NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

FREE SPACE OPTICS COMMUNICATIONS FOR LOW-POWER HANDHELD MOBILE DEVICES

by

James D. Miller

September 2020

Thesis Advisor: Gurminder Singh
Co-Advisor: Peter R. Ateshian

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** September 2020	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE
FREE SPACE OPTICS COMMUNICATIONS FOR LOW-POWER HANDHELD MOBILE DEVICES

5. FUNDING NUMBERS

6. AUTHOR(S) James D. Miller

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release. Distribution is unlimited.

12b. DISTRIBUTION CODE
A

13. ABSTRACT (maximum 200 words)

This research demonstrates a machine learning (ML) approach to array-based free-space optical (FSO) communication using mobile devices. Modern warfighters need non–radio frequency (RF) communication methods to eliminate the risks associated with RF communication, such as detection, eavesdropping, and jamming. FSO communications promises tremendous throughput among other advantages, such as low-probability of intercept/detect and resistance to jamming. However, atmospheric conditions significantly reduce achieved performance by introducing fading and noise on the channel. To increase channel resilience and throughput, we employ spatial codes using an array of lasers at the transmitter and train several ML models on the channel alphabet to provide efficient decoding at the receiver. We compare the performance of a Single Shot Detection (SSD) MobileNet model with a You-Only-Look-Once model during the training process, and we demonstrate data transfer over a proof-of-concept system using the trained SSD MobileNet model. We detail the hardware and software implementation for the proof-of-concept, which uses handheld mobile devices and an array of low-cost, low-power lasers. Future experimentation is planned to incorporate forward-error correction and testing over greater distances under realistic conditions.

14. SUBJECT TERMS
free space optical, mobile device, machine learning, ML, radio frequency, laser, mobile devices, communications, non-RF communication, You-Only-Look-Once, YOLO, Single Shot Detection, SSD, MobileNet, free-space optical, FSO

15. NUMBER OF PAGES
85

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

FREE SPACE OPTICS COMMUNICATIONS FOR LOW-POWER HANDHELD MOBILE DEVICES

James D. Miller
Lieutenant, United States Navy
BS, University of South Alabama, 2004

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 2020

Approved by: Gurminder Singh
Advisor

Peter R. Ateshian
Co-Advisor

Gurminder Singh
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This research demonstrates a machine learning (ML) approach to array-based free-space optical (FSO) communication using mobile devices. Modern warfighters need non–radio frequency (RF) communication methods to eliminate the risks associated with RF communication, such as detection, eavesdropping, and jamming. FSO communications promises tremendous throughput among other advantages, such as low-probability of intercept/detect and resistance to jamming. However, atmospheric conditions significantly reduce achieved performance by introducing fading and noise on the channel. To increase channel resilience and throughput, we employ spatial codes using an array of lasers at the transmitter and train several ML models on the channel alphabet to provide efficient decoding at the receiver. We compare the performance of a Single Shot Detection (SSD) MobileNet model with a You-Only-Look-Once model during the training process, and we demonstrate data transfer over a proof-of-concept system using the trained SSD MobileNet model. We detail the hardware and software implementation for the proof-of-concept, which uses handheld mobile devices and an array of low-cost, low-power lasers. Future experimentation is planned to incorporate forward-error correction and testing over greater distances under realistic conditions.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION 1
A. PROBLEM STATEMENT 2
B. OBJECTIVE 2
C. SCOPE 2
D. MOTIVATION 3
E. THESIS STRUCTURE 3

II. BACKGROUND 5
A. FREE SPACE OPTICS 5
1. Electromagnetic Spectrum 6
2. Spectrum Regulations 8
3. Atmospheric Effects 8
4. Transmitter and Receiver Characteristics 10
5. Pulse Width Modulation 10
6. Light Amplification as a Communication 11
7. Multiple Input Multiple Output (MIMO) 12
B. ELECTRONIC WARFARE 12
1. Jamming 12
2. LPI/LPD/LPE 13
3. Emissions Control (EMCON) 13
C. HANDHELD MOBILE DEVICES FOR MILITARY USE 13
1. Power Consumption in Mobile Devices 14
2. 2D and Quick Response Codes 15
D. REAL-TIME COMPUTER VISION 16
1. TensorFlow Lite 16
2. IBM Watson Machine Learning 18
3. Open Source Computer Vision 19
E. SUMMARY 19

