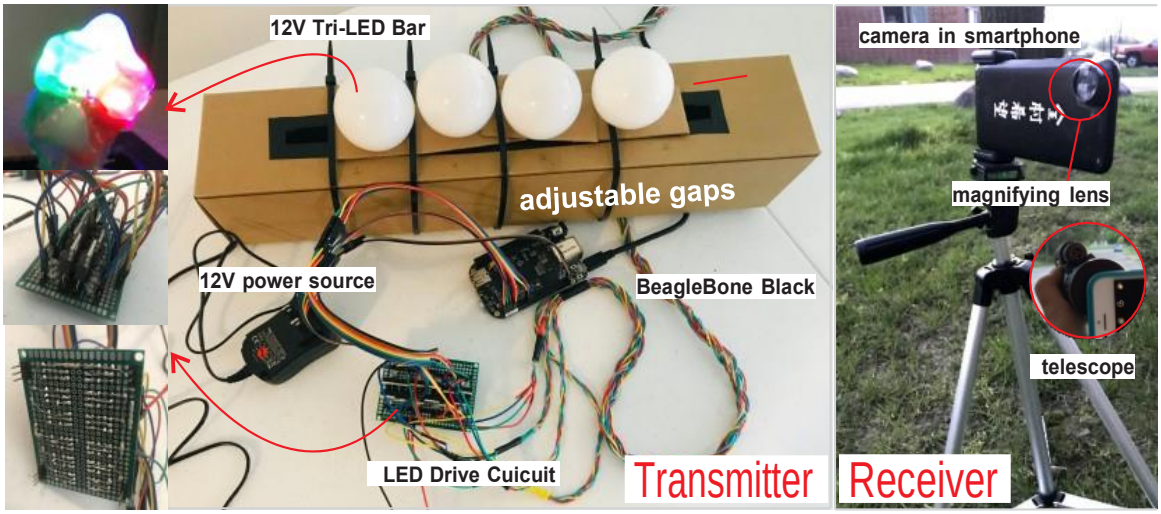
Although vehicles are in high-speed motion, the speed of light is 3 x 10^8 m/s which is overwhelmingly faster than the vehicles' speed. Therefore, the optical signals from the transmitter can be recorded in real time on their RainbowRow strips. The main impact of high speed motion is the varied sphere shape and size with different motion speed, which is sometimes a positive situation instead of a negative situation.

3.7 Implementation and Evaluation

Transmitter. We implement a low-cost RainbowRow prototype, as shown in Figure 3.17 (a). The transmitter consists of a BeagleBone Black MCU, slef-implemented fast LED drivers with MOSFET transistors, and a 12V self-made Tri-LED bar, total cost is under **$100**. Each transmission unit consists of a red, a green, and a blue LED bulbs with white sphere cover.

Receiver. The receiver is a commercial smartphone (VIVO Y71A or iPhone 7) with an additional commercial magnifying / telescope lens(< **$10**) and performs decoding via OpenCV. Some commercial smartphones already have **several camera modules** with magnifying and telescopic lens such as Huawei Mate 30, iPhone 13 and Samsung S22.

Setup. The RainbowRow implementation is shown in Figure 3.17 (a). We conduct experiments on our prototypes in two real use case settings: indoor office (Figure 3.15), and vehicular network (Figure 3.16). We also conduct an ablation study and a diversity robustness evaluation scenario (Figure 3.17 (b)). We set different rotation/viewing angles, distances, day or night, with/without



(a) RainbowRow implementation with commercial devices

(b) experiment scenarios for ablation study and other comparisons

Figure 3.17 RainbowRow implementation including Tx & Rx and experiment scenarios.

lens, relative motion speed and camera parameters settings for a comprehensive evaluation.

3.7.1 RainbowRow in Indoor Office.

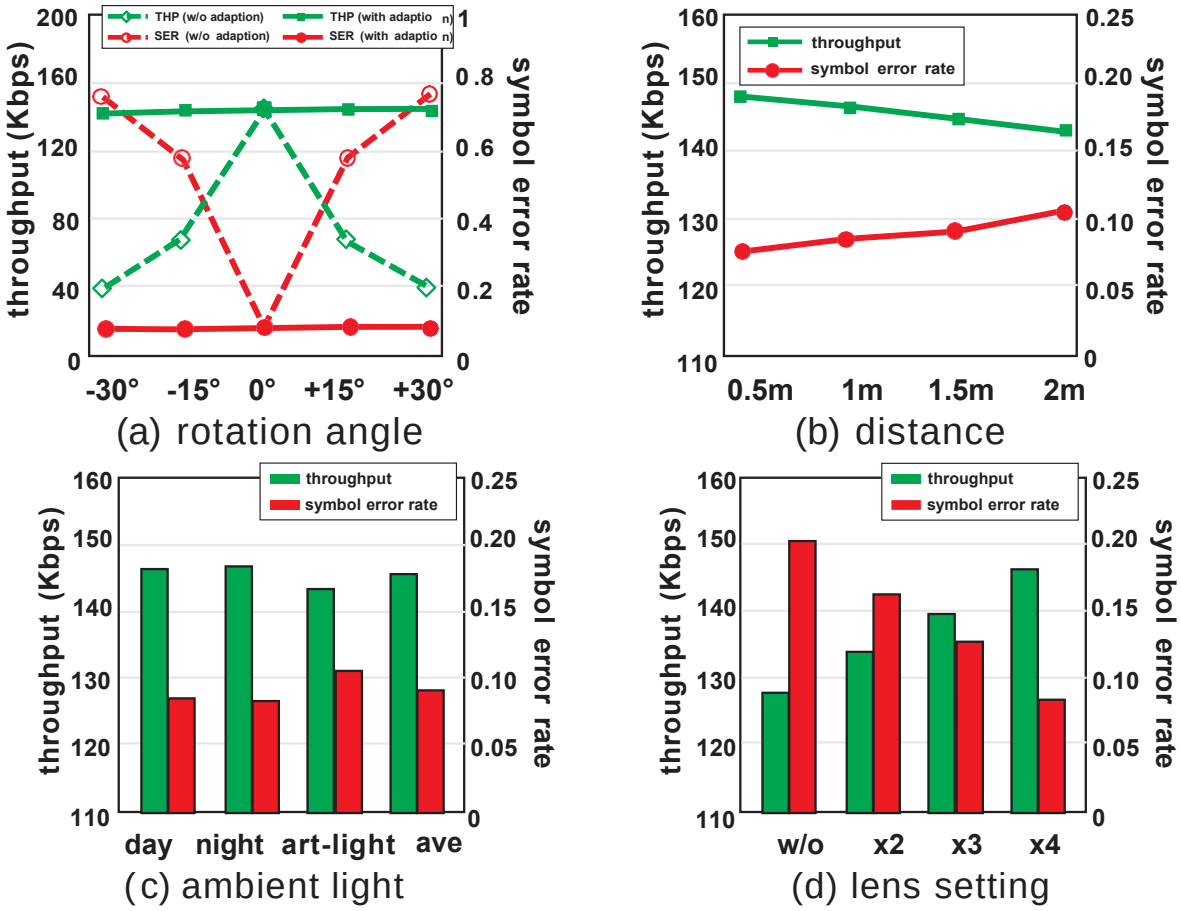


Figure 3.18 RainbowRow performance for indoor office use case.

We set transmission frequency at **10 KHz** while adjusting other settings to study their impacts to the achieved throughput in an indoor office.

Throughput vs. Rotation Angle We set the rotation angle (Figure 3.15) with 5 settings: -30, -15, 0, +15, and +30. We keep the distance at 1m during day time with the same lens setting. As shown in Figure 3.18 (a), RainbowRow achieves the highest throughput of 146 Kbps at 0 and decreased with the increased absolute value of rotation angle. We also present the data rate with centrosymmetric adaptation as contrast. The results demonstrate our centrosymmetric adaptation effectively addresses the rotation angle mismatch problem.

Throughput vs. Distance. We set 4 distances: 0.5m, 1m, 1.5m, and 2m. We keep the rotation angle at 0 during day time with the same lens setting. As shown in Figure 3.18 (b), the achieved data rate slightly decreased with the increased distance from the transmitter to the receiver from 148 Kbps at 0.5m to 143 Kbps at 2m, a change of only 5 Kbps.

Throughput vs. Ambient Light. We conduct experiments at day, night, and with an artificial light source (the added light by human) scenario to study the influence of ambient light. We set rotation angle at 0. We keep the distance at 1m during day time with the same lens setting. As shown in Figure 3.18 (c), there is no significant performance difference among three settings and RainbowRow achieves 146.4 Kbps, 146.7 Kbps, and 143.2 Kbps separately.

Throughput vs. Lens. We also evaluate the influence of different lens settings. We conduct experiments during the day with the rotation angle at 0. We keep the distance at 1m. As shown in Figure 3.18 (d), the achieved throughput increased with the use of magnification. These results demonstrate that using the magnifying lens can successfully address the problem of long distance within 2m.

3.7.2 RainbowRow in Vehicular Networks.

We set the transmission frequency at **10 KHz** while adjusting other settings to study their impacts to the achieved throughput in vehicular networks.

Throughput vs. Viewing Angle. We set the viewing angle (illustrated in Figure 3.16) with 5 settings: -60, -30, 0, +30, and +60. We keep the distance at 4m during day time with the telescope. As shown in Figure 3.19 (a), RainbowRow achieves the highest throughput at 128 Kbps at 0 and **did not** decrease with the increased absolute value of viewing angle. In contrast to other PD-based tight directional requirements, such as the ability to only follow the vehicle in the same lane, RainbowRow has a broad viewing angle between the transmitter and the receiver.

Throughput vs. Distance. We set the distance with 4 settings: 4m, 6m, 8m, and 10m. We keep the viewing angle at 0 during day time with the telescope. As shown in Figure 3.19 (b), the achieved data rate increases with the increased distance from the transmitter to the receiver from 128 Kbps at 4m to 133 Kbps at 10m. The reason is the telescope adaptation is suited better for

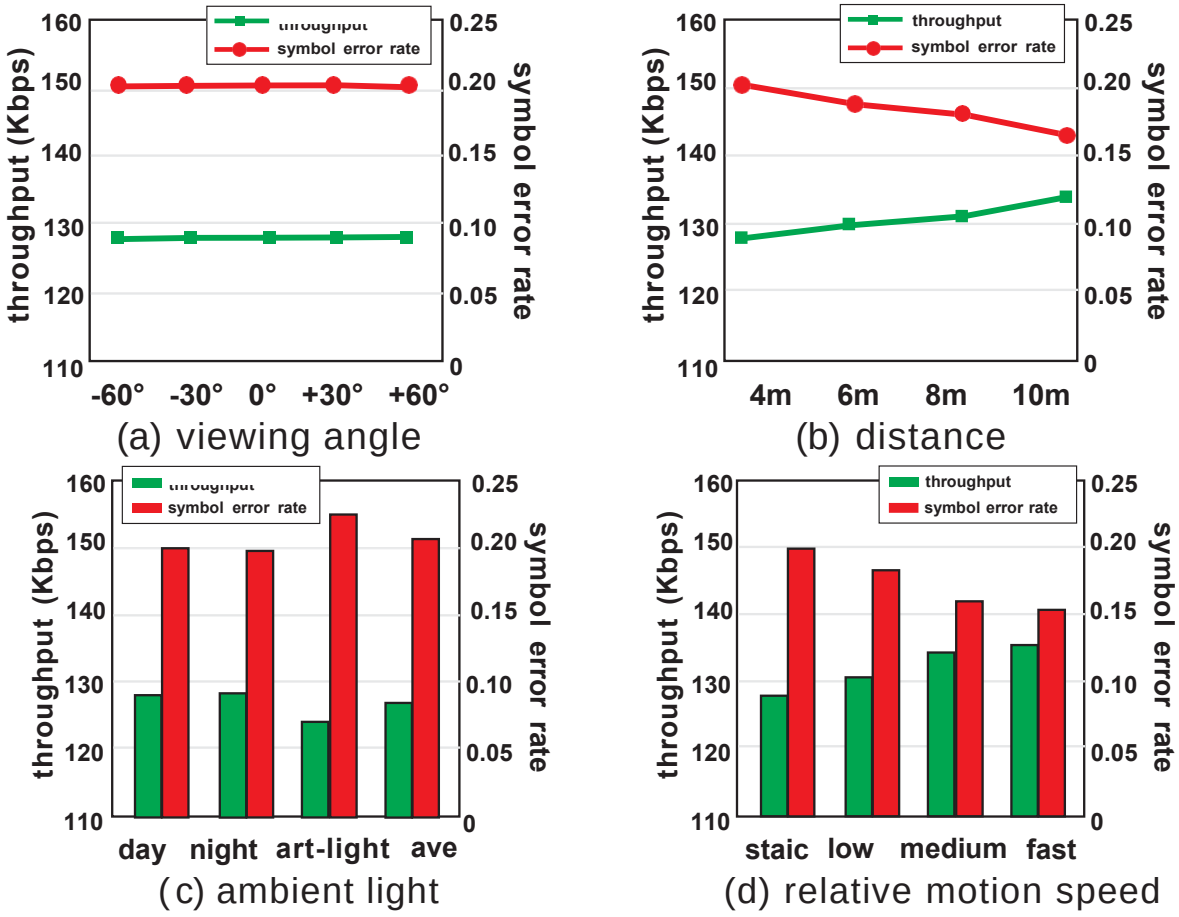


Figure 3.19 RainbowRow performance for vehicular network use case.

longer distance.

Throughput vs. Ambient Light. We conduct experiments at day, night and with an artificial light source scenario to study the influence of ambient light. We set viewing angle at 0. We keep the distance at 4m during day time with the same lens setting. As shown in Figure 3.19 (c), there is no significant performance difference among three settings. RainbowRow achieves 128 Kbps at day, 128.3 Kbps at night, and 124 Kbps with the artificial light source.

Throughput vs. Relative Motion Speed. We set 4 camera speeds in a horizontal direction to simulate the motion between vehicles. We keep the distance at 2m during day time with the same lens setting. As shown in Figure 3.19 (d), there is no significant performance difference among three settings. However, the captured shape of fast motion speed becomes larger than the static shape, which can even help to decode better due to the increased strip length and the strip number,

as shown in Figure 3.16 (c).

Summary. These results verify that RainbowRow with specific adaptations is suitable for both indoor office and vehicular network with benefits: (1) over 120 Kbps data rate with flexible distance up to 10m; (2) secure indoor communication and broader-view vehicular communication; (3) with no additional energy consumption due to its synchronous lighting function; and (4) robust with rotation/viewing angles and ambient light, and (5) low cost and easy to deploy due to the already mounted LED bulbs and cameras.

3.7.3 Ablation Study.

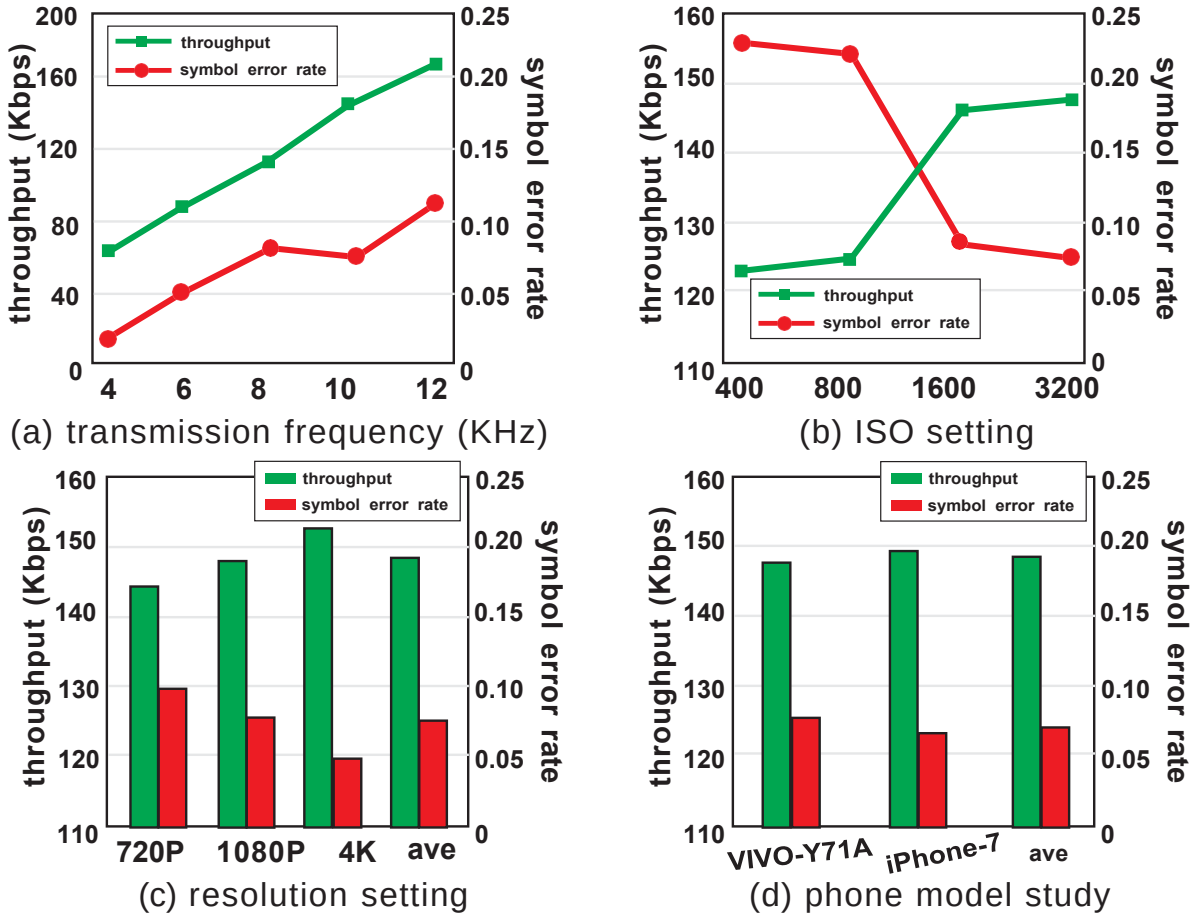


Figure 3.20 Ablation study for Rainbow in different camera parameter setting.

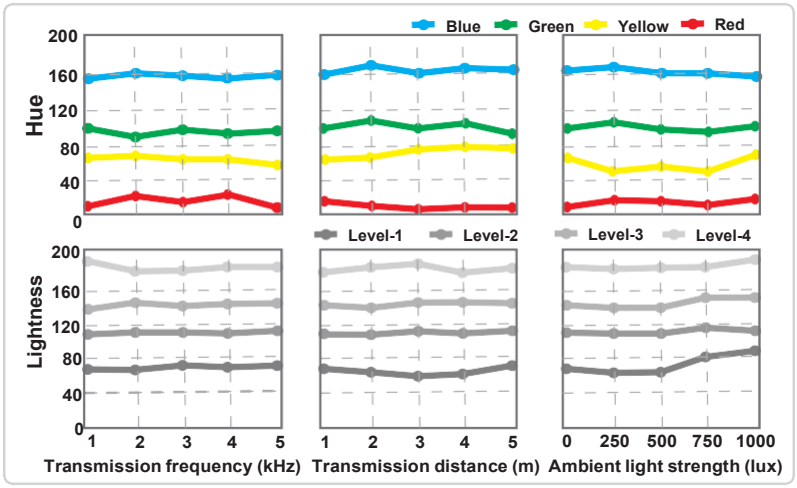
Transmission frequency. We set distance at 0.5m during day time and set different transmission frequencies from 4KHz to 12KHz. As shown in Figure 3.20 (a), RainbowRow can achieve

up to **170** Kbps. The data rate increases with increasing transmission frequency. **ISO:** We set transmission frequency at 10KHz during day time with 0.5m distance and set different ISO from 400 to 3200. As shown in Figure 3.20 (b), the data rate increases with increasing ISO value. **Resolution:** We set transmission frequency at 10KHz during day time with 0.5m distance and set different resolutions in [720P, 1080P, 4K]. As shown in Figure 3.20 (c), the data rate increases with increasing resolution. **Different phones:** We set transmission frequency at 10KHz during day time with 0.5m distance and set resolution at 1080P while using two commercial phones. As shown in Figure 3.20 (d), the achieved data rate are similar with the same parameter.

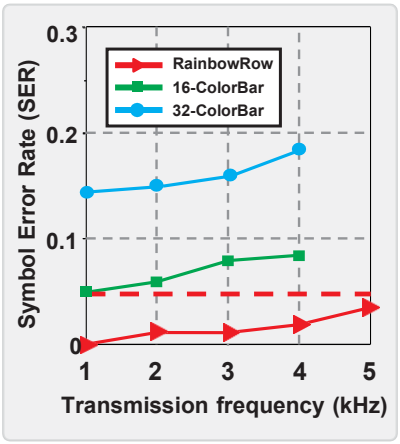
3.7.4 Comparison with existing work.

With existing LED-OCC. As shown in Figure 3.21 (a), both hue and lightness keep proper gaps for robust decoding in varied transmission frequencies, distances, and ambient light. Combining these, the proper symbol distance modulation assists the SER (symbol error rate) reduction and the throughput improvement compared with other high-order modulation methods such as 16-ColorBar and 32-ColorBar[38], as shown in Figure 3.21 (b)- (c). The throughput of RainbowRow is higher than 4-ColorBar and 4-CASK and even higher than the high-order 32-ColorBar and 8-CASK among all frequencies. The throughput of RainbowRow is about **10X** of 4-ColorBar and 4-CASK with the same diversity order. When the frequency is **5KHz**, RainbowRow can achieve up to **72Kbps**.

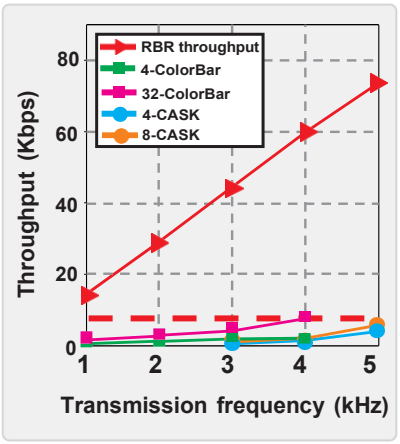
With other approaches. Although the current achieved data rate of over 120Kbps within 10m does not compete with similar range RF techniques such as Bluetooth at 1Mbps within 10m, RainbowRow is more secure due to its LoS propagation in the physically individual space with the great potential for dense spatial multiplexing and simpler interference control compared to RF-techniques. We also build a radar map for comparison among RainbowRow with other approaches: (1) LiFi (LED-PD), (2) RF-based, (3) screen-camera (LCD-OCC) in 8 aspects with their performance ranking: (1) data rate, (2) distance, (3) security, (4) energy efficiency, (5) flexibility, (6) low-price, (7) broad view, and (8) broad bandwidth, as shown in Figure 3.21 (d). These results show our RainbowRow generally outperforms than the existing approaches by its practical data rate, long distance, secure feature, energy-efficient, suitable for numerous use cases,



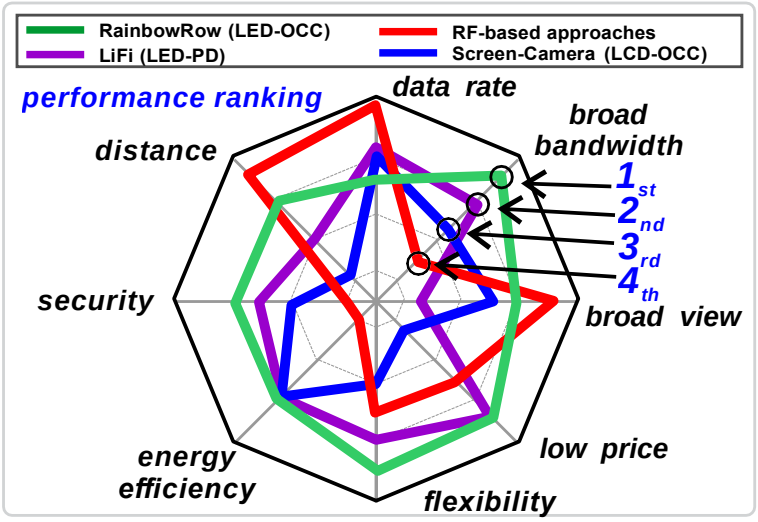
(a) robustness of 4-order spectrum and amplitude



(b) SER comparison



(c) throughput comparison



(d) radar map comparison

Figure 3.21 The comparison of RainbowRow with the existing LED-OCC modulation and other related work.

low-cost, broad view and uncrowded spectrum.

3.8 Discussion and Summary

Some Concerns. (1) **Additional lens**. A smartphone's inbuilt telescope camera may be able to take a high-resolution image of the moon instead of needing a separate telescope due to the quick development of camera technology in smartphones and mobile devices. (2) **Energy consumption**. Similar to RF approaches, our RainbowRow can also design the wake-up mechanism to turn on/off the OCC function to avoid the always-on camera imaging. The LED bulbs in RainbowRow are energy-efficient and also offer simultaneous illumination capabilities instead of RF's sole communication. (3) **Practical Use Cases.** Our RainbowRow can be deployed in many scenarios, such as indoor/outdoor lighting, traffic signs, and vehicle lamps, lighthouses, and underwater/drone communication because of the wide deployment and low-cost of LED and commercial cameras.

Future Directions. (1) MAC and Handover. The dense deployment of LEDs for OCC small cells require multiple user access[15, 55, 105]. RainbowRow should allow users to switch from different optical cells for handovers. It is essential to design handover mechanisms for seamless communications and smooth mobility, and need to be studied appropriately in the future. **(2) Higher-speed potential.** Our RainbowRow is a worthwhile first attempt by utilizing low-order spatial diversity (i.e, 4) and gains up to a maximum of 170 Kbps, which is 20X of the existing 1D rolling LED-OCC approaches (i.e., < 8 Kbps). In the future, we could explore higher-order RainbowRow to boost LED-OCC's data rate further (e.g, 16 or even 64) by MCU with more control ports and fast synchronous controlling ability.

In summary, we propose *RainbowRow*, the first to utilize spatial diversity in 2D rolling blocks to boost LED-OCC's data rate for real-world applications. We model 2D rolling blocks and explore the modulation design combining this spatial diversity with other diversities for improved data rate. Furthermore, we address technical challenges in optical imaging management at both the transmitter and the camera. Then, we deploy RainbowRow testbed in 2 real-world use cases with practical adaptations. Our comprehensive experiments and results demonstrate that our *RainbowRow* protocol can achieve a throughput over 120 Kbps at up to 10m and outperforms

existing LED-OCC (<8Kbps, <1m). We believe that RainbowRow can be the beginning of LED-OCC in bridging the performance gap for future high-speed applications.

CHAPTER 4

3D SPATIAL DIVERSITIES ENABLED UNDERWATER NAVIGATION

Underwater optical wireless communication techniques hold great promise, offering a broad bandwidth and long communication range in comparison to existing expensive underwater communication methods like acoustic and RF-based techniques. This makes them particularly suitable for underwater navigation assistance, especially in dive and rescue operations. Adopting passive optical tags for object and human identification, as well as location-based services, proves to be a practical solution in these scenarios.

However, existing optical tags, such as barcodes or QR codes, typically employ one or two-dimensional designs, which can limit their robust decoding capability and full-directional localization capabilities required for underwater navigation tasks. To address this limitation, we propose a novel passive 3D optical identification tag-based positioning scheme for underwater navigation. Our unique UOID (Underwater Optical Identification) tag enables users to determine their current orientation by utilizing the arc of clockwise positioning elements. Additionally, the tag employs perspective principles to estimate underwater distances accurately. By incorporating these enhancements, our UOID tag overcomes the limitations of existing passive optical tags, providing a more effective and reliable solution for underwater navigation tasks.

4.1 Motivation

The ocean, other natural and man-made water areas (e.g. lakes, rivers, ponds, pools, reservoirs) account for more than 71% of the surface area of Earth. Although sea exploration has been undertaken throughout history, much of the underwater world remains a mystery that still needs to be explored by humans[89, 91]. Nowadays, there has been a growing research interest in numerous water-based applications such as climate change monitoring, oceanic animals study, oil rigs exploration, lost treasure discovery, unmanned operations, scuba diving, search/rescue, and underwater navigation assistance[134]. Additionally, it is reported by Market Reports that the Global Scuba Diving Equipment market was valued at USD 1127 million in 2020 and is projected to reach USD 1503 million by 2027[87]. Most of these applications require reliable, flexible, and

fast underwater communication to provide a safe and comfortable experience. However, despite the rapid development and progress of terrestrial and space communication, high-speed underwater wireless communication (**UWC**) is still not fully explored[89, 76, 62, 9].

There are significant differences between underwater and terrestrial scenarios, such as a harsh environment and lack of infrastructure deployment. When signals propagate in water, wireless communication faces challenges: water turbulence, limited power supply, unusable GPS, marine animal block issues. Today's most popular UWC techniques adopt acoustic, radio frequency (RF), and optical waves as wireless mediums. However, acoustic signals are generated by high-power sonar (sound navigation and ranging) equipment with a long communication range, but with the cost of high communication latency. As for RF-based UWC techniques, they have low latency but still face high energy consumption issues with a minimal communication range due to severe interference of seawater with the electromagnetic waves[46, 146, 54, 63, 89, 134].

Underwater navigation poses significant challenges due to the limitations of GPS and the cost associated with other acoustic/RF-based methods [89]. Traditionally, divers have relied on waterproof compasses and pre-dive location information from guides, which is not an intelligent, reliable, or flexible solution [121, 45, 66]. Drawing inspiration from terrestrial navigation, an alternative approach involves using waterproof signage systems to display location information for underwater navigation. However, this method faces difficulties as finite-sized map images or messages are challenging to locate and read underwater due to the harsh optical conditions. In light of these challenges, there is a need for innovative solutions that can provide reliable and efficient underwater navigation assistance, taking advantage of the unique characteristics of the underwater environment.

An alternative solution to address the challenges of underwater navigation is to utilize passive tags along with a portable tag reader, providing embedded and clear navigation information. Passive optical tags, such as barcodes and QR codes, are already popular in our daily lives [81, 138]. However, their short communication range makes them ineffective for underwater navigation since users may struggle to locate the tags and scan them underwater. Increasing the tag size could

enhance the communication range, but this approach comes with drawbacks, such as higher costs and potential disruption to the original ecological environment. Thus, it's essential to explore more efficient and environmentally friendly ways to improve the communication range of passive tags for underwater navigation without compromising on cost and ecological impact.

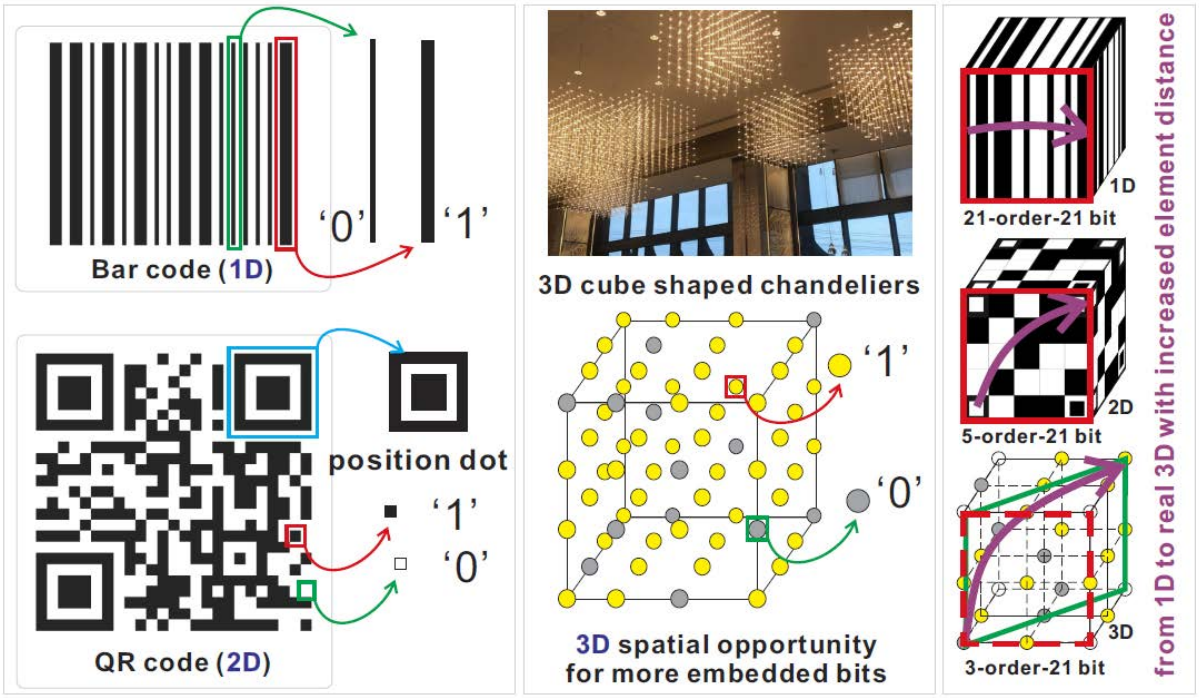


Figure 4.1 Existing optical tags and 3D spatial diversity.

When discussing passive tags we define a high-order tag as containing more than five elements per dimension. For example the barcode in the left of Figure 4.1 contains 16 columns, or 16 elements in its one dimension. We also define a low-order tag as having five or less elements per dimension. High-order tags, however, are not feasible for underwater navigation because as the number of elements increases the error rate also increases due to the necessity for elements to be physically closer to each other. On the other side, the amount of embedded data in a low-order barcode or QR code is not rich enough for underwater navigation.

Motivation: (1) Acoustic and RF-based UWC is not feasible because of drawbacks such as high latency, low communication range, or need for an external power source. (2) High-order optical tags cannot be reliably used for underwater navigation because of their error rates and

short communication range. (3) Existing optical tags only utilize 1D/2D spatial diversity for data embedding[111]. Even the 3D versions of Bar/QR codes shown in Figure 4.1 have limited element distances and ignore 3D spatial diversity. As a result, there will be more error bits in decoding, especially in muddy underwater scenarios. (4) Existing bar/QR codes, even in 3D, have limited scanning angles and require the user to move to directly face the surface of the codes, which is inconvenient for underwater navigation activities. (5) We can use 3D spatial features to provide underwater positioning based on the perspective principle, which states that objects such as cubes are observed differently at different distance and angles.

To address the problems above, we design **U-Star**, an underwater signage system based on passive 3D optical identification tags for underwater navigation, as illustrated in Figure 4.2. U-Star consists of UOID tags and the AI-based mobile tag reader. UOID tags are hollowed-out cubes which consist of data elements and positioning elements. The data elements are positioned with proper non-Line-of-Sight spacing on the UOID tag. The positioning elements are set in different clockwise color sequences along the six faces of the UOID. The U-Star tag reader is built on waterproof mobile devices with standard, commercial cameras.

4.2 Background and Related Work

4.2.1 Underwater Navigation

Underwater navigation is important for human-related underwater activities, such as scuba diving and underwater accident rescue. Natural underwater navigation requires the diver to utilize physical contours and characteristics of dive sites and combine basic compass skills to find the path to their destination[45, 43, 7]. Natural underwater navigation is similar to terrestrial navigation, the diver first needs to known his/her current location based on the site map or underwater physical features of dive sites and then guide him/herself to their destination based on the information on map or prior knowledge. However, natural underwater navigation relies highly on diver's familiarity with dive sites. If unfamiliarity or any confusion with dive sites, it is very dangerous for divers, to the point that many have lost their lives.

Many researchers have made efforts to improve underwater navigation[88, 46, 65, 47]. However,

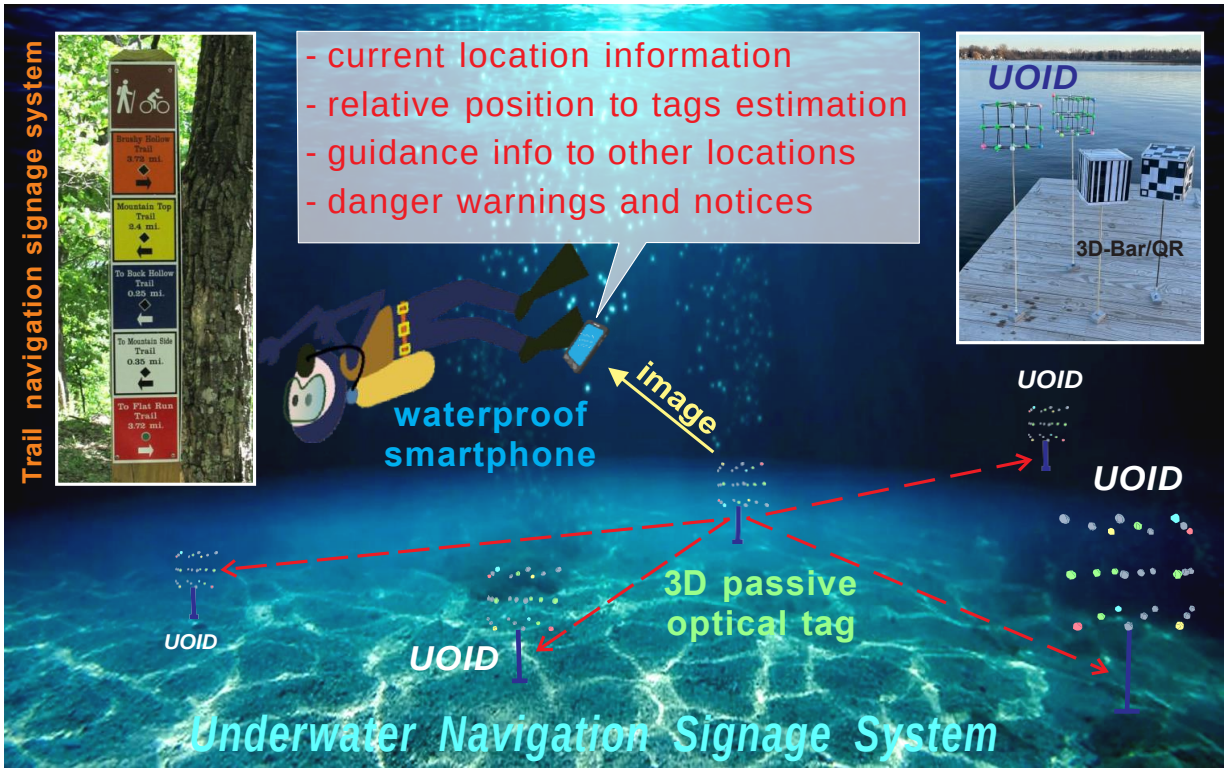


Figure 4.2 U-Star underwater navigation illustration.

these are based on acoustic and RF techniques that incur significant drawbacks, including high power consumption, expensive price, long latency, or short communication ranges. To combat these issues, we explore setting underwater, on-site visible signage tags to provide site location information and navigation guidance. Our approach is inspired by traditional terrestrial navigation techniques such as tour maps and location marks in hiking trails[71, 70] and offers new and innovative techniques for underwater navigation.

However, it is not practical to just place the signage tags underwater in a similar fashion to terrestrial navigation. This is because it is not as easy for users to move to directly face the tags as it is on land, the hostile underwater optical environment, and that the lengthy communication distance [47, 44] makes effectively reading the signage impossible. The optical tags used in underwater navigation need three features: (1) **Easily observed**. The color and brightness are striking enough to be observed by users at long distances (10m-20m) and the content on the tag should be visible from practically every angle. (2) **Enough data capacity**. The data embedded in the tags needs

to be large enough to record both the location information and guidance advice. (3) **Positioning ability**. The tag needs to provide relative position information to the user. Feature (1) is more based on material and color choice, specifically, to suit the underwater scenario. Feature (1) also relies on the hollowed-out structure of the tag design for the real 3D passive optical tag. Features (2) and (3) are in the category of optical wireless communication and we discuss below.

4.2.2 Existing passive 1D/2D optical tags

Barcodes and QR codes are widely used machine-readable optical tags in our daily lives. Barcodes, invented in 1951, represent data using parallel lines with varying widths and spacing [113]. They became commercially successful in supermarket checkout systems. Later, two-dimensional (2D) variants known as matrix codes were developed, capable of representing more data per unit area [111, 100]. One of the popular matrix codes is the QR (Quick Response) code, widely used in various aspects of life, such as mobile payment, social E-cards, electronic tickets, access control. High-order QR codes, like the version 40 QR code (177x177), can embed 23,648 bits [100].

However, in underwater navigation scenarios, using high-order bar/QR codes is not suitable due to their limited scanning angles, restricted data element distance, and the challenging optical environment's quality. These limitations make it difficult for users to see and scan the codes effectively underwater. Thus, there is a need for more robust and underwater-friendly optical identification tags that can address the unique challenges of navigation in aquatic environments.

These bar/QR codes only focus on 1D and 2D spatial diversity and ignore the potential opportunity of three-dimensional spatial diversity in optical tag data embedding. Even with the 3D version of Bar/QR codes (six planes of the cube are covered with the same bar/QR codes to ensure consistent content at various angles), the user can record up to three repeat bar/QR codes, which does not increase data element distances and does not fully take advantage of 3D spatial diversity in data embedding. Our 3D optical tag design is inspired by 3D cube-shaped chandeliers, but improved and modified for the data and communication needs of underwater scenarios. Each element inside of a 3D light cube can denote bits **1** and **0** via its **On** and **Off** status, as opposed to linear or

matrix dots on a surface in a bar/QR code. Although images of the 3D optical tag captured by our tag reader is a 2D pixel matrix, we can restore the 3D optical tag based on perspective principles. When compared to optical tags with the same tag size and the same amount of embedded data (e.g., 1D, 2D codes, and surface 3D tags with 1D/2D codes attached), our proposed 3D hollowed-out cube improves data element distance by leveraging 3D spatial diversity in data embedding. In our U-Star system[141], we design UOID, passive 3D optical identification tags, to utilize the 3D spatial diversity to increase the distances among data elements for robust and full-directional underwater decoding.

4.2.3 Optical Positioning and Perspective

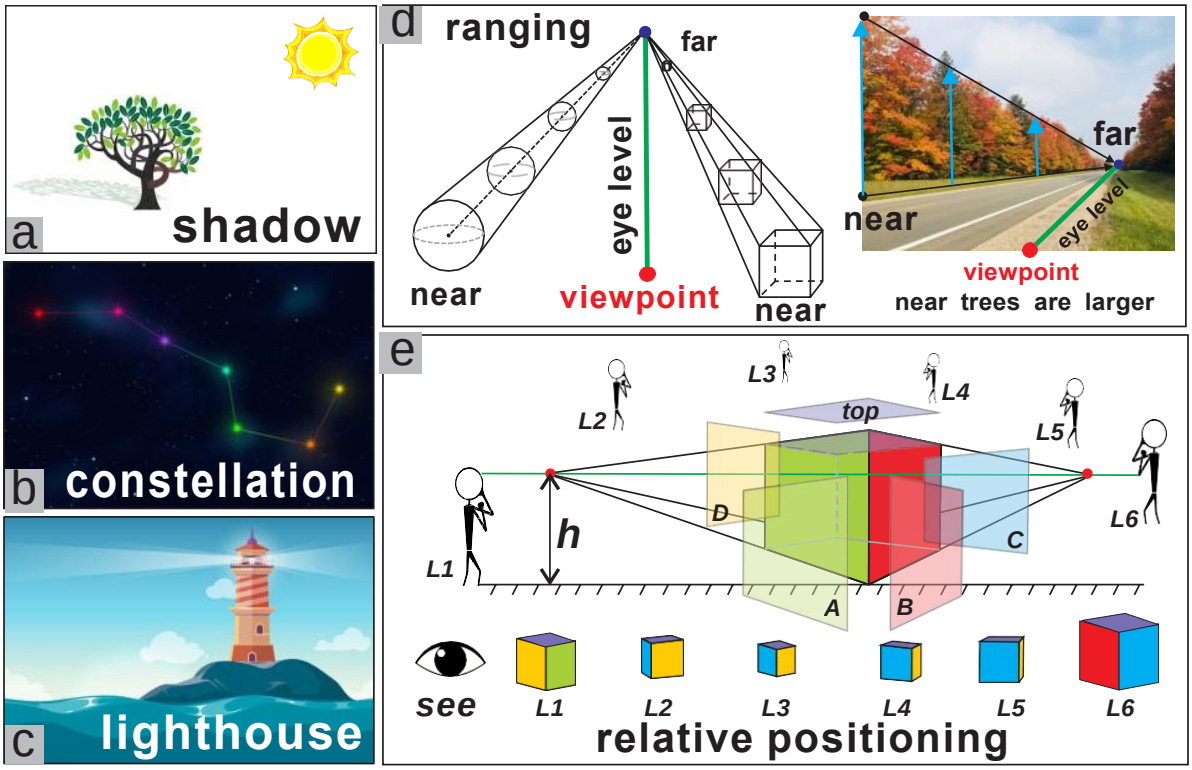


Figure 4.3 Perspective principle for positioning.

It is very common for humans to utilize natural or human-made luminous objects for positioning, as shown in Figure 4.3. For example, we can determine orientation by observing the direction of the shadows during the day time due to the sun's movement and the direction of the Big Dipper at the night because the orientation of the Big Dipper is unchanged and always pointing to the

Earth's North Pole[10, 25]. In addition to orientation and localization based on natural optical objects, lighthouses are an example of human-made optical object based positioning. The basic functions of lighthouses are to guide ships, indicate dangerous areas and help ships to determine their positions[94]. For underwater scenarios, researchers have also made many efforts to design optical-based underwater positioning mechanisms and systems. Akhoundi et al. design RSS (Received Signal Strength) based optical positioning systems that calculate location based on the received optical signal from multiple anchors[4]. In other work[135], the authors proposed a ToA and RSS-based underwater localization system. However, these works require a significant power supply and expensive devices with high-accuracy sensors.

Perspective principles are traditionally used in vision and art[74, 96]. Creatively, we can utilize the perspective principles for ranging and relative positioning. The perspective principle simply describes the visual relationship between the observer and the observed object: (1) increasing the distance between the observer and the object results in a reduced size of the observed object, as shown in Figure 4.3 (d); (2) varying the angle from the view point to the object results in a variable shape and observed content of the observed object, as shown in Figure 4.3 (e). Our U-Star design also utilizes UOID tags as fixed underwater beacons utilizing 3D spatial diversity for optical ranging and orientation guidance besides its data embedding.

Compared with existing work, our UOID tags are based on passive optical wireless communication and therefore utilize natural light sources to present data and provide relative positioning without energy consumption concerns. The tag readers are also commercial camera-based devices instead of expensive sensors.

4.3 Our Approach: U-Star

Our proposed underwater navigation system consists of two parts, as shown in Figure 4.4: (1) 3D passive optical tags: UOID tags, and (2) AI-based mobile tag reader.

UOID tags. UOID tags are anchored underwater with fixed facing direction. They are made of fluorescent materials and can absorb light from natural underwater environment or users' flashlight. There are data elements and positioning elements in UOID tags, which are assigned with proper

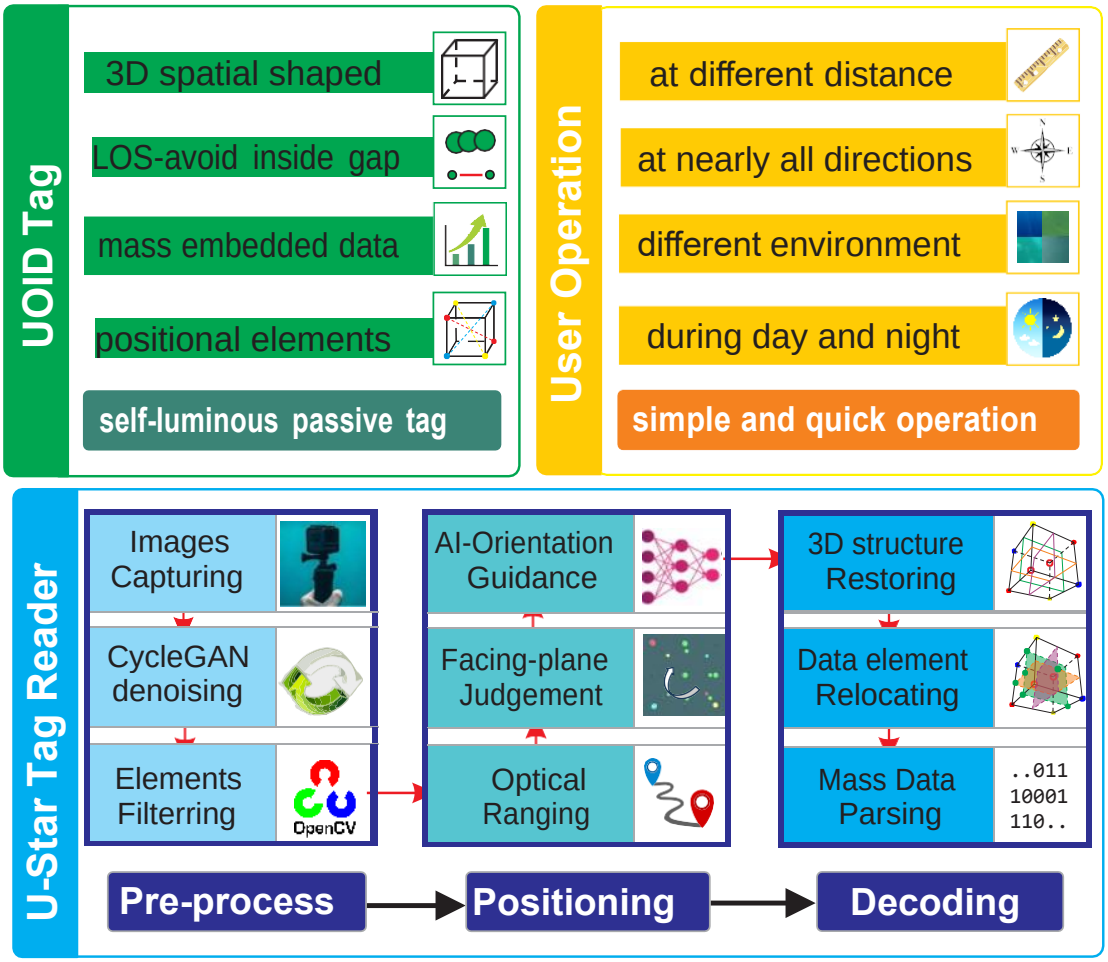


Figure 4.4 U-Star system diagram including UOID tag, user operation, and tag reader.

spacing to eliminate LoS blockage in the tag's 3D spatial domain to present data.

Tag reader. The tag readers are based on commercial smart devices such as smartphones or sports cameras. These devices can capture images of UOID tags and perform underwater, robust, and real-time data parsing and relative positioning by its onboard computation abilities. The U-Star tag readers have three key modules: (1) CycleGAN denoising based pre-processing, (2) CNN based relative positioning, (3) 3D restoring based decoding.

User operation and navigation procedure. The detailed U-Star underwater navigation procedures are: (1) The diver, equipped with a tag reader, looks for luminous UOID tags. (2) The diver uses waterproof tag reader to take pictures of a specific UOID tag at current location. (3) The tag reader performs image style transformation for denoising, then the tag reader can determine

diver's relative position including the distance estimation and orientation guidance. (4) The diver knows where he/she is now and can navigate him/herself to new sites based on the pre-recorded data from the backup database that the tag reader can query with the embedded data from the UOID tag (which we call a query code). The user operation is simple and quick and can be performed at different distances with all directions in different environments and time.

4.3.1 Challenges and Solutions

(1) **LoS blockage.** When capturing tag images, some inside elements are blocked by their front elements due to lights' line-of-sight propagation. We address this by assigning elements with proper spacing and a machine learning based restoration. (2) **Harsh optical environment.** The underwater environment decreases the quality of captured UOID images, and thus makes them hard to decode. We design CycleGAN based algorithms to transfer unclear images into clear images (Unity3D-style images) before decoding. (3) **Underwater relative positioning.** The UOID tag is expected to help determine the distance between the user and the tag as well as user's current orientation for relative positioning. We propose clockwise positioning arc schemes to denote planes and a CNN method to infer relative position. (4) **3D decoding.** The tag reader needs to restore each element to a standard 3D space from a random 2D image during decoding. We utilize the perspective principle to reconstruct the 3D structure for data parsing.

4.3.2 Advances Compared with Prior Art

(1) Same tag order with more embedded bits. Despite the fact that the user can capture the information of one and up to three surface planes of a 3D version of existing Bar/QR codes in N-order, the decoded bits are the same as the bits in one plane. The embedded bits in an N-order barcode are roughly N. The embedded bits in an NxN QR code are roughly N^2-4 bits. The embedded bits in an NxNxN UOID tag are N^3-6 bits. The amount of embedded bits in a UOID tag increases exponentially compared to the same order 1D/2D optical tags. Even their 3D versions cannot compare to the UOID tags (e.g., 3-order UOID embeds 7x and 4.2x bits of the same order Bar and QR code).

(2) Same tag size & data with larger element distance. The larger the average element

distance and the broader the distribution of element distances, the better the detection performance and the less the error bits. We measure distances between all 21 data elements in 3D versions of the Bar/QR, and UOID tag that have the same embedded bits and tag size (edge is 19cm). Data element distances in Bar and QR codes are all smaller than 20cm, however the data element distances in UOID tags are completely distributed in a greater range of [5, 30] cm.

Our **contributions** can be summarized as follows:

(1) This is the first work to employ passive **3D** optical identification tags for underwater navigation. We model 3D spatial diversity and utilize it to increase the distance of data elements in our proposed UOID tags for simple and robust underwater navigation.

(2) We propose a passive 3D optical identification tag based positioning scheme for underwater navigation. Our UOID tag can help user to determine their current orientation by the arc of clockwise positioning elements and estimate the underwater distance due to perspective principles.

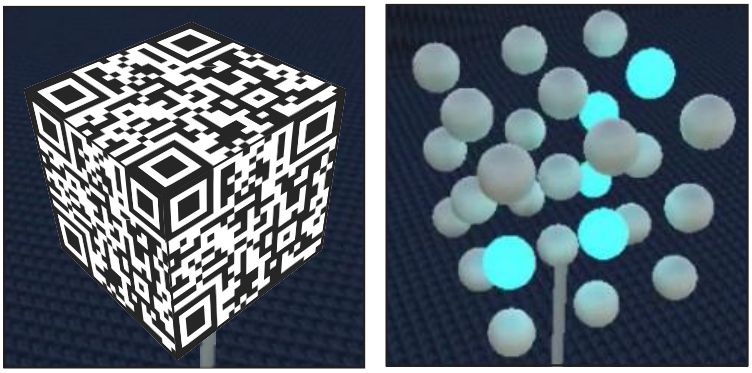
(3) We propose AI-based mobile algorithms at the tag reader for robust UOID decoding. We design CycleGAN based underwater denoising, CNN-based relative positioning, and real-time data parsing algorithms without significant computation overhead, latency or energy concerns.

(4) We implement U-Star and evaluate its performance on UOID tag prototypes in different underwater scenarios. Our experiment results show that a 3-order UOID tag can embed 21 bits of data with a BER of 0.003 at 1m and less than 0.05 at a distance of up to 3 m. We also make fair comparison with existing optical tags (Bar, QR) to show the superiority of our UOID tags in underwater navigation. U-Star also achieves over 90% accuracy for both optical ranging at up to 7m and orientation guidance.

4.4 Passive 3D Optical Tag

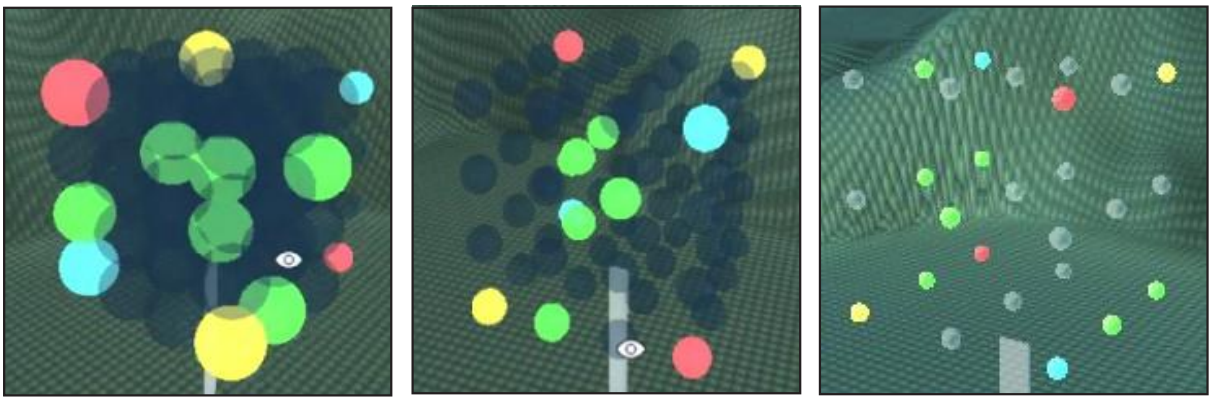
4.4.1 3D Spatial Diversity Exploration

As shown in Figure 4.5, we use a 3D cube instead of a 2D matrix to represent more bits in an optical tag. Naturally, there are two methods to embed data in a 3D cube: (1) embed data on its six surfaces, (2) embed data on both its surfaces and inside space (i.e., hollowed-out), which fully utilize the 3D spatial diversity. For method (1), the tag reader can only capture the dots on 1 and up



(a) surface 3D (b) spatial 3D

Figure 4.5 Surface/real 3D.



(a) 4-order tag without & with spacing (b) 3-order with spacing

Figure 4.6 Proper spacing to combat LoS.

to 3 surfaces due to the line-of-sight (LoS) characteristic of light. Method (1) cannot also guarantee that the embedded data captured at different angles is always the same (unless all 6 planes cover the same content) due to the potential of capturing different surfaces, which means that the tag's decoded data will change without consistency. Additionally, Method (1) results in smaller data element distances and a shorter communication range.

Thus we choose method (2) to embed data in our UOID design. However, the LoS issue can also occur if we embed data inside of a 3D cube due to mutual blockage among elements physically near each other. As shown in Figure 4.6, the 4-order (4x4x4) tag without proper spacing will have the mutual blockage issue. Three factors affect the blockage: (1) Tag order. As the order of tags

increases (3-order, 4-order, 5-order), more and more blockage occurs. Similarly, as the order of tags decreases, so does blockage. (2) Element size. The smaller the element size, the less blockage. (3) Mutual Spacing. The larger the mutual spacing of elements, the less blockage. We discuss a 3-order UOID tag with fixed element size and we address the mutual blockage by extending the spacing among nearby nodes to guarantee the tag reader can capture all elements in most cases.

4.4.2 UOID Tag Design

Positioning and data elements. In our UOID tag design, there are two types of elements: positioning elements and data elements, as shown in Figure 4.7. The positioning elements are on six vertex points with three pairs of colors. The positioning elements help determine the relative position of the user and assist in reconstructing the 3D cube for data parsing. The data elements make up most of the elements in a UOID tag for data embedding. They are located at the two remaining vertex points as well as inside of the tag itself.

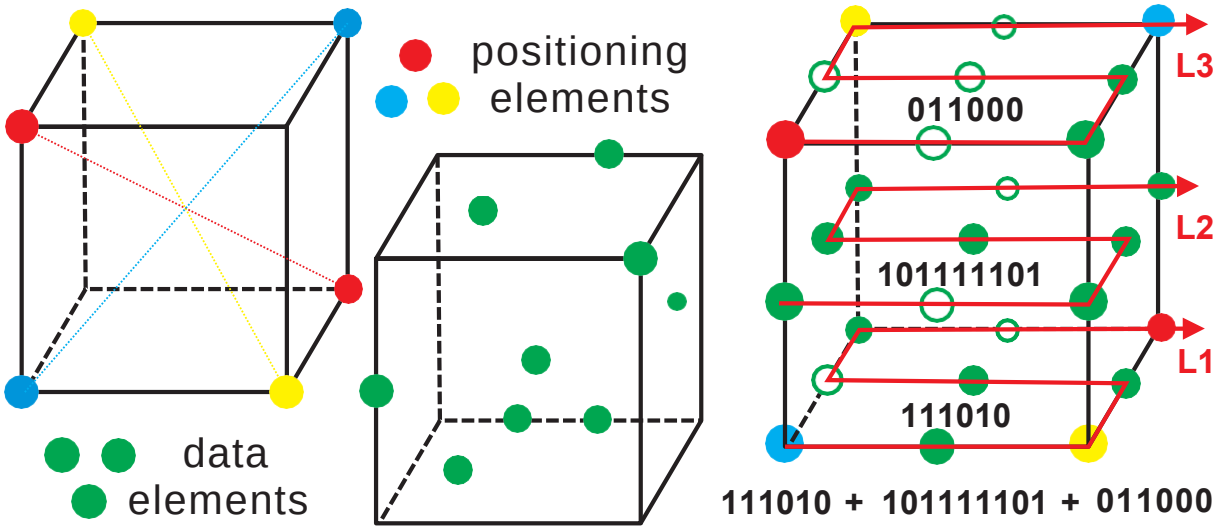


Figure 4.7 Two element types in UOID.

Positioning elements. As shown in Figure 4.7, each pair of colored elements are at a pair of vertex points. Thus, each plane of the cube has three different colored positioning elements. They can denote six surfaces based on the generated clockwise arc color sequence, Figure 4.10 (a)). Then the tag can determine which surfaces the user is facing based on captured surfaces of the

tag and determine orientation based on the perspective principle to support underwater navigation. Furthermore, these positioning elements can help to reconstruct the 3D structure from captured 2D images based on the perspective principle for data parsing. The reason for using three instead of four positioning elements to denote a plane are: (1) Three dots can already determine a surface. Four dots will sacrifice the positions that could be used for assigning data elements and thus decrease the embedded data. (2) Fewer overall colors is desirable, as more colors will increase the color detection error for decoding due to the fewer hue gaps.

Data elements. The data elements of our UOID are assigned to various 3D spatial locations. There are three layers L1, L2, and L3. For each layer, we assign data elements in an '**S**' shape. If the data element is colored green, the embedded bit is **1**, if the data element is not colored, the embedded bit is **0**. As illustrated in Figure 4.7, L1 embeds bits '111010', L2 embeds bits '101111101' and L3 embeds bits '011000'. This 3-order UOID tag embeds a total of 3^3-6=21 bits, '111010 101111101 011000'. We set the current angle of view to be the standard coordinate system for data parsing. With the assistance of positioning elements (Figure 4.16), we can map the tag images from any angle of view into the standard coordination system and then conduct the mass data parsing.

4.4.3 Underwater-specific Tag Design

Color Choices. Light with different wavelengths/color have different absorption rates in water. As shown in Figure 4.8 (a), the green and blue light have less absorption in deeper underwater environments such as a depth of 20 m[140, 89]. However, considering most commercial underwater activities do not exceed depths greater than 10 m, the color choices (red, yellow, green, and blue) above in the UOID tag are reasonable (for deeper underwater navigation, finer-grained blue and green can be chosen). As shown in Figure 4.8 (b), these four colors also have sufficient hue value gap to decrease the wrong detection of colors during decoding[148]. The green light has the longest emission time after 5s of being shined by a flashlight as shown in Figure 4.8 (c). Because data elements are the most numerous and important elements, we set them to green.

Luminous powder. Our UOID tags are passive, without any power supply. As illustrated in

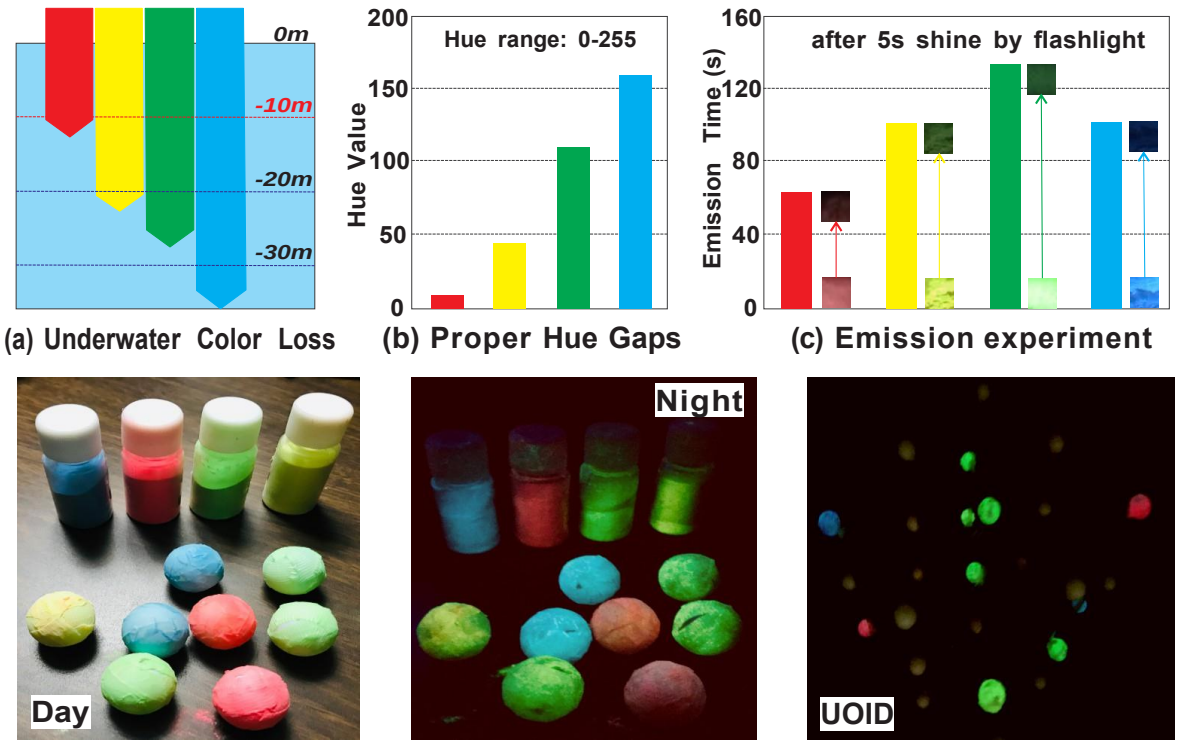


Figure 4.8 Color choices and luminous powder.

Figure 4.8 (d), we coat the elements in luminous powder, which is cheap and nontoxic to marine animals. As shown in Figure 4.8 (c), the luminous powder with our chosen colors can keep emitting light more than 60 seconds (1 min) after being shined by a flashlight for 5 seconds in our experiments. This ensures that the UOID tags work by absorbing natural underwater light and emitting light in specific colors, allowing us to see and scan UOID tags at any time of day or night.

4.5 Underwater Positioning

4.5.1 Optical Ranging

For underwater navigation, the perception and estimation of distance is very important. Our UOID tags can give the user a rough feeling of the distance between themselves and the tag. We use the rough size of the captured tag to infer the current distance from the user to the tag. The estimated relative distance has no relation with the angle of capturing images by the user.

As shown in Figure 4.9, We can estimate the distance based on the captured tag size because the

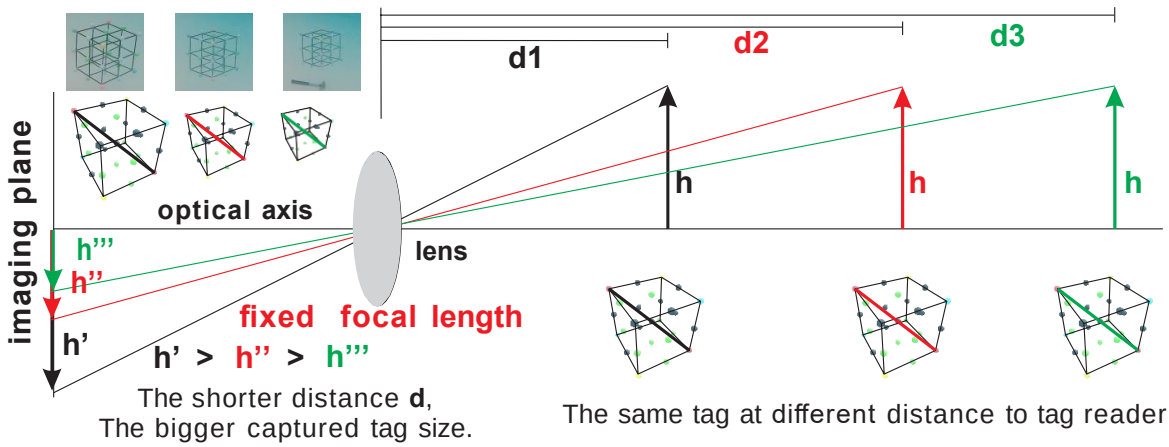


Figure 4.9 3D spatial perspective based optical ranging.

tag size increases when the user is getting closer to the tag due to the spatial perspective principle. We first collect the captured images (camera is set with fixed focal length) at different distances and use this dataset to train the CNN model for classification offline. Then we can use the trained CNN model to predict and estimate the current distance from the user to the tag in real-time.

4.5.2 Orientation Guidance

We map the six planes of the UOID tag onto six different clockwise color arcs which start from the non-positioning element: Yellow(Y)-Blue(B)-Red(R) maps to Plane 1, BRY to Plane 2, RBY to Plane 3, YRB to Plane 4, RYB to Plane Top, and BYR to Plane Bottom as shown in Figure 4.10. The UOID tag is fixed underwater (i.e., a specific plane of the UOID tag always faces in a specific direction), and thus the user/tag reader can know his/her orientation based on the plane of the UOID the user is currently facing. For example, as shown in Figure 4.11, Plane 1 is facing South. That means if the user is facing Plane 1, the user can know his/her current orientation is directed North.

For underwater navigation, the Plane Top and Bottom faces do not provide value to orientation decisions. Additionally, North, East, South, and West are not sufficiently descriptive for navigation. Therefore, we define 8 user facing orientations: North (facing Plane 1), Northwest (facing Plane 1 & 2), West (facing Plane 2), Southwest (facing Plane 2 & 3), South (facing Plane 3), Southeast (facing Plane 3 & 4), East (facing Plane 4), Northeast (facing Plane 4 & 1) as shown in Figure 4.11. Naturally, we can determine the plane the user is facing based on the color arc detected in images.

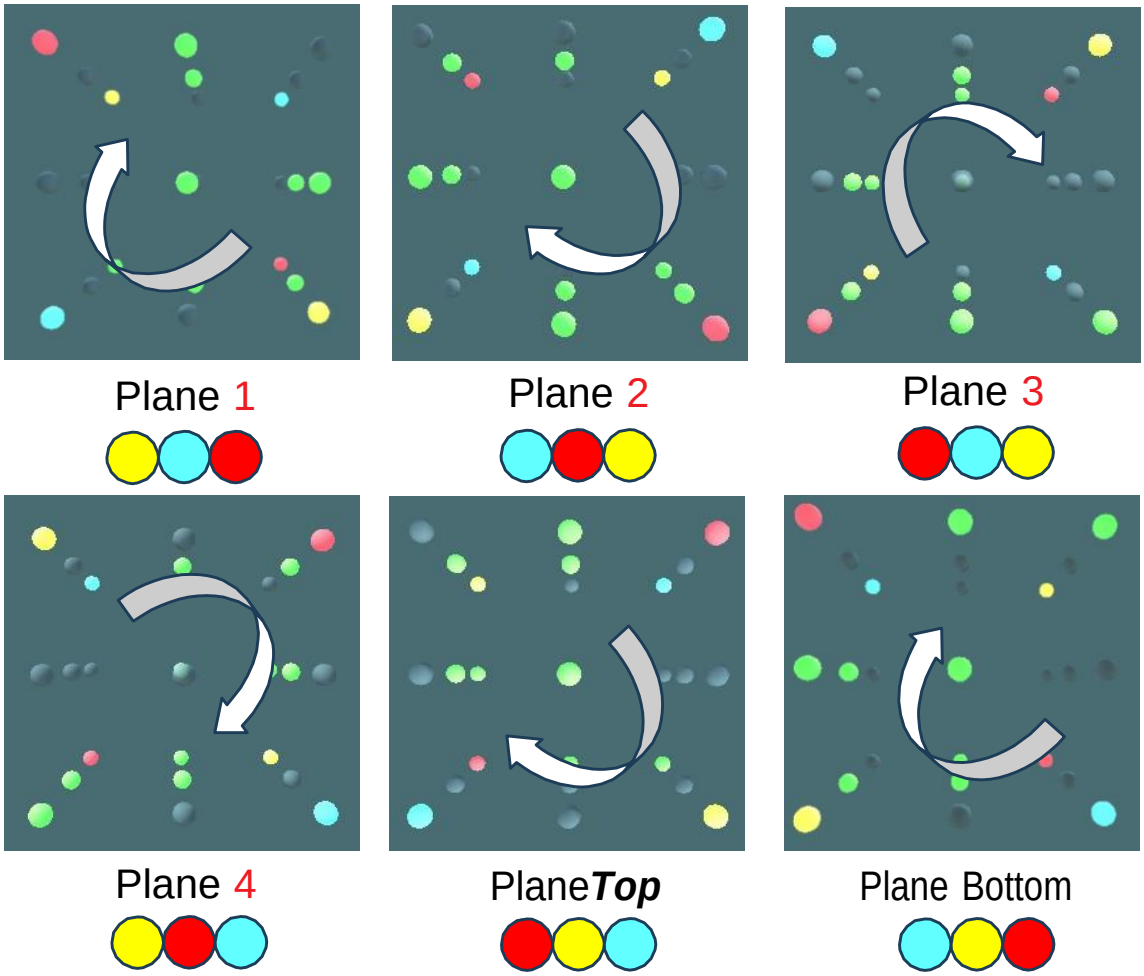


Figure 4.10 Positioning elements for the plane decision.

However, due to the small size of elements in captured images, it is hard to judge which plane the user is facing. Thus, we employ CNN models to learn plane features offline and then predict the plane in the captured image in real-time, similar to the AI method used in the optical ranging procedure.

4.6 AI-based Mobile Tag Reader

4.6.1 CycleGAN based Denoising

CycleGAN is a popular deep learning method and is mostly used for image style transforming which can convert images between Style X and Style Y. For example, to generate a monet-style image from a real world picture or vice versa[150]. We adopt a lightweight CycleGAN to convert

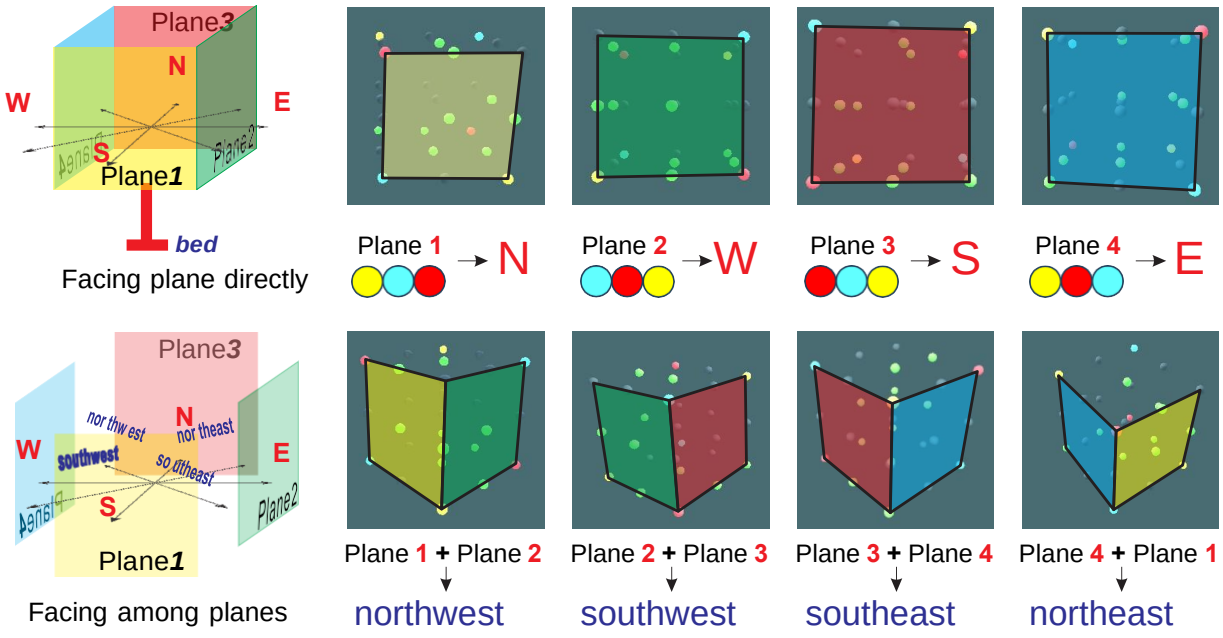


Figure 4.11 The orientation guidance principle illustration.

the real underwater images taken of real, physical UOIDs created for U-Star (Style X) into clear Unity3D-style images created in the Unity3D game engine (Style Y) for further processing. The images in real underwater scenarios have a random and different background (i.e., with noise) for UOID tags. The images in the Unity 3D version have clear and pure backgrounds (i.e., there is no noise from the background in these images). Thus, we can utilize CycleGAN to convert real-world images with noise (Style X) to Unity3D-version images without noise (Style Y) to perform underwater denoising as shown in Figure 4.12.

In our CycleGAN-based denoising, instead of the typical unpaired datasets, we create the partial-paired datasets, the Real UOID tags (60 images) and the Virtual UOID tags (60 images), for each underwater environment setting in the CycleGAN training procedure, as shown in Figure 4.12. Partial-paired means the positioning elements are paired between the real UOID tag images and the Unity3D version images of the training datasets, while the inside data elements are not paired. Partial-paired CycleGAN denoising guarantees mostly correct conversion of both the tag structure, data elements and the color of positioning elements.

To train the CycleGAN efficiently, we use three different types of losses to train our Cycle-GAN.

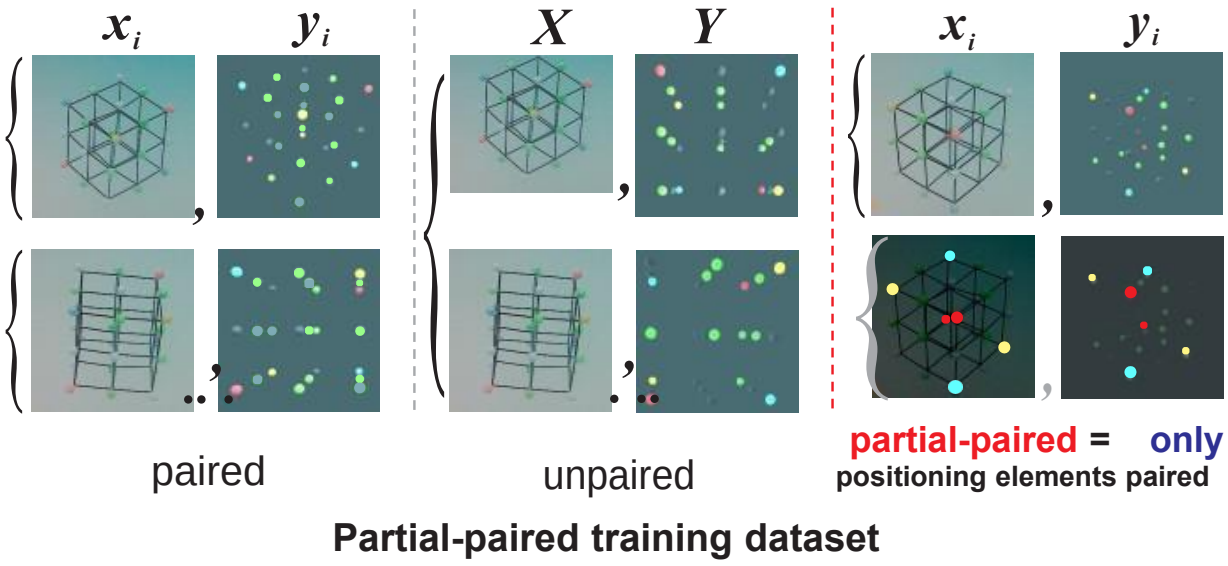


Figure 4.12 CycleGAN based denoising from real underwater tag images to the Unity3D version tag images.

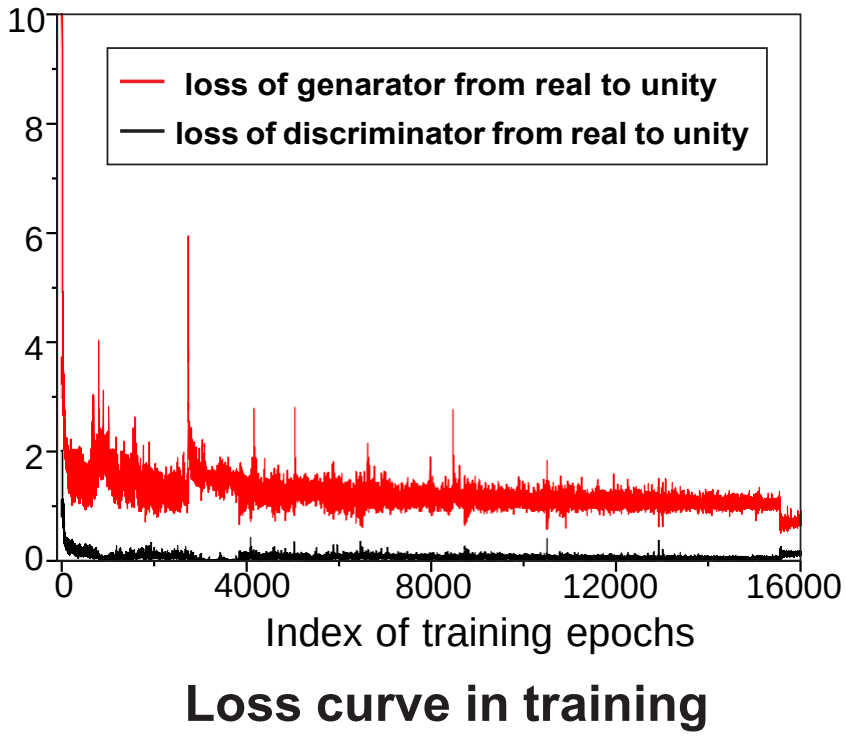


Figure 4.13 CycleGAN based denoising: training loss curves.

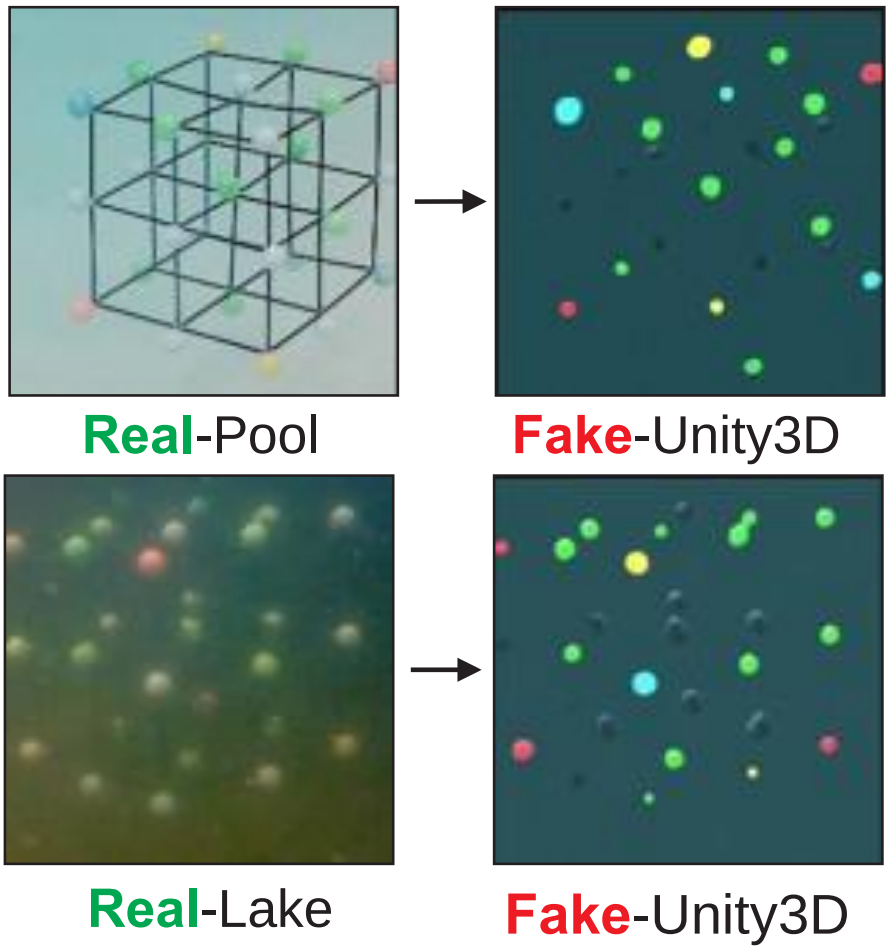


Figure 4.14 CycleGAN based denoising: result samples.