III. EXPERIMENTAL ARCHITECTURES 21
A. MOBILE MIMO COMMUNICATIONS USING TF-LITE WITH IBM ML 21
1. Free Space Optics Transmit Process 22
2. Free Space Optics Receive Process 28
B. MIMO COMMUNICATIONS USING OPENCV WITH ML 31
C. SUMMARY 37

IV. TESTING AND ANALYSIS 39
A. MACHINE LEARNING EXPERIMENT USING TF-LITE AND IBM WATSON 39
B. MACHINE LEARNING EXPERIMENT USING OPENCV AND IMAGEAI 40
C. SUMMARY OF RESULTS 50

V. CONCLUSIONS AND RECOMMENDATIONS 51
A. DISCUSSION OF FINDINGS 51
B. RECOMMENDATIONS FOR FUTURE WORK 52
1. Initialize Communicates Using Machine Learning 52
2. Continued Study in Error Detection and Correction Methods 53
C. CONCLUSION 53

APPENDIX A. ARDUINO TRANSMIT CODE 55

APPENDIX B. ML USING OPENCV AND IMAGEAI CODE.................................59

LIST OF REFERENCES 61

INITIAL DISTRIBUTION LIST 63

LIST OF FIGURES

Figure 1. Illustrates only certain electromagnetic waves are able to penetrate the earth's atmosphere. Source: [5]. 9

Figure 2. Illustrates a typical FSO communications link. Source: [6]. 10

Figure 3. PWM laser modulation spaced over time. 11

Figure 4. Structure of a QR code symbol. Source: [12] 15

Figure 5. Flow diagram of a TensorFlow-Lite deep learning implementation model on an Android mobile device. Source: [13]. 17

Figure 6. Proof-of-concept system diagram 23

Figure 7. Example 3D and photographic image types used to train the object detection model of light patterns. 24

Figure 8. Diagram of the SSD MobileNet model process. 26

Figure 9. Transmission process from the user to user and character identification snapshot. 29

Figure 10. Demonstration of the custom-built chat application. 30

Figure 11. YOLOv3 network architecture schematic. Source: [20] 33

Figure 12. Shannon-Hartley capacity theorem and descriptions. Source: [16]. 36

Figure 13. Loss during SSD MobileNet model training. 40

Figure 14. Loss during model training displayed over 23 cycles. 42

Figure 15. Object loss function is the sum of localization, confidence, and classification loss. Source: [21]. 43

Figure 16. Model evaluation settings and mAP results. 44

Figure 17. Byte pattern reflected from FSO transmission with a 98% probability. 46

Figure 18. Byte pattern reflected from FSO transmission with a 96% probability. 47

Figure 19. Image of a byte pattern reflected from FSO transmission with a 96% probability. 48

Figure 20. Example image of a false positive pattern detection. 49

LIST OF TABLES

Table 1. Electromagnetic spectrum..7

Table 2. Probability and pattern box points of captured objects..............................45

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ASCII	American Standard Code for Information Interchange
Bps	Bits per second
COP	Common operating picture
COTS	Commercial-off-the-shelf
EM	Electromagnetic
EMCON	Emissions controlled
Fps	frames per second
FSO	Free space optics
Gbps	Gigabit per second
IDE	Integrated development environment
IED	Improvised explosive device
KBps	Kilobytes per second
Laser	Light amplification by stimulated emission of radiation
LED	Light emitting diode
LoS	Line-of-sight
LPE	Low probability of exploitation
LPD	Low probability of detection
LPI	Low probability of interception
M2M	Machine-to-machine
Mbps	Megabits per second
ML	Machine learning
MIMO	Multiple input multiple output
NAVEODTECH	Naval explosive ordnance disposal technology
NLoS	Non-line-of-sight
OpenCV	Open source computer vision library
PWM	Pulse width modulation
QR	Quick response
R-CNN	Region-based convolutional neural network
RF	Radio frequency
SISO	Single input single output