More specifically, we apply an identity loss (L_{id}) for generator network, a GAN loss (L_{GAN}) for the Discriminator, and a cycle loss (L_{cycle}) for the cycle step.

$$L_{CycleGAN} = L_{id} + \lambda_1 L_{GAN} + \lambda_2 L_{cycle} \quad (4.1)$$

Both identity loss and GAN loss are using L1 loss, while the cycle loss is applied by a MSE loss. We summed those three losses together with different prior assigned weights (λ_1 and λ_2) to help the model converge. The value of λ_1 and λ_2 are selected empirically, in our case, we use 10 and 5 for λ_1 and λ_2 respectively. By integrating the three losses together, we feed the pairwise training images to the CycleGAN and train the generators and discriminators. The loss curve in the training of the generator and discriminator (from real images to Unity3D-style images) are shown

in Figure 4.13. The varying trend of the loss curves show the conversion from the real underwater UOID tag images to the Unity3D-style UOID tag images converges successfully.

The examples of original captured underwater images and the denoised images are shown in Figure 4.14. We can see that underwater images from both a pool and lake can be successfully denoised and converted to Unity3D-style images with a mostly correct tag structure, color, and element positioning. The CycleGAN denoising also removes the physical UOID frame components to reduce the LoS blockage. Although there are a few elements with unmatched colors, we can correct them based on the original image easily. The next steps of relative position determination and data parsing can then be based on these converted Unity3D-style UOID tag images to lessen the influence of harsh underwater optical environment.

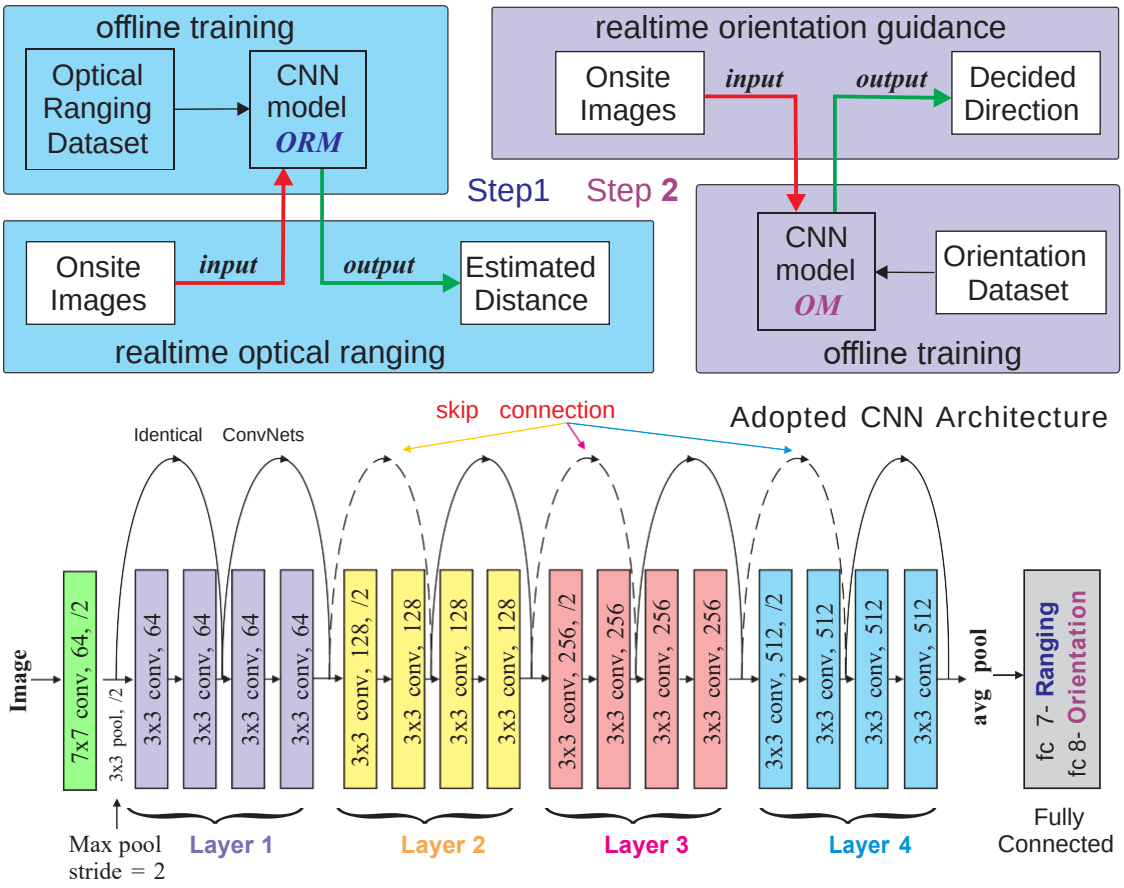


Figure 4.15 CNN based relative positioning of optical ranging, orientation guidance and adopted ResNet-18 network architecture.

4.6.2 CNN based Relative Positioning

We adopt CNN-based deep learning methods to determine the relative position instead of non-deep, traditional computer vision methods to simplify the task and decrease the computation overhead. It is difficult to calculate relative distance directly with different underwater backgrounds, which requires several steps: (1) locate the tag in the image using AI or CV methods, (2) calculate the tag size, and (3) utilize the distance estimation relation to calculate the estimated distance. In comparison, we choose a CNN model because it does not necessitate detecting the tag in the image or calculating the tag size. Instead, we directly output the prediction distance in different underwater environments using the trained CNN model and captured images of UOID tags.

We create two datasets (1) optical ranging dataset (280 images of Unity3D version and 280 images of real underwater), and (2) orientation dataset (320 images of Unity3D version and 320 images of real underwater) for the offline CNN training. The reason using both real underwater tag images and Unity3D version tag images in training is to increase the generality of the prediction model. Then we use CycleGAN denoised tag images for real-time relative position determination. As shown in Figure 4.15, our CNN models, ORM (optical ranging model) and OM (orientation model), adapt the ResNet-18 architecture. ResNet-18 is a neural network architecture that adds a skip connection between disconnected layers, such that the input of deep layers will not only take the output from its preceding layer, but also from its former layers which may contain original data. Such design effectively copes with gradient vanishing problem in DNNs[35], and further increases the depth of network with fewer additional parameters. ResNet has demonstrated superior performance on image classification tasks [17, 18, 1], which is particularly suitable for our goal that distinguishing relative position both optical ranging and orientation guidance. We follow the ResNet-18 design due to its efficiency and high accuracy on image classification tasks. Specifically, we retain all of the convolutional and pooling layers, and modify the output feature of the last fully connected layer to match the number of possible options (i.e., 7 for ORM and 8 for OM).

4.6.3 Data Parsing via Perspective Principle

The data elements in captured images are different when the user is at different relative positions to the UOID tag. To decode the embedded data in the tag, the tag reader needs to know the 3D locations of data elements in a standard coordinate system to then perform decoding.

Restore 3D structure. Based on three pairs of positioning elements, the tag reader can restore the 3D structure of UOID tag based on captured 2D images in six steps shown in Figure 4.16 (a): (1) obtain Unity3D-style UOID image after CycleGAN based denoising, (2) filter out three pairs of positioning elements via computer vision tools, (3) decide which positioning element for each pair is in the front or rear based on element size, (4) find one of the two remaining vertices, (5) find the other remaining vertex, and (6) decide which of remaining vertices is front or rear based on the element size of nearby positioning elements. Finally, we can reconstruct the 3D structure based on the total of 8 vertices of the 3D cube.

For step (4), there are two sub-steps: **(4-1)** Extend line Y1R2 and R1Y2 to find the intersection point IP1(not shown in the figure). Then connect B2 with IP1, which is the cross line of plane Y2R1**B2** and Y1R2**B2**. **(4-2)** Extend line Y1B2 and B1Y2 to find the intersection point IP2. Then connect R2 with IP2, which is the cross line of plane B2Y1**R2** and B1Y2**R2**. Then we can find the first vertex, which is the intersection point of B2IP1 and R2IP2. The sub-steps for step (5) are similar to the sub-steps in step (4).

Data element location restoration. As shown in Figure 4.16 (b) and (c), we can restore the location of data elements by matching the filtered data element and locations of each element calculated based on the positioning elements. If the specific filtered data element is near or at the specific calculated location from the restored 3D structure, it signifies a match. Then we can denote that this location has a data element as bit **1** while other vacant calculated data element locations will be decoded as bit **0**. Then the tag reader decodes the embedded data and generates the bitstream based on the data assignment rule illustrated in Figure 4.7.

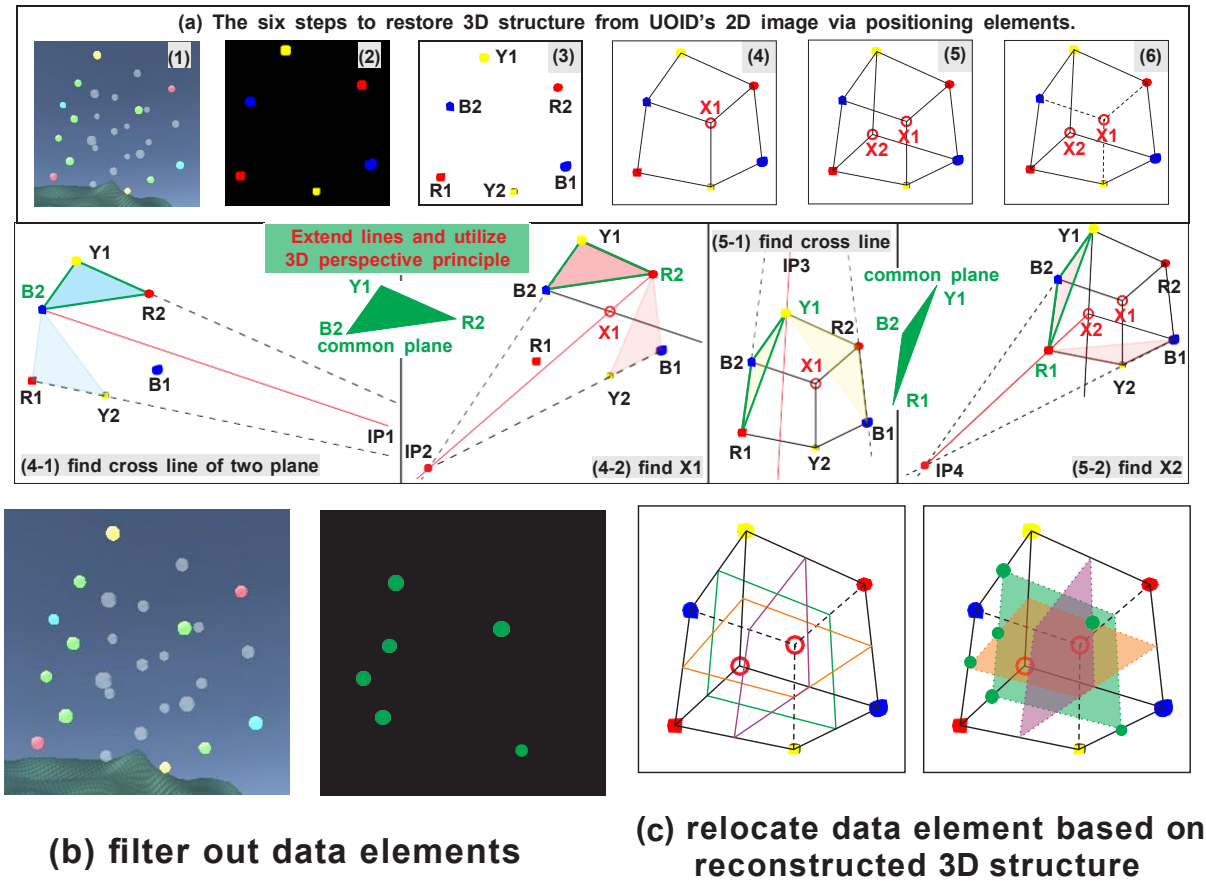


Figure 4.16 The illustration of 3D structure restoring and data parsing is based on the perspective principle.

4.7 Implementation and Evaluation

4.7.1 UOID Tags

We implement two versions of UOID tags. One is a virtual NxNxN UOID tag created in the Unity3D cross-platform game engine to simulate UOID tags of various order and also different permutations of embedded data within tags of the same order. We also implement multiple physical 3x3x3 UOID tags for use underwater.

Virtual UOID tag. The elements in our virtual UOID tags are translucent with fluorescent effects and are assigned with the proper spacing, as shown in Figure 4.10 and Figure 4.16.

Real UOID tag. As shown in Figure 4.17 (a), the UOID tags can be observed well during both the day and night because they absorb natural light and emit light. For the elements of our physical UOID tags we employ soft plastic balls (ϕ = 2cm) glazed with fluorescent powder and attach them

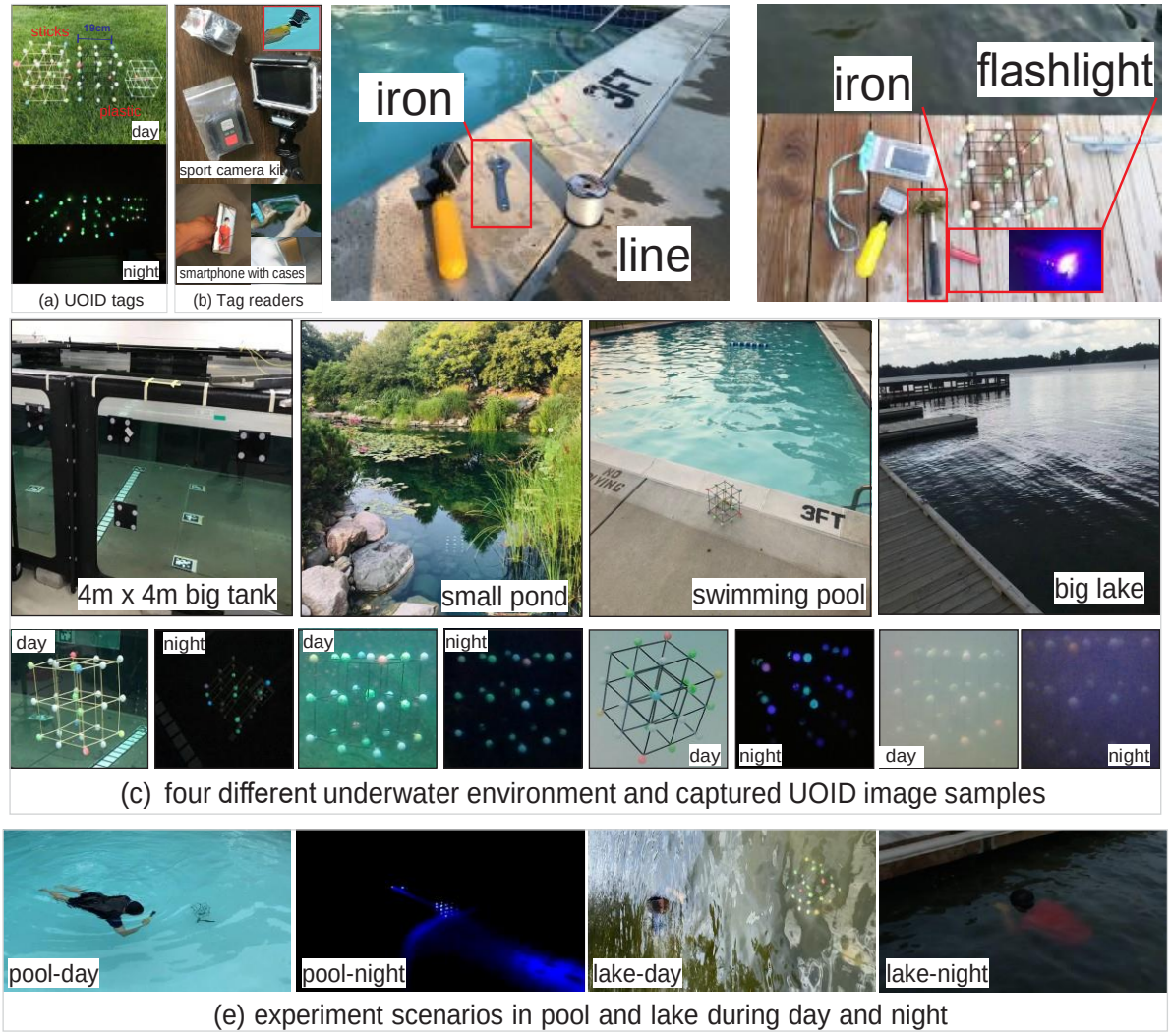


Figure 4.17 U-Star system implementation, setup and experiment scenarios in day and night.

on 3 types of cube structure frames for exploration (sticks, black and transparent plastic). Finally we choose the black plastic frame-based UOID tags (edge: 19cm, weight: 14g) for evaluation.

4.7.2 Tag Reader

There are many commercial smart devices that can be adopted for use in our U-Star system. Some of these include underwater sports cameras and smartphone with transparent, waterproof cases, as shown in Figure 4.17 (b). These commercial camera devices are popular and cheap. In our experiment, we use the Campark sport camera, which costs less than $50 and set it at a fixed focal length.

4.7.3 Setup

Different underwater environment. Figure 4.17 (c) shows four underwater environments (indoor big tank, outdoor small pond, swimming pool, and big lake) and captured images of UOID tags.

Tag fixation and flashlight. We fix the UOID tags at the bottom of a body of water, i.e., a specific UOID plane always faces a specific direction. We use iron and connection pole to sink and fix the UOID tag underwater. During the night, the user can use a flashlight for underwater lighting to activate the UOID tags.

We evaluate three performance aspects of our U-Star system: (1) relative positioning, (2) data parsing, (3) comparison with existing optical tags. In addition, we conduct an underwater navigation case study in a 4m x 10m indoor pool with 4 UOID tags. Finally, we evaluate other aspects such as cost/price, computing overhead, and latency.

4.7.4 Accurate Relative Positioning.

We evaluate the relative positioning performance in three aspects: optical ranging accuracy, orientation guidance accuracy (both at 100^{th} epoch), and their training loss in [5, 200] epochs.

Optical ranging. We have 7 different distance settings: 1m, 2m, 3m, 4m, 5m, 6m, and 7m. As shown in Figure 4.18 (a), due to the considerable tag size difference, the ranging accuracy of 1m and 7m distance settings are 100% for both with and without CycleGAN denoising. After CycleGAN denoising, the ranging accuracy improves significantly and reaches nearly 100% for other distance settings. The results show that the trained CNN model for optical ranging performs well to estimate the distance from the user to the tag with CycleGAN denoising. The results show our current U-Star prototypes can provide up to 7 meters of optical ranging with average accuracy nearly 100%.

Orientation guidance. We provide eight recognized orientations for underwater navigation: North(N), North West(NW), West(W), South West(SW), South(S), South East(SE), East(E), and North East(NE). As shown in Figure 4.18 (b), no matter what was the user is facing (any of the eight recognized orientations) the accuracy of our orientation classification is always 100% when

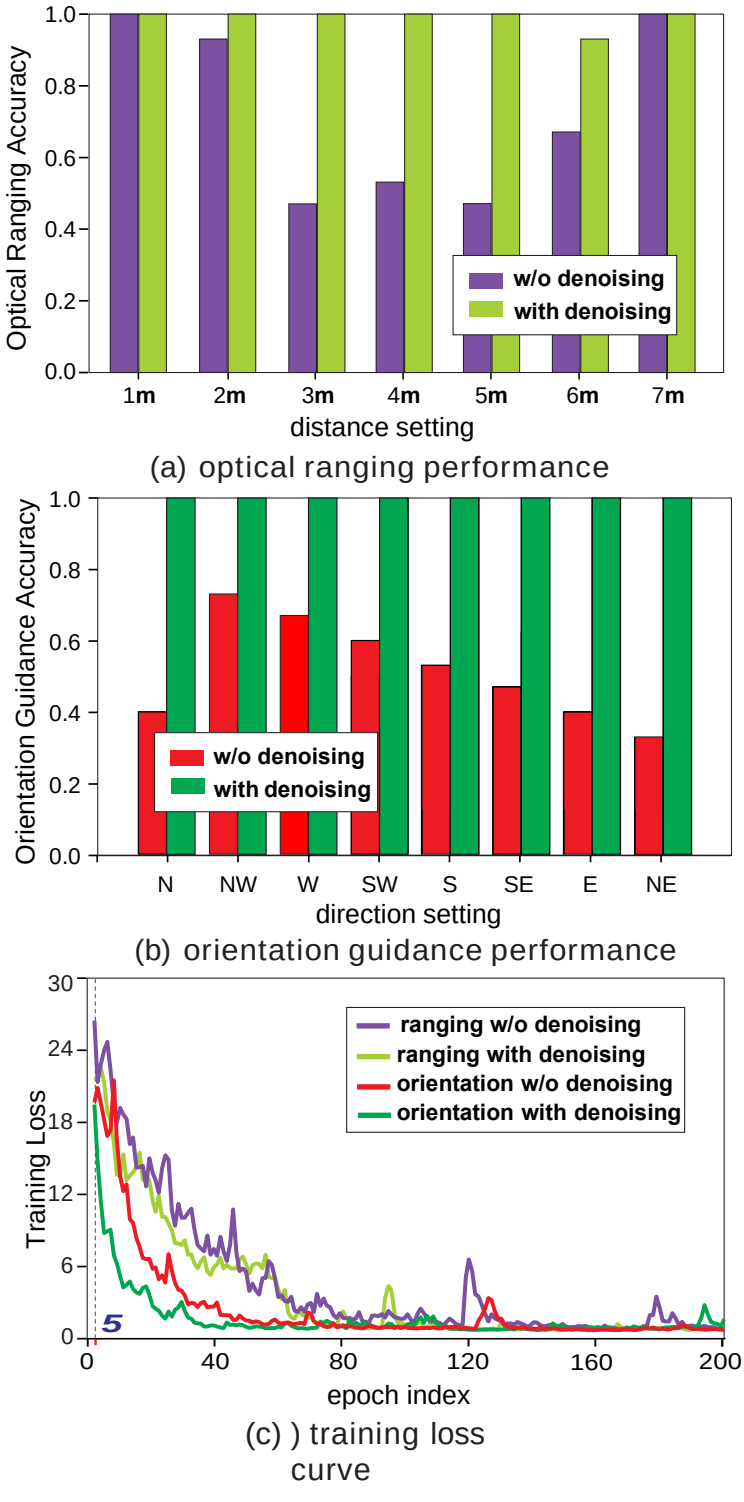


Figure 4.18 Relative positioning performance in aspects of optical ranging, orientation guidance and training loss.

performing orientation guidance with CycleGAN based denoising. We also present orientation guidance performance without CycleGAN based denoising for comparison. The results show that the performance with CycleGAN denoising is better than without CycleGAN denoising. This shows that the CycleGAN based denoising helps the CNN model to improve the orientation guidance performance by decreasing the impact of harsh water conditions. The results show that our U-Star system can provide accurate orientation guidance amongst all eight orientations.

Training loss. For relative positioning, we also measure the loss in CNN based training for optical ranging and orientation guidance separately. As shown in Figure 4.18 (c), the optical ranging training loss curves both with/without denoising are above the orientation training loss curves during the training process. This means that features (tag size) in the optical ranging dataset are not as rich as the features (positioning elements and their various permutations) in the orientation dataset. The curves with CycleGAN denoising are beneath those without CycleGAN denoising during the entire training process no matter the optical ranging training or orientation training. That means that using the CycleGAN denoising can help decrease training loss more quickly and limit the impact of harsh underwater optical conditions for relative positioning.

4.7.5 Robust Data Parsing

We use our tag reader to capture images of four real UOID tags A1, A2, B1, B2 with random capturing poses in different distances, water conditions, and time of day to evaluate the decoding performance of U-Star. A1 and B1 embed raw bits without error correction codes. A2 has 3, 5, and 3 common data bits with A1 in layers 1, 2, and 3 respectively. A2 also has 3, 4, and 3 Hamming ECC parity bits in layers 1, 2, and 3. B2 has 3, 5, and 3 common data bits with B1 in layers 1, 2, and 3 respectively, and also has 3, 4, and 3 Hamming ECC parity bits in layers 1, 2, and 3. Hamming ECC[33] can correct 1 error bit per bitstream, thus, for a total of 3 error bits correction capability for a tag. The bits in A1, A2, B1, B2 are shown in Table 4.1.

We define the BER as the average bit error ratio of the entire embedded valid data bits in two UOID tags with different data embedding (i.e., two tags: A1 and B1 or two tags: A2 and B2). Each BER value is calculated using 30 captured images and we use it as a metric to evaluate the decoding

Tag	bits in 1st layer	bits in 2nd layer	bits in 3rd layer
A1	101101	110010001	001011
A2	101101	111110011	010101
B1	001101	100111010	101001
B2	010101	101100111	101101

common data bits of A1 & A2 or B1 & B2 data bits without ECC Hamming ECC parity bit Valid data bit

Table 4.1 Embedded bits in 4 UOID tags: A1, A2, B1 and B2.

performance of our U-Star system. Besides the difference between UOID tags with and without Hamming ECC codes[110], we also compare BER performance with and without CycleGAN based denoising as comparison.

In different communication distances. We adjust the distance of the tag to the tag reader to be 1m, 1.5m, 2m, 2.5m, and 3m in clean water (pool) during the daytime. As shown in Figure 4.19 (a), the BER remains low, consistently less than 0.09 after CycleGAN denoising in all distance settings. We found that the best data parsing distance for current U-Star prototypes is 1m, as the BER is 0. The BER performance without CycleGAN denoising is significantly worse than with CycleGAN denoising at 3m. This confirms that the CycleGAN denoising works well, especially at longer distances. Both with and without CycleGAN denoising, the BER with ECCs is lower than for without ECCs. The BER is 0.003 at 1m and continues to be less than 0.05 up to 3m with Hamming ECC and CycleGAN denoising simultaneously.

In different water conditions. We explore four water conditions during the day in experiments: indoor tank with clean water, small pond, swimming pool, and big lake, as shown in Figure 4.17 (c). We conduct experiments at a distance of 1m (the best capturing distance for data parsing of the current U-Star prototype mentioned above). As shown in Figure 4.19 (b), without CycleGAN denoising, our data parsing performs best in the pool and worst in the pond. This is because the pool is clean enough for data parsing without the denoising process and the small pond makes the color of the elements change too much. After CycleGAN based denoising, the BER decreased significantly in all four water conditions. The Hamming ECC codes decreased the BER even further, resulting in a BER lower than 0.07 for all four water conditions. Notably, the tank, pool,

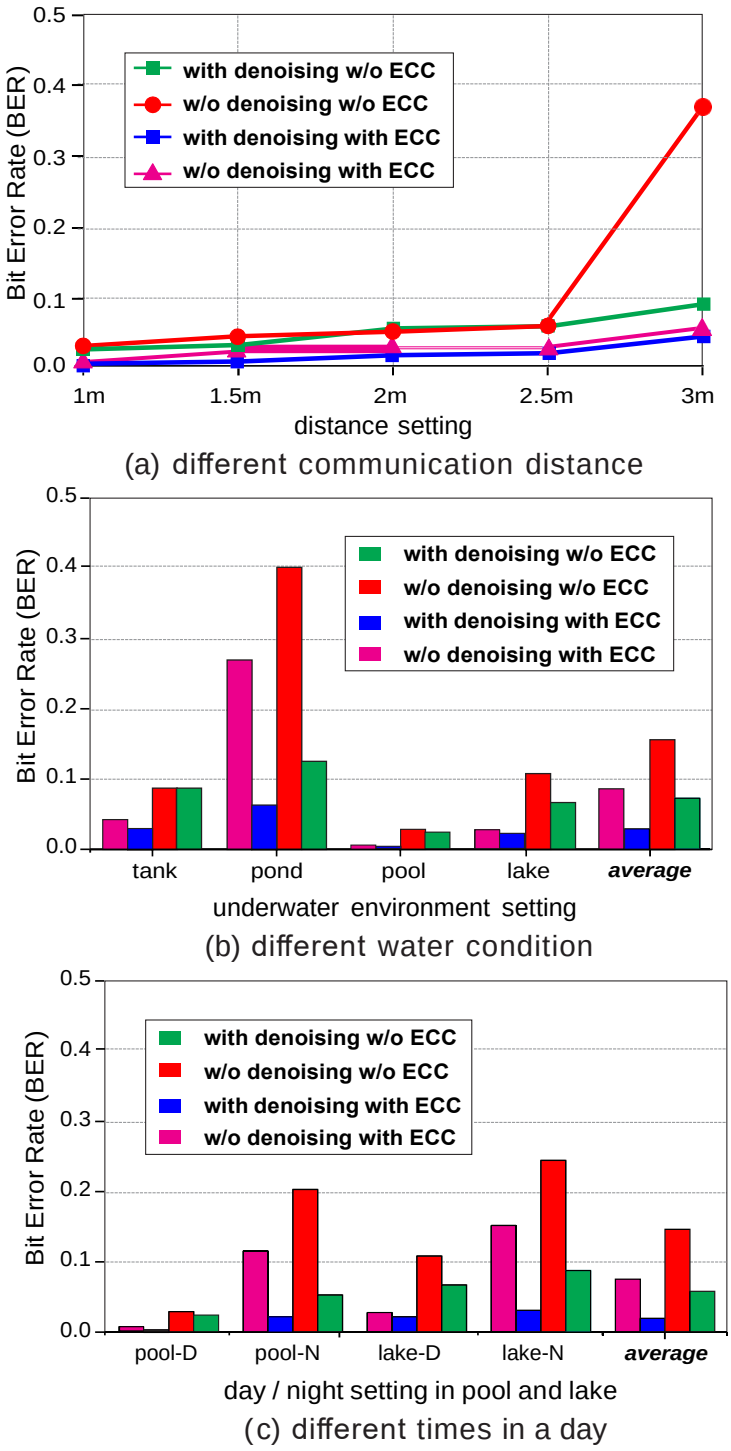


Figure 4.19 Decoding performance with different communication distance, water condition, and time (day/night).

and lake situations show a BER approaching 0. The average BER decreases from 0.16 to 0.03 after CycleGAN denoising and Hamming error correction. In summary, the BER in four different water conditions is all low enough with CycleGAN based denoising and Hamming error correction for robust data parsing.

In different times of the day. We conduct experiments during both day and night at a distance of 1m in the swimming pool and lake. As shown in Figure 4.19 (c), the BER in the daytime is lower than in the night for both the pool and lake. Even with a flashlight shining to activate the UOID tag, the current UOID tag only has luminous powder covering the element surface, which is not as bright as in the day time. Moreover, at night, the BER without denoising in the lake is worse than the clean pool, because the emitted light from the UOID tag is too weak to go through more muddy water in the lake. After CycleGAN based denoising and Hamming error correction, the BER in all four settings decreased significantly and is lower than 0.03. The results show that the current U-Star system performs data parsing well with CycleGAN based denoising and Hamming error correction both day and night.

4.7.6 Comparison with Existing Optical Tags

We implement the 3D version of existing Bar/QR codes with the same 21 embedded data bits (101101 110010001 001011) and the same tag size (cube edge: 19cm) as our UOID tag for a fair comparison across various aspects. The data alignment, implemented tags and the comparison experiment scenarios are shown in Figure 4.20 (a). We conduct experiments and make comparisons in the five aspects below to demonstrate the superiority and necessity of our designed UOID tags over existing optical tags for underwater navigation.

(1) Same tag order with more embedded bits. Despite the fact that the user can capture the information of one and up to three surface planes of a 3D version of existing Bar/QR codes in N-order, the decoded bits are the same as the bits in one plane. The embedded bits in an N-order barcode are roughly N. The embedded bits in an NxN QR code are roughly N^2-4 bits. The embedded bits in an NxNxN UOID tag are N^3-6 bits. As shown in Figure 4.20 (b), the amount of embedded bits in a UOID tag increases exponentially compared to the same order 1D/2D optical

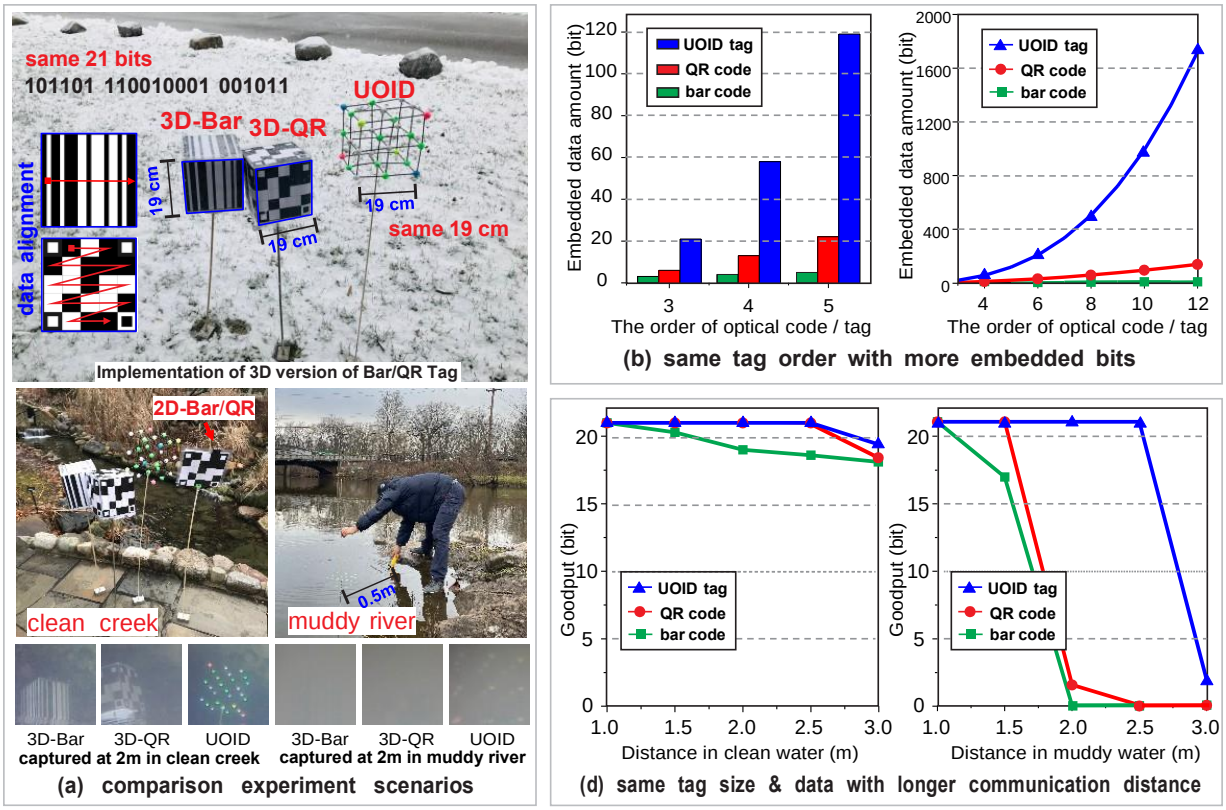


Figure 4.20 Comparison between UOID tags with existing optical tags. (a) Experiment scenarios, (b) Data improvement, (d) Better goodput performance.

tags. Even their 3D versions cannot compare to the UOID tags (e.g., 3-order UOID embeds 7x and 4.2x bits of the same order Bar and QR code).

(2) Same tag size & data with larger element distance. The larger the average element distance and the broader the distribution of element distances, the better the detection performance and the less the error bits. We measure distances between all 21 data elements in 3D versions of the Bar/QR, and UOID tag that have the same embedded bits and tag size (edge is 19cm). As shown in Figure 4.20 (a) and Figure 4.21 (c), data element distances in Bar and QR codes are all smaller than 20cm, however the data element distances in UOID tags are completely distributed in a greater range of [5, 30] cm.

(3) Same tag size & data with longer communication range. We also investigate the goodput performance of three tags mentioned above in two different underwater scenarios: clean creek and muddy river at varying distances from 1m to 3m. In clean creek, all three tags perform well and

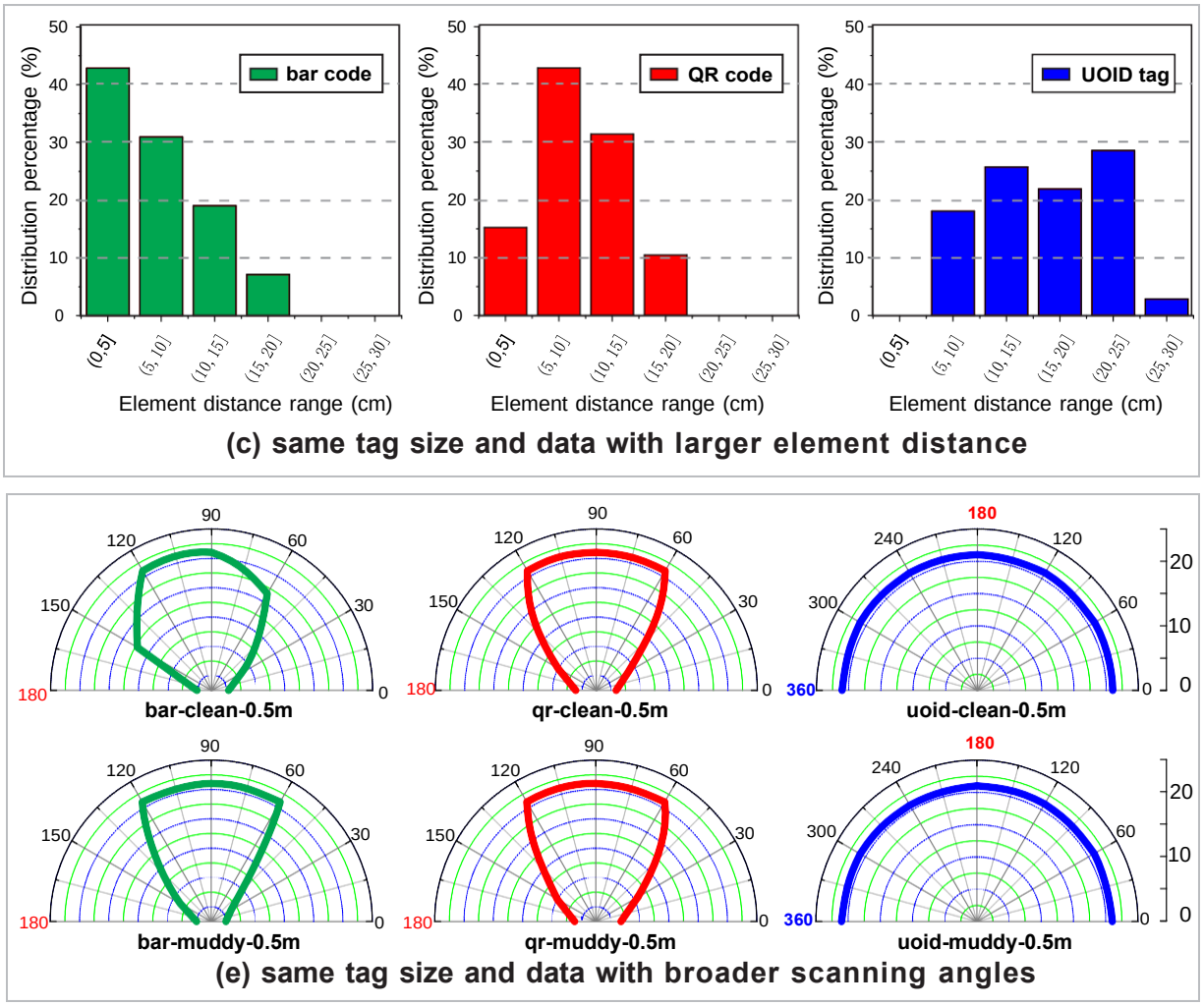


Figure 4.21 Comparison between UOID tags with existing optical tags. (c) Broader element distance, and (e) Full-directional scanning.

produce more than 17 bits of goodput at up to 3m, as illustrated in Figure 4.20 (a) and (d). However, in the muddy river, the goodput of Bar and QR codes in the 3D version drops dramatically after 1.5m, whereas the UOID tag maintains its high goodput until 2.5m.

(4) Same tag size & data with broader scanning angles. Furthermore, for all three of the aforementioned tags, we evaluate the goodput performance with varying scanning angles at 0.5m under the clean creek and muddy river. As shown in Figure 4.20 (a) and Figure 4.21 (e), the usage view range of the existing optical tags has also been increased from less than 120° to 360° of UOID tags for both clean creek and muddy river.

(5) Other benefits of UOID design. When compared to the 2D plane (the version of 1D Bar

and 2D QR codes in our daily life, shown in the left middle of Figure4.20 (a)) and confined 3D cube (3D version of Bar/QR) to maintain tag's location and orientation in flowing water or current (i.e., creek, river, tide), the hollowed-out UOID can lessen influence of water current to allow it to flow through the tags and maintain stabilization.

4.7.7 Case Study with Multiple UOID tags

The usage of our U-Star signage system is similar to barcode/ QR code adopted in auto-supermarket systems. The data embedded in codes are the query codes used for searching a backup database with records for all offered goods. Due to the large enough storage ability on the mobile device, the ability to embed more query codes will result in better navigation. Our 3-order UOID tag can embed $2^{3x3x3-6}$ = 2,097,152 possible query codes. Even with Hamming ECC parity bits that sacrifice 10 (3+4+3=10) bits, there are still 11 data bits available for embedding 2^{11} = 2,048 query codes. As shown in Figure 4.22, we implement four UOID tags with Hamming error correction codes in the case study, and their 11 valid data bits match to distinct query codes in range of [0, 2047] in the backup database. The database stores the current absolute location information, the guidance information, and risk warnings such as "shark near" which can queried via the related query codes. Our demo in a 4m x 10m indoor pool, the user dives at the start site of B and plan to go to the destination site of C and then back.