SSD	Single shot detector
YOLO	You-only-look-once

EXECUTIVE SUMMARY

Free space optics (FSO) communications provide an attractive alternative to conventional RF techniques. FSO communications provide several advantages, such as higher bandwidth, low probability of interception (LPI), low probability of detection (LPD), and low probability of exploitation (LPE). It can also provide immunity from interference and jamming. However, the FSO channel in the atmosphere is subject to a variety of environmental effects that reduce the overall capacity of the channel. The effects of atmospheric turbulence are a key loss factor, creating relatively long duration fades at the receiver [1]. Given the high data-rates possible with FSO, fading due to atmospheric turbulence causes extensive error rates over a single-input/single-output system [2].

To overcome these issues, several mitigation techniques have been implemented. Many of these include various forms of multiple input/multiple output systems, which incorporate features such as spatial diversity and multiplexing, aperture or relay diversity, and space-time coding. Multiplexing arrays of lasers to achieve greater throughput leads to the additional challenge of inter-channel interference due to beam divergence, as noted in [3].

The application of machine learning (ML) to wireless communications is outlined in [4], where the entire physical layer is replaced with a deep-learning auto encoder. A similar technique was found to overcome inter-symbol interference when applied to the unsupervised learning of radio-frequency signals [5]. More conventionally, machine-learning is used to recognize visual patterns after undergoing distortions. Visual codes have been used to transmit data wirelessly over short distances.

This research project incorporates elements from a number of these FSO communication techniques and applies ML to demonstrate a low-cost, low-overhead proof-of-concept solution. We use an array of low-energy, eye-safe lasers combined with a camera to establish a multiple-input/single-output channel using handheld mobile devices which takes advantage of transmitter diversity for increased resilience and throughput. This setup transmits data by alternating visual patterns at high frequencies. To combat the inter-channel interference induced by beam divergence, we train an ML model on a limited channel alphabet in order to efficiently decode these patterns upon receipt. Two different ML models are implemented, and the performance characteristics of each are examined through a series of experiments.

For the basic proof-of-concept, user input is entered via a smartphone text-based interface. This data is encoded and a sequence of patterns are modulated onto eight lasers configured in a grid pattern, using on-off keying (OOK). A ninth laser is also included for timing purposes, making a 3x3 array. These patterns are projected across free space onto a smooth surface in line-of-sight (LoS) of the receiving device.

The receiving smartphone device senses incoming data via its on-board camera. ML-based image-recognition software is used to associate the received patterns to the channel library. The decoded data is then passed to a display module that outputs text within the smartphone application. The user interface for both transmission and receipt is hosted on Galaxy S9 SM-G9600 devices running Android 8.0.0. The transmitting smartphone is connected to the transmission device via a USB on-the-go cable, capable of transfer speeds up to 480 Mbps.

The transmission device consists of nine Adafruit Class 3R 5mW laser diodes in a small fabricated box, placed in a three by three pattern. These eye-safe Class 3 laser diodes were chosen due to safety considerations, effectively reducing the communication distance to less than 100 feet as a function of beam output power. System control is provided by an Arduino Uno, which receives input from the smartphone application, and modulates the corresponding encoded patterns onto the attached laser array. Using this setup, we achieved a maximum transmission rate of 115.2 Kbps. The receiving Galaxy S9 device can capture a maximum of 960 frames per second (fps) with its on-board camera; however, it can only maintain 960 fps for two seconds. After that, the camera maintains 30 fps. Sampling at twice the achievable data rate, this yields an overall transmission speed of approximately 15 fps, or a maximum of 120bps if no forward error correction is employed.