When the user scans Tag B at the start location, the user will be given the current absolute location (i.e., facing North and at (2m, 0.5m) in the coordinate system) as well as information about its nearby nodes (i.e., D is the nearest tag with 4.5m relative distance to B's NorthEast direction, A is 5.3m away from B to B's EastSouth direction, and C is 9m away from B to B's East direction) to help navigate himself to other spots.

The user intends to visit Tag D first. He looks for a bright dot around 4.5m away (the optical ranging of UOID provides him a sense of underwater distance) at the NorthEast direction of Tag B. If he cannot find his way, he will travel to another nearby node such as Tag A.

After confirmation of D's existence, he moves to Tag D and repeats the similar procedure to go to Tag A first (compared with 8.2m to C, the distance to A is 5m and A is the nearest uncovered

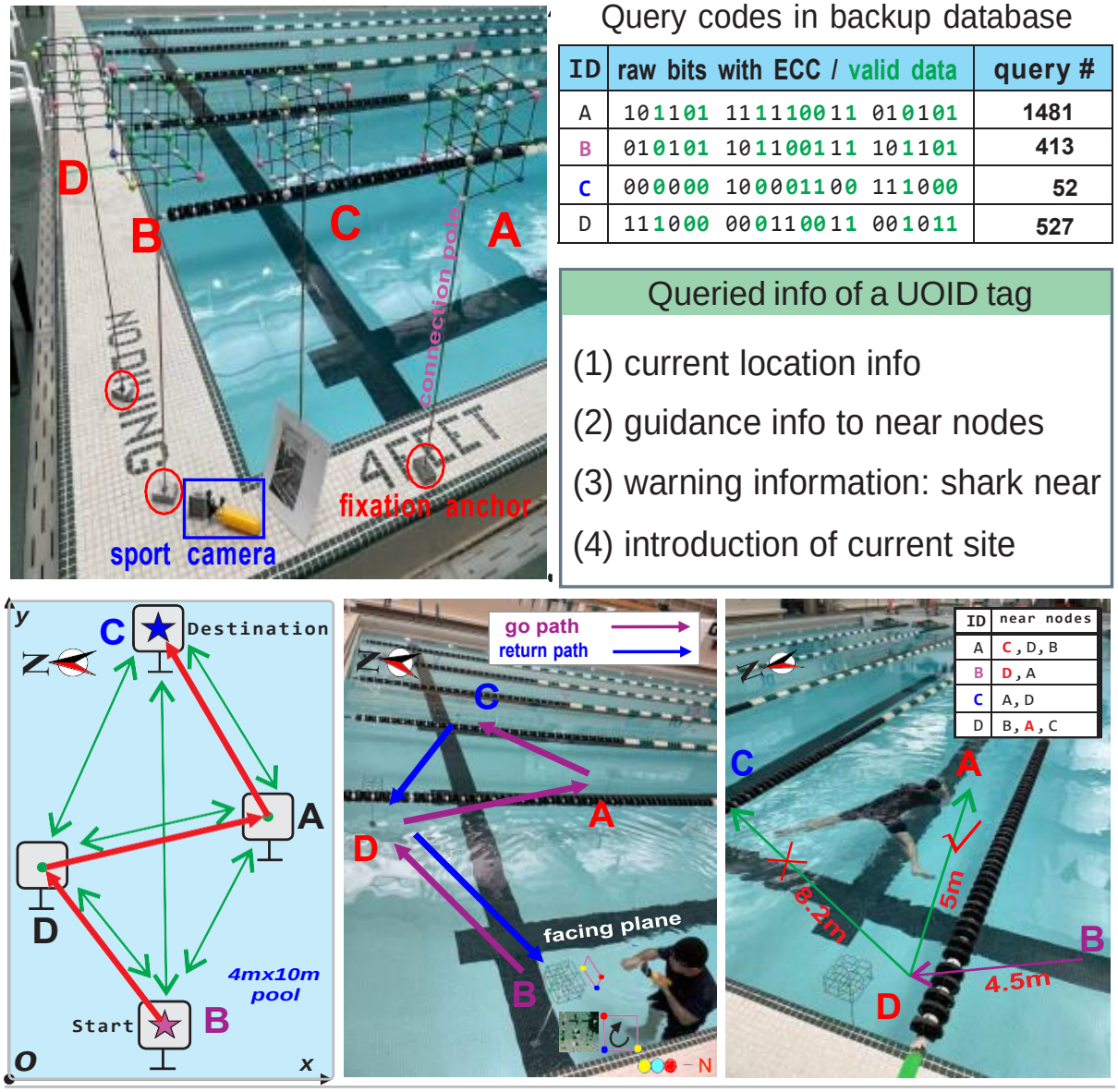


Figure 4.22 Underwater navigation case study of U-Star in a 4mx10m indoor pool with 4 UOID tags and backup database.

node to D). And next, from A, he finally reaches at destination C.

His path (locally optimal) is B-D-A-C and return path is C-D-B (effective path) while globally optimal path C-B may not work due to he may not confirm B's existence from C. By following the procedures above, he achieves self-guided underwater navigation easily and effectively, regardless of the start and destination tags.

4.7.8 Other Concerns

Cost and price. As shown in Figure 4.23, the main cost of the U-Star system is the tag reader, while the UOID tag is very cheap (less than $3 for each). For practicality, the tag reader can be replaced with the user's own smartphones covered with a waterproof case, which is less than $4. Considering multiple UOID tags deployed underwater, the U-Star system with 20 UOID tags costs less than $100 for an underwater site with an area of $1km^2$ ($7m$ x $7m$ x 20).

Device	Material	Cost ($)
One 3x3x3 UOID tag	element balls	< 1
	stick / plastic	< 0.5
	hot melt glue	< 0.5
	double-side tap	< 0.5
	luminous powder	< 0.5
	Total for a tag	≈ 3
One tag reader	sport camera	30 (basic)
	smart phone	self-contained
	waterproof case	3.5 (Amazon)
U-Star with 20 tags		< 100

Figure 4.23 Cost & price.

Computation overhead. For underwater situations, battery is limited and not easy to replace. The tag reader should not conduct complex computations that consume energy too fast. The training processes are offline, the real-time tasks are denoising, optical ranging, orientation guidance, and decoding. As shown in Figure 4.24, the denoising requires the most memory resources and decoding required the fewest memory resources. For all four tasks, they require a combined 430 MiB of memory and is not a computational burden for a commercial smart device.

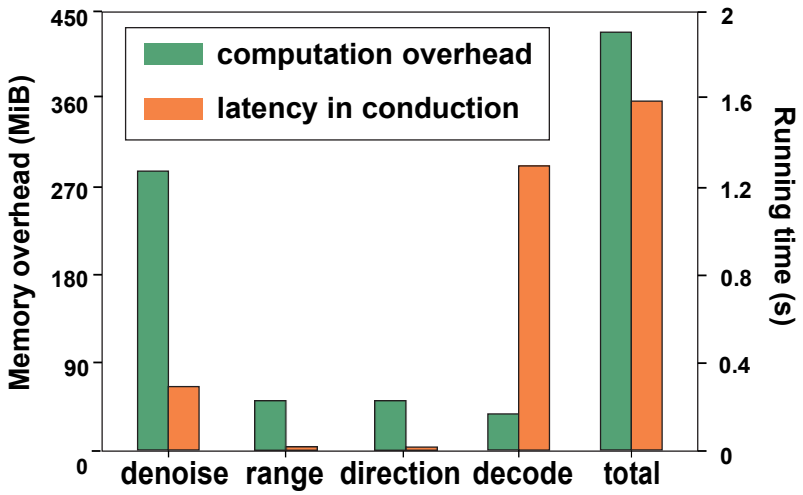


Figure 4.24 Overhead.

Latency. For underwater navigation tasks, time can be important to improve the user experience and even save people's lives. Compared with state-of-art underwater navigation systems, including audio-based systems, U-Star has nearly no time delay in signal propagation due to the fast propagation of light. Thus we only consider the computational latency. As shown in Figure 4.24, optical ranging and orientation guidance have the lowest running time of 0.002 s, while decoding has the longest running time at 1.25 s. All four tasks consume 1.59 s total, which is still quick enough for a good user experience.

4.8 Discussion and Summary

Usage instruction of scanning UOID. Even with appropriate spacing between data elements in UOID tags, there is some LoS blockage at certain scanning angles. However, by slightly adjusting capturing poses without moving the user's location, it is simple to avoid blockages and capture all data elements.

The number of guidance directions. Our current U-Star prototype can provide user orientation guidance in 8 directions, which is sufficient for practical underwater navigation. U-Star, however, may be updated to finer-grained orientation guidance using a same CNN training with more directions (e.g., 16 directions).

UOID deployment. Because GPS is unusable for underwater scenarios, the positions of

deployed UOID tags are identified and saved in a backup database on shore at a one-time deployment cost. We can use spring installation techniques to fix UOID tags on the underwater floor with little regard for location and orientation fluctuation caused by tide and flow. They can make the tag flexible when subjected to tide power and automatically resume its suspected position when it becomes static, much like how tall building dampers maintain stability and extend tag usage lifetime.

System robustness and potential side-effect on marine animals. (1) moss/scum removing: Because moss grows slowly, we can periodically (e.g., every month) remove the accumulated moss and maintain UOID tags as part of underwater infrastructure maintenance. We can utilize an ultrasonic technique to remove moss touchlessly while causing no harm to the UOID tags or other marine life. (2) luminous powder: To prevent pollution and harm to marine life, we apply non-toxic, non-radioactive, and long-lifespan (more than 15 years) luminous powder wrapping with waterproof glues. (3) marine debris: We can use integrated molding technology and 3D printing techniques in the future to produce recycled, solid and not easily damaged UOID tags to avoid marine debris.

Applications benefited by U-Star. (1) Recreation scuba diving. (2) Underwater rescuing. In addition to using fixed UOID tags as infrastructure for safe underwater activities, we can attach smaller size UOID tags (which store people's identifying information) on top of underwater helmets as mobile UOID tags for persons participating in underwater activities. As a result, rescuers can scan UOID tags to identify people and learn about on-site situation (how many people and who are in danger or need rescue). The trapped people, on the other hand, can scan larger UOID tags on rescuers to actively seek help and instructions from rescuers. (3) Future directions combined with Augmented Reality. We can update the tag reader side from current sport camera/smart phone to AR goggles to show guidance info in more direct and visual manner instead of small display on smartphone for user experience of WYSIWYG, "see UOID, see INFO".

In summary, we implement the U-Star system for simple and robust underwater navigation. We investigate 3D spatial diversity for data embedding with wider element distances and additionally use it for relative positioning. We address challenges in system design and implementation, e.g,

combating harsh underwater environments and 3D structure restoration for data parsing. Finally, we conduct experiments based on virtual and real UOID tags in multiple underwater scenarios. Our 3-order UIOD prototype can embed 21 bits and achieves a BER of 0.003 at 1m and less than 0.05 at up to 3 m with approaching 100% relative positioning precision.

CHAPTER 5

HAND POSE RECONSTRUCTION VIA 3D SPATIAL DIVERSITIES

Smart homes, medical devices, education systems, and other emerging cyber-physical systems offer exciting opportunities for sensing-based user interfaces, especially those utilizing fingers and hand gestures as system input. However, existing vision-based approaches, which rely on time-consuming image processing, often adopt a low 60 Hz location sampling rate (frame rate) for real-time hand gesture recognition. Additionally, they may not perform well in low-light environments or have limited detection range.

To address these challenges, we propose RoFin, a novel system that leverages the 3D spatial-temporal diversities of optical signals for fine-grained finger tracking and hand pose reconstruction. RoFin stands out as a low-cost and privacy-protected solution, enabling real-time 3D hand pose reconstruction with fine-grained finger tracking capabilities. It works effectively in various distance ranges and under diverse ambient light conditions, providing a more versatile and robust approach to hand gesture recognition and tracking.

Figure 5.1 RoFin can better record jitter of writing[68].

5.1 Motivation

Some researchers attach on-body sensors (e.g., accelerators, gyroscopes.) to each finger and joint to measure the spatial position variation of fingers. Other studies utilize wireless signals such as radio frequency signals, acoustic signals, and light signals (e.g., soli[61], FingerIO[75],

and Ali[59]) for hand-free gesture recognition. However, these methods require the expensive or specific devices and have limited sensing distance less than 0.5m.

Vision-based hand gesture identification approaches are widely popular, using similar processing techniques as human eyes to detect hand morphology with a perception frequency of about 60Hz. The accuracy of vision-based hand gesture recognition exceeds 80% with the aid of deep learning [137]. However, these vision-based methods have several drawbacks: (1) They are not effective in low-light conditions or for long detection ranges due to the limited amount of light reflected from the hand to the camera's image sensor. (2) The low sampling rate (e.g., 60 Hz) of cameras when tracking fingers is similar to the limited perception ability of human eyes, making it challenging to capture the detailed motion trajectory of trembling hands, as observed in patients with Parkinson's disease. (3) Vision-based approaches involve high processing costs and latency, mainly due to the need for recognizing hand morphology with about 20 hand joints. (4) The captured frames of the scenes with hands raise privacy concerns, particularly in sensitive circumstances.

Commercial cameras and LEDs are deployed everywhere, enabling optical camera communication (OCC) a reality in our daily lives. The rolling shutter in commercial cameras exposes one row of pixels and generates a whole image row by row. A clear strip effect appears when the switching speed of the light wave from the transmitter is equal to or slightly less than the rolling shutter speed. Many researchers have tried to improve data rates by collecting data in rolling strips rather than the entire image frame. However, these systems[124, 122, 125, 147, 148] only exploit rolling shutter for communication instead of sensing such as inside-frame fine-grained location tracking with high sampling rate (rolling shutter speed, e.g, 5 KHz) instead of one sample (1Hz).

To overcome these limitations, our proposed system, **RoFin**, leverages 3D spatial-temporal diversities of optical signals to offer fine-grained finger tracking and hand pose reconstruction. By doing so, RoFin addresses the drawbacks associated with traditional vision-based hand gesture recognition approaches and provides a low-cost, real-time, and privacy-protected solution. RoFin consists of wearable gloves and a commercial camera, as shown in Figure 5.2. Each glove finger and the wrist is attached to one low-power LED node controlled by Arduino Nano (<$10).

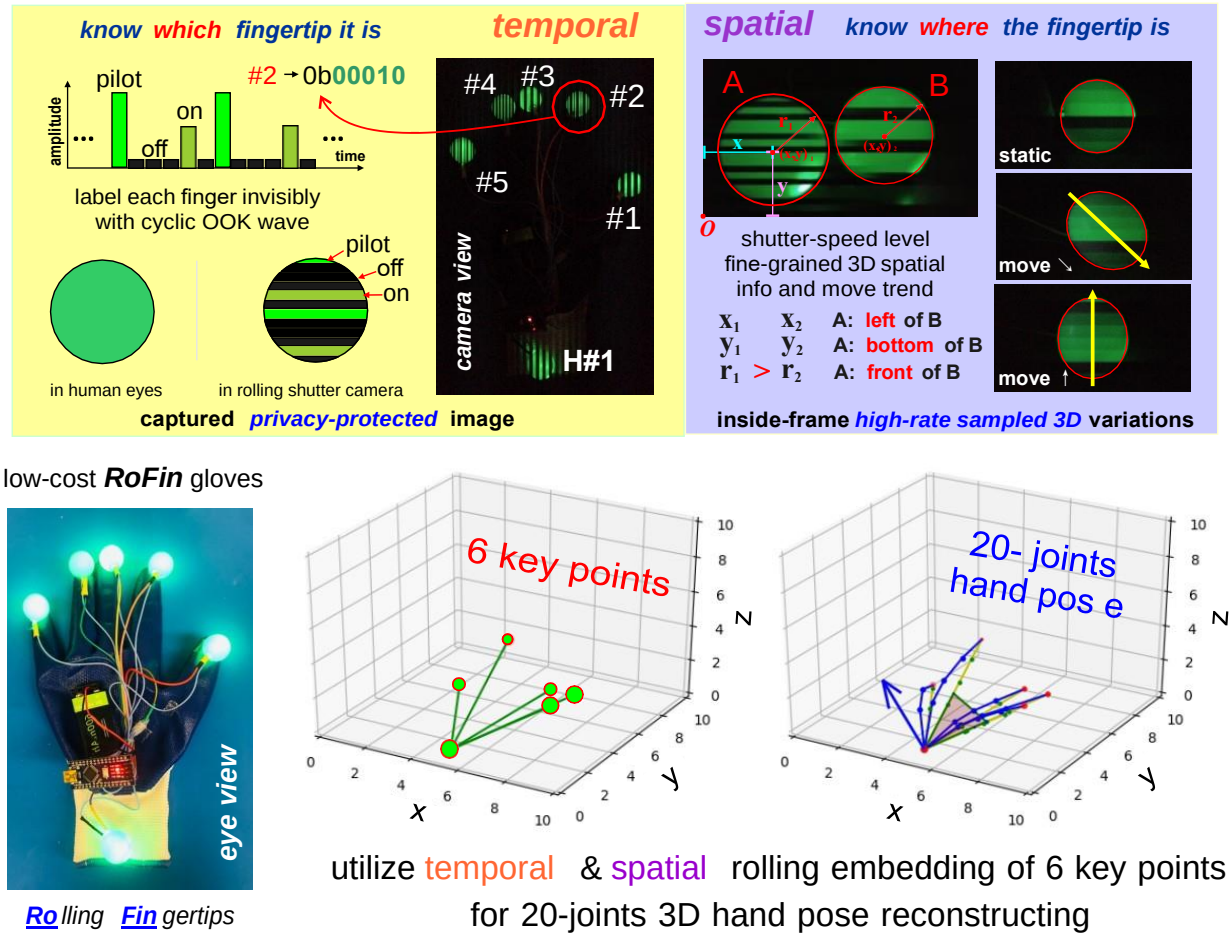


Figure 5.2 3D hand pose reconstructing via 6 temporal-spatial 2D rolling patterns.

5.2 Background and Related Work

5.2.1 Vision-based 3D Hand Pose Recognition

Numerous works adopt cameras to recognize hand poses. In general, these computer vision approaches can be classified into 2 categories. (1) Hand image searching in pre-computed databases with machine-learning assistance. These methods capture hand images and then query pre-computed 3D hand models to determine the best-matched hand pose[107, 8, 56]. (2) Calculate 3D coordinates of hand joints directly and then identify the hand pose by optimizing an objective function. These methods represent the hand with a 3D hand model and adopt an optimization strategy to speed up hand pose prediction[97, 137, 86] However, these existing vision-based hand pose recognition methods are based on complete hand morphology such as hand silhouettes and

numerous joints (e.g., 20 joints) with non-trivial tracking and computation overhead. Furthermore, vision-based approaches sample the location variation at the frame update level while the frame rate is set $\leq$ 60 fps instead of higher to be compatible with time-consuming image processing.

In contrast, RoFin takes a different approach, enabling 3D hand pose reconstruction using only six 2D rolling spots: the five fingertips and the wrist point of the hand. By relying on fewer tracking points and employing a lightweight pose reconstruction algorithm called HPR, RoFin[143, 142] achieves real-time hand pose reconstruction with an average time cost of 13.8 ms. Additionally, even with a limited 60 fps frame rate, RoFin can sample numerous inside-frame points instead of only one, as in vision-based approaches. This enhanced sampling granularity greatly improves the accuracy and precision of finger tracking.

5.2.2 Strip Effect in Rolling Shutter Camera

Cameras commonly found in our everyday smart devices utilize a low-cost technique called **rolling shutter** to reduce the readout time of pixels from the entire image frame. In a rolling shutter camera, the exposure occurs one row of pixels at a time, generating the complete image row by row. However, this rolling shutter mechanism can cause a noticeable **strip effect** when the switching speed of the light wave from the transmitter matches or slightly exceeds the rolling shutter speed. This strip effect allows for the sequential capture of optical signals containing transmitted data in a symbol period, enabling optical camera communication (OCC) techniques such as CASK, ColorBar, and others[124, 122, 148]. These OCC techniques leverage the rolling shutter phenomenon to facilitate communication by capturing and interpreting the transmitted optical signals in a series of rolling strips.

The high-rate sampling ability of a rolling shutter camera is not fully utilized in current vision-based finger tracking and hand pose recognition approaches. These methods typically only sample one location of a specific objective (e.g., a fingertip) in a frame, despite the rolling shutter camera's capability to capture numerous location samples during a frame period.

In contrast, RoFin maximizes the potential of these numerous location samples by employing active LED spheres attached to the fingertips. By tracking the location variation of the center of

each LED sphere in 3D space during one frame period, RoFin can achieve fine-grained inside-frame finger tracking granularity. This is particularly useful in scenarios involving high-motion status (e.g., shaking), as RoFin can generate deformed ellipses to record the finger's movement accurately. This capability enhances user experiences in activities such as virtual painting and writing.

Moreover, RoFin's fine-grain tracked virtual writing traces of Parkinson patients enable more precise trace optimization compared to vision-based methods, which rely on coarse-sampled traces. This capability allows RoFin to provide a more accurate and valuable tool for assisting patients with Parkinson's disease in their writing and other motor activities.

5.3 Our Approach: RoFin

RoFin **first** exploits **2D temporal-spatial rolling** fingertips for ***(1)*** *active optical labeling for fingers/hands*, ***(2)*** *fine-grained inside-frame finger tracking with rolling shutter speed*, and ***(3)*** *real-time 3D hand pose reconstructing*. Each LED node covered with same-size sphere emits distinct light waves as optical label, which is invisible to human eyes but perceptive by rolling shutter cameras for robust finger identification. Based on the captured spots (deformed ellipses) via rolling shutter at high sampling rate (e.g., 5 KHz), RoFin can parse fine-grained 3D locations and inside-frame variations of fingertips (left/right, up/down, and front/rear). Finally, RoFin reconstructs 3D hand pose consisting of 20 points by tracking only 6 key points (5 fingertips and 1 wrist point) for less latency and computation overhead.

Composition. RoFin system consists of two parts. **(1) RoFin gloves** are commercial insulating gloves where each fingertip and the wrist are attached with a low-power LED component covered with a plastic ball. These LED components are controlled by an Arduino Nano to generate identical LED waves to indicate different fingertips. **(2) RoFin reader** is based on commercial cameras (e.g., smartphones, web cameras). These cameras use adjustable focal length lenses and rolling shutters with configurable shutter rates.

Workflow. (i) The user puts on RoFin gloves and makes some hand poses. **(ii)** After setting the rolling shutter rate and focal length, RoFin reader captures the continuous 2D rolling spots of six key points (5 fingertips and 1 wrist point) frame by frame. **(iii)** RoFin reader identifies

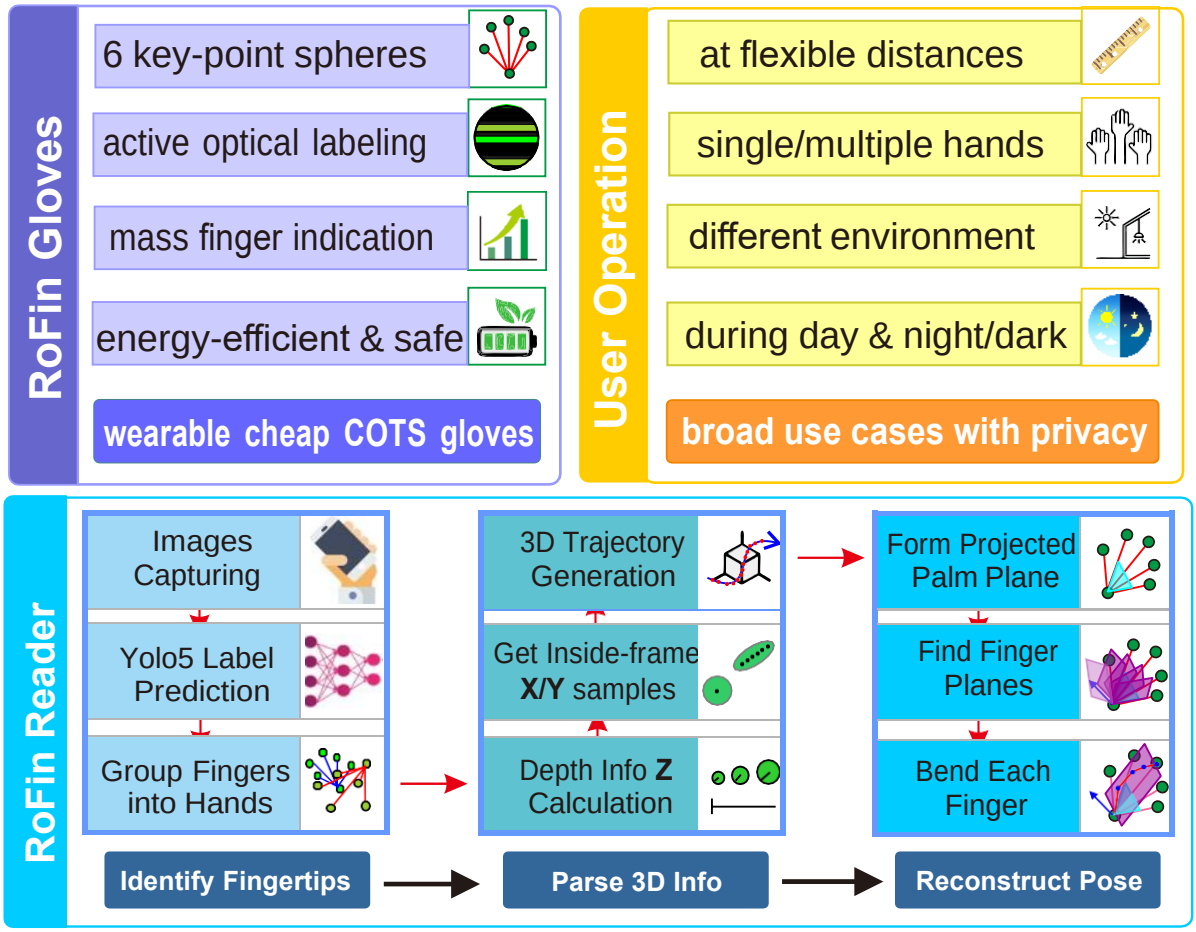


Figure 5.3 RoFin system overview: composition, workflow and three main tasks.

each fingertip/wrist point via lightweight CNN model with bounding boxes (i.e, YOLOv5). **(iv)** RoFin parses the 3D location variations of each key point based on captured deformed ellipses in each frame with the granularity of strip width. **(v)** Finally, RoFin reconstruct 3D hand pose via lightweight HPR algorithm based on the parsed label and its fine-grained 3D location.

3 Main Tasks. At the high level, RoFin responds to two questions: **(1) identify which fingertip** it is, and **(2) locate position and its inside-frame variation** of this fingertip with sampling rate at rolling shutter speed. RoFin further **(3) reconstructs 3D hand pose** via HPR algorithm based on outputs from (1) and (2).

5.3.1 Challenges and Solutions

However, we must address three significant technical challenges in developing RoFin:

C1: Each finger from multiple hands must have a distinct and robustly identifiable label, even in varied ambient light and at long distances.

C2: Deciphering the fine-grained 3D fluctuation of fingertips based on the 2D shape (i.e., a distorted ellipse) recorded during a frame period poses a considerable challenge.

C3: RoFin relies on tracking only six key points of a hand to reduce overhead. However, accurately reconstructing a 20-point 3D hand pose from these limited 6 key points in real-time presents a significant challenge.

Our **contributions** can be summarized as follows:

(1) RoFin is the first work to exploit rolling shutter effect for 3D hand pose reconstructing. We indicate each fingertip and wrist point with asynchronous cyclic optical labels. Then we adopt a lightweight CNN model with bounding boxes to identify fingertips and wrist points. Our active optical labeling overcomes the limitations of the vision-based technique and is appropriate for the identification of multiple hands in low-light and long-range detection scenarios.

(2) We creatively utilize inside-frame high sampling via rolling shutter to track several fingertips' 3D location variation instead of only one 2D location sample in a frame to enhance tracking granularity further while vision-based approaches only use one 2D location sample during one frame period. The improved finger tracking ability has potentials for the virtual writing for Parkinson's suffers, better user experience for virtual writing/painting in AR/VR/MR.

(3) Based on the finger identification and parsed 3D location info of 6 key points (5 fingertips and 1 wrist point) from (1) and (2), we design a real-time and lightweight 20-point 3D hand pose reconstructing algorithm HPR from tracked 6 key points. HPR can efficiently reconstruct a 3D hand pose by direct calculation instead of redundancy points' tracking while not sacrificing the reconstructing accuracy.

(4) We implement RoFin with commercial devices and evaluate its performance of (i) finger identification performance in different settings, (ii) inside-frame tracking enhancement in comparison to the vision-based approach, and (iii) hand pose reconstructing error with Leap Motion as the benchmark and its reconstruction latency. We also discuss the potential use cases of RoFin such

as multi-user interaction for meta, virtual writing or health monitoring for Parkinson suffers, hand pose commands for smart home.

5.4 Active Optical Labeling

5.4.1 Temporal Rolling Patterns

The light source emits optical signals which varied with time sequences at rolling shutter speed level during one frame period can be recorded row by row in the captured image frame by the rolling shutter camera. Only when the rolling shutter rate is similar to the transmission frequency, however, can we clearly see the distinct rolling strips, as illustrated in Figure 5.4.

We can utilize captured rolling spots with distinct strip textures as active optical labels to **indicate fingertips**. However, optical signals have multiple light features varied with temporal sequences such as amplitude, color, frequency. Which ought to be used in rolling patterns for RoFin? We explored and the captured images are shown in Figure 5.4.

- **Amplitude.** We can adjust brightness of the light source with time sequences[20, 152, 118]. The light amplitude fluctuation is vividly captured sequentially.
- **Transmission Frequency.** We may also alter the ON/OFF switching speed of the light wave.
- **Color Spectrum.** We could transmit the light with different wavelengths. The captured rolling strips are colourful and vary in the same way of color fluctuation with time sequences as the light source does.

Choice. It requires RGB LED and complicated modulation to achieve color spectrum diversity. Complex modulation and a longer time period to present complete frequency variation (i.e, only partial of the complete pattern could be presented on the captured spot of sphere with limited width) are both necessary for transmission frequency diversity. To indicate multiple fingertips, **amplitude** variation is more suitable compared with different colors or transmission frequency which require more complex devices (i.e., multi-color LED, high-clock-rate MCU) and control overhead. Thus, we apply single-color LEDs with Pulse Width Modulation (PWM).

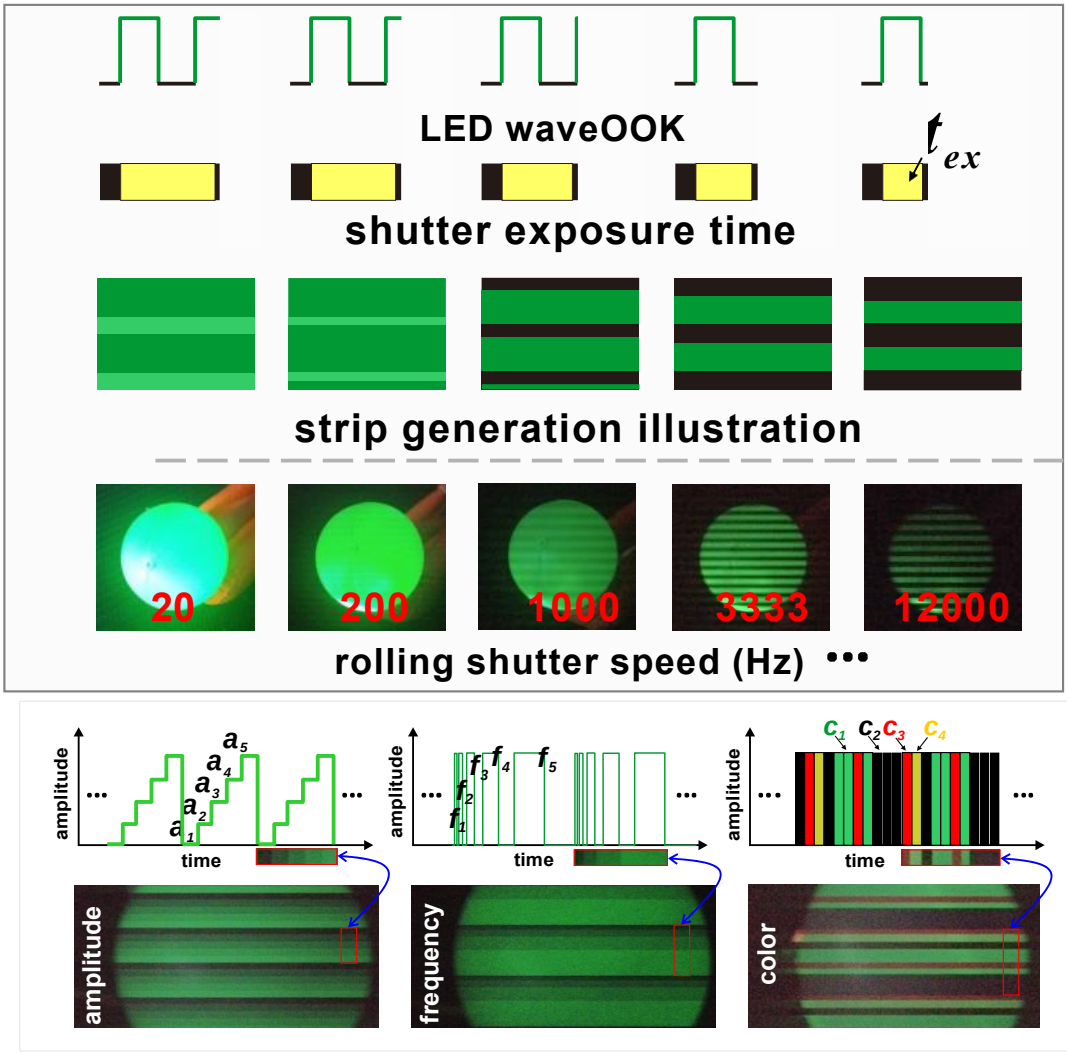


Figure 5.4 Captured strips impacted by shutter speed. Light feature selection for temporal rolling patterns.

5.4.2 Fingertip and Hand Indication

Each attached LED element can emit the different amplitude waves as the active optical labels. However, we can not synchronously control each light source to let them start temporal rolling pattern at the same time. Additionally, because of their various positions inside the field of view (FOV) of the camera, the recorded rolling strip may begin at a different time. These asynchronous problems make it difficult for the RoFin reader (i.e., camera) to recognize the embedded identification information from different light sources (i.e., LEDs). Thus, we design asynchronous Cyclic-Pilot On-Off-Keying (CP-OOK) labeling scheme for different fingertips from multiple hands

and wrist-assisted hand indication.

· **CP-OOK based Fingertip Indication**

The optical label consists of two parts: **(1) CP (cyclic pilot)**, takes one symbol period at the beginning, and **(2) indication sequences**, formed via 5 (can be extended) OOK (On-Off Keying) symbols, as shown in Figure 5.5 (a). Aside from the Off symbol (dark), the optical label design has two amplitude levels: the CP symbol has the highest brightness, whereas the On symbol has the lower brightness of CP.

Instead of the normal very long preamble[2], we designed a short pilot (i.e., CP). Because the number of rolling strips revealed in the finger pattern (the circle or ellipse) is restricted, we must ensure that at least one complete optical label is shown in each rolling pattern for further decoding. Furthermore, to improve the robustness of these optical labels in variable environment, we set a total of 2 non-dark amplitudes (Am_{CP}, and Am_{On}) instead of additional amplitude levels (e.g., 5 amplitude levels in amplitude shit keying).

We encode the index of each finger with its binary number into OOK symbols, as shown in Figure 5.5 (a). When the finger index is 11, for example, the binary number is *01011* and the indication sequence is [*Off, On, Off, On, On*]. The length of the indication sequence is determined by the number of fingers being tracked. 3 OOK symbols can represent up to 8 fingers, enough for 1 hand. 4 OOK symbols can represent 16 fingers, enough for 3 hands. In general, N OOK symbols can represent 2^N fingers that are appropriate for $2^N/5$ hands. The transmission frequency of light waves have the same or slightly slower frequency than the rolling shutter, and thus these optical labels are clearly recorded for further finger identification, as shown in Figure 5.5 (b).

· **Wrist-assisted Hand Indication**

We assign each finger from multiple hands of multiple users with a finger index as illustrated in Figure 5.5 (c). For example, there are users A, B, and so on. We assign the A's right hand as the hand #1, A's left hand as hand #2. And we assign the B's right hand as hand #3, and the rest can be done in the same manner. We evaluate three hands (A's right hand and left hand, B's right hand). We assign these fingers with indication index from 1 to 15 finger by finger as shown in Figure 5.5

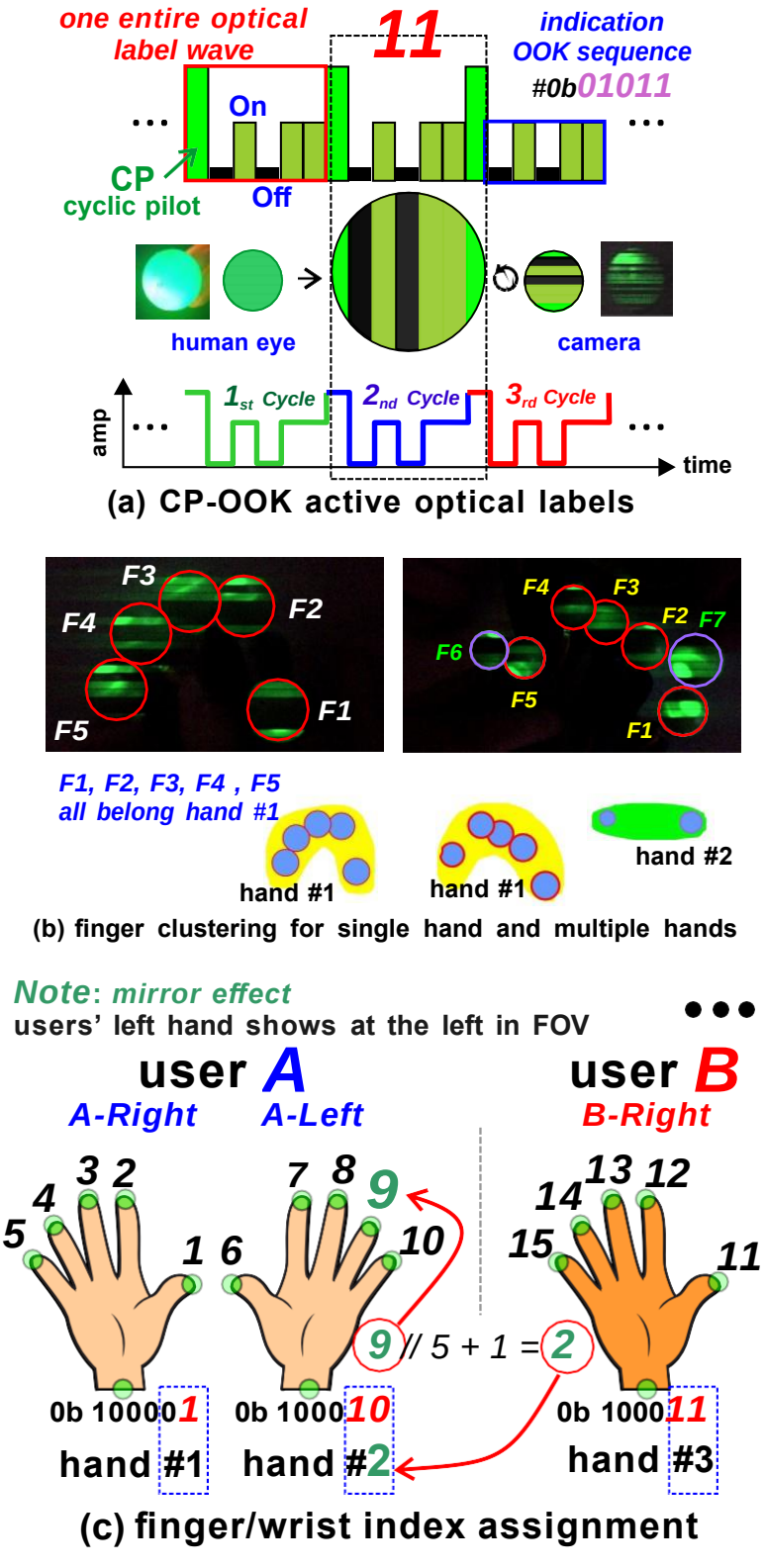


Figure 5.5 The scheme design of CP-OOK fingertip indication and wrist point labeling.

(c).

However, only 5 fingertips are not enough to determine a hand in a 3D space. Besides the five fingertips in a hand, we also attach one LED node covered with same-size sphere at the end of the wrist. This additional wrist point has more vital meaning for hand pose reconstructing in comparison to any of the five fingertips. Furthermore, different hands should have distinct indications for these 6 key points of each hand (5 fingertips an 1 wrist point) to differentiate hands and correctly reconstruct each hand pose when they are shown in the camera view at the same time.

Based on the analysis above, the indication of the wrist should have more significant indication than the fingertips but not introduce additional non-trivial overhead (e.g, use different light features: colored-LED, different modulation schemes: FSK). To achieve this design goal, we use the same CP-OOK modulation technique in fingertip indication, but set the **leftmost** indication bit as **1** while the remaining bit sequence as the wrist indication for differentiation.

Given three hands #1, #2, and #3, it requires 4-bit indication sequence to denote 15 fingers. Thus, originally, the binary number for finger #11 is 1011. But we set its indication sequence as 01011, which is [*Off, On, Off, On, On*], to make it compatible for wrist point indication. For the wrist point from #2, its binary number is 10. Following the rule above, the indication sequence of this wrist point is set as 10010, which is [*On, Off, Off,* ***On****,* ***Off***].

5.5 3D Spatial Parsing

Although vision based approach can use higher frame rate (e.g., 120 fps, 240 fps) for sampling, the image processing is still time-consuming. Thus, vision based approaches can not achieve the faster hand pose reconstructing as the faster frame rate and normally set the frame rate at about 60 fps for real-time user experience. Different with vision based approach which only use one 2D location sample $(x,\ y)$ in each frame, RoFin tracks numerous 3D location samples (the inside-frame trajectory of the sphere's center, the deformed ellipse) with high sampling rate (i.e., rolling shutter speed). Thus RoFin has more sensitive perception ability of rapid or subtle motion changes (e.g., writing jitters from Parkinson suffers) than vision approaches with the same frame rate. However, it is challenge to parse the 3D coordinates (x, y, and z) via the deformed ellipses.

5.5.1 Depth Info Estimation: Z

Perspective Principle. We keep the LED light source fixed in the FOV, but as we move the light source closer or farther to the camera, the size of the captured spot grows and shrinks separately due to perspective principle. Based on the size of the captured spot, we could calculate the depth info (Z, i.e., the front and back), as shown in Figure 5.6 (a).

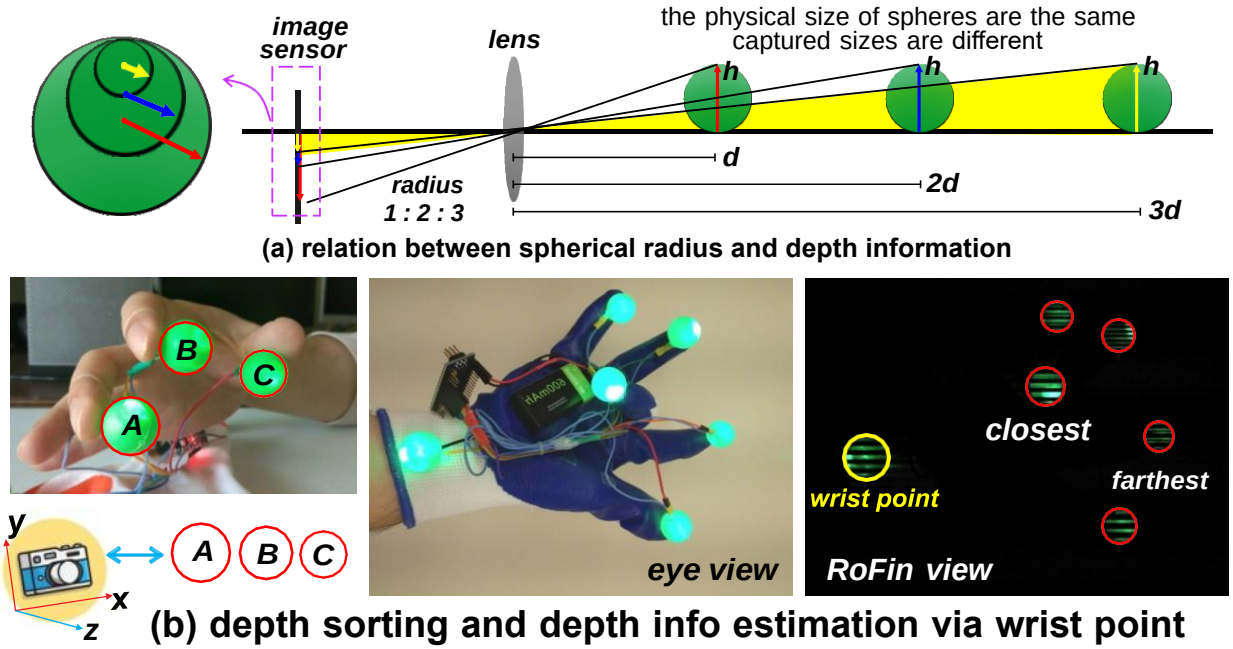


Figure 5.6 Absolute depth calculation via perspective principle by using wrist point as the reference.

Absolute Depth Calculation. The wrist point is designed not only for assistance for hand indication, its captured diameter ϕ_w (unit: pixel) can also be used to calculate the absolute distance of key points d to the camera. As shown in Figure 5.6 (b), the distances to the camera has the relations: $\frac{1m}{d} = \frac{\phi_w}{\phi_{1m}}$. Thus, the absolute distance d from the wrist point to the camera can be formulated as $d = \frac{\phi_{1m}}{\phi_w}$. To do so, we measure and store the captured spot diameter of wrist point at 1m as reference for depth info estimation of all six key points using the same manner.

Coordination Transformation. We set the center of wrist point is the origin of 3D coordinate system. As shown in the right of Figure 5.2, the z value of the five fingertips is set as their physically relative depth distance value to the wrist point. The center (x, y) of each fingertip's spot shown in

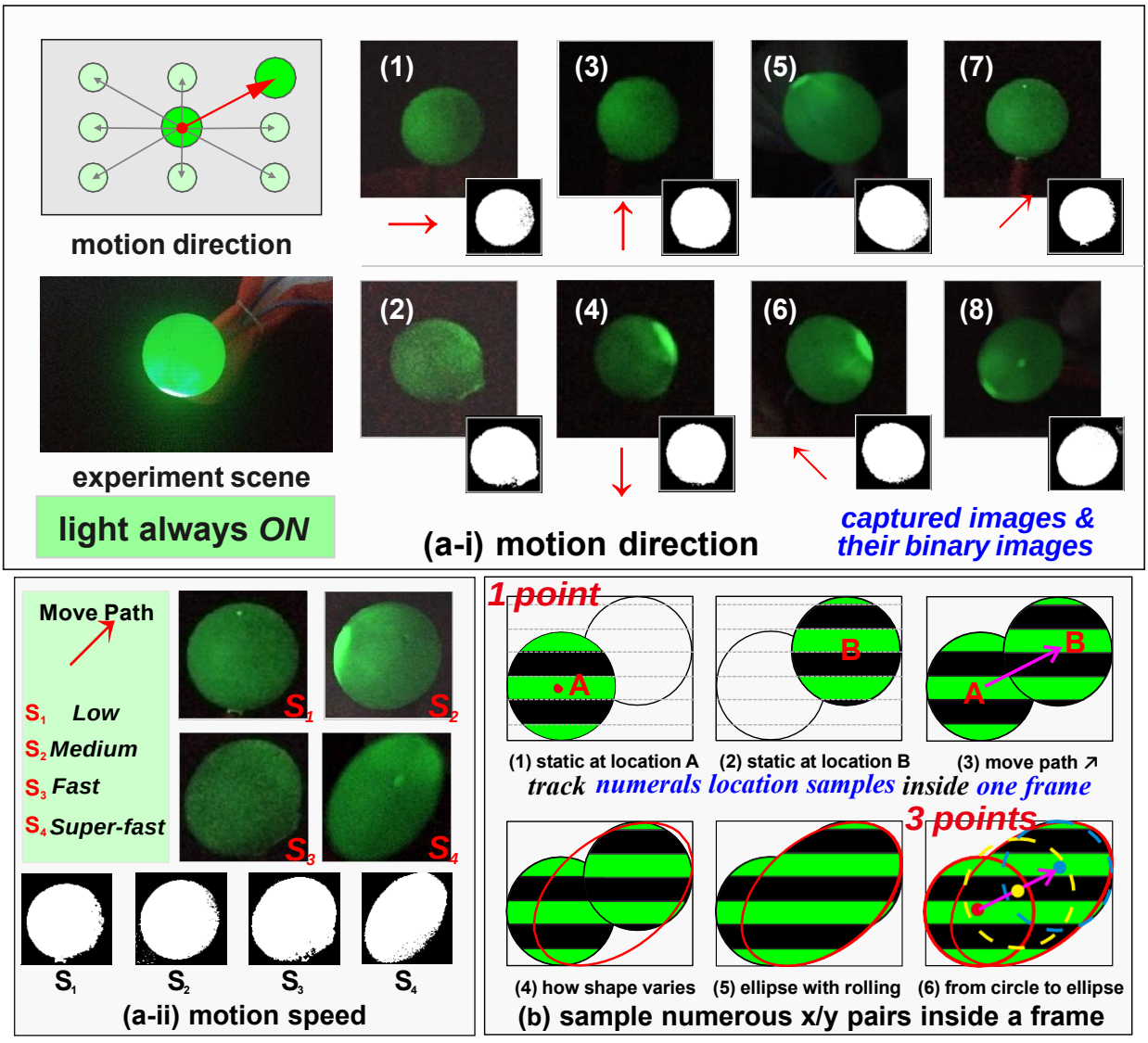


Figure 5.7 (a) Impacts of the shape variation of deformed ellipse: (i) motion direction and (ii) motion speed. (b) The sphere center's location variation is recorded in the deformed ellipse with the granularity of strip width.

the image plane is the pixel value which we need to convert to the physical distance as well. We also use the pixel value range of the wrist point's diameter which maps to the 19 mm of the plastic sphere as the reference to convert the relative X/Y value of each fingertip's center into their relative physical distance to the wrist point separately.

5.5.2 Inside-frame Fine-grained X/Y Tracking

Why high-rate inside-frame sampling? The objectives are mostly in mobile with random trajectory in real-life situations (e.g., vehicles, drones, or fingers). For example, it is required for numerous location samples in unit time to recover the real trajectory of fingertip as brush in virtual writing/painting. Either a long, random curve that is drawn quickly or a small curve requires more samples to capture more details. However, existing vision-based approaches sample the location variation at the level of frame update. Besides, the frame rate is set to about 60 fps instead of higher frame rate considering the time-consuming image processing. To break this gap, we creatively propose to utilize rolling shutter effect for numerous inside-frame location samples.

Impact of Motion Direction. We move the light source with different directions while keep the light source with the fixed distance to the camera plane and the motion speed. For example: (1) and (2) from left to right (→) and reversed (←); (3) and (4) from bottom to top (↑) and reversed (↓); (5) and (6) from upleft to bottomright (↘) and reversed (↖); and (7) and (8) from bottomleft to upright (↗) and reversed (↙). As shown in Figure 5.7 (a-i), the captured spot shape changes to an ellipse rather than the previous circle and its long axis can reflect the moving direction of the light source.

Impact of Motion Speed. We set 4 levels of motion speed of the light source (i.e, low, medium, fast, and super-fast) with the same motion direction (↗) and fixed distance to the camera plane. As the motion speed increases, so does the length of the ellipse's long axis, as shown in Figure 5.7 (a-ii).

Numerous Inside-frame X/Y Location Samples. The captured circle or ellipse's pixel index range in columns and rows reflects the horizontal and vertical location information independently. The circle shape means the fingertip/wrist point is not moving or moving slow in the image plane

during the entire frame period, and its center location (x, y) can be treated as its location in horizontal and vertical directions. The deformed ellipse records the detailed inside-frame motion with the sample rate at rolling shutter speed, as illustrated in Figure 5.2.

5.5.3 Finger's Tracking among Frames.

As shown in Figure 5.7 (a-i), the opposite moving direction of the light source has the same rolling pattern shape (i.e, the ellipse with similar long axis direction) in the single frame. For example, there are 3 frames in Figure 5.8 (a): frame1, frame2, and frame3. In frame2, the light source may move with possible trends as (↗) or (↙) and thus we can not determine fingertip's motion with separate frame.

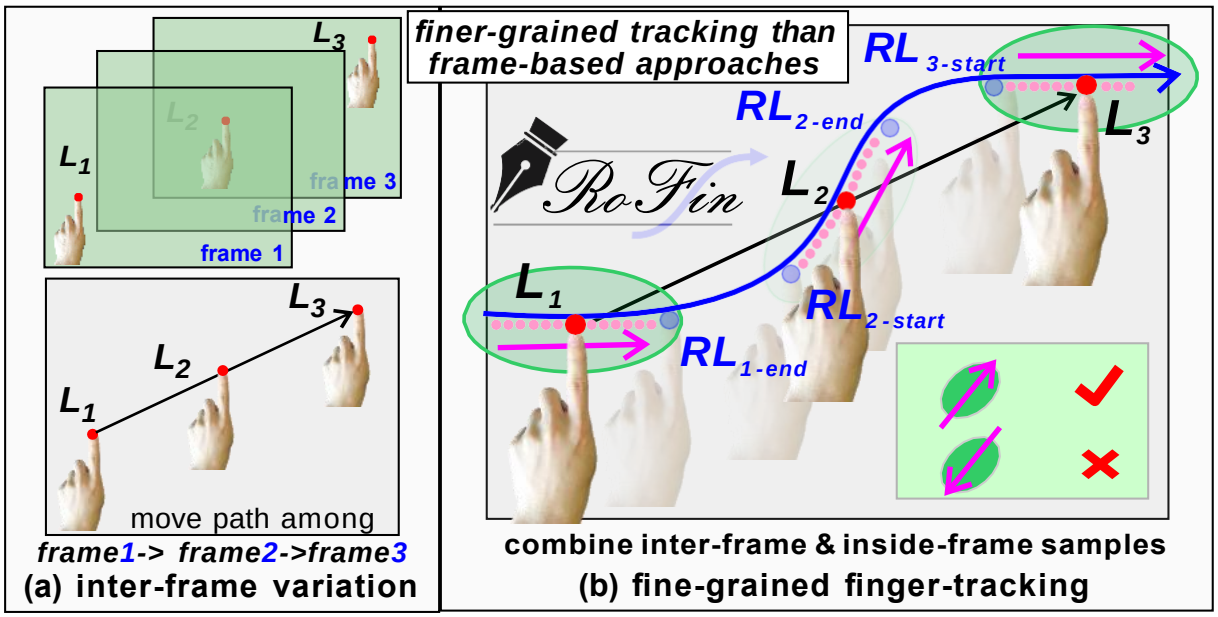


Figure 5.8 RoFin's Finger Tracking among frames combined with numerous inside-frame samples.

Moving Trend Determination. However, if we combine the inside-frame moving direction candidates with two continuous frames, we can know the finger's moving trend. Because these frames are continuously generated with time sequences, the end position of finger pattern in previous frame will be close to the start position of finger pattern as shown in Figure 5.8 (b). Thus, we can determine the finger's moving trend by finding the closest positions of finger pattern in two continuous frames. In this example, the position point RL_{1-end} and $RL_{2-start}$ are the closest position

points between two continuous frames frame1 and frame2.

Moving Trajectory Generation. More importantly, the moving trend determination method is a one-time initialization phase that only requires one frame duration to determine the end positions of each finger pattern and record them as the start positions for the next frame. In this example, using the finger pattern position in frame3, we can know the point $RL_{3-start}$ is the start point. Then we can track finger locations by combining these numerous inside-frame samples and updating them frame by frame, as illustrated in Figure 5.8 (b). Finally, we can generate a finer-grained moving trajectory in RoFin than the vision-based approach.

5.6 Hand Pose Reconstructing

5.6.1 Identify Rolling Labels via CNN

Traditionally, we could decode these optical labels via the amplitude thresholds. However, due to the variable optical environment, it is difficult to configure the thresholds dynamically. Even in the same ambient light settings, the captured rolling pattern for each finger requires different thresholds for decoding. Furthermore, the amplitude gap between the CP and On symbols is narrowed dramatically in strong ambient light and could cause numerous decoding errors.

Convolutional Neural Networks (CNN) are widely applied in computer vision object classification due to their great robustness and accuracy. The benefits include: (1) Offline training and online identification can reduce latency for real-time finger label parsing; (2) even in high ambient light and difficult to distinguish CP and On, the CNN model can learn the features in the repeating dark and bright rolling strips.

We adopt YOLOv5 for our optical labels identification with their related bounding boxes. YOLO (You Only Look Once) models are commonly used for objects detection since their fast inference with high accuracy. The network structure of YOLOv5 consists of EfficientNet backbone structures, BiFPN (Bi-directional Feature Pyramid Network) layers to extract object's features effectively, as shown in Figure 5.9 (a). Then these features are fed through the prediction nets for both objective's class and location of boxes as output.

We capture 90 images of 3 RoFin gloves in 3 different ambient light strengths with 10 images for

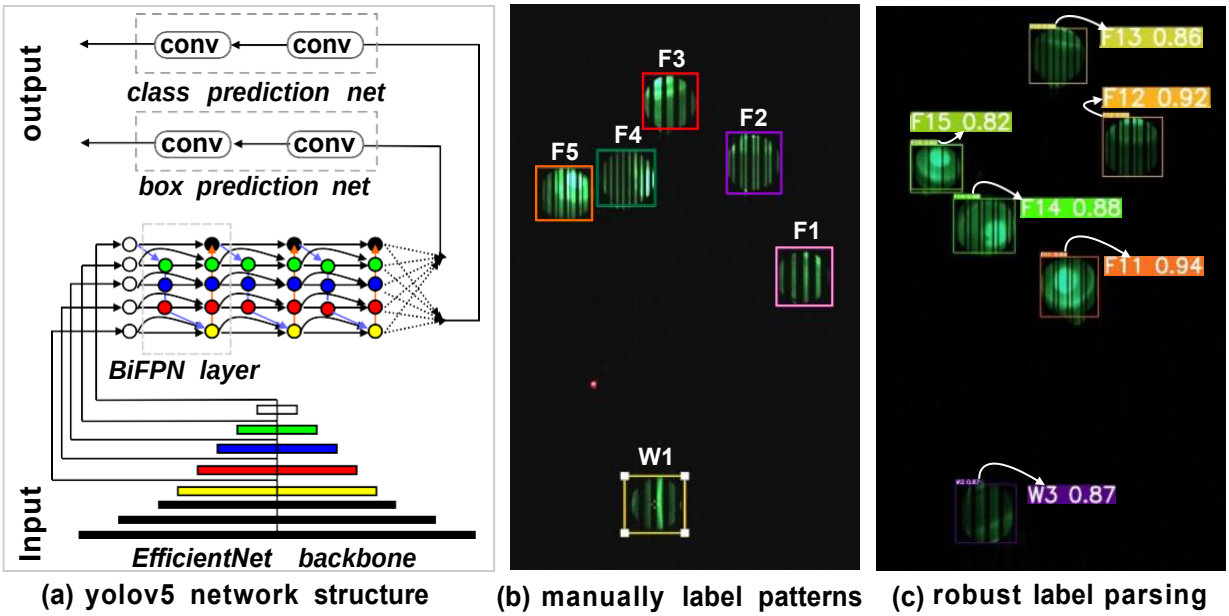


Figure 5.9 Label parsing via YOLOv5.

each setting. Then, we manually label each rolling pattern with 18 class labels (i.e., F1-F15, W1-W3), as shown in Figure 5.9 (b). Then, we adopt data augmentation via the gray-scale modification to increase the size of training dataset. Finally, we use the trained model to infer the rolling pattern's label with bounding boxes. As shown in Figure 5.9 (c), the trained model can output the label accurately with a high confidence ratio. Besides, these outputted bounding boxes include each sphere's x,y, and radius for 3D spatial parsing and further hand pose reconstructing.

5.6.2 Cluster Fingers and Wrists into Hands

Grouped 6 Key Points of a Hand. Based on the identified fingertips and wrists from multiple hands above, we can calculate their hand belonging separately. And then we can easily cluster fingertips and wrist points from one hand together. For example, the fingers which have indication numbers in [1, 2, 3, 4, 5] and the wrist point with an indication number of 1 should be grouped in hand #1 due to their calculated hand index being the same, which is 1. As shown in Figure 5.10, the wrist labeling avoids the wrong finger clustering with the wrist point from another hand and thus guarantees further accurate hand pose reconstruction.

3D Coordinates of 6 Key Points. The 6 key points with 3D coordinates clustered into one

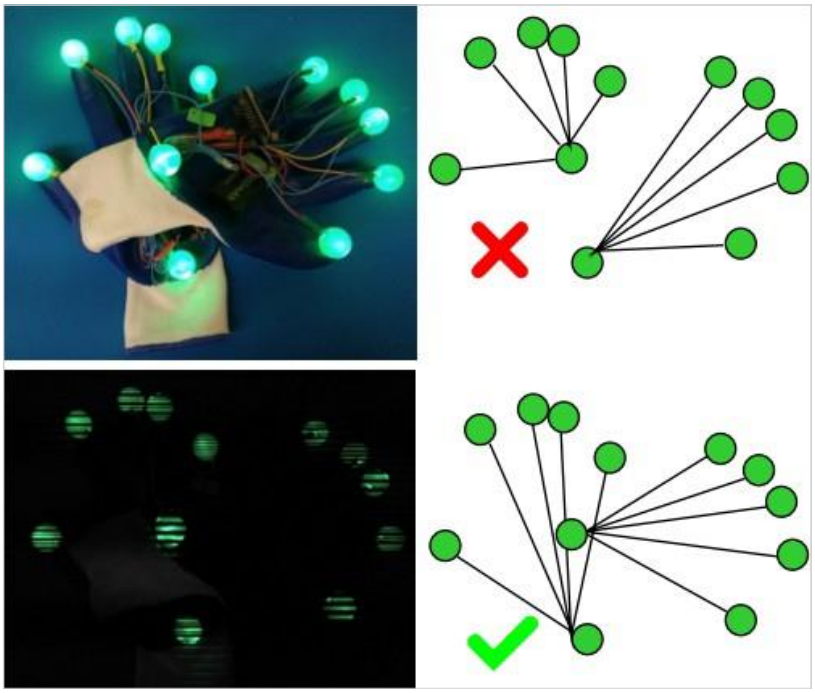


Figure 5.10 Finger clustering with the correct wrist point into a hand.

hand will be input into the HPR model and then the HPR model outputs the reconstructed 3D hand pose in real time. Different from fine-grained finger tracking with numerous inside-frame sampled points in an image frame, the real-time hand pose reconstructing requires only one 3D location sample for each of six key points per frame for processing.

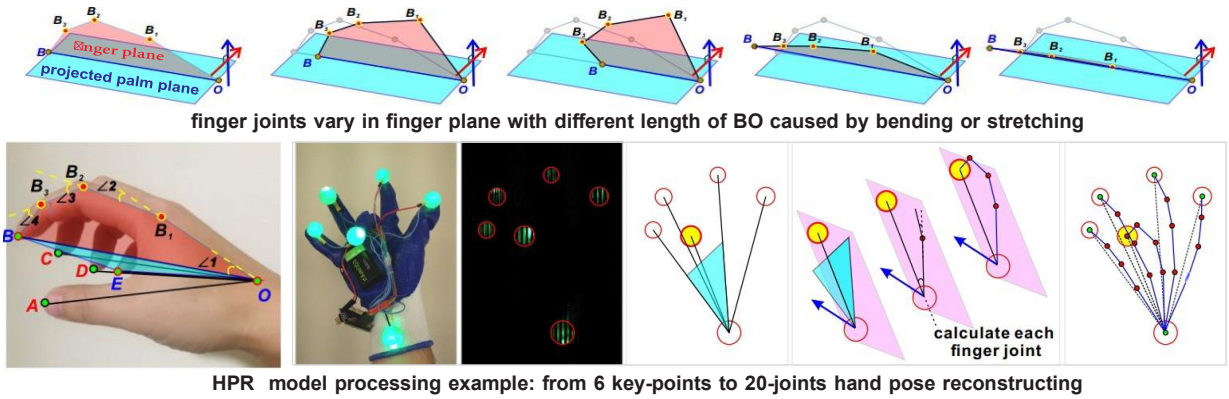


Figure 5.11 Hand and the illustration of the HPR (hand pose reconstructing) model for hand pose reconstruction via six tracked 3D key points.

5.6.3 Lightweight HPR Model

Given 3D positions of 5 fingertips and 1 wrist point from a hand, the 3D hand pose is definite and thus we can reconstruct 3D hand pose. In comparison to vision-based approach, our approach tracks only 6 key points instead of 20 points for less tracking and computation overhead. However, it is challenging to reconstruct a 20-joints hand pose via restricted 6 key points in real time. To overcome this challenge, we design the lightweight HPR model illustrated below.

According to six key points with 3D coordinates (the wrist point p_O, the tip of thumb p_A, the tip of index finger p_B, the tip of middle finger p_C, the tip of ring finger p_D, and the tip of little finger p_E), as shown in Figure 5.11, how can we reconstruct a 20-joints hand pose? The intuitive answer is to calculate the 3D location of the other 14 points: p_{A_1}, p_{A_2} (i.e., we simplify the thumb finger with 2 joints), p_{B_1}, p_{B_2}, p_{B_3}, p_{C_1}, p_{C_2}, p_{C_3}, p_{D_1}, p_{D_2}, p_{D_3}, p_{E_1}, p_{E_2}, and p_{E_3}.

The Plane of Projected Palm. As shown in Figure 5.11, the fingers and the palm can be projected on the plane which we defined as projected palm P_{palm}. Actually, the tips of the index finger and the little finger and the wrist point consists of the P_{palm} (i.e., P_{BOE}).

The Plane Formed by Finger Joints. The joints of a finger form a finger plane. These finger planes (except the thumb finger plane) are perpendicular to the plane P_{palm}. For example, joints of the index finger: p_{B_1}, p_{B_2}, p_{B_3}, p_B, and the wrist point p_O generates the finger plane $P_{OB_1B_2B_3B}$. And $P_{OB_1B_2B_3B} \perp P_{palm}$ (i.e., $P_{OB_1B_2B_3B} \perp P_{BOE}$). In contrast to finger planes above, the thumb finger plane is almost parallel to the plane P_{palm} (i.e., $P_{OA_1A_2A} \perp P_{BOE}$). Thus we can find these 5 finger planes based on the known plane P_{BOE}, as shown in Figure 5.11.

Given the 5 connection lines between each fingertip to the wrist point (i.e., l_{OA}, l_{OB}, l_{OC}, l_{OD} and l_{OE}) and the calculated finger planes $P_{O\underline{A_1}\underline{A_2}A}$, $P_{O\underline{B_1}\underline{B_2}\underline{B_3}B}$, $P_{O\underline{C_1}\underline{C_2}\underline{C_3}C}$, $P_{O\underline{D_1}\underline{D_2}\underline{D_3}D}$, and $P_{O\underline{E_1}\underline{E_2}\underline{E_3}E}$, we can determine the unknown 14 joints (underlined) on the finger planes via following two rules.

- We can simplify the finger bending because each finger section from one finger bends with a similar angle or proportional angle, as shown in Figure 5.11.

- The length from the fingertip to the wrist point l_{con} equals the sum of each finger section's projection to the line l_{con}. Thus, we can calculate the bending angle and further find each unknown joint location.

As shown in Figure 5.11, the finger joints of the index finger vary in its finger plane with different lengths of l_{con} (i.e., l_{OB}). Thus, given a value of variable l_{con}, the 3D locations of other joints from this finger are fixed and can be calculated. For example, we know the length of each finger section of the index finger (i.e., l_{OB_1}, $l_{B_1B_2}$, $l_{B_2B_3}$, and l_{B_3B}) by the initial measurement step. Given the calculated l_{OB} (i.e.,l_{con}), the unknown bending angle for index finger $\angle\alpha$ can be calculated by the equation below:

$l_{OB_1} \times \cos 2\alpha + l_{B_1B_2} \times \cos\alpha + l_{B_2B_3} \times \cos\alpha + l_{B_3B} \times \cos\alpha = l_{OB}$.

5.7 Implementation and Evaluation

5.7.1 RoFin Gloves

We implement three wearable RoFin gloves for experiments as shown in Figure 5.12. The main components in one pair of RoFin gloves are shown in Table 5.1: lightweight insulated breathable gloves, 2 Arduino Nano MCU, 12 green LEDs wrapped with 12 green plastic balls (ϕ = 19mm), and a 9V li-ion battery for power-supply. The total weight of one pair of RoFin glove is **132g** (including two batteries' weight of 60g) while the total price is only **26.3**$.

Component	**Price** (USD)	**Details**
insulated gloves	0.6 x 2 = 1.2	for each: 24cm x 15cm, 18g
Arduino Nano	10 x 2 = 20	ATmega328P, 5V, 16M
LED	0.02 x 12 = 0.24	5mm, green, 20000mcd, 20mA
plastic cover	0.08 x 12 = 0.96	19mm, green, lightweight
battery	2 x 2 = 4	rechargeable batteries cost about 7x2 = 14$
Total price	26.3	mass produced, cheaper the price

Table 5.1 Components in one pair of RoFin gloves.

5.7.2 RoFin Reader

There are numerous commercial smart devices widely available and reasonably priced that can be used as our RoFin reader including smart phones, drone cameras, and even underwater sports

cameras. In our experiments, we use commercial smartphones such as iPhone 7, VIVO Y71A, and Samsung s20, as shown in Figure 5.12 (b).

We evaluate the RoFin's performance in three folds. **(1) label identification** with different ambient light settings, distances, cameras. **(2) inside-frame tracking performance** in contrast to vision-based method. **(3) hand reconstruction performance** with Leap Motion as the benchmark. Then we also discuss about RoFin's use cases and other concerns such as privacy, power consumption.

5.7.3 Robust Label Parsing

In this subsection, we evaluate the label parsing performance under different settings: (1) ambient light [low, medium, strong], (2) sensing distance [0.5m, 1.5m, 2.5m], (3) different hands [#H1, #H2, #H3], (4) different labels [F1-F15, W1-W3], and (5) different cameras [iPhone 7, VIVO-Y71A, Samsung s20], as shown in Figure 5.12 (b) and (c).

(1) Impact of Ambient Light. We use the trained model to predict the labels in the captured images in three different ambient light environment at the same distance 0.5m with 3 hands. As shown in Figure 5.13 (a), the label parsing achieves the best accuracy under the strong ambient light at 0.94 and the average accuracy of 0.91. These results demonstrate RoFin's label parsing works robustly under varied ambient light even in the darkness and outperforms than vision-based approaches, which can not work in the darkness and lack of identification ability.

(2) Impact of Sensing Distance. We predict the labels in the captured images in three sensing distance settings under the same strong ambient light setting. The average accuracy of label parsing is shown in Figure 5.13 (b). The accuracy of label parsing drops slowly with increased distance. RoFin achieves the best accuracy of 0.93 at 0.5m and 0.77 at 2.5m. These results demonstrate RoFin works robustly under varied sensing distance even at 2.5m, which outperforms than vision-based approaches with limited distance (i.e, 1m).

(3) Impact of Different Hands. We also evaluate the label parsing performance of six labels from different hands. As shown in Figure 5.13 (c), The hand #1 and #3 achieve the high prediction accuracy more than 0.96 while the hand #2 achieves the lowest accuracy of 0.77. The reason is

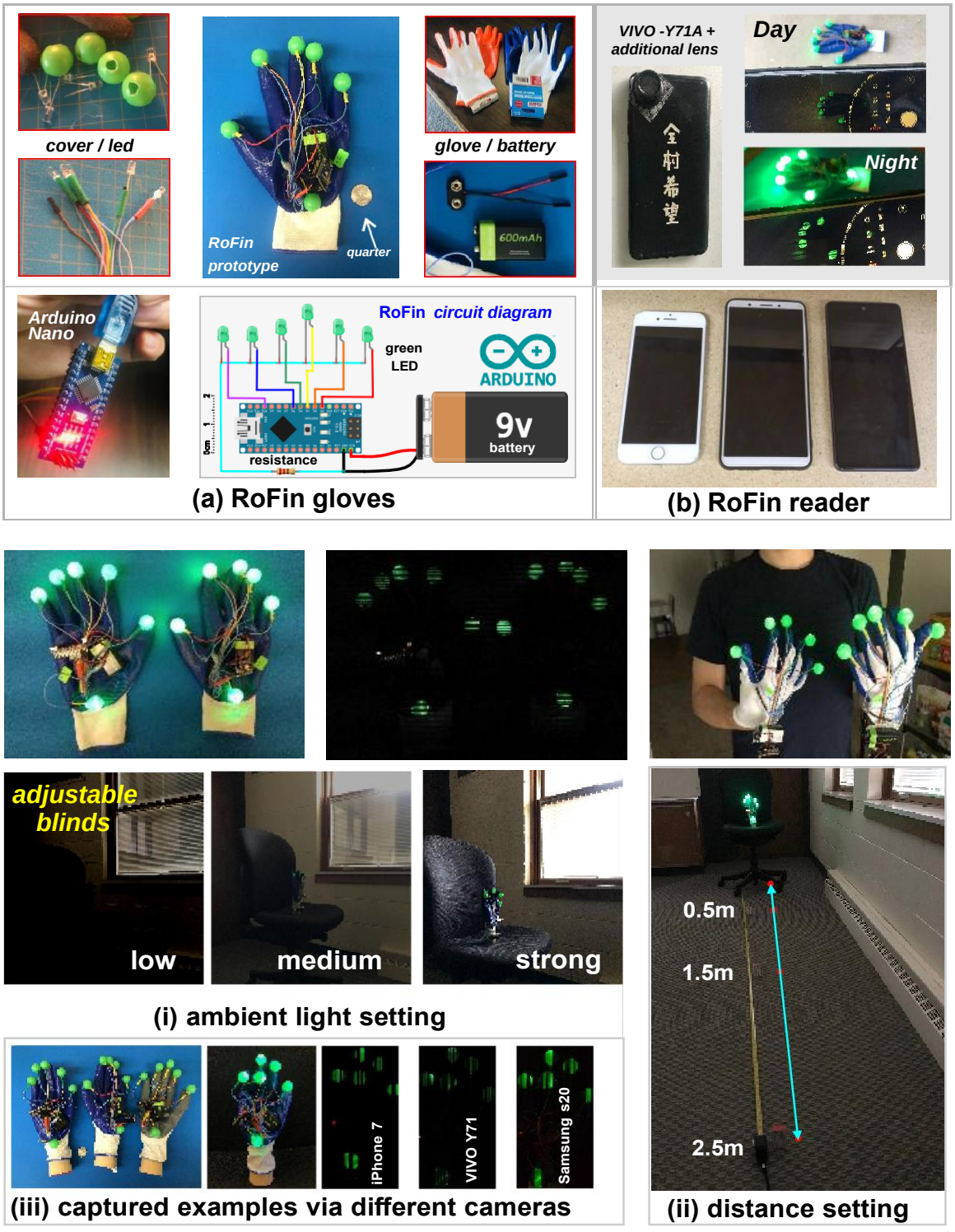


Figure 5.12 System implementation: RoFin gloves (prototype & circuit diagram), RoFin reader (commercial cameras) and experiment scenarios (varied ambient light strength and distances from 0.5m to 2.5m).

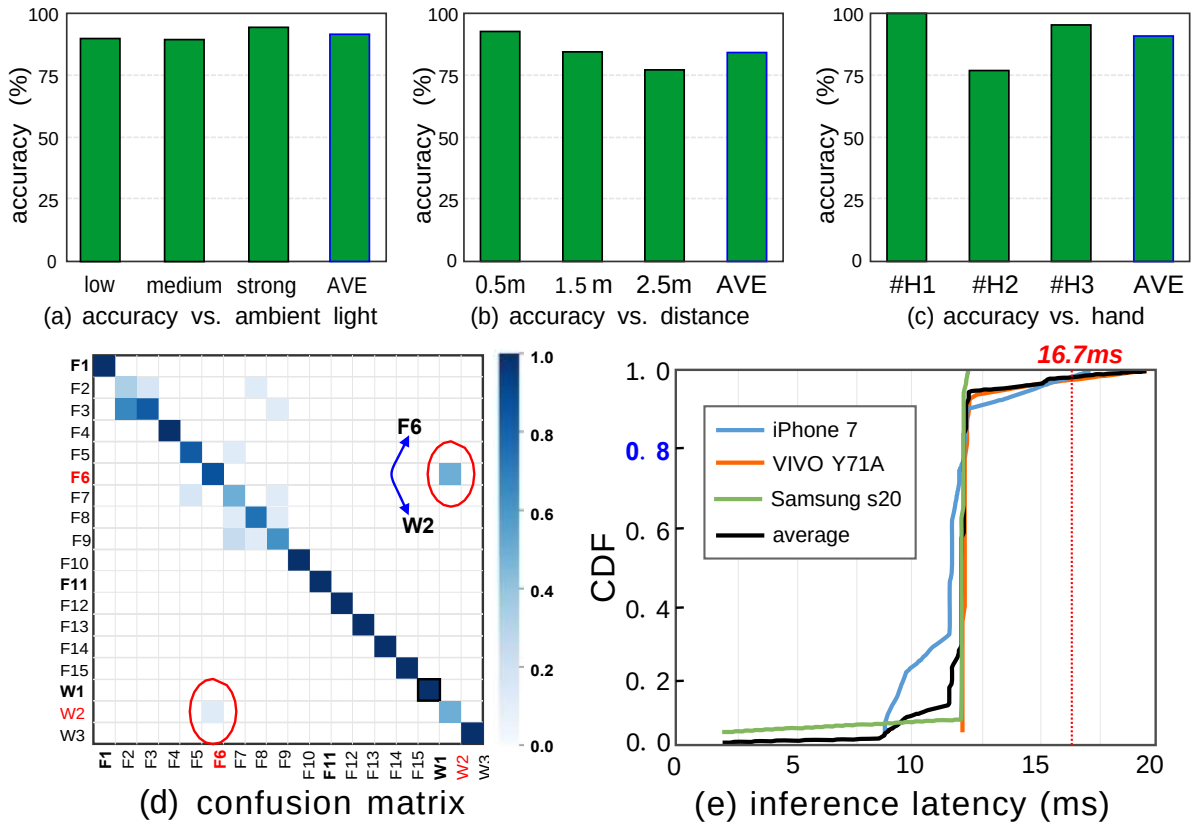


Figure 5.13 Label parsing accuracy performance in varied settings and latency evaluation.

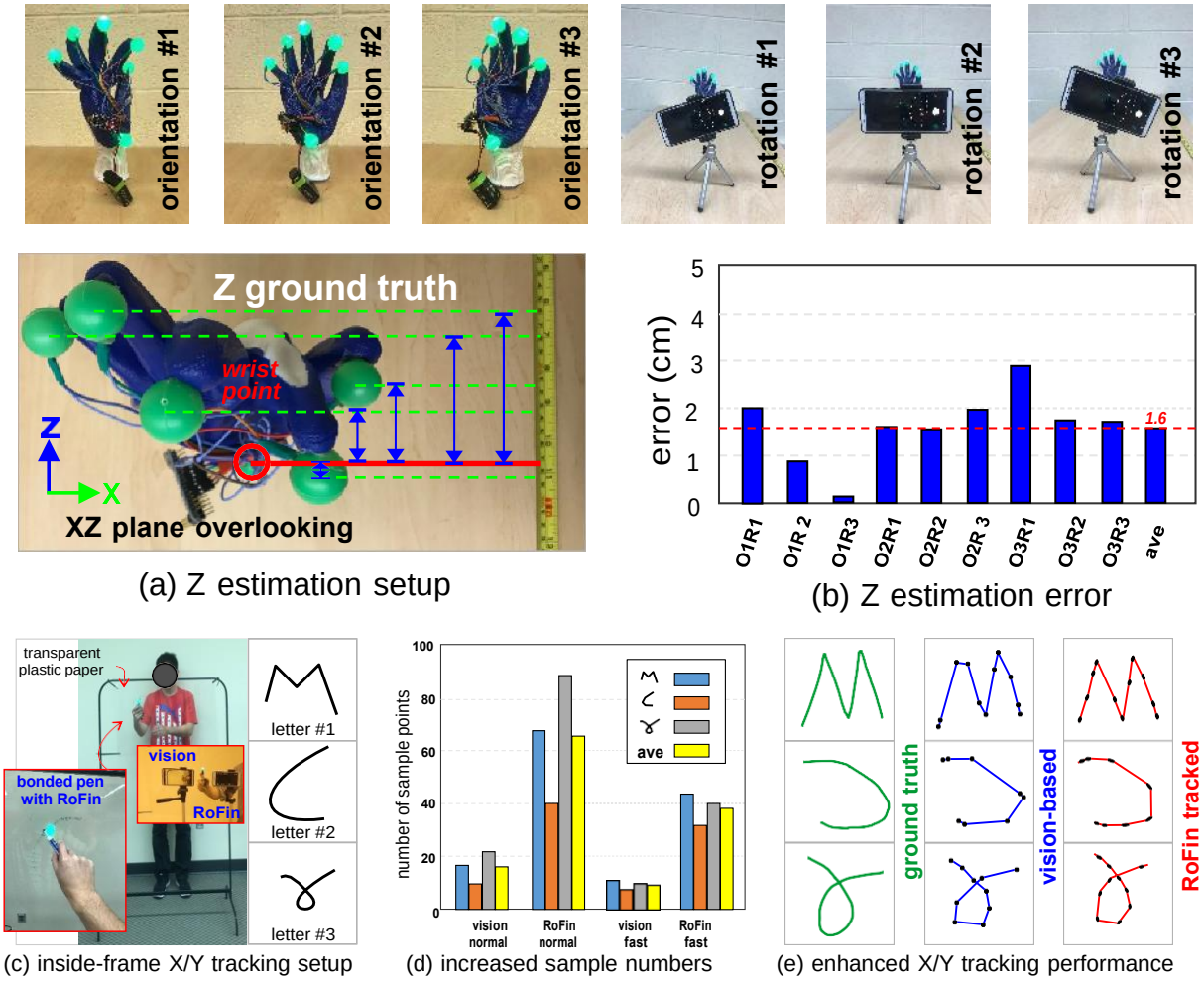
the F6-F10 from hand #2 have more confused rolling patterns than hand #1 and #2. Even though, the average label parsing accuracy still achieves 0.91, which demonstrates the effectiveness of our optical labeling and parsing scheme.

(4) Impact of Different Labels. We also present the confusion matrix of the trained label parsing model for 18 different classes (i.e., F1-F15,W1-W3) in Figure 5.13 (d). It shows the labels from hand #2 are easier to be identified as other labels than hand #1 and #3, which is consistent with the results in Figure 5.13 (c). It also shows that the rolling pattern of W2 [CP, On, Off, Off, On, Off] is confused by F6 [CP, Off, Off, On, On, Off]. That is because the reversed rolling patterns of F6 [Off, On, On, Off, Off, CP] (i.e, [..Off, On, On, Off, Off, CP, Off, On, On, Off, Off, CP..]) has the high similarity with the W2 when the amplitude of CP is similar to On symbol.

(5) Impact of Different Cameras. We use the trained model to parse the labels captured by different cameras of commercial smartphones [iPhone 7, VIVO Y71A, and Samsung s20]

to measure their label parsing latency performance. As shwon in Figure 5.13 (e), the labels captured by iPhone can be parsed with the shortest time while the average parsing latency of these different cameras are about 12ms (i.e., 83Hz), and less than the 16.7ms (i.e., 60 Hz). These results demonstrate RoFin achieves the real-time label parsing.

5.7.4 Enhanced Inside-frame Tracking



(a) Z estimation setup (b) Z estimation error

(c) inside-frame X/Y tracking setup (d) increased sample numbers (e) enhanced X/Y tracking performance

Figure 5.14 Z estimation performance and the enhanced inside-frame tracking performance.