While speed was not the primary design factor in this proof-of-concept design, this data-rate is clearly not adequate for a fully operational system. Typical commercial FSO systems transmit at a rate of 10 Gbps. Experimental FSO systems are competing with fiber optics systems and have demonstrated successful transmission rates of 120 Gbps [6]. An optimal optical receiver will need to capture and process data at similarly high rates.

Our research demonstrates the feasibility of designing an array-based, mobile FSO communications channel using ML to perform pattern recognition and decoding of optical symbols. However, both experiments were limited in the number labels and training sets available since we had to create and label the entire dataset using unique light patterns that do not share characteristics or similarities with a pre-built model. Limitations in computing power and time also meant that we only were able to train a small subset of the overall channel alphabet. More generally, another limiting factor for mobile devices the time it takes to process the received images. While TensorFlow Lite optimizes the computationally demanding operations required by deep neural networks, the process still requires significant processing time compared to traditional methods, such as using a photodiode receiver to receive FSO communication.

Additionally, the proof-of-concept was intentionally limited in scope by severely restricting the test conditions. The system was run indoors in a dark environment to illuminate the laser patterns for the camera to focus. These conditions were chosen to isolate the objects detected and identified by the ML application. The proof-of-concept architecture was designed to demonstrate ability and determine a protocol for mobile-to-mobile FSO communications and not to test the system's capabilities under hostile environmental factors—ideally, the system must be able to transfer data under harsh environmental conditions, to include varying atmospheric and noise conditions.

References

[1] R. D. Halbing, C. R. Hershberger, J. A. Stalder, L. C. Andrews, and R. L. Phillips, "Fading effects due to scintillation caused by atmospheric turbulence in a wireless optical communication link," in *Optical Science and Technology, SPIE's 48th Annual Meeting.*, 2004, pp. 33-43.

[2] V. W. S. Chan, "Free-space optical communications," *J. Lightwave Technol*., vol. 24, pp. 4750–4762, Dec 2006.

[3] J. Akella, M. Yuksel, and S. Kalyanaraman, “Multi-channel communication in free-space optical networks for the last-mile,” *15th IEEE Workshop on Local and Metropolitan Area Network*, Jun. 2007, pp. 43–48.

[4] D. Xu, X. Liu, and A. Vasilakos, “Traffic-Aware resource provisioning for distributed clouds,” *IEEE Cloud Computing*, vol.2, no. 1 pp. 30 – 39, Jan 2015.

[5] J. Lewis, “Lund University develops 5 trillion FPS film camera,” *Vision Systems Design*, vol. 22, no. 6, pp. 9–12, Jun 2017.

[6] M. Singh and J. Malhotra, “Performance investigation of high-speed FSO transmission system under the influence of different atmospheric conditions incorporating 3-D orthogonal modulation scheme,” *Optical and quantum electronics*, vol. 51, no. 9, pp. 1–13, 2019.

ACKNOWLEDGMENTS

I want to thank my thesis advisors, Dr. Gurminder Singh and Mr. Peter Ateshian, for their patience, guidance, and words of encouragement as I worked to complete this thesis. I appreciate the time and energy they spent in helping me accomplish my goals and finish this thesis. I would like to thank Paul Keeley and Jerome Nilmeier for their assistance and knowledge as subject matter experts.

To my family, I would like to thank my loving wife, Ramona, for her unwavering support. Her encouragement, enthusiasm, patience, and understanding were, in no small part, integral to my successful completion of this thesis. I would like to thank my son, Daniel, for his cheerful support and motivating spirit to help complete the task.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

In military applications, there are scenarios where traditional radio frequency (RF) based communication is not permitted to prevent detection, eavesdropping, and jamming. There is a need for alternate communication modalities that eliminate the risks associated with RF communication. Optical communication being inherently non-RF can provide such an alternative. The ability to operate within the information environment is essential to the success of military operations. This is where free space optics (FSO) can be used for its capacity for high bandwidth and the ability to provide communications in RF hostile environments. FSO communications offer high bandwidth-resilient mobile communications with the capability to support the warfighter in RF hostile and hazardous environments were traditional RF transmissions are not permitted. The study can benefit military communications capabilities by exploring optical communication techniques between mobile systems.