In this subsection, we evaluate the accuracy of Z estimation and the enhanced inside-frame tracking of X/Y.

· **Z Estimation Performance**

Setup. We set a wooden hand model worn RoFin glove at the desk, as shown in Figure 5.14 (a).

The fingertips are separated with different distances to the camera image plane (XY plane). The hand model keeps the same pose but with 3 different orientations to the camera. We also set camera with 3 different rotations to capture the RoFin glove. Then we measure the distance between the fingertips' projected points on the desk to the camera plane as the Z ground truth.

Z Estimation Accuracy. As shown in Figure 5.14 (b), although the error of estimated depth info Z via RoFin varies with the different hand orientation and camera rotation, RoFin achieves the average estimation error of 1.6 cm when sensing distance is 0.5m.

· X/Y Tracking Performance

Setup. We bond one fingertips of the RoFin glove with a pen (blue marker) and draw on the transparent plastic paper hanging parallel to the camera's image plane, as shown in Figure 5.14 (c). We also set two cameras at the fixed distance 0.5m when the user is drawing. One camera follows the traditional vision based approach which captures the video as usual with 60 fps frame rate while the other camera (RoFin reader) captures the video of the rolling patterns with the same 60 fps frame rate but with high rolling shutter rate (8KHz). Thus we track 3 traces of user's drawing at the same time: (1) ground truth on the plastic paper, (2) vision approach tracked trace, (3) RoFin tracked trace.

X/Y Tracking Enhancement. We ask the user to draw 3 different letters: (1) M with more straight lines, (2) C with curve, (3) a rotated α with more complex curve with two writing speed: (1) normal speed, and (2) faster speed. As shown in Figure 5.14 (d) and (e), the RoFin tracked 4 times of location points for the same letter, which significantly enhances the granularity of tracking trace in compared to vision-based tracking. Besides, RoFin achieves more accurate trace tracking than vision based method among all three different letters due to its fine-grained inside-frame sampling.

In a nutshell, it demonstrates that our low-cost RoFin provides accurate Z estimation and enhanced X/Y tracking.

5.7.5 Real-time Hand Pose Reconstruction

We define 10 hand poses as shown in Figure 5.15 for hand pose reconstruction evaluation. We capture the images of the wooden hand worn the RoFin glove with RoFin reader for different

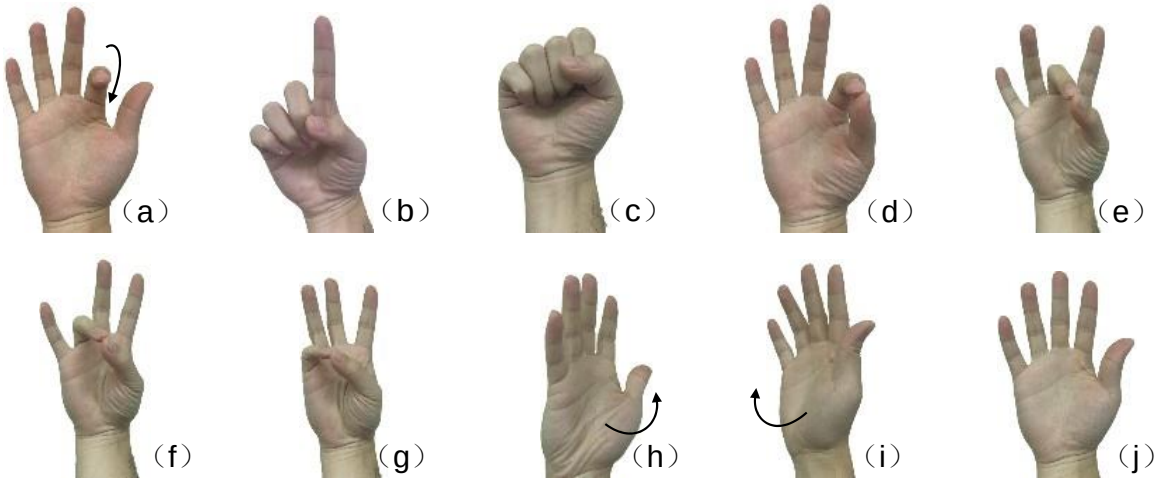


Figure 5.15 10 defined hand poses: (a) bend index finger, (b) point with index finger, (c) close the fist, (d-g) pinch thumb with Index, Middle, Ring, and Little finger, (h) turn palm to the left, (i) turn palm to the right, (j) the palm.

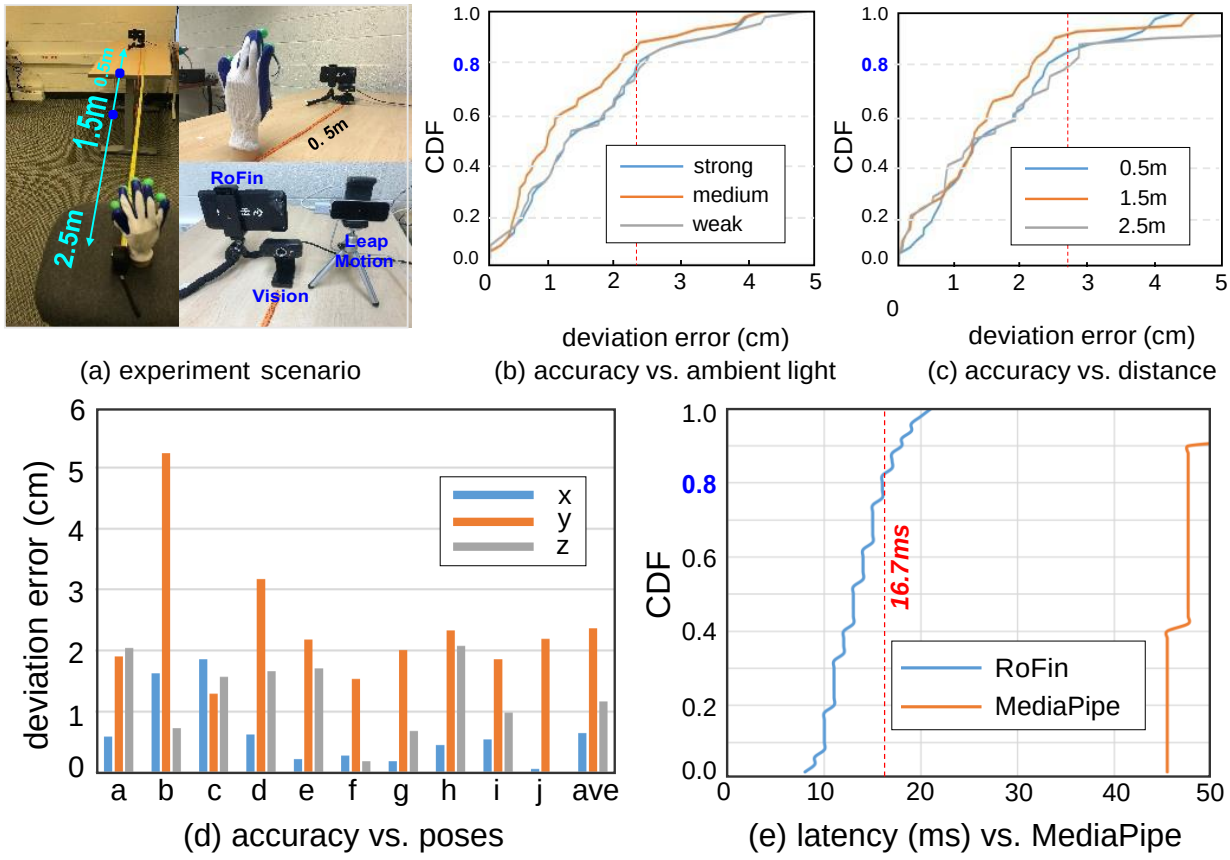
hand poses. Then we run HPR model and evaluate its accuracy and latency with Leap Motion as benchmark.

· **Reconstructing Accuracy**

Impact of Ambient Light. We define the deviation error as the average difference of x,y,z between RoFin with Leap Motion. As shown in Figure 5.16 (b), the average deviation error of three ambient light settings [low, medium, strong] under 0.5m has the similar distribution and the most deviation error is distributed less than 22 mm. Among three ambient light settings, the medium ambient light achieves the best performance due to the RoFin reader can capture the most clear contours of six key points' spheres.

Impact of Sensing Distance. As shown in Figure 5.16 (c), the average deviation error of three distances [0.5m, 1.5m, 2.5m] are similar and are mostly distributed in 28 mm. The deviation error of 1.5m achieves the best performance with the average deviation error of 14 mm while the 2.5m setting achieves the largest average deviation error of 19 mm. These results demonstrate our HPR model works well up to 2.5 m while the vision approaches usually work within 1 m and Leap Motion works within 0.5m.

Impact of Different Poses. We also evaluate the reconstructing deviation error of 10 hand



(a) experiment scenario (b) accuracy vs. ambient light (c) accuracy vs. distance

(d) accuracy vs. poses (e) latency (ms) vs. MediaPipe

Figure 5.16 Hand pose reconstructing performance.

poses defined above. As shown in Figure 5.16 (d), the reconstructed y has the largest deviation error compared with x and z, especially for the hand pose (b), point with index finger. The reason is that the finger planes of the ring finger, the little finger are not exactly as assumed in our simplified HPR model that their finger planes perpendicular to the projected palm plane. Among 10 hand poses, the pose (j) achieves the lowest average deviation error in hand pose reconstructing of 7.6 mm.

· **Reconstructing Latency.**

As for hand pose reconstructing, the main advantage of RoFin compared with vision-based approaches is its less tracked key points and flexible and long sensing distance. We evaluate the hand pose reconstructing latency and make comparison with the vision based approach Media Pipe ran on the same platform: Thinkpad T480 with Intel(R) Core(TM) i7-8650U CPU for different hand poses under the same 0.5m distance and strong ambient light setting.

As shown in Figure 5.16 (e), the latency of the RoFin HPR model is distributed less than 21 ms with the average latency of 13.8 ms (72Hz), which is less than 16.7 ms (60Hz). The vision based Media Pipe achieves 47.5 ms latency in average. Although the finger label parsing requires about 12ms for each image frame, the label parsing module and the HPR module can still run in pipe-line manner to achieve the real-time processing. These results demonstrate that our HPR model can achieve real-time hand pose reconstructing due to its only tracking 6 key points with simplified HPR model.

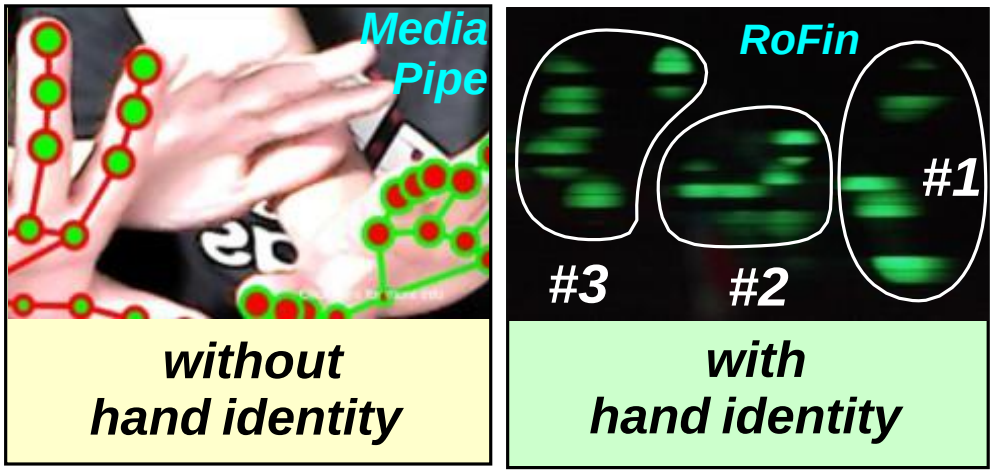
5.7.6 Use Cases

In this subsection, we provide three potential use cases for RoFin gloves in aspect of RoFin's three main features: (1) fingers/hands identification. (2) fine-grained inside-frame X/Y tracking. (3) real-time hand pose reconstructing.

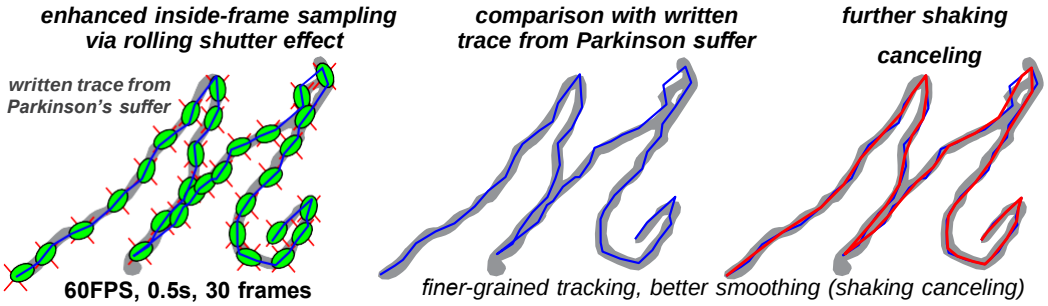
Multi-user interaction for AR/VR/MR. RoFin can track inside-frame X/Y location samples at rolling shutter rate and thus provide the ability of fine-grained finger tracking, especially for the high-speed motion or small-scale motion. Multiple users can use their fingertips to write or paint virtually at the same time in front of the camera. Thus, RoFin can be used as the user interface with better user experience for AR/VR/MR with privacy protection of users due to they only want the camera to capture the trace instead of the face, as shown in Figure 5.17 (a).

Virtual Writing or Health Monitoring for Parkinson's Suffers. Our RoFin system can track fine-grained writing trace including the subtle trembling while the vision-based approaches (1Hz inside-frame sampling rate and about 60 fps frame rate) can not track it clearly as human eyes. The Parkinson's suffers can use our RoFin glove to virtually write characters. Then we can use RoFin tracked fine-grained trace to better smooth the trace (e.g, connect the middle points among two trace sub-lines), as shown in Figure 5.17 (b). Besides, the tracked fine-grained trace can be also utilized as the medical diagnosis and health monitoring.

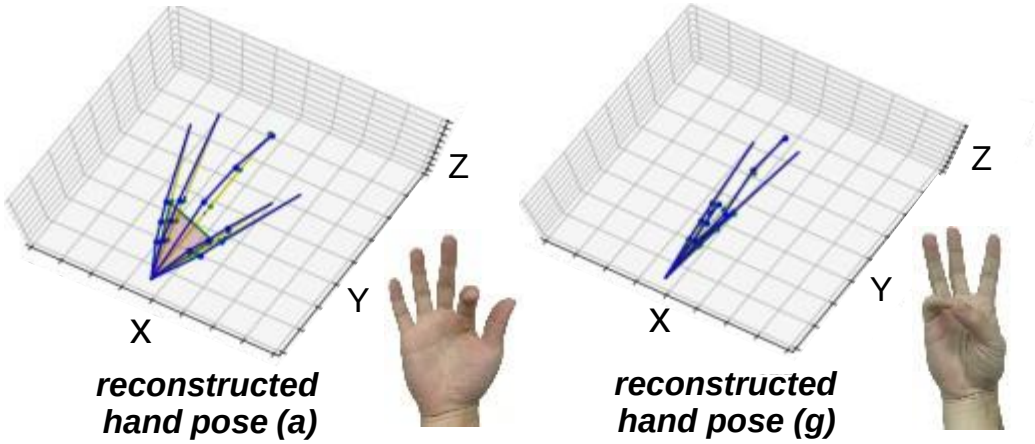
Hand Pose Commands for Video Games/Smart Home. RoFin achieves real-time hand pose reconstructing with less computation overhead and high accuracy. With the similar use cases as other hand gesture recognition approaches, our low-cost RoFin system can be used as the hand pose



(a) multi-user MR interaction



(b) fine-grained trampling tracking for further smoothing



(c) hand pose commands

Figure 5.17 Three possible use cases for our low-cost RoFin: (1) multi-user MR interactions with identification and protected privacy, (2) finer-grained tracking of writing of Parkinson's suffer[68], (3) real-time hand pose commands.

command input interface for video games, smart home. Figure 5.17 (c) shows reconstructed hand pose examples via RoFin's HPR model.

5.8 Discussion and Summary

Non-vision based Solutions. Our RoFin outperforms the vision-based approach in several aspects: (1) provide finger indication, (2) finer-grained finger tracking with the same frame rate setting, (3) less key points tracking and faster hand pose reconstruction, (4) long work distance and robust under varied ambient lights, (5) privacy protection and low cost. As for the non-vision based solutions, there are two types: (1) on-body sensor based approaches[12, 52, 48], and (2) hand-free approaches[120, 61, 75, 59]. Compared with our RoFin, these approaches have some limitations: (1) requirement of specific or expensive sensors and devices instead of commercial LED nodes, such as mmWave chips, FBG sensors, (2) limited sensing distances within the near hand area (i.e., within 0.5m), (3) lack of finger or hand identification ability and can not serve multiple users with user identification.

Privacy Leakage. Vision approaches such as human eyes, Media Pipe, as well as the Leap Motion integrated with camera can cause the privacy leakage. One example is shown in Figure 5.18. The user conducts the hand pose command while his/her hand holds a bank card. The Leap Motion and the Media Pipe cause the leakage of sensitive information (i.e., cvv number) which may result in the property loss.

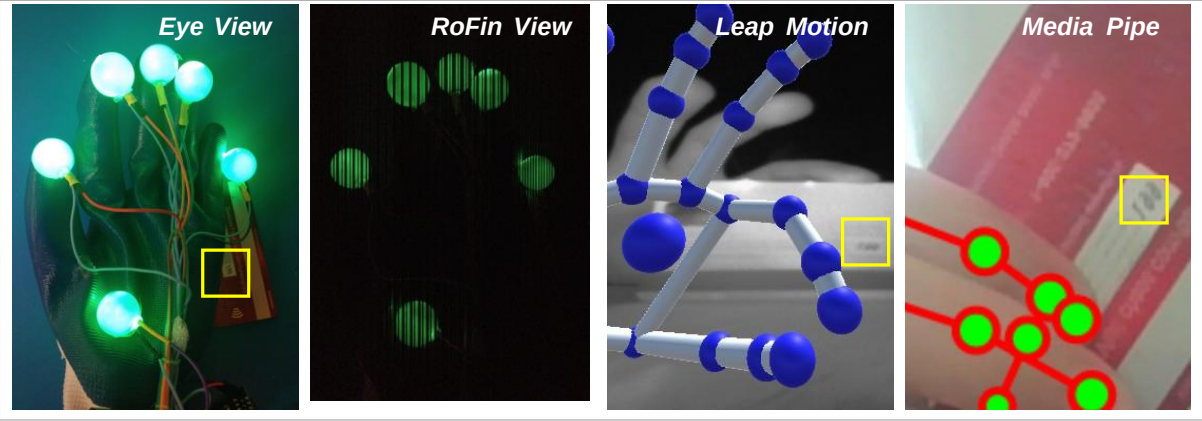


Figure 5.18 Sensitive data leakage of vision-SOTA.

Power Consumption and Safety. Our RoFin gloves are made of electric insulation rubber gloves, and the voltage at the LED node side is less than 3V, ensuring the safety of users who wear gloves. The current through one RoFin glove's circuit is **75** mA, and the power consumption is **225** mW. Based on our 600mAh and 9V li-ion battery, one RoFin glove can work for approximately 5.4 Wh / 225 mW = **24** hours before needing to be recharged.

Limitation of RoFin. Compared with hand-free approach such as vision based method, our current RoFin prototype requires the user to wear gloves attached with plastic spheres and has wires and battery. This limitation can be relieved by ergonomic design, textile technique, energy harvesting, or even passive labeling optimization in the future.

Future Direction. (1) optimize the spheres and explore back-scatter based passive fingertips' labeling. We can decrease the sphere size and exploit energy harvesting techniques for decreased weight and ease to use. (2) update HPR model. We cam improve HPR for hand poses in which finger planes are not perpendicular to the projected palm plane (e.g., hand pose (b)). (3) extend RoFin for body gesture recognition. The core idea of RoFin can be extended for human body gesture reconstructing easily with predictable benefits.

In summary, we exploit the 2D temporal-spatial rolling to construct 3D hand pose. We address technical challenges in RoFin system design and implementation, e.g, fingertips active optical labeling, fine-grained 3D information parsing of rolling fingertips, and lightweight 20-joints 3D hand pose reconstructing via 6 tracked key points. Then we undertake studies using RoFin gloves in a variety of circumstances. The results demonstrate our RoFin can robustly identify fingers, parse fine-grained 3D info, and achieve real-time hand pose reconstruction. Our RoFin is a low-cost but effective solution for human computer interactions with promising use cases.

CHAPTER 6

4D SPATIAL-TEMPORAL DIVERSITIES IN SWARMING DRONES

Drones have become increasingly popular in both the industry and research communities due to their numerous advantages, such as low cost, small size, adaptability, ease of use, and a wide range of potential applications. However, the current control method for swarming drones relies on stand-alone modes and centralized radio frequency control from a ground-based base station, which lacks drone-to-drone communication. This approach has several drawbacks, including crowded RF spectrum with mutual interference, high latency, and a lack of on-site drone-to-drone interactions.

To address these limitations, we propose PoseFly, an AI-assisted Optical Camera Communication (OCC) system designed for drone clusters. OCC offers several benefits, including high spatial multiplexing capability, Line of Sight (LoS) security, broader bandwidth, and an intuitive vision-based manner. By leveraging the rolling shutter effect in drone sensing and communication, PoseFly provides drone identification, on-site localization, quick-link communication, and lighting functionalities. This innovative approach offers a more efficient and reliable solution for sensing and communication within drone clusters, enhancing their overall performance and capabilities.

6.1 Motivation

Drones, one type of unmanned aerial vehicle (UAV), attract more attention because of their advantages over manned aircraft, including their small size, low cost, simplicity of operation, and broad potential applications[112, 53, 103, 93, 79]. Drones are now used in a variety of fields, such as aerial photography, plant protection, express deliveries, transportation, animal monitoring, surveying and mapping, power inspection, disaster relief, news reporting, selfies, film and television production. Drones are projected to play significant roles in integrative development for sensing, communication, and computing in the near future due to ongoing advances in artificial intelligence and their superior mobility. According to Verified Market Research, the size of the global drones market, which was expected to be worth USD 19.23 billion in 2020, would increase to USD 63.05 billion by 2028 with a CAGR of 16.01 percent between 2021 and 2028[28].

Nonetheless, the current approach to drone's control relies on centralized base station (CBS)

from the ground. This technique has several limitations, including RF spectrum congestion, which causes interference, significant latency, and the absence of real-time drone-to-drone interactions on-site. The transmission between the drones and the CBS in centralized control can naturally be avoided by the on-site interactions among drones in distributed manner. We could use RF to establish distributed drone-to-drone communication. However, due to Non-Line-of-Sight (NLoS) propagation, eavesdroppers can easily detect RF signals, and there is nontrivial multi-path effects and caused mutual interference[6, 36]. Even though there is no back-and-forth communication cost between drones and the CBS in RF based distributed drone-to-drone communication, the growing drone population may cause the RF spectrum to become crowded, which could lead to more localization errors owing to retransmission and lag.

There are two main issues for localization of drones with high mobility: **(1)** computing a drone's appropriate localization information, including distance, posture, speed, and so on; and **(2)** promptly receiving the computed localization information. Actually, we can use on-site posture features of a drone (transmitter) and compute at the receiving side (another drone) instead of transmitter's IMU to reduce transmission overhead. For instance, when a flock of geese is flying together, goose A (receiver) observes goose B (transmitter) and processes B's posture features in A's brain rather than goose B computing its own position and notifying A.

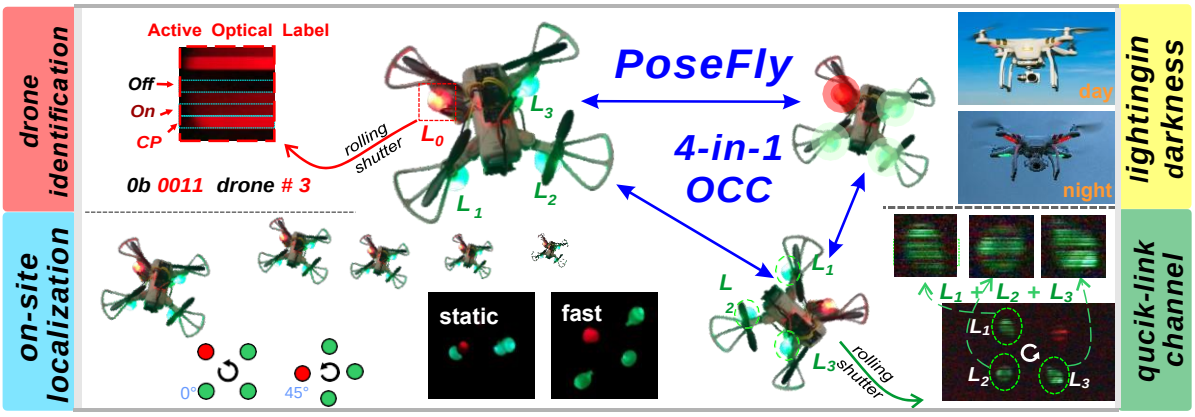


Figure 6.1 PoseFly: 4-in-1 OCC for swarming drones, similar to geese flying and their relative localization and collaboration.

To overcome the limitations of existing work, we introduce **PoseFly**, a novel approach that

leverages the 2D spatial-temporal diversities of rolling shutter cameras for on-site drone positioning. As depicted in Figure 6.1. PoseFly makes use of four inexpensive LEDs with plastic covers. One of these LEDs is red, and the remaining three are green. The red LED in the front-left corner of the drone emits unique cyclic OOK (On-Off Keying) waves, serving as an optical identification for each drone. As a result, drones with inbuilt cameras can easily identify one another. Furthermore, when coupled with green LEDs, the red LED aids in locating. PoseFly precisely calculates the positions of the drones and enables rapid data flow between them via Optical Camera Communication (OCC) links by evaluating changes in the arrangement of these LEDs.

6.2 Background and Related Work

6.2.1 Drone Identification

Vision based methods could be used to identify drones. For example, the camera can take an image of a drone and identify it based on its shape and features. Then the reader uses the greyscale image of the scene and detect the drone based on its silhouette[104]. However, these systems cannot work well at night, as the captured image of drones are not clear enough, nor do they work at longer distances. RF systems can identify drones in a few ways. Drones typically communicate at a much higher frequency than other mobile devices. If the RF connection is monitored, the used frequency could be utilized to determine if a device is a drone or not. However, other wireless devices could communicate at the same frequency and thus it will cause the wrong identification[78]. Instead of the clear images with complete morphology needed by computer vision or confused RF spectrum indication, PoseFly[144] simply requires one active LED node which holds the indication sequences and can work well in both day and night.

6.2.2 Drone Localization

We present the related work of drone localization below and illustrated in Figure 6.2. **(1) RF.** Current RF-based drone localization methods are based on received signal strength or time difference of arrival. By monitoring the signal strength of an emitter or the change in time of its arrival, a receiver could determine the direction and speed of the drone. However, interference in the

path can corrupt the localization results [77]. **(2) Vision.** The vision based localization approaches use cameras to record several frames of scene, then detect a drone and calculate its velocity and future position[90]. While this is certainly effective, it has non-trivial processing overhead, especially for image processing of morphology with varied background when the drone is flying.**(3) IMU.** Drones can also measure their own localization date (e.g., position, and velocity) via inner measurement unit (IMU) and send them out to other drones. However, these messages would need to be sent constantly and received through long distances.Thus, the IMU based methods have non-trivial send-out communication overhead and time delay, especially when there are numerous drones with severe interference[98].**(4) GPS.** Although GPS system can provide accurate location information, they also have send-out cost and cannot work well in urban areas, caves, tunnels. **(5) LiDAR.** As for LiDAR system, they can provide on-site localization of nearby drones. However, it has high-energy consumption.

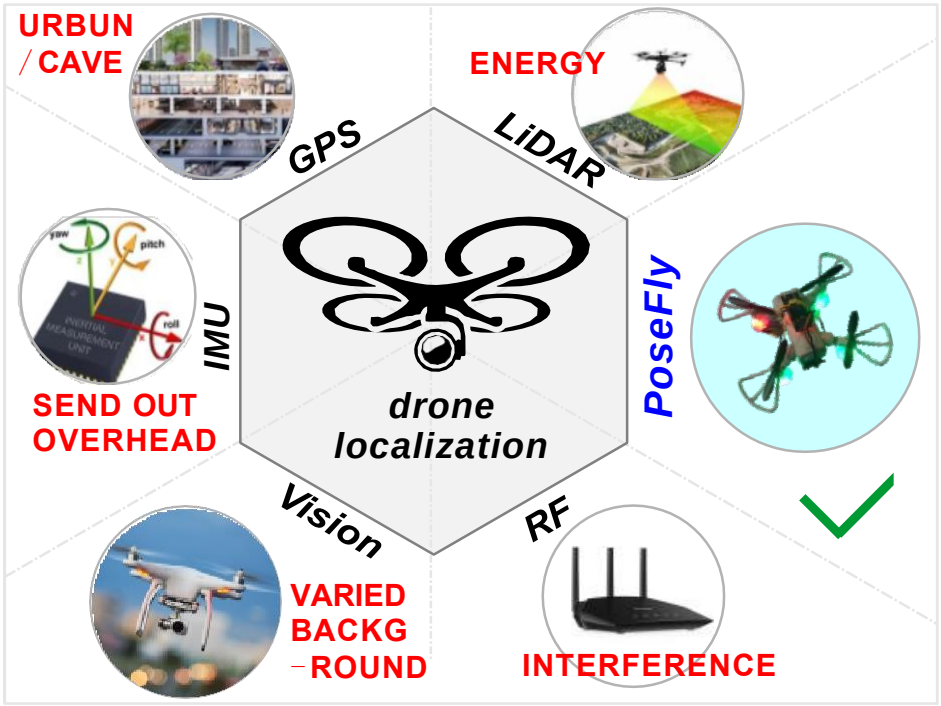


Figure 6.2 Drone localization approaches: GPS, IMU, vision, RF, LiDAR, and PoseFly.

In contrast to above mentioned drone localization approaches, PoseFly only requires one frame image to determine velocity and orientation. PoseFly uses 4 LED nodes to illustrate which direction

the drone is facing, allowing orientation to be found. Velocity can also be found through the orbs, as the faster the drone moves, the more the orbs will deform in one direction. It is free from interference from multiple drones thanks to the spatial diversity of millions pixels from the camera to capture them into different image zones. The illuminated balls allow PoseFly to work during day and night over flexible distances. Considering these energy efficient LED balls also provide lighting function, PoseFly is a green localization approach. Moreover, the localization of PoseFly does not have the send-out cost due to the reader capture the drone's image (the light propagates at high speed of 3×10^8m/s) and then process it locally. Besides, PoseFly's on-site localization only relies on the drones themselves and thus can work in caves/tunnels where GPS can not work.

6.2.3 Drone Communication

Today, most drones communicate via radio frequency medium. RF signals can travel over relatively long distances. However, RF systems can be prone to eavesdroppers, jammers, and interference [29]. The RF signal is sent though the open space and anybody can listen or send their own confounding signals. PoseFly is based on the Line-of-Sight propagation manner and thus the signals can be blocked out to attackers out of the swarming drones and makes it more secure than RF-based communication. Similarly, jammers must send more light directly into the receiver to jam the camera.

6.3 Our Approach: PoseFly

Our proposed **PoseFly**, is composed of two parts, as illustrated in Figure 6.3: (1) commercial LED based PoseFly Transmitter, (2) AI-assisted commercial camera based PoseFly Reader. One drone can equip both transmitter and receiver as a transceiver.

PoseFly transmitter. PoseFly transmiter consists of 4 commercial low-power LED components attached on each corner of a four-rotor drone. These 4 LEDs, one is red while the others are green, are covered with plastic balls of the same color and controlled by an Arduino Nano.

PoseFly receiver/reader. PoseFly reader is based on commercial cameras, which can be the mounted cameras on the drones. These cameras use adjustable focal length lenses and configurable rolling shutter rates and frame rate.

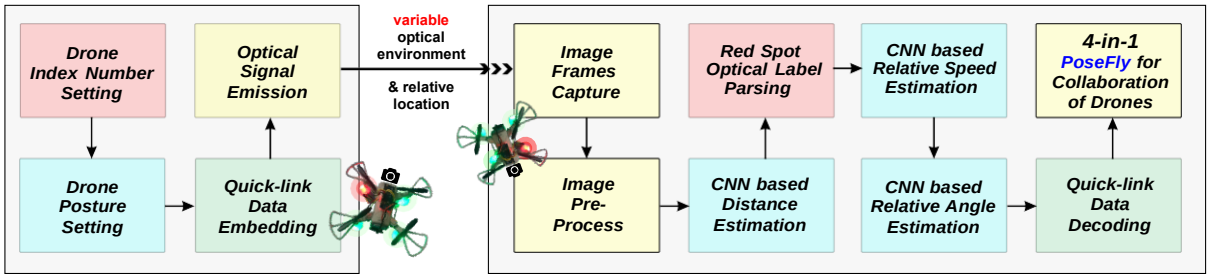


Figure 6.3 The system overview including transmitter and receiver, and the workflow of PoseFly.

Four Integrated Functions: (1) Drone identification: The red LED generates OOK waves with cyclic pilots to indicate the index of a drone in the drone cluster. For example, the OOK wave [on, off, off, on] indicates the index of the drone is 0b1001, which is # 9. **(2) Drone on-site localization**: The PoseFly reader can estimate distance from the transmitter to the reader based on the size of captured four LEDs. Furthermore, the reader can conduct on-site angle parsing based on generated shape and color pattern of four LEDs. Additionally, the shape of the rolling spot varies from normal circle to ellipse with different motion speed of drones, which can help the reader to conduct speed estimation. **(3) Drone quick-link**: At the same time, the other three green LEDs create the quick-link channel among nearby drones by fast on-off switching. **(4) Lighting**: These LED components provide lighting function at the dark environment or night.

Workflow: As shown in Figure 6.3, these four functions are achieved at different distance between two drones step by step. **(1)** Firstly, when a drone, Drone A, notices there is a bright spot, which is another drone, Drone B, based on B's **lighting function** in long distance (>20m) via its camera. **(2)** Then Drone A will fly closer to B based on its **distance estimation** (<20m) function and conduct the **drone identification** (<12 m) to know the index number of Drone B in the cluster of drones. **(3)** Later, Drone A flies closer to B and performs finer-grained localization of B such as the estimation of **motion speed and posture angle** of B. **(4)** When these two drones require mutual data sharing, they can fly closer within 4m and utilize the **quick-link** channel to share the information such as the fly instructions, on-site posture info of other drones.

There are three main **technical challenges**, as illustrated in Figure 6.4 and outlined below:

C1: Robust identification of drones at long distances. Unlike geese, drones cannot easily

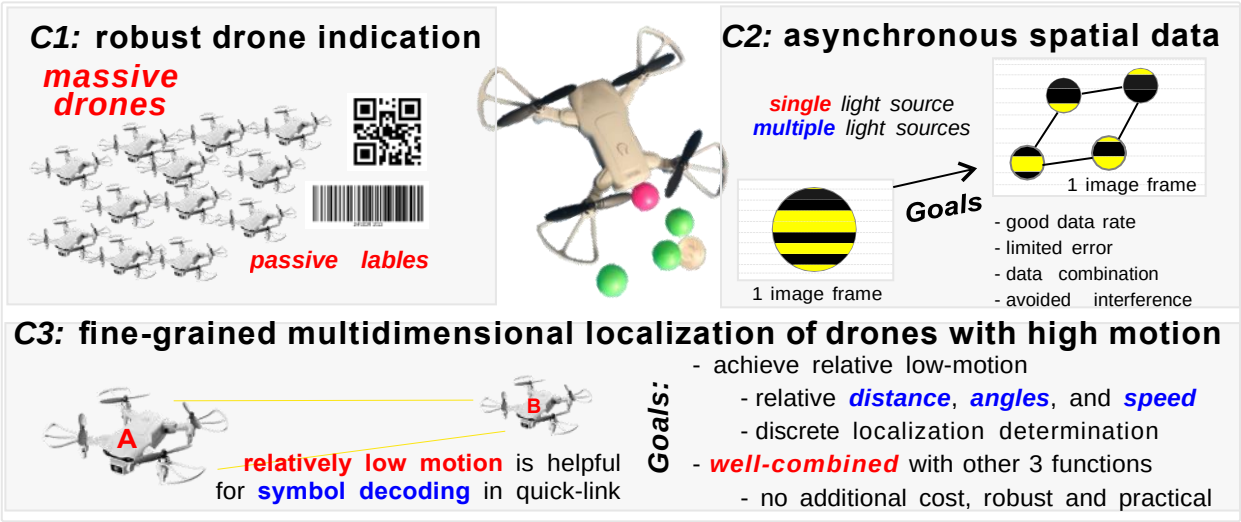


Figure 6.4 Three main challenges in PoseFly: robust drone indication, asynchronous spatial data combination, and localization with high motion.

recognize other drones with similar appearances through visual recognition alone. To address this, we propose attaching optical marks or labels on drones. However, traditional static marks or existing bar/QR codes are passive and can only work within a limited recognition distance, typically around 1 meter.

C2: Lightweight yet precise localization (distance, speed, angle). Geese can sense the posture of other geese using various vision features, such as the head, wings, and feet. However, applying the same method for sensing the drone's posture would introduce non-trivial computation overhead, which is not desirable for real-time applications.

C3: Decoding asynchronized rolling strips in rolling spots with random locations in a frame. The rolling strips generated in each rolling spot are not synchronized for decoding with flying drones. This asynchronous nature poses a challenge in efficiently and accurately decoding the information from the rolling strips, particularly when they appear at random locations within a frame.

Our **contribution** can be summarized as follows:

(1) This is the first work to exploit rolling patterns for on-site drone posture parsing, including relative distance, speed and angle estimation, which was solely used for optical camera communi-

cation before.

(2) We thoroughly investigate the spatial rolling patterns and design the 4-in-1 PoseFly, an AI-assisted approach for drone identification, drone localization, drone communication, and lighting with commercial LEDs and cameras.

(3) We address challenges via cyclic pilots and OOK for active optical labeling and robust quick-link communication. We adopt CNN models for accurate and robust identification, localization at the receiver side.

(4) We evaluate PoseFly on our implemented prototypes in both day and night with varying distance and motion speed. Experiment results show that PoseFly can identify drones with nearly 100% accuracy within 12m while providing accurate pose parsing (100% distance estimation within 20m, 100% speed and angle estimation within 4m). Additionally, PoseFly provides averagely 5 Kbps quick-link channel at up to 4m.

6.4 Drone Identification

For drone interactions, drone detection is critical. However, current optical labels like barcodes and QR codes are passive and only function at close ranges of a few centimeters. To overcome this limitation, we design active optical labels for drone identification in long distance (up to 12m). We present our active optical label design at transmitter side and the CNN based robust label parsing solution below.

6.4.1 High-capacity Optical Labeling

Rolling Shutter strip Effect. The global shutter exposes the entire scene at once. The rolling shutter in commercial CMOS cameras, in contrast, exposes one row of pixels while concurrently creating an entire image row by row. Figure 6.5 illustrates the rolling shutter strip effect, which happens when the rolling shutter speed and the switching speed of the light wave from the transmitter are about equal. Thus, temporal optical signals carrying transmitted data during symbol periods can be successively collected as rolling strips.

CP-OOK Label Wave Design. In PoseFly, each drone is identified by an optical label that regularly emits distinct amplitude waves that are invisible to human eyes (the On-Off switching rate

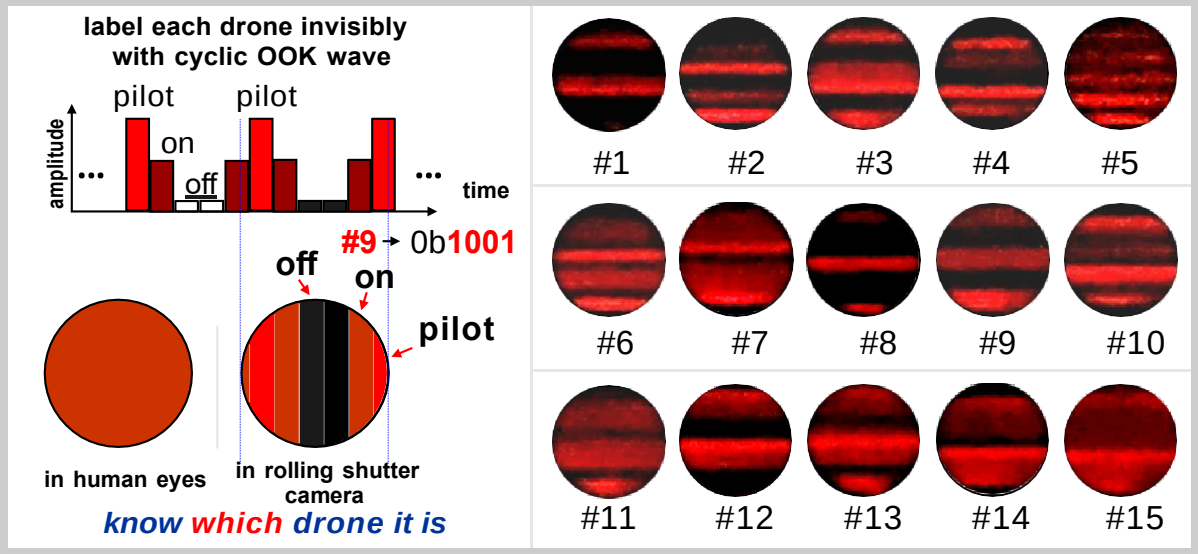


Figure 6.5 Rolling strip effect and cyclic CP-based active optical label design: 4 OOK symbols denote up to 16 drones.

is too high such as more than KHz frequency to be sensed by human eyes[124, 2]). The optical label is comprised of two components: **(1) CP (cyclic pilots)**, which begins with one symbol period with adjustable symbol period (strip width) and is used to distinguish an entire optical label, and **(2) indication symbols**, which are made up of four (or more) OOK (On-Off Keying) symbols. There are two amplitude levels besides darkness in the Off symbol, generated by PWM (pulse width modulation) control: the On symbol has a lower brightness than the CP symbol while the CP symbol has the highest brightness.