This thesis discusses the development FSO communications used by mobile systems capable of operating within RF-denied environments and without the receiver having to be in direct line-of-sight of the transmitter. This research demonstrates dynamic-information exchange between mobile systems using optical methods to transmit dynamic encoded data using machine learning methods. This thesis describes the principles and techniques used in the development of mobile communications within a non-RF environment. Our goal of this research is to explain the capabilities and limitations of FSO communications within an RF denied environment, especially with regards to mobile devices.

We continued the development of an existing design using arrays of low-energy, eye-safe lasers or LEDs, enabling us to design a system to support mobile device requirements, increased information density, and robust communication with error correction capability. An array of lasers/LEDs can send data by oscillating patters at high frequencies. In addition, our system designs were developed to demonstrate multiple array low-energy FSO mobile communications using machine learning algorithms for image detection.

A. PROBLEM STATEMENT

The goal of this thesis is to develop a mobile FSO communications system that uses machine learning (ML) and is capable of operating within RF-denied environments. This thesis addresses the following research questions:

1. **Principal question:** How might we create a system architecture to support ML optical communications between mobile devices within a non-RF environment?
2. **Secondary question:** How do we develop an architecture using commercial-off-the-shelf (COTS) components to support mobile optical communications?

B. OBJECTIVE

The objective of this thesis is to research, develop, and examine mobile FSO communications architectures to improve communication FSO methods. The thesis builds upon the previous FSO multiple input multiple output (MIMO) communications architecture to improve error correction encoding. In addition, the thesis develops an architecture for mobile FSO MIMO communications using mobile devices. Achieving this objective will provide a proof-of-concept demonstrating the capabilities and identify limitations of FSO communications that can be used within RF denied environments.

C. SCOPE

The thesis scope consists of the research, development, design, construction, and testing of FSO communications for mobile systems. The project research includes architecture design to optimize M2M optical communications that used arrays of low-energy, eye-safe lasers or LEDs, and machine learning algorithms for data transmission and interference correction. The project proof-of-concept demonstrates QR like format research for optical transmissions to improve encoding data techniques. The methodologies explored in this research will facilitate future work in the creation of advanced capabilities for FSO communications using machine learning algorithms.

D. MOTIVATION

The use of technology has grown in our daily life. This also applies not least to military operations. Communication has long played an important role in military warfare. FSO communications provide high-speed resilient communications to the warfighter in a RF communications degraded environment.

The study can benefit military communications capabilities by exploring optical communication techniques between mobile systems. Research includes optimization methods to improve data encoding to improve optical communications between mobile devices. Potential to advance the knowledge and/or practice associated with FSO communication systems. By developing new methodology, this project adds to the body of knowledge in non-RF and optical-based communication.

E. THESIS STRUCTURE

This thesis explores the concept of FSO communications for mobile devices. The thesis is divided into five different chapters. Chapter II reviews prior work within academia and the current literature on FSO communications and capabilities. It provides background knowledge on the subject.

Chapter III outlines the methodology of this thesis. It defines the methods of research and the experimental architecture used for the experiments. This chapter describes the method for using a machine learning design to develop an operational mobile FSO communications system that demonstrates the usefulness of the machine learning FSO architecture. Two methods for machine learning are designed. Each method's advantages and disadvantages are tested and analyzed in Chapter IV.

Chapter IV contains the results and analysis from the experiments conducted. In this chapter, we analyzed the results of mobile FSO communication methods and provided conclusions based on the outcomes. Testing and analysis were conducted on machine learning methods using single shot detector (SSD) and you-only-look-once (YOLO). In addition, we tested single input single output (SISO) pulse width modulated communications over low-cost mobile devices.

Chapter V contains a summary of the research presented and interprets the findings. This chapter provides conclusions and recommendations for further research on FSO and machine learning methods.

II. BACKGROUND

This chapter describes the foundation of FSO communications by means of handheld mobile devices and the usage of FSO in RF-denied environments. It will provide a foundation by covering the basics of FSO communications, electromagnetic spectrum, FSO advantages and the challenges faced in using optical communications. This chapter addresses both terrestrial and exterritorial FSO communication methods. We will discuss how quick response (QR) code methods for sending data and identifying signal location and orientation can be used to improve FSO communications. In addition, this chapter provides a background in electronic warfare. It explores the use of handheld mobile FSO communication devices and the benefits of their use in electronic warfare. Finally, we explore how neural networks and real-time computer vision methods are used for image recognition and detection in optical communication systems.

A. FREE SPACE OPTICS

In recent years, FSO communication has emerged as an attractive alternative solution for high-speed data communications, as opposed to traditional RF communications. Traditional RF covers frequencies from 30 Hz to microwave frequencies extending to 300 GHz. FSO is the transmitting of modulated light signals through air, liquid and vacuum. FSO communication features include high bandwidth capability, immunity from jamming, low probability of interception (LPI), low probability of detection (LPD), low probability of exploitation (LPE) license-free spectrum, high data rate, rapid deployment time, and Low power requirements compared to RF.

FSO communication systems provide point-to-point LoS and non-line-of-sight (NLoS) depending on power and transmitter type, wireless connectivity through the use of light propagation within a free space medium. In the context of communications, an FSO medium is defined as air, water, vacuum, outer space and boundaries, as opposed to using a material medium such as a transmission line or fiber-optic cable. FSO systems combine valuable capabilities of fiber optics and RF systems by providing high data rates and nonmutual interference, and LPI/LPD/LPE security with wireless connectivity, quick installation, and low cost [1].

1. Electromagnetic Spectrum

The electromagnetic (EM) spectrum is a physical medium through which military forces conduct operations. Modern military forces use the EM spectrum as a method for communications and a tool for situational awareness. The EM spectrum includes all types of EM light radiation ranging from extremely low frequencies, 3 Hz, to Gamma Rays, 300 EHz. Table 1 list designated band of radio frequencies in the electromagnetic spectrum from 3 Hz to 300 EHz. All EM radiation waves travel at the speed of light in a vacuum and can be reflected, refracted, and diffracted. Radiation is energy that spreads out as it travels. Waves in the EM spectrum differentiate by frequency and wavelength. RF and visible light are ideal methods for communications due to their capability to travel through the earth's atmosphere and space. Light waves visible to the human eye represent a limited range of wavelengths within the EM spectrum.

Traditional RF technologies are extensively used in wireless-based communications systems. FSO communications operate outside of the traditional RF boundaries. Compared with traditional radio waves, FSO have multiple advantages in wireless communications. An advantage of FSO communications include frequencies measuring hundreds of terahertz. FSO frequencies operate within the electromagnetic spectrum as wavelengths between 100 nm to 3100 nm and include the entire visible spectrum.

Table 1. Electromagnetic spectrum.

Class			Frequency	Wavelength
Ionizing Radiation	Y	Gamma Rays	300 EHz	1 pm
	HX	Hard X-Rays	30 EHz	10 pm
			3 EHz	100 pm
	SX	Soft X-Rays	300 PHz	1 nm
	EUV	Extreme Ultraviolet	30 PHz	10 nm
Visible Light	NUV	Near Ultraviolet	3 PHz	100 nm
			300 THz	1 μm
	NIR	Near Infrared	30 THz	10 μm
	MIR	Mid Infrared	3 THz	100 μm
	FIR	Far Infrared	300 GHz	1 mm
Microwaves and Radio Waves	EHF	Extremely High Frequency	30 GHz	1 cm
	SHF	Super High Frequency	3 GHz	1 dm
	UHF	Ultra High Frequency	300 MHz	1 m
	VHF	Very High Frequency	30 MHz	10 m
	HF	High Frequency	3 MHz	100 m
	MF	Medium Frequency	300 kHz	1 km
	LF	Low Frequency	30 kHz	10 km
	VLF	Very Low Frequency	3 kHz	100 km
	ULF	Ultra Low Frequency	300 Hz	1000 km
	SLF	Super Low Frequency	30 Hz	10000 km
	ELF	Extremely Low Frequency	3 Hz	100000 km