High Indication Capacity. We embed drone's binary index into OOK indication symbols. The binary number is *1001* when the drone index is 9 with indication symbols of [*On, Off, Off, On*]. The amount of drones in the drone cluster determines how long the indication symbols are. 4 OOK symbols can indicate up to 16 drones. In general, N OOK symbols can represent 2^N numbers for 2^N drones, which is promising for high-capacity indication and identification of drone swarms. Although some drones may be very close and appear in the FOV of the camera at the same time, different optical labels can notify the observing drone who they are.

6.4.2 CNN based Robust Label Parsing

Traditionally, the amplitude threshold was used to decode these optical labels. But it is difficult to configure the threshold dynamically due to drones' nonlinear movement, long distance and the dynamic optical environment. For the following reasons, we adopt convolutional neural network (CNN)-based label parsing in PoseFly to avoid the complexity and decoding overhead: (1) Online identification and offline training can reduce latency for real-time drone label parsing; (2) the CNN model can learn the features in the repeated dark and bright rolling strips even in conditions where it is difficult to distinguish the amplitude of CP and On.

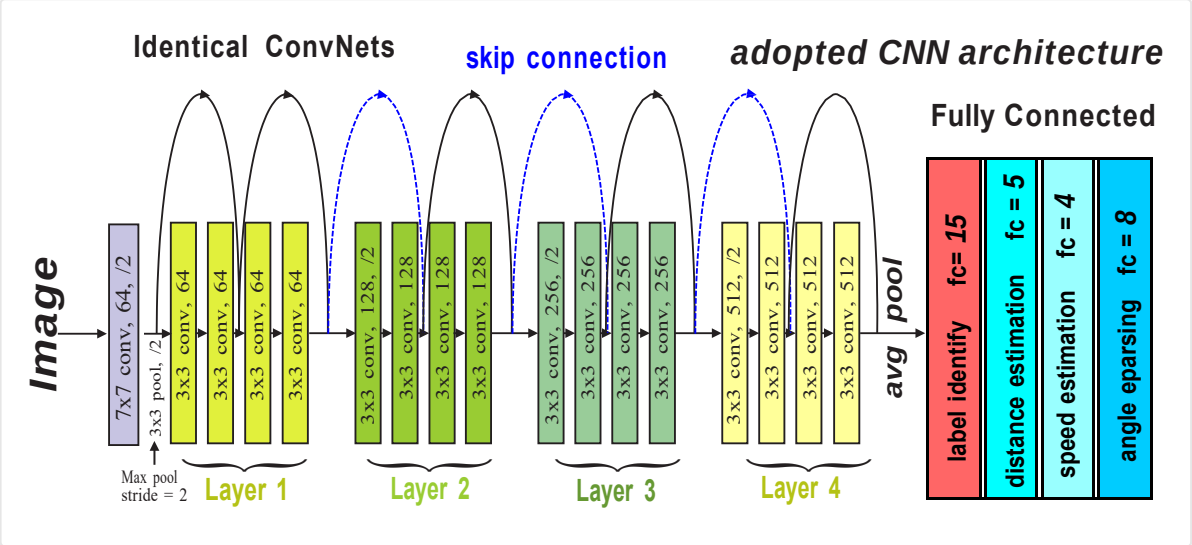


Figure 6.6 Adopted CNN networks in PoseFly: ResNet-18 with modified fully connected layers.

We capture real images of optical labels from 15 drones at various distances in day and night to use as training data. The CNN models adopted in PoseFly shown in Figure 6.6 use the ResNet-18 architecture. They are the Drone Identification Model (DIM), Distance Estimation Model (DEM), Speed Estimation Model (SEM), and Angle Parsing Model models (APM). PoseFly has demonstrated exceptional performance on image classification tasks including [17, 18, 1], which is extremely appropriate for our objective of identifying rolling strip patterns and the created shape with color patterns. The last fully connected layer's output feature is modified to meet the number of options (e.g., 15 in DIM, 5 in DEM, 4 in SEM, and 8 in APM) while keeping other layers the same.

6.5 Drone Localization

The on-site drone localization (pose parsing) in our proposed PoseFly consists of three parts: (1) distance estimation, (2) relative speed estimation, and (3) on-site angle parsing. We present challenges and design details below.

6.5.1 Relative Distance Estimation

For drone localization, the perception and estimation of distance is very important for the interactions among flying drones. For example, accurate estimation of distance between two drones can avoid unexpected collisions and keep the specific flight formations similar to geese flying for complex collaboration and tasks. The quadrangle generated by the four LED spots in our PoseFly transmitter can give another drone a rough sensing of the distance between themselves. We use the rough size of the captured quadrangle of drone to infer the current relative distance between two drones.

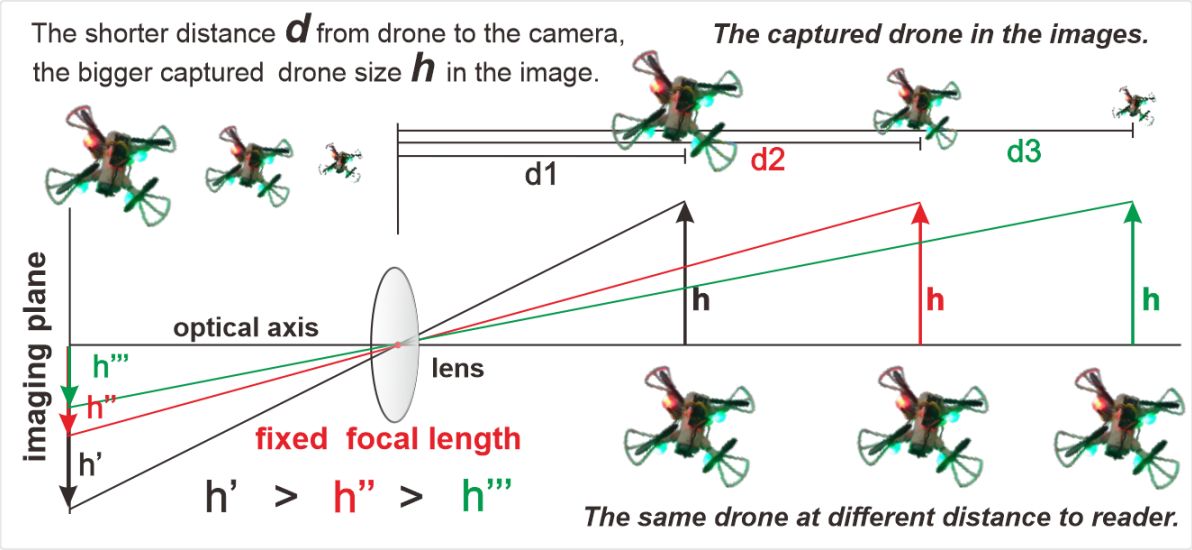


Figure 6.7 Distance estimation via perspective principle: longer distance, smaller captured drone size.

As shown in the bottom of Figure 6.7, we can estimate the distance based on the captured drone size because the drone size increases when the drone is getting closer to the other drone due to the spatial perspective principle. We first collect the captured images (camera is set with fixed

focal length) at different distances and use this data set to train the CNN model for classification offline. Then we can use the trained CNN model to predict and estimate the current relative distance between two drones in real-time.

To filter out the strong ambient light and emphasize the 4 colored spots, we set the rolling shutter with a high shutter speed such as 4000 Hz in our experiments. In our current version of PoseFly, we set 5 distances: 4m, 8m, 12m, 16m, and 20m. The captured quadrangles in day and night with random poses are shown in Figure 6.12 (c).

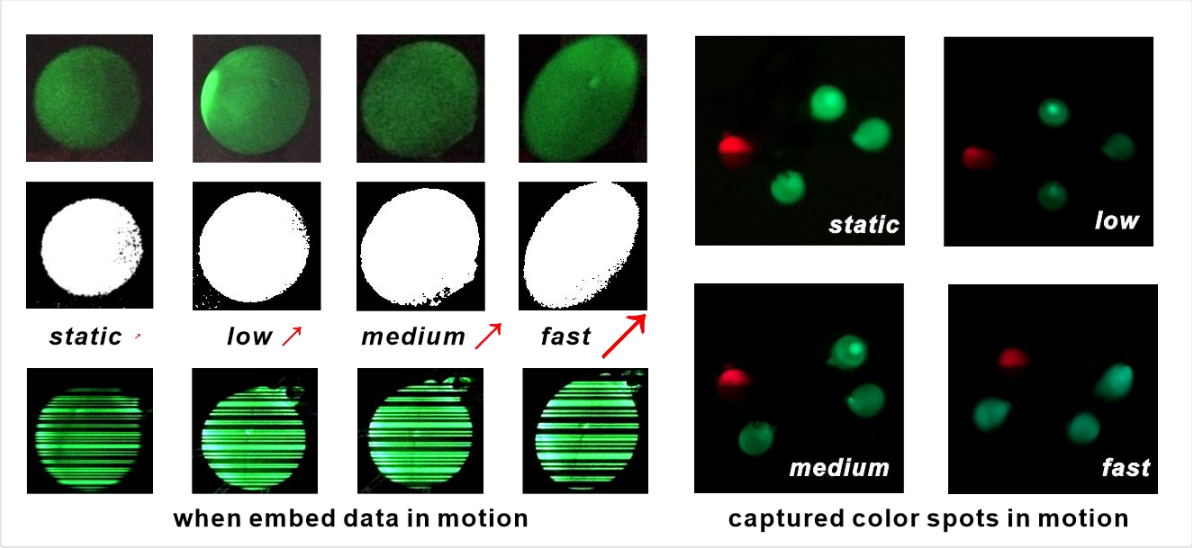


Figure 6.8 Relations with motion speed and varied spot shapes: fast the speed, larger shape variation of the spot.

6.5.2 Relative Speed Estimation

The same as distance estimation, the drone speed is critical for drones' collaboration and accident avoidance. In PoseFly, we exploit our discovered relation among motion speed and the varied shape of the spot generated by one of four LEDs.

First, we explore the relation between different motion speed and the captured spot shape at the same distance between the camera and the light source. We set different motion speeds of the light source to simulate the drone's different motion speed and capture the shape of generated spot. As shown in Figure 6.8, we set 4 levels of movement speed of the light source (i.e, static, low, medium,

and fast) and move the light source with the same movement path (↗) without movement in the front and back direction, the shape of captured rolling patterns changes. As the speed of the light source increases, the shape morphs from a circle to an oval with speed, so does the length of the ellipse's long axis for both light sources embedding and without embedding data.

In PoseFly, we captured images of the shapes of each spot generated by four LEDs speed estimation within 4m. To make the SEM more robust, we capture these images in day and night with 4 different motion speeds with random moving paths and used as training dataset for SEM.

6.5.3 Relative Angle Parsing

We model the drone as a rigid body and use the four LEDs to denote the plane of the bottom plane of the drone. The red LED is mounted at the left-front corner of a drone and it can be treated as the positioning element to denote the facing angle of the drone.

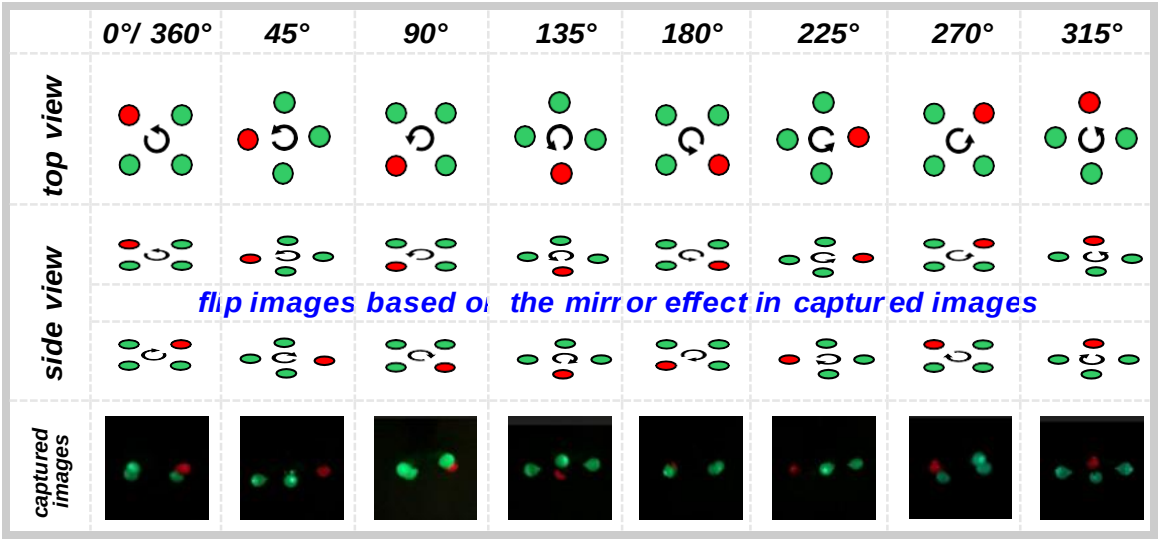


Figure 6.9 On-site angle parsing via colored-arc variation.

As shown in Figure 6.9, we define the relative angle is 0° when the camera captures a drone's tail end. Then the captured red spot rotated 45° in clockwise direction. Using the same rule, we totally define 8 relative angle statues: [0° or 360°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°]. Naturally, we can determine the relative angle of the captured drone based on the position of red spot in the color arc detected in images. However, due to the small size of LED spots in captured

images, it is hard to judge the relative angle. Thus, we employ CNN models to learn relative angle features offline and then predict the relative angle in the captured image in real-time, similar to the AI method used in previous optical label parsing, distance estimation, and relative speed estimation.

Similarly, we set high rolling shutter speed to avoid the ambient light when we capture the images of color arcs. The captured images for training at 4m in day and night are shown in the bottom of Figure 6.9.

6.6 Drone Quick-Link

The sensed postures of nearby drones can be stored locally for the usage of drone itself. At the same time, this posture information can also be shared to nearby drones and extend the communication ranges by using some drones as the relay nodes. Thus, even if some drones are far away or blocked by other drones due to LoS (line-of-sight), they can still communicate with each other. To achieve this goal, we design a quick-link channel for data sharing and communication and present the details of the PoseFly quick-link below.

6.6.1 Modulation Design

Quick-link is one type of OCC, which provides data sharing ability for a small amount of burst data[2]. In PoseFly, we design quick-link to provide a robust optical channel with the similar data rate level (hundreds of bps to several Kbps) besides other 3 functions synchronously. The challenge here is that the captured three green spots are randomly located in a captured image frame due to the high motion of the drone and varied among frames. Thus, even though we successfully recorded the data in one of the three green spots, we are unable to identify which spot it is and cannot eventually complete the correct decoding. Furthermore, different with optical labels, if we adopt PWM and use amplitude shift keying, it will sacrifice the transmission bandwidth and the decreased data rate significantly.

In PoseFly, firstly we can determine which green spot (i.e., L_1, L_2, or L_3) based on the colored arc in captured image. For the modulation in each green spot, we design CP (cyclic preamble) based cyclic OOK data sequences with only bright and dark amplitude levels for robust quick link. The CP takes the same duration with the CP in optical labels. The CP in green spots are dark strips

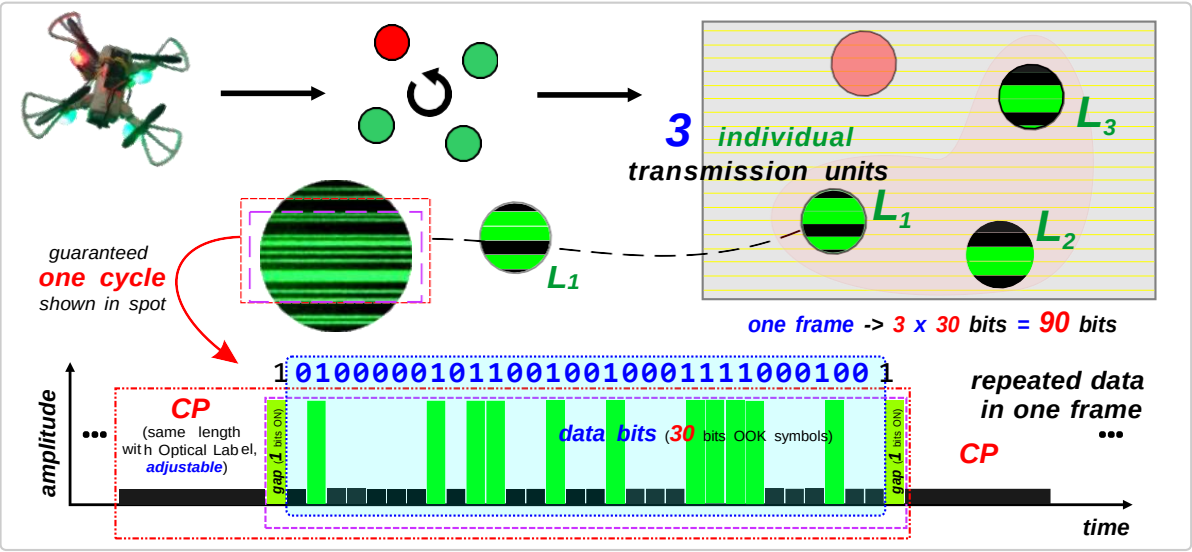


Figure 6.10 Quick link modulation design in PoseFly.

with adjustable width. The symbol length of OOK data sequences is set as 32 bits while setting the beginning symbol and the end symbol as On as gaps between CP and valid data symbols shown in Figure 6.10. The data sequence may contain the same length of dark strips as the CP which may make it hard to recognize the CP during rolling strips. Nevertheless, we can set the CP to have a long symbol length to prevent this from happening to confuse decoding. For example, if we set CP with 8 symbol periods, the possibility of the inside data sequence containing 8 continuous Off symbols is (30-8) / $C_{30}^{8} \approx 4\text{x}10^{-6}$, which is low enough for potential conflicts. Thus, we set the CP as 8 continuous Off symbols. The data amount embedded in each spot depends on how many rolling strips are in it and total data amount in one image frame is the sum of number of strips in all three spots. In each frame, we embed the different data into three green LEDs and choose proper symbol duration of OOK and CP to guarantee there is over one entire cyclic CP and data sequence in one spot.

To robustly detect the data symbols between CP, PoseFly performs quick link communication within 4m. As shown in Figure 6.11, whatever the position of the three spots is in a captured frame with different motion, the strips are clear. So, using the three transmission units that were recorded in each frame, we could collect the data from each green spot and then reconstruct the bit stream. Finally, the data is transferred via the quick link provided by PoseFly, frame by frame.

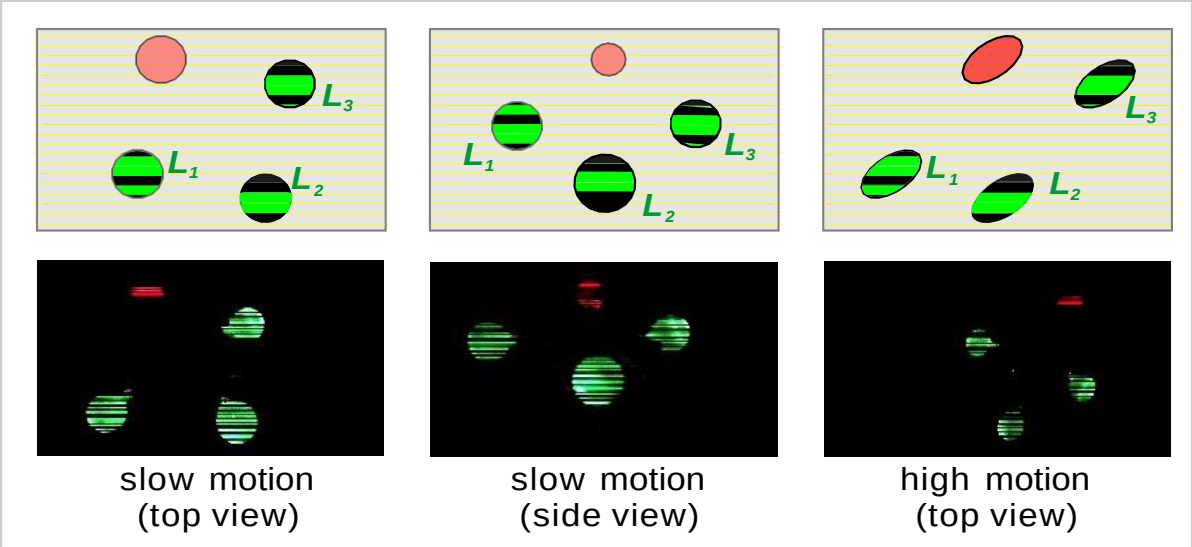


Figure 6.11 Robust symbol detection when drones are flying.

In our prototype, each image frame embeds 30×3 (the number of spots) = 90 valid OOK data symbol (i.e., 90 bits). And the camera frame rate is set as 60 frame per second, the quick link in our proposed PoseFly can achieve the 60×90 = 5400 bits per second data rate, which is 5.4 Kbps, enough for quick link communication among drones to send commands, urgent messages, pose information of drones.

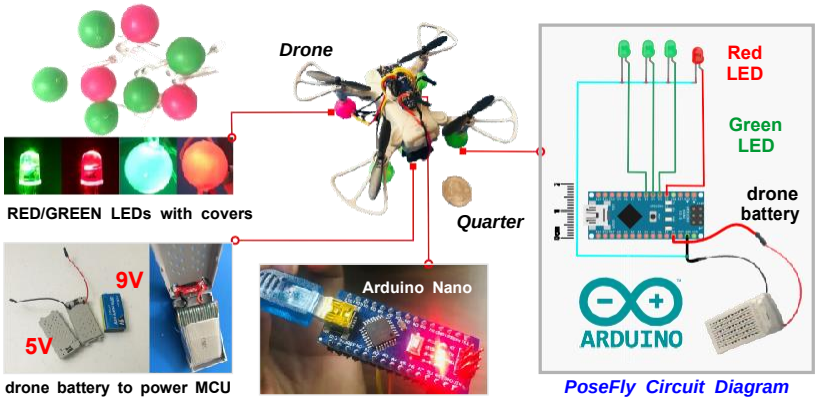
6.7 Implementation and Evaluation

6.7.1 Transmitter

We implement the PoseFly transmitter prototype for experiments as shown in Figure 6.12. The main components in one PoseFly prototype are shown in Table 6.1: entry-level drone, 1 Arduino Nano MCU, 1 red and 3 green LEDs wrapped with 1 red and 3 green plastic balls (ϕ = 19mm). The total weight of added components in PoseFly except the drone is **25g** (we use the battery of drone itself for powering the Arduino Nano) while the total price except the drone is only about **12**$.

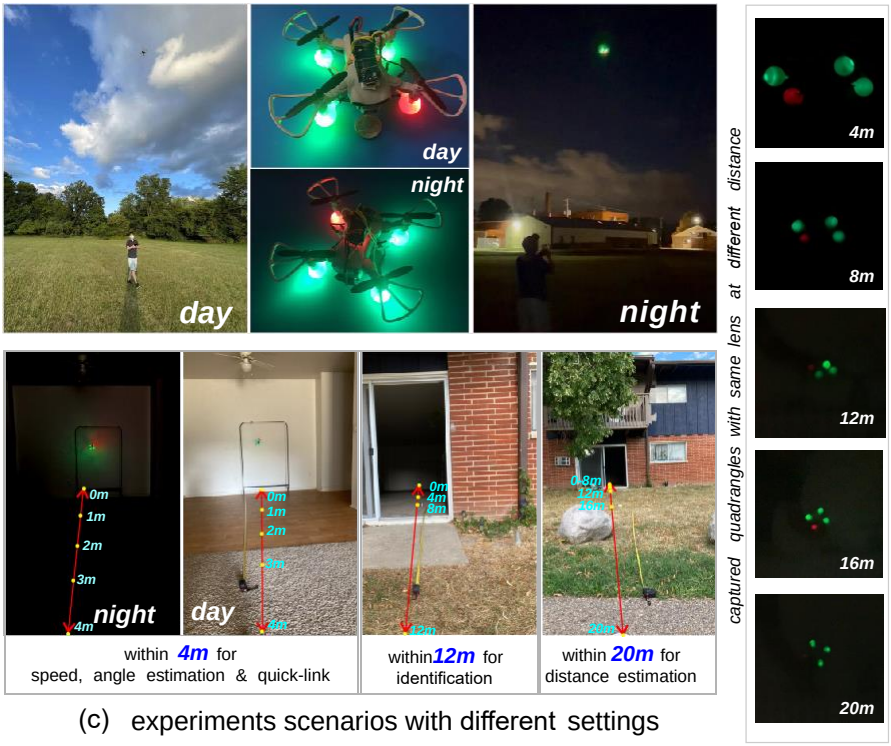
6.7.2 Receiver

There are numerous commercial smart devices that can be used as the PoseFly reader in our prototype. As shown in Figure 6.12 (b), these commercial camera devices are widely available and reasonably priced such as VIVO Y71A, and the iPhone 7 we used. To extend the distance for



(a) Transmitter design in PoseFly and implementation

(b) Receiver design in PoseFly and used additional lens



(c) experiments scenarios with different settings

Figure 6.12 PoseFly implementation including transmitter (a) and receiver (b). The experiment scenarios and setup (c).

usage of PoseFly, we use commercial portable lens for smartphone photographing, the price of the lens we used is about 5$. This universal 20x lens can capture the clear images of objects in long distance. In real use scenarios, PoseFly receivers are the mounted cameras similar to cameras in our prototype.

Component	Price ($)	Details
entry-level drone	40	size: 14cm x 14cm, 125g
Arduino Nano	10	ATmega328P, 5V, 16M
LED	0.1	5mm, gree/red, 20000mcd, 20mA
plastic cover	0.3	19mm, green, lightweight
portable lens	≈ 8	Bostionye 20x mobile lens
Total price	< 60	mass produced, cheaper the price

Table 6.1 Components in PoseFly.

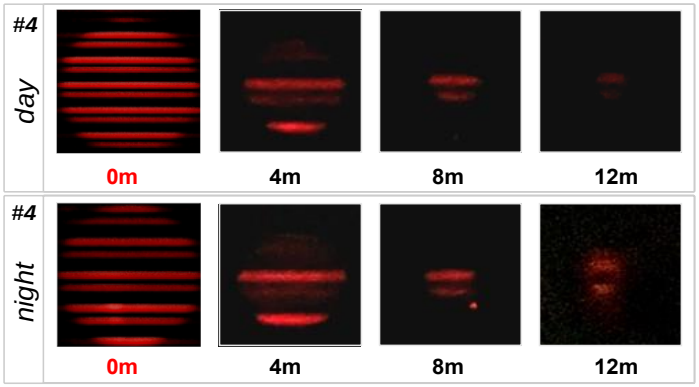
6.7.3 Setup

Drone size. The drone used in our prototypes is tiny sized: 14cm×14cm. In the future, we can equip PoseFly to bigger drones (e.g., 1m×1m) to have better performance such as longer distance and higher data rate because of stronger LED power and higher number of strips shown in LED spot.

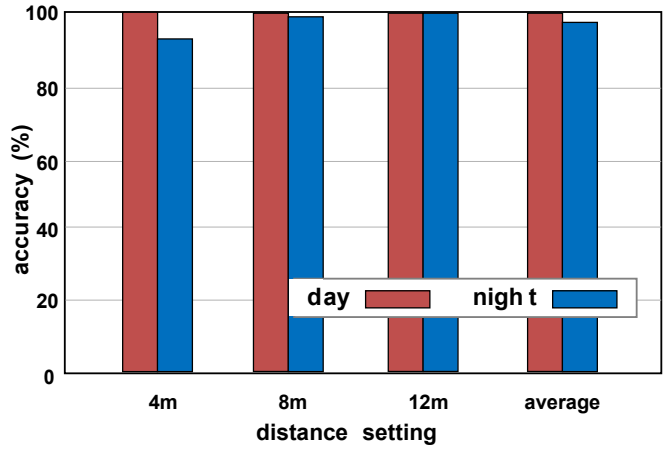
Different optical environment. Figure 6.12 (c) shows the scenarios of our implemented PoseFly transmitter flying in two environment (day and night). Figure 6.12 (c) also shows the experiment scenarios in day and night with different distance.

Simulate the drone flying. In our experiments, we hold the drone in hand or hang it on a hanger and simulate it is flying with different distances, angles, and speeds to the PoseFly receiver (smartphone) in day and night.

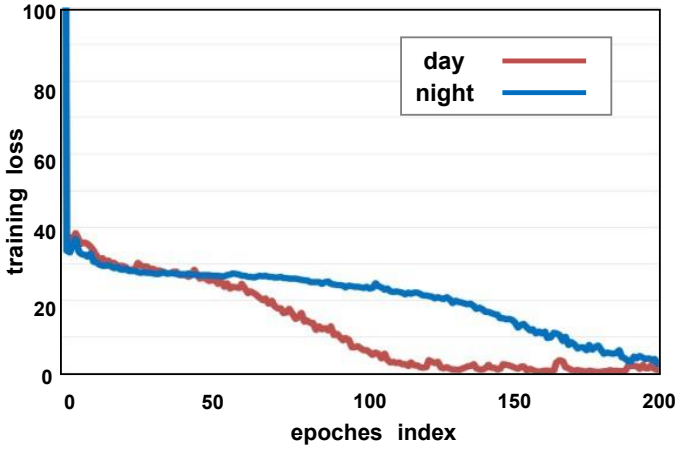
We evaluate PoseFly's performance based on our implemented testbed in three folds: (1) the drone identification accuracy performance, (2) the drone localization accuracy performance including distance, speed, and angle estimation, (3) quick-link performance. Finally, we measure the computation overhead and running time caused latency for each function and make comparisons among PoseFly and the state-of-the-art approaches.



(a) captured optical label #4 at different distance



(b) identification accuracy of optical labels



(c) training loss curves for day and night

Figure 6.13 Drone identification: (a) captured optical labels of #4 in different distance, (b) optical label identification accuracy in both day and night, (c) training loss curves in epoches from [0, 200] in both day and night.

6.7.4 Identification Accuracy

In our experiment, we evaluate the identification accuracy of 15 active optical labels with index number in range of [1, 15]. We capture the optical labels shown in the red LED spot at 3 distance settings: 4m, 8m, and 12m in both day and night time with random postures of the drone.

We capture 10 images for each setting (a specific optical label, a specific distance, day/night setting), thus totally $10\times15\times3\times2 = 900$ images as training dataset. The sampled images of label #4 are shown in Figure 6.13 (a). We evaluate the label identification accuracy performance at day and night, and their training loss in [0, 200] epochs.

Although the number of strips displayed on the cover become less with the increased distance from the drone to the camera and hard for recognizing by human eyes as shown in Figure 6.13 (a), the cyclic rolling pattern is still good enough for CNN to be classified which is demonstrated by Figure 6.13 (b). The identification accuracy of 15 optical labels achieves average 100% in day time and more than 97% at night. The training loss curve for data set of day time drops faster and earlier than the night as shown in Figure 6.13 (c). The reason is that it is harder to distinguish amplitudes between CP and On symbols at the night due to the fusion of optical signals.

6.7.5 Localization Accuracy

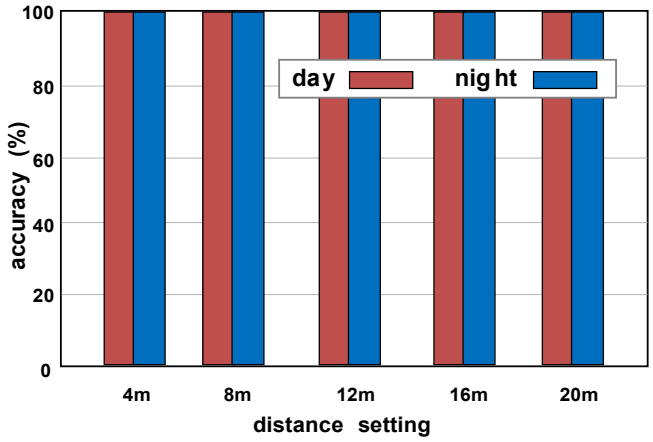
(1) Distance Estimation.

We evaluate the distance estimation accuracy of 5 settings in [4m, 8m, 12m, 16m, 20m]. We capture the spot shape of the drone with random postures and speed in both day and night time. We capture 10 images for each setting (a specific distance, day/night setting), thus totally $10\times5\times2 = 100$ images as the training dataset.

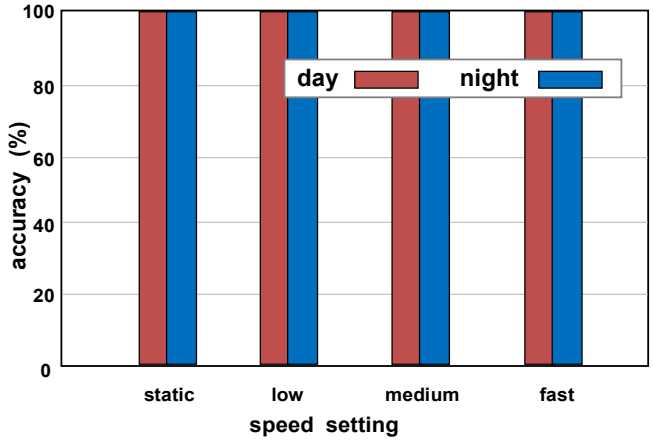
As shown in Figure 6.14 (a), the distance estimation accuracy during day time among all distance settings achieves 100%, which demonstrates our PoseFly can provide within 20m distance ranging among drones in day time. Similarly, PoseFly also works well for distance estimation at night with 100% accuracy within 20m.

(2) Relative Speed Estimation.

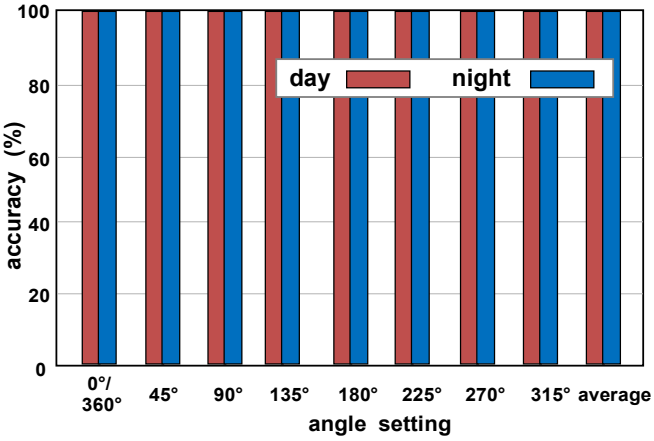
We evaluate the speed estimation accuracy of 4 settings in [static, low, medium, fast]. We



(a) distance estimation performance



(b) relative speed estimation performance



(c) relative angle estimation performance

Figure 6.14 Drone localization performance: (a) distance estimation accuracy, (b) speed estimation accuracy, (c) angle estimation accuracy in both day and night with models saved at 200^{th} epoch.

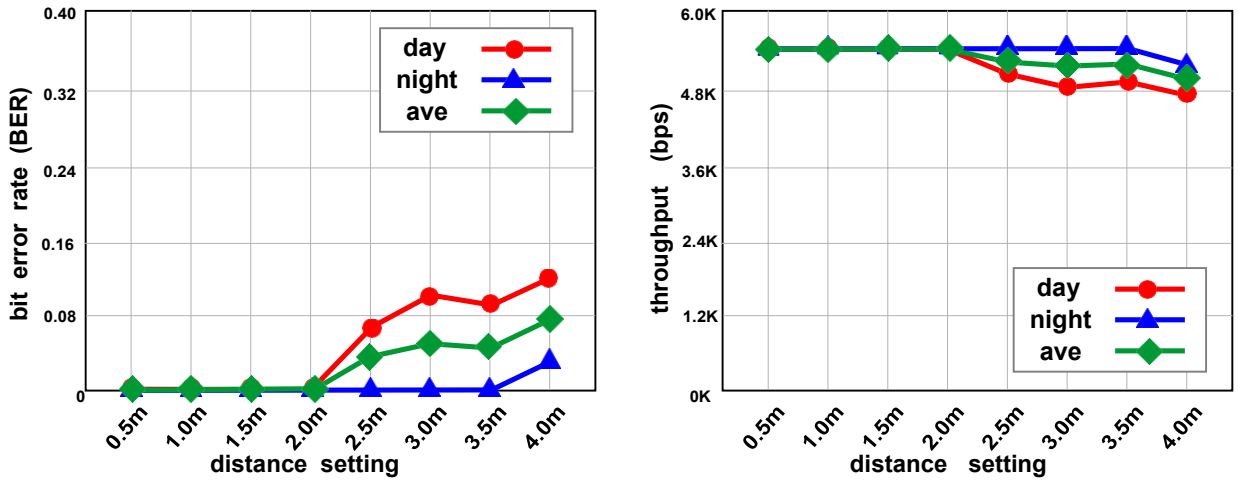
capture the spot shape of drone with random postures at 4m both day and night time. We capture 10 images for each setting (a specific speed, day/night setting), thus totally 10×4×2=80 images as the training dataset.

As shown in Figure 6.14 (b), the speed estimation accuracy during the day time among all four speed settings achieves 100% for both day and night, which demonstrates our PoseFly can provide robust relative speed estimation among drones.

(3) Relative Angle Parsing.

We evaluate the relative angle estimation accuracy of 8 settings in [0° or 360°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°]. We capture the spot shape of the drone with random speed at 4m both day and night time. We capture 10 images for each setting (a specific relative angle, day/night setting),thus totally 10×8×2 = 160 images as the training dataset. As shown in Figure 6.14 (c), the CNN model saved at 200^{th} epoch can classify the drones with different relative angles of 8 options accurately for both day and night with estimation accuracy of 100% within 4m sensing distance.

To sum up, our AI-assisted drone pose parsing/localization works well for all three aspects during day and the night in different distances for the flying drones.



(a) quick-link bit error rate (b) quick-link throughput

Figure 6.15 Quick link performance: (a) BER, and (b) throughput.

6.7.6 Quick-link Evaluation

We evaluate the Quick-link performance within 4m (0.5m, 1m, 1.5m, 2m, 2.5m, 3m, 3.5m, 4m) in both day and night. We set the shutter speed properly (12Khz) with transmission frequency to capture clear rolling strips shown on the three green spots in each frame and set the frame rate as 60FPS. For each setting (a specific distance, day/night setting), we capture 10 images, thus totally 8×2×10 = 160 images to measure its BER and achieved throughput.

BER performance. We decode OOK data sequence inside of two CPs. As shown in Figure 6.15 (a), the bit error rate in each frame is **0** within **2m** for both day and night. With the increased distance, the BER increased as well due to the weaker optical signals at longer distances. Nevertheless, our prototype still achieves the average BER less than **0.08** at **4m**. The reason the BER in day is higher than the BER in the night is that the lower amplitude gap of captured On symbols and Off symbol in day due to the strong ambient light than at the night for the same distance.

Throughput performance. The valid data bits in each frame is the sum of valid data in three green spots, which is calculated by 30 bits (32-2)×3×frame rate (60 FPS)×BER. As shown in Figure 6.15 (b), our PoseFly achieves **5.4 Kbps** within **2m** for both day and night. Although the throughput drops with increased transmission distance, the dropped data amount is limited. Even at **4m**, our PoseFly still achieves the average throughput over **5 Kbps**. Although the captured spot

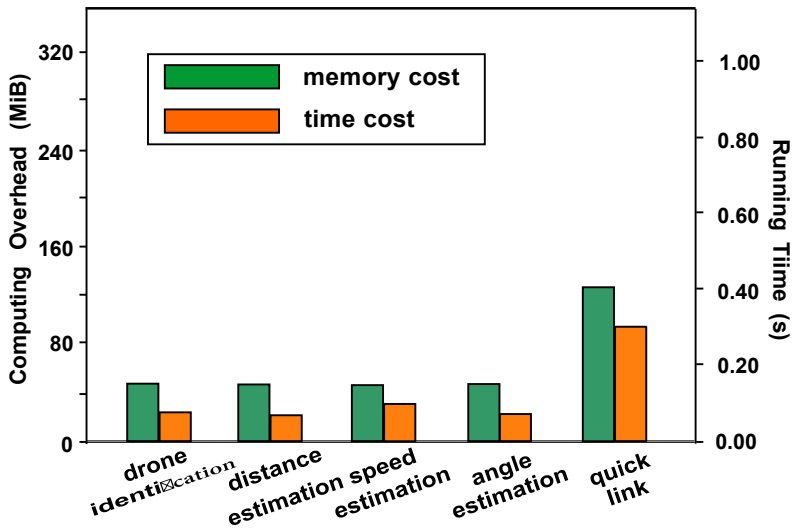


Figure 6.16 Computation cost and latency evaluation.

size is decreased by the increased distance, we can still capture the complete and differentiable strips at 4m with lens.

6.7.7 Overhead

Computation overhead. For drones, battery is limited. LEDs provide lighting function and are energy efficient. Thus, we only consider the computation overhead at reader side. The reader should not conduct complex computations and consume energy too fast. The training processes are offline, the drone identification, distance, speed, and angle estimations are real-time tasks conducted with few computation cost for each step by step when the drone is flying. As shown in Figure 6.16, the quick link requires the most memory resources due to more narrow strips in decoding compared with CNN based tasks mentioned above. For all these tasks, they require a combined 313 MiB of memory and is not a computational burden for a commercial smart device.

Latency. For collaboration tasks among drones, time can be important to improve the efficacy and efficiency. Compared with state-of-art drone localization systems, including audio-based systems, PoseFly has nearly no time delay in signal propagation due to the fast propagation of light. Thus we only consider the computational latency. As shown in Figure 6.16, the drone identification, drone on-site localization (distance, speed, angle estimation) have the low running time of about 0.07 s - 0.09s for each. These functions can be run in pipeline manner (i.e., totally 0.07s-0.09s) and thus achieve the real-time on-site pose parsing. For example, given two drones with 20m/s relative speed, after drone A completes its pose paring function for drone B, the parsed distance may only have 20m/s$\times$0.09 = 1.8 m distance estimation error. The distance estimation in PoseFly is designed for discrete distance ranges [4m, 8m, 12m, 16m, 20m], and 1.8m distance estimation error is acceptable and practical. Different with real-time on-site drone pose parsing, the quick link function is designed for information sharing (e.g., roughly which drones are nearby, some broadcast commands) if needed which is not strictly require real-time communication. Thus, 0.31 s is acceptable, which is similar to the collaborations among geese.