Benefits of optical communication include no spectrum regulations or license requirements, immunity to electromagnetic interference, large bandwidth capacity, rapid deployment time, and relatively low usage cost. Optical systems communicate at a higher frequency than traditional RFs and will not interfere with traditional RFs operating in the same environment. Unlike traditional RF transmissions that broadcast with greater threat surface, FSO communications operate using a narrow laser beam that is not accessible unless viewed directly making the communications difficult to detect and intercept. In addition, FSO can be used with encryption to provide a secure communications method for satellites, mobile devices, forward operating units, ship, submarines, and autonomous vehicles. Due to its versatility, FSO systems have been deployed in local networks at a distance of a few meters to high-powered satellite systems operating at a distance over thousands of kilometers [2].

2. Spectrum Regulations

Traditional RF communications have the potential to cause interference and require frequency management and are fall under government regulations. Government regulations can limit bandwidth usage, set equipment requirements, and require licensing. Unlike traditional RF communications, FSO communications are not regulated. FSO systems can operate in close proximity to each other without causing interference. The need to manage FSO system stems from human eye safety requirements rather than spectrum management. There are four main classes for lasers: Class 1, Class 2, Class 3 and Class 4. Laser are classified based on their potential for causing injury. All lasers used in this thesis are Class 1. Lasers use visible light and do not possess eye or skin hazards.

3. Atmospheric Effects

Laser beam propagation can be degraded as it travels through space by atmospheric phenomena, such as scattering, scintillation, and atmospheric attenuation. Fog, rain, snow and clouds can have a damaging impact on FSO performance and cause scattering of optical signals. Fog is the main contributing factor compared to other atmospheric conditions in optical scattering limiting communication distances. Military considerations include dust, smoke, and ionization.

An optical wave propagating through the atmosphere will experience intensity fluctuations which can reduce the link availability and may introduce errors. Scintillation is the rapid fluctuations in intensity. It is caused by small temperature changes in the atmosphere, creating refraction fluctuations, such as optical turbulence [3].

FSO systems are capable of longer distances and greater speeds than microwave communications. Researchers at the German Aerospace Center have demonstrated a data transmission rate of 1.72 terabits per second across a distance of 10.45 kilometers [4]. Only optical and radio waves are suitable for satellite communications. The Earth's atmosphere prohibits other types of electromagnetic radiation from space from reaching Earth's surface. Most waves in the electromagnetic spectrum are absorbed into the atmosphere [5].