6.8 Discussion and Summary

Comparison with Existing Work. *(1) Passive optical label.* Compared with passive optical label such as bar code and QR codes[111] with the similar size (2cmx2cm) as the red cover in our prototype, we measured that these passive optical labels are only workable within 50cm. *(2) RF based localization.* RF based localization can provide distance estimation error within about several meters with a localization time of more than 70 seconds while not providing other aspects of drone pose parsing in our PoseFly such as angle and speed estimation[92]. *(3) RF/OCC communication.* RF techniques can provide long communication distance, however, they face the severe interference when there are massive drones. Existing OCC approaches can achieve similar several Kbps throughput ability, however, they did not provide optical labeling, and on-site localization functions[38, 37].

Other Concerns. *(1) discrete value.* Current PoseFly provides discrete relative localization instead of continuous relative distance/angle/speed value. However, PoseFly is designed for swarming drones' collaboration which does not require the exact value of relative positioning, the similar to the geese flying. *(2) modulated ambient light.* Although there are modulated light such as LiFi (>100KHz) transmitters, our PoseFly can filter them out them via spatial diversity of millions of camera pixels and different frequency (about 10 KHz). *(3) frame gap loss.* The transmitted data in quick-link channel are repeated for broadcast and thus the frame gaps caused data loss will not impact the final decoded data.

In summary, we propose PoseFly for simple and robust on-site drone pose parsing via optical camera communication. We design a color-arc scheme and investigate spatial embedding ability of rolling shutter cameras and first exploit it for drone localization including relative distance, speed, and angle estimations. Besides, we design active optical labels with cyclic pilot and data sequences in frame-level for high-capacity drones indication and quick-link communication for real-time and smooth collaborations among drones. Finally, we conduct experiments on implemented prototype in various scenarios. The solid experiments show that our PoseFly can achieve near 100% accuracy for drone identification at up to 12m, 100% drone localization as well as 5 Kbps average data rate

with average BER lower than 0.08 at up to 4m for both day and night. These results demonstrate our PoseFly works well.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

Because of the rapid growth of the limited and crowded RF bandwidth for high-speed wireless communication services, there has been a boom in research and industry interest in optical wireless communication (OWC). The new technology ushers in a new potential world of fast and ubiquitous wireless communications and enables integrated sensing and communication, as well as new challenges in developing OWC techniques.

First, we propose LiFOD to improve the data rate in LiFi system. We exploit Compensation Symbols previously only used for dimming to indicate bit patterns in modulation as dimming side-channel. We addressed challenges including greedy bit pattern mining, compensation redesign and relocation. LiFOD utilizes 1D temporal diversity in data embedding.

Second, we propose RainbowRow to boost the data rate of optical camera communication. We exploit 2D rolling blocks in optical imaging to transmit more bits for each optical symbol. By redesigning the transmitter with linear LED bulbs and addressing optical singals' interference, RainbowRow achieves 20× data rate improvement than the existing OCC systems.

Third, we embed data bits with the 3D spatial manner to overcome the limitations of existing passive optical tags. U-Star is a cost-effective and practical underwater self-navigation solution for large-scale applications. We utilize deep learning and color-arc designs to address challenges such as underwater denoising, relative positioning, and robust decoding.

Besides communication, we also exploit 3D spatial-temporal diversities for optical wireless sensing in RoFin. We design low-cost RoFin gloves with 6 key points and utilize rolling shutter effect to construct the hand pose in real time. Our proposed RoFin can also provide fine-grained finger tracking for numerous applications such as virtual writing for Parkinson suffererss.

Finally, we propose PoseFly, which utilizes 4D (3D spatial with 1D temporal) diversities for on-site pose parsing of drones. PoseFly is designed as a low-cost, but effective integrated optical sensing and communication framework for large-scale drone networks with 4 functions, including

massive drone indication, quick-link channel, lighting, and multi-level drone positioning.

These studies and outcomes from our implemented prototypes validate the ideas we advocated. These results demonstrate that our explored multi-dimensions of spatial-temporal diversities in optical wireless communication can indeed improve the performance of OWC systems. Our work may enable numerous applications in the future as the promising techniques fornext generation wireless communication and networks.

7.2 Ongoing Work

Following the explored projects, the ongoing project we are working on is HotSys, which focuses on systems of holographic optical tags for scalable and collaborative mobile infrastructures. Most existing OWC based techniques for vehicular mobile systems adopt omnidirectional beamforming. This requires strict beam alignment, which leads to a limited communication field of view and lacks relative positioning capabilities [130]. Therefore, we propose HotSys, a system of Holographic Optical Tags to overcome this limitation to support scalable and collaborative mobile systems, which may include Vehicle to Everywhere (V2X) Systems, as shown in Figure 7.1.

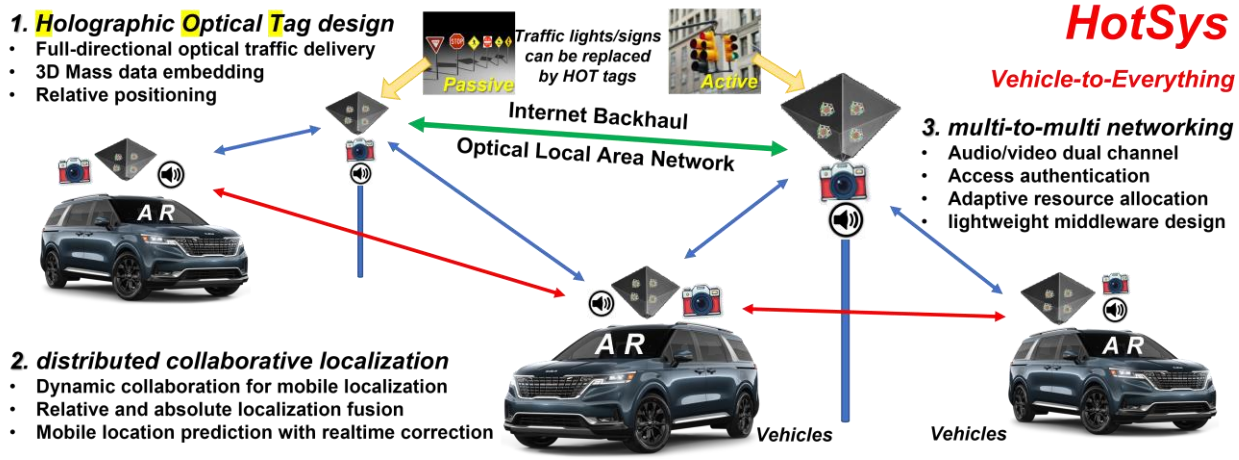


Figure 7.1 Research Objectives of HotSys: (1) holographic optical tag design, (2) distributed collaborative localization, (3) middleware design for multi-to-multi networking.

HotSys tags are virtual 3D tags embedded with data and positioning elements in 3D space. The images of a HotSys virtual 3D tag are delivered in multiple directions via a new multi-

direction reflector. The HotSys tags attach to individual vehicles for simultaneous multi-to-multi communications (i.e., multi-to-multi communications means that a node can transmit to and receive from multiple directions at the same time, as shown in Figure 7.1). Multi-to-multi communications using the HotSys tags will not require beam alignment concerns and therefore exploit data embedded in 3-dimensional space for fast and robust data transmission. The system will include middleware to enable collaborative positioning identification of the mobile vehicles within the system. As a result, HotSys tags on the vehicles will be composed into a distributed system to construct a reliable and accurate localization system and a scalable collaborative communication system. The prototype of HotSys tag is shown in Figure 7.2.

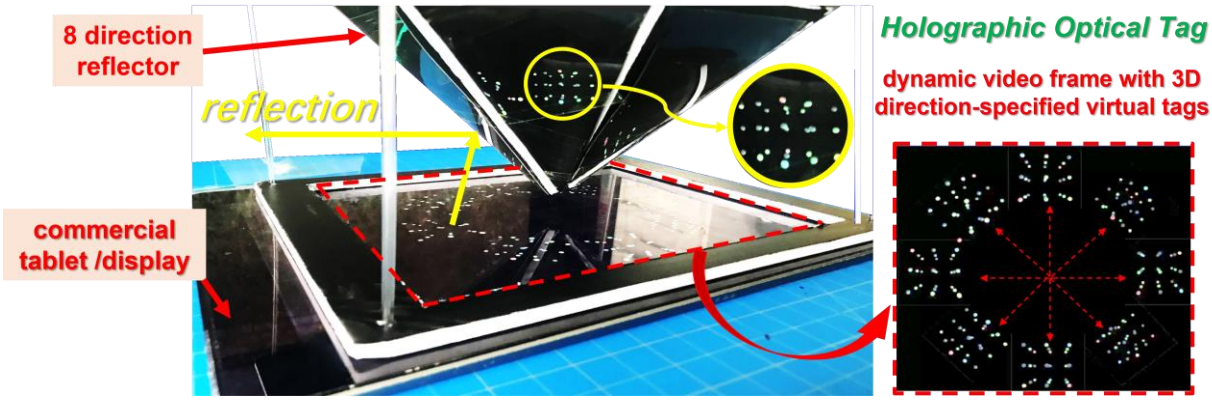


Figure 7.2 The design illustration of Holographic Optical Tags.

7.3 Future Work

In the future, our research will continue to explore the multi-dimensions of spatial-temporal diversities to further enhance optical wireless communication (OWC) and enable novel OWC sensing techniques. These advancements have a wide range of potential application scenarios, including cellular connectivity, smart homes, V2X communication, underwater communication, e-health, space communication, smart shopping, and more. However, it is important to note that while these applications mainly focus on the user side, we must also pay attention to the infrastructure side. There are related technologies, such as data center optical networks, virtualized radio access networks, MIMO (Multiple Input Multiple Output)[50], Full-Duplex spectrum[5],

beamforming[119], and smart surfaces[22], which form the backbone and foundation to enable and support the diverse applications mentioned above.

Integrating research efforts in both user-side applications and advanced infrastructure technologies is crucial to fully harness the potential of optical wireless communication and achieve next-generation wireless networks. By focusing on both aspects, we can create a comprehensive ecosystem that addresses the challenges and opportunities of optical wireless communication.

On the user-side, exploring diverse application scenarios and developing innovative solutions for areas like smart homes, underwater communication, and e-health will lead to practical implementations of optical wireless communication in everyday life. Simultaneously, advancing infrastructure technologies such as data center optical networks, virtualized radio access networks, MIMO, Full-Duplex spectrum, beamforming, and smart surfaces will provide a strong foundation to support the increasing demands of optical wireless communication networks.

Combining these research efforts will lead to a well-rounded and future-proof approach to optical wireless communication, enabling efficient and reliable wireless communication systems that cater to the diverse needs of modern society. This integration will play a vital role in shaping the next-generation wireless landscape and unlocking new possibilities for communication and connectivity.

BIBLIOGRAPHY

[1] COCO. https://paperswithcode.com/dataset/coco, 2014.

[2] Ieee standard for local and metropolitan area networks–part 15.7: Short-range optical wireless communications. *IEEE Std 802.15.7-2018 (Revision of IEEE Std 802.15.7-2011)*, pages 1–407, April 2019.

[3] Yun Ai, Aashish Mathur, Gyan Deep Verma, Long Kong, and Michael Cheffena. Comprehensive physical layer security analysis of fso communications over málaga channels. *IEEE Photonics Journal*, 12(6):1–17, 2020.

[4] Farhad Akhoundi, Amir Minoofar, and Jawad A Salehi. Underwater positioning system based on cellular underwater wireless optical cdma networks. In *2017 26th Wireless and Optical Communication Conference (WOCC)*, pages 1–3. IEEE, 2017.

[5] Muhammad Amjad, Fayaz Akhtar, Mubashir Husain Rehmani, Martin Reisslein, and Tariq Umer. Full-duplex communication in cognitive radio networks: A survey. *IEEE Communications Surveys & Tutorials*, 19(4):2158–2191, 2017.

[6] Lorenzo Bertizzolo, Salvatore D'Oro, Ludovico Ferranti, Leonardo Bonati, Emrecan Demirors, Zhangyu Guan, Tommaso Melodia, and Scott Pudlewski. Swarmcontrol: An automated distributed control framework for self-optimizing drone networks. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, pages 1768–1777, 2020.

[7] Azzedine Boukerche and Peng Sun. Design of algorithms and protocols for underwater acoustic wireless sensor networks. *ACM Computing Surveys (CSUR)*, 53(6):1–34, 2020.

[8] Yujun Cai, Liuhao Ge, Jianfei Cai, and Junsong Yuan. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 666–682, 2018.

[9] Charles J. Carver, Zhao Tian, Hongyong Zhang, Kofi M. Odame, Alberto Quattrini Li, and Xia Zhou. Amphilight: Direct air-water communication with laser light. *GetMobile: Mobile Comp. and Comm.*, 24(3):26–29, January 2021.

[10] Darren Caulfield and Kenneth Dawson-Howe. Direction of camera based on shadows. In *Proceedings of the Irish Machine Vision and Image Processing Conference*, pages 216–223. Citeseer, 2004.

[11] Nan Cen, Neil Dave, Emrecan Demirors, Zhangyu Guan, and Tommaso Melodia. Libeam: Throughput-optimal cooperative beamforming for indoor visible light networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 1972–1980. IEEE, 2019.

[12] Weiya Chen, Chenchen Yu, Chenyu Tu, Zehua Lyu, Jing Tang, Shiqi Ou, Yan Fu, and Zhidong Xue. A survey on hand pose estimation with wearable sensors and computer-vision-based methods. *Sensors*, 20(4):1074, 2020.

[13] Ramesh Kumar Chidambaram and Rammohan Arunachalam. Automotive headlamp high power led cooling system and its effect on junction temperature and light intensity. *Journal of Thermal Engineering*, 6(6):354–368, 2020.

[14] Mostafa Zaman Chowdhury, Moh Khalid Hasan, Md Shahjalal, Md Tanvir Hossan, and Yeong Min Jang. Optical wireless hybrid networks: Trends, opportunities, challenges, and research directions. *IEEE Communications Surveys & Tutorials*, 22(2):930–966, 2020.

[15] Mostafa Zaman Chowdhury, Md Tanvir Hossan, Amirul Islam, and Yeong Min Jang. A comparative survey of optical wireless technologies: Architectures and applications. *IEEE Access*, 6:9819–9840, 2018.

[16] CIE. chromaticity diagram, 1931.

[17] CIFAR10.https://paperswithcode.com/sota/image-classification-on-cifar-10, 2009.

[18] CIFAR100.https://paperswithcode.com/sota/image-classification-on-cifar-100, 2009.

[19] Minhao Cui, Yuda Feng, Qing Wang, and Jie Xiong. Sniffing visible light communication through walls. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pages 1–14, 2020.

[20] Minhao Cui, Qing Wang, and Jie Xiong. Breaking the limitations of visible light communication through its side channel. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pages 232–244, 2020.

[21] Francisco J. Escribano, José Sáez Landete, and Alexandre Wagemakers. Chaos-based multicarrier VLC modulator with compensation of LED nonlinearity. *IEEE Trans. Communications*, 67(1):590–598, 2019.

[22] Roberto Flamini, Danilo De Donno, Jonathan Gambini, Francesco Giuppi, Christian Mazzucco, Angelo Milani, and Laura Resteghini. Toward a heterogeneous smart electromagnetic environment for millimeter-wave communications: An industrial viewpoint. *IEEE Transactions on Antennas and Propagation*, 70(10):8898–8910, 2022.

[23] Ander Galisteo, Diego Juara, and Domenico Giustiniano. Research in visible light communication systems with openvlc1. 3. In *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, pages 539–544. IEEE, 2019.

[24] Ander Galisteo, Qing Wang, Aniruddha Deshpande, Marco Zuniga, and Domenico Giustiniano. Follow that light: Leveraging leds for relative two-dimensional localization. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*, pages 187–198, 2017.

[25] Jazmine Gaona and Ray Oltion. Natural navigation. 2013.

[26] Jun Gong, Yang Zhang, Xia Zhou, and Xing-Dong Yang. Pyro: Thumb-tip gesture recognition using pyroelectric infrared sensing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pages 553–563, 2017.

[27] GSA. U.s. general services administration, 6.15 lighting. https://www.gsa.gov/node/82715, 2021.

[28] GSA. Verified market research. https://www.verifiedmarketresearch.com/product/drones-market/, 2022.

[29] Lav Gupta, Raj Jain, and Gabor Vaszkun. Survey of important issues in uav communication networks. *IEEE Communications Surveys & Tutorials*, 18(2):1123–1152, 2015.

[30] Harald Haas, Liang Yin, Yunlu Wang, and Cheng Chen. What is lifi? *Journal of lightwave technology*, 34(6):1533–1544, 2015.

[31] MA Hadi. Wireless communication tends to smart technology li-fi and its comparison with wi-fi. *American Journal of Engineering Research (AJER)*, 5(5):40–47, 2016.

[32] C Haldoupis and K Schlegel. Characteristics of midlatitude coherent backscatter from the ionospheric e region obtained with sporadic e scatter experiment. *Journal of Geophysical Research: Space Physics*, 101(A6):13387–13397, 1996.

[33] Richard W Hamming. Error detecting and error correcting codes. *The Bell system technical journal*, 29(2):147–160, 1950.

[34] J. Hao, Y. Yang, and J. Luo. Ceilingcast: Energy efficient and location-bound broadcast through led-camera communication. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, 2016.

[35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[36] Justin Hu, Ariana Bruno, Drew Zagieboylo, Mark Zhao, Brian Ritchken, Brendon Jackson, Joo Yeon Chae, Francois Mertil, Mateo Espinosa, and Christina Delimitrou. To centralize or not to centralize: A tale of swarm coordination. *arXiv preprint arXiv:1805.01786*, 2018.

[37] P. Hu, P. H. Pathak, H. Zhang, Z. Yang, and P. Mohapatra. High speed led-to-camera communication using color shift keying with flicker mitigation. *IEEE Transactions on Mobile Computing*, 19(7):1603–1617, 2020.

[38] Pengfei Hu, Parth H Pathak, Xiaotao Feng, Hao Fu, and Prasant Mohapatra. Colorbars: Increasing data rate of led-to-camera communication using color shift keying. In *proceedings of the 11th ACM conference on Emerging Networking experiments and technologies*, pages 1–13, 2015.

[39] Pei Huang, Jun Huang, and Li Xiao. Exploiting modulation scheme diversity in multicarrier wireless networks. In *2016 13th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pages 1–9. IEEE, 2016.

[40] Neminath Hubballi and Mayank Swarnkar. *bitcoding*: Network traffic classification through encoded bit level signatures. *IEEE/ACM Transactions on Networking*, 26(5):2334–2346, 2018.

[41] RD Hunsucker and HF Bates. Survey of polar and auroral region effects on hf propagation. *Radio Science*, 4(4):347–365, 1969.

[42] Ayesha Ijaz, Lei Zhang, Maxime Grau, Abdelrahim Mohamed, Serdar Vural, Atta U Quddus, Muhammad Ali Imran, Chuan Heng Foh, and Rahim Tafazolli. Enabling massive iot in 5g and beyond systems: Phy radio frame design considerations. *IEEE Access*, 4:3322–3339, 2016.

[43] Tariq Islam and Seok-Hwan Park. A comprehensive survey of the recently proposed localization protocols for underwater sensor networks. *IEEE Access*, 2020.

[44] Mohammad Jahanbakht, Wei Xiang, Lajos Hanzo, and Mostafa Rahimi Azghadi. Internet of underwater things and big marine data analytics—a comprehensive survey. *IEEE Communications Surveys & Tutorials*, 2021.

[45] Fahad Jalal and Faizan Nasir. Underwater navigation, localization and path planning for autonomous vehicles: A review. In *2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST)*, pages 817–828. IEEE, 2021.

[46] Junsu Jang and Fadel Adib. Underwater backscatter networking. In *Proceedings of the ACM Special Interest Group on Data Communication*, pages 187–199. 2019.

[47] Ruhul Khalil, Mohammad Babar, Tariqullah Jan, and Nasir Saeed. Towards the internet of underwater things: Recent developments and future challenges. *IEEE Consumer Electronics Magazine*, 2020.

[48] Jun Sik Kim, Byung Kook Kim, Minsu Jang, Kyumin Kang, Dae Eun Kim, Byeong-Kwon Ju, and Jinseok Kim. Wearable hand module and real-time tracking algorithms for measuring finger joint angles of different hand sizes with high accuracy using fbg strain sensor. *Sensors*, 20(7):1921, 2020.

[49] Aleksandra Kostic-Ljubisavljevic and Branka Mikavica. Challenges and opportunities of vlc application in intelligent transportation systems. In *Encyclopedia of Information Science and Technology, Fifth Edition*, pages 1051–1064. IGI Global, 2021.

[50] Erik G Larsson, Ove Edfors, Fredrik Tufvesson, and Thomas L Marzetta. Massive mimo for next generation wireless systems. *IEEE communications magazine*, 52(2):186–195, 2014.

[51] Hui-Yu Lee, Hao-Min Lin, Yu-Lin Wei, Hsin-I Wu, Hsin-Mu Tsai, and Kate Ching-Ju Lin. Rollinglight: Enabling line-of-sight light-to-camera communications. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, pages 167–180, 2015.

[52] Yongjun Lee, Myungsin Kim, Yongseok Lee, Junghan Kwon, Yong-Lae Park, and Dongjun Lee. Wearable finger tracking and cutaneous haptic interface with soft sensors for multi-fingered virtual manipulation. *IEEE/ASME Transactions on Mechatronics*, 24(1):67–77, 2018.

[53] Bin Li, Zesong Fei, and Yan Zhang. Uav communications for 5g and beyond: Recent advances and future trends. *IEEE Internet of Things Journal*, 6(2):2241–2263, 2018.

[54] Chenning Li, Hanqing Guo, Shuai Tong, Xiao Zeng, Zhichao Cao, Mi Zhang, Qiben Yan, Li Xiao, Jiliang Wang, and Yunhao Liu. Nelora: Towards ultra-low snr lora communication with neural-enhanced demodulation. In *Proceedings of ACM SenSys*, 2021.

[55] Juan Li, Xu Bao, Wance Zhang, and Nan Bao. Qoe probability coverage model of indoor visible light communication network. *IEEE Access*, 8:45390–45399, 2020.

[56] Rui Li, Zhenyu Liu, and Jianrong Tan. A survey on 3d hand pose estimation: Cameras, methods, and datasets. *Pattern Recognition*, 93:251–272, 2019.

[57] Tianxing Li, Chuankai An, Zhao Tian, Andrew T Campbell, and Xia Zhou. Human sensing using visible light communication. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 331–344, 2015.

[58] Tianxing Li, Chuankai An, Xinran Xiao, Andrew T Campbell, and Xia Zhou. Real-time screen-camera communication behind any scene. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, pages 197–211, 2015.

[59] Tianxing Li, Xi Xiong, Yifei Xie, George Hito, Xing-Dong Yang, and Xia Zhou. Reconstructing hand poses using visible light. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):1–20, 2017.

[60] You Li and Javier Ibanez-Guzman. Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Processing Magazine*, 37(4):50–61, 2020.

[61] Jaime Lien, Nicholas Gillian, M Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. Soli: Ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics (TOG)*, 35(4):1–19, 2016.

[62] Chi Lin, Yongda Yu, Jie Xiong, Yichuan Zhang, Lei Wang, Guowei Wu, and Zhongxuan Luo. Shrimp: a robust underwater visible light communication system. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, pages 134–146, 2021.

[63] Huaiyin Lu, Ming Jiang, and Julian Cheng. Deep learning aided robust joint channel classification, channel estimation, and signal detection for underwater optical communication. *IEEE Transactions on Communications*, 69(4):2290–2303, 2020.

[64] Philip Lundrigan, Neal Patwari, and Sneha K. Kasera. On-off noise power communication. In *The 25th Annual International Conference on Mobile Computing and Networking*, MobiCom '19, New York, NY, USA, 2019. Association for Computing Machinery.

[65] Chengcai Lv, Binjian Shen, Chuan Tian, Shengzong Zhang, Liang Yu, and Dazhen Xu. Signal design and processing for underwater acoustic positioning and communication integrated system. In *2020 IEEE 3rd International Conference on Information Communication and Signal Processing (ICICSP)*, pages 89–93. IEEE, 2020.

[66] Nino E Merencilla, Alvin Sarraga Alon, Glenn John O Fernando, Elaine M Cepe, and Dennis C Malunao. Shark-eye: A deep inference convolutional neural network of shark detection for underwater diving surveillance. In *2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, pages 384–388. IEEE, 2021.

[67] Raed Mesleh, Hany Elgala, and Harald Haas. Led nonlinearity mitigation techniques in optical wireless ofdm communication systems. *Journal of Optical Communications and Networking*, 4(11):865–875, 2012.

[68] Micrographia. U.s. general services administration, 6.15 lighting. https://en.wikipedia.org/wiki/Micrographia_%28handwriting%29, 2022.

[69] Muhammad Sarmad Mir, Borja Genoves Guzman, Ambuj Varshney, and Domenico Giustiniano. Passivelifi: rethinking lifi for low-power and long range rf backscatter. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, pages 697–709, 2021.

[70] Olga Mirgorodskaya, Olesya Ivanchenko, and Narine Dadayan. Using digital signage technologies in retail marketing activities. In *Proceedings of the International Scientific Conference-Digital Transformation on Manufacturing, Infrastructure and Service*, pages 1–7, 2020.

[71] András J Molnár. Trailsigner: A conceptual model of hiking trail networks with consistent signage planning and management. In *Information Modelling and Knowledge Bases XXXII*, pages 1–25. IOS Press, 2020.

[72] Mohammed SA Mossaad, Steve Hranilovic, and Lutz Lampe. Visible light communications using ofdm and multiple leds. *IEEE Transactions on Communications*, 63(11):4304–4313, 2015.

[73] N Muraleedharan, Anna Thomas, S Indu, and BS Bindhumadhava. A traffic monitoring and policy enforcement framework for http. In *2020 Third ISEA Conference on Security and Privacy (ISEA-ISAP)*, pages 81–86. IEEE.

[74] Zhang Nan, Zhang Fan, and Enmao Liu. Design of a shared platform for interactive public art from perspective of dynamic vision. In *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pages 37–42. IEEE, 2020.

[75] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. Fingerio: Using active sonar for fine-grained finger tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1515–1525, 2016.

[76] Ibrahima N'Doye, Ding Zhang, Mohamed-Slim Alouini, and Taous-Meriem Laleg-Kirati. Establishing and maintaining a reliable optical wireless communication in underwater environment. *IEEE Access*, 9:62519–62531, 2021.

[77] Phuc Nguyen, Taeho Kim, Jinpeng Miao, Daniel Hesselius, Erin Kenneally, Daniel Massey, Eric Frew, Richard Han, and Tam Vu. Towards rf-based localization of a drone and its controller. In *Proceedings of the 5th workshop on micro aerial vehicle networks, systems, and applications*, pages 21–26, 2019.

[78] Phuc Nguyen, Mahesh Ravindranatha, Anh Nguyen, Richard Han, and Tam Vu. Investigating cost-effective rf-based detection of drones. In *Proceedings of the 2nd workshop on micro aerial vehicle networks, systems, and applications for civilian use*, pages 17–22, 2016.

[79] Phuc Nguyen, Hoang Truong, Mahesh Ravindranathan, Anh Nguyen, Richard Han, and Tam Vu. Matthan: Drone presence detection by identifying physical signatures in the drone's rf communication. In *Proceedings of the 15th annual international conference on mobile systems, applications, and services*, pages 211–224, 2017.

[80] U.S. Department of Energy. Lighting Choices to Save You Money. https://www.energy.gov/energysaver/lighting-choices-save-you-money, 2022.

[81] Hao Pan, Yi-Chao Chen, Lanqing Yang, Guangtao Xue, Chuang-Wen You, and Xiaoyu Ji. mqrcode: Secure qr code using nonlinearity of spatial frequency in light. In *The 25th Annual International Conference on Mobile Computing and Networking*, page 27. ACM, 2019.

[82] Kun Qian, Yumeng Lu, Zheng Yang, Kai Zhang, Kehong Huang, Xinjun Cai, Chenshu Wu, and Yunhao Liu. Aircode: Hidden screen-camera communication on an invisible and inaudible dual channel. In *NSDI*, pages 457–470, 2021.

[83] Qualcomm. Making 5g nr a reality: leading the technology inventions for a unified, more capable 5g air interface. *White paper*, 2016.

[84] E Ramadhani and GP Mahardika. The technology of lifi: A brief introduction. In *IOP Conf. Series: Materials Science and Engineering*, volume 3, pages 1–10, 2018.

[85] A Rammohan and C RameshKumar. Investigation on light intensity and temperature distribution of automotive's halogen and led headlight. In *2017 International conference on Microelectronic Devices, Circuits and Systems (ICMDCS)*, pages 1–6. IEEE, 2017.

[86] Razieh Rastgoo, Kourosh Kiani, and Sergio Escalera. Sign language recognition: A deep survey. *Expert Systems with Applications*, 164:113794, 2021.

[87] Market Reports. Globle scube diving equipment industry research report, growth trends and competitive analysis 2021-2027. https://www.marketreportsworld.com/ global-scuba-diving-equipment-industry-18271751, 2021.

[88] Aleksandr Rodionov, Petr Unru, and Aleksandr Golov. Long-range underwater acoustic navigation and communication system. In *2020 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE)*, pages 60–63. IEEE, 2020.

[89] Nasir Saeed, Abdulkadir Celik, Tareq Y Al-Naffouri, and Mohamed-Slim Alouini. Underwater optical wireless communications, networking, and localization: A survey. *Ad Hoc Networks*, 94:101935, 2019.

[90] Krishna Raj Sapkota, Steven Roelofsen, Artem Rozantsev, Vincent Lepetit, Denis Gillet, Pascal Fua, and Alcherio Martinoli. Vision-based unmanned aerial vehicle detection and tracking for sense and avoid systems. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1556–1561. Ieee, 2016.

[91] Giuseppe Schirripa Spagnolo, Lorenzo Cozzella, and Fabio Leccese. Underwater optical wireless communications: Overview. *Sensors*, 20(8):2261, 2020.

[92] Zhambyl Shaikhanov, Ahmed Boubrima, and Edward W Knightly. Autonomous drone networks for sensing, localizing and approaching rf targets. In *2020 IEEE Vehicular Networking Conference (VNC)*, pages 1–8. IEEE, 2020.

[93] Abhishek Sharma, Pankhuri Vanjani, Nikhil Paliwal, Chathuranga M Wijerathna Basnayaka, Dushantha Nalin K Jayakody, Hwang-Cheng Wang, and P Muthuchidambaranathan. Com- munication and networking technologies for uavs: A survey. *Journal of Network and Computer Applications*, 168:102739, 2020.

[94] Truman R Strobridge. *Chronology of Aids to Navigation and the Old Lighthouse Service, 1716-1939*. Public Affairs Division, United States Coast Guard, 1974.

[95] Sanjib Sur, Ioannis Pefkianakis, Xinyu Zhang, and Kyu-Han Kim. Towards scalable and ubiquitous millimeter-wave wireless networks. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, pages 257–271, 2018.

[96] Witold Szymański and Maurycy Kin. The perspective transformation in illusionistic ceiling painting of late baroque. *Teka Komisji Architektury, Urbanistyki i Studiów Krajobrazowych*, 15(1):104–112, 2019.

[97] Andrea Tagliasacchi, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. Robust articulated-icp for real-time hand tracking. In *Computer graphics forum*, volume 34, pages 101–114. Wiley Online Library, 2015.

[98] Lei Tao, Tao Hong, Yichen Guo, Hangyu Chen, and Jinmeng Zhang. Drone identification based on centernet-tensorrt. In *2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, pages 1–5. IEEE, 2020.

[99] Zhao Tian, Kevin Wright, and Xia Zhou. The darklight rises: Visible light communication in the dark. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 2–15, 2016.

[100] Sumit Tiwari. An introduction to qr code technology. In *2016 international conference on information technology (ICIT)*, pages 39–44. IEEE, 2016.

[101] CAIDA UCSD. SIGCOMM'17 anonymized internet traces. https://www.caida.org/data/passive/passive_dataset.xml, 2017.

[102] CAIDA UCSD. CAIDA'19 anonymized internet traces. https://www.caida.org/data/passive/passive_dataset.xml, 2019.

[103] Hanif Ullah, Nithya Gopalakrishnan Nair, Adrian Moore, Chris Nugent, Paul Muschamp, and Maria Cuevas. 5g communication: an overview of vehicle-to-everything, drones, and healthcare use-cases. *IEEE Access*, 7:37251–37268, 2019.

[104] Eren Unlu, Emmanuel Zenou, and Nicolas Riviere. Using shape descriptors for uav detection. *Electronic Imaging*, 2018(9):128–1, 2018.

[105] Suseela Vappangi and VV Mani. Concurrent illumination and communication: A survey on visible light communication. *Physical Communication*, 33:90–114, 2019.

[106] Qing Wang, Marco Zuniga, and Domenico Giustiniano. Passive communication with ambient light. In *Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies*, pages 97–104, 2016.

[107] Robert Wang, Sylvain Paris, and Jovan Popović. 6d hands: markerless hand-tracking for computer aided design. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 549–558, 2011.

[108] X. Wang, J. P. Linnartz, and T. Tjalkens. An intelligent lighting system: Learn user preferences from inconsistent feedback. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, UbiComp '16, page 1620–1626, New York, NY, USA, 2016. Association for Computing Machinery.

[109] Zeyu Wang, Zhice Yang, Qianyi Huang, Lin Yang, and Qian Zhang. Als-p: Light weight visible light positioning via ambient light sensor. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 1306–1314. IEEE, 2019.

[110] WiKi. Hamming code. https:en.wikipedia.orgwikiHamming_code, 2021.

[111] WiKi. Barcode. https://en.wikipedia.org/wiki/Barcode#Matrix_(2D)_barcodes, 2022.

[112] WiKi. Unmanned aerial vehicle. https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle, 2022.

[113] Norman J Woodland and Silver Bernard. Classifying apparatus and method, October 7 1952. US Patent 2,612,994.

[114] Hongjia Wu, Qing Wang, Jie Xiong, and Marco Zuniga. Smartvlc: When smart lighting meets vlc. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*, pages 212–223, 2017.

[115] Hongjia Wu, Qing Wang, Jie Xiong, and Marco Zuniga. Smartvlc: Co-designing smart lighting and communication for visible light networks. *IEEE Transactions on Mobile Computing*, 19(8):1956–1970, 2019.

[116] Xiping Wu, Mohammad Dehghani Soltani, Lai Zhou, Majid Safari, and Harald Haas. Hybrid lifi and wifi networks: A survey. *IEEE Communications Surveys & Tutorials*, 23(2):1398–1420, 2021.

[117] Yue Wu, Purui Wang, Kenuo Xu, Lilei Feng, and Chenren Xu. Turboboosting visible light backscatter communication. In *Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '20, page 186–197, New York, NY, USA, 2020. Association for Computing Machinery.

[118] Yue Wu, Purui Wang, Kenuo Xu, Lilei Feng, and Chenren Xu. Turboboosting visible light backscatter communication. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, pages 186–197, 2020.

[119] Zhenyu Xiao, Lipeng Zhu, Yanming Liu, Pengfei Yi, Rui Zhang, Xiang-Gen Xia, and Robert Schober. A survey on millimeter-wave beamforming enabled uav communications and networking. *IEEE Communications Surveys & Tutorials*, 24(1):557–610, 2021.

[120] Huichuan Xu, Daisuke Iwai, Shinsaku Hiura, and Kosuke Sato. User interface by virtual shadow projection. In *2006 SICE-ICASE International Joint Conference*, pages 4814–4817. IEEE, 2006.

[121] Beiya Yang and Erfu Yang. A survey on radio frequency based precise localisation technology for uav in gps-denied environment. *Journal of Intelligent & Robotic Systems*, 103(3):1–30, 2021.

[122] Y. Yang, J. Hao, and J. Luo. Ceilingtalk: Lightweight indoor broadcast through led-camera communication. *IEEE Transactions on Mobile Computing*, 16(12):3308–3319, 2017.

[123] Yanbing Yang, Jie Hao, and Jun Luo. Ceilingtalk: Lightweight indoor broadcast through led-camera communication. *IEEE Transactions on Mobile Computing*, 16(12):3308–3319, 2017.

[124] Yanbing Yang and Jun Luo. Boosting the throughput of led-camera vlc via composite light emission. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pages 315–323. IEEE, 2018.

[125] Yanbing Yang and Jun Luo. Composite amplitude-shift keying for effective led-camera vlc. *IEEE Transactions on Mobile Computing*, 19(03):528–539, 2020.

[126] Yanbing Yang, Jun Luo, Chen Chen, Wen-De Zhong, and Liangyin Chen. Synlight: synthetic light emission for fast transmission in cots device-enabled vlc. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 1297–1305. IEEE, 2019.

[127] Yang Yang, Zhimin Zeng, Julian Cheng, and Caili Guo. Spatial dimming scheme for optical ofdm based visible light communication. *Optics express*, 24(26):30254–30263, 2016.

[128] Yang Yang, Zhimin Zeng, Julian Cheng, and Caili Guo. A novel hybrid dimming control scheme for visible light communications. *IEEE Photonics Journal*, 9(6):1–12, 2017.

[129] Zhice Yang, Zeyu WANG, Jiansong Zhang, Chenyu Huang, and Qian Zhang. Polarization-based visible light positioning. *IEEE Transactions on Mobile Computing*, 18(3):715–727, 2019.

[130] Ibrar Yaqoob, Latif U Khan, SM Ahsan Kazmi, Muhammad Imran, Nadra Guizani, and Choong Seon Hong. Autonomous driving cars in smart cities: Recent advances, requirements, and challenges. *IEEE Network*, 34(1):174–181, 2019.

[131] Kai Ying, Zhenhua Yu, Robert J Baxley, Hua Qian, Gee-Kung Chang, and G Tong Zhou. Nonlinear distortion mitigation in visible light communications. *IEEE Wireless Communications*, 22(2):36–45, 2015.

[132] Fahad Zafar, Masuduzzaman Bakaul, and Rajendran Parthiban. Laser-diode-based visible light communication: Toward gigabit class communication. *IEEE Communications Magazine*, 55(2):144–151, 2017.

[133] Fahad Zafar, Dilukshan Karunatilaka, and Rajendran Parthiban. Dimming schemes for visible light communication: the state of research. *IEEE Wireless Communications*, 22(2):29–35, 2015.

[134] Zhaoquan Zeng, Shu Fu, Huihui Zhang, Yuhan Dong, and Julian Cheng. A survey of underwater optical wireless communications. *IEEE communications surveys & tutorials*, 19(1):204–238, 2016.

[135] Bo Zhang and Hoi Dick Ng. An experimental investigation of the explosion characteristics of dimethyl ether-air mixtures. *Energy*, 107:1–8, 2016.

[136] Chi Zhang and Xinyu Zhang. Pulsar: Towards ubiquitous visible light localization. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, pages 208–221, 2017.

[137] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*, 2020.

[138] Kai Zhang, Yi Zhao, Chenshu Wu, Chaofan Yang, Kehong Huang, Chunyi Peng, Yunhao Liu, and Zheng Yang. Chromacode: A fully imperceptible screen-camera communication system. *IEEE Transactions on Mobile Computing*, 2019.

[139] Lan Zhang, Kebin Liu, Xiang-Yang Li, Cihang Liu, Xuan Ding, and Yunhao Liu. Privacy-friendly photo capturing and sharing system. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 524–534. ACM, 2016.

[140] Weidong Zhang, Lili Dong, Xipeng Pan, Peiyu Zou, Li Qin, and Wenhai Xu. A survey of restoration and enhancement for underwater images. *IEEE Access*, 7:182259–182279, 2019.

[141] Xiao Zhang, Hanqing Guo, James Mariani, and Li Xiao. U-star: An underwater navigation system based on passive 3d optical identification tags. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, pages 648–660, 2022.

[142] Xiao Zhang, Griffin Klevering, Juexing Wang, Li Xiao, and Tianxing Li. Rofin: 3d hand pose reconstructing via 2d rolling fingertips. *Proceedings of 21st ACM International Conference on Mobile Systems, Applications, and Services, conditionally accepted*, 2023.

[143] Xiao Zhang, Griffin Klevering, and Li Xiao. Exploring rolling shutter effect for motion tracking with objective identification. In *Proceedings of the Twentieth ACM Conference on Embedded Networked Sensor Systems*, pages 816–817, 2022.

[144] Xiao Zhang, Griffin Klevering, and Li Xiao. Posefly: On-site pose parsing of swarming drones via 4-in-1 optical camera communication. In *2023 IEEE 24th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–10. IEEE, 2023.

[145] Xiao Zhang, James Mariani, Li Xiao, and Matt W Mutka. Lifod: Lighting extra data via fine-grained owc dimming. In *2022 19th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pages 73–81. IEEE, 2022.

[146] Xiao Zhang and Li Xiao. Effective subcarrier pairing for hybrid delivery in relay networks. In *2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pages 238–246. IEEE, 2020.

[147] Xiao Zhang and Li Xiao. Lighting extra data via owc dimming. In *Proceedings of the Student Workshop*, pages 29–30, 2020.

[148] Xiao Zhang and Li Xiao. Rainbowrow: Fast optical camera communication. In *2020 IEEE 28th International Conference on Network Protocols (ICNP)*, pages 1–6. IEEE, 2020.

[149] Run Zhao, Dong Wang, Qian Zhang, Xueyi Jin, and Ke Liu. Smartphone-based handwritten signature verification using acoustic signals. *Proceedings of the ACM on Human-Computer Interaction*, 5(ISS):1–26, 2021.

[150] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[151] Shilin Zhu, Chi Zhang, and Xinyu Zhang. Automating visual privacy protection using a smart led. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, pages 329–342, 2017.

[152] Shilin Zhu, Chi Zhang, and Xinyu Zhang. Lishield: Create a capture-resistant environment against photographing. In *Proceedings of the 9th ACM Workshop on Wireless of the Students, by the Students, and for the Students*, pages 23–23, 2017.