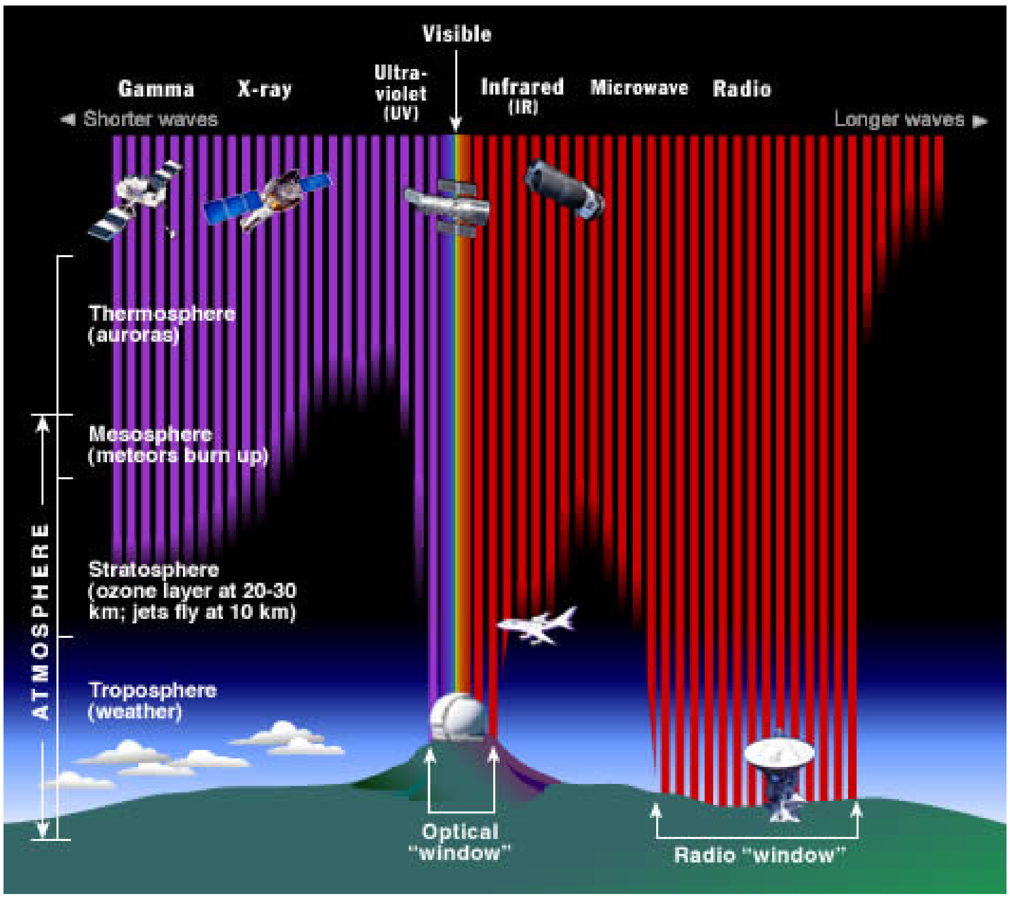


Figure 1. Illustrates only certain electromagnetic waves are able to penetrate the earth's atmosphere. Source: [5].

4. Transmitter and Receiver Characteristics

A typical FSO system consists of an optical transmitter, receiver and space as a medium. FSO communication performance is dependent on the capability of optical transmitters and receivers. This thesis examines FSO system architectures that transmit and receive defined light patterns or ones and zeros. The FSO TensorFlow-Lite model receiver utilizes a machine learning algorithm to detect and identify transmitted light patterns as data.

As shown in Figure 2, the transmitter inputs a series of 1s and 0s to be sent over free space by using a laser diode. Laser diodes convert the electrical energy into light by using a simple on-off keying to modulate the laser diode by turning the current "ON" and "OFF". In this example, the data is represented by a 1 or 0 and is transmitted by on-off keying of light from a laser diode. The receiver uses a photodiode to convert the light back into an electrical current of "1101".

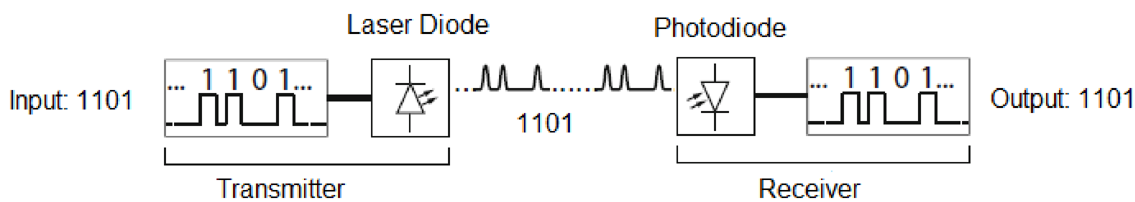


Figure 2. Illustrates a typical FSO communications link. Source: [6].

5. Pulse Width Modulation

Pulse width modulation (PWM) is the process of controlling the optical output of a laser by a manipulated source to divide the optical single into discrete parts. For our FSO laser communications, this is done by altering the pulse duty cycle and repetition frequency. FSO communications with PWM control the altering duty cycle and repetition frequency of the pulse of the light steam. PWM lasers emit pulses of light that are spaced in time, as shown in Figure 3. The pulse duration width is the interval between time measured across a pulse. The period is the interval between time measured from the start of one pulse to the next. FSO PWM communication systems do not emit light between pulses.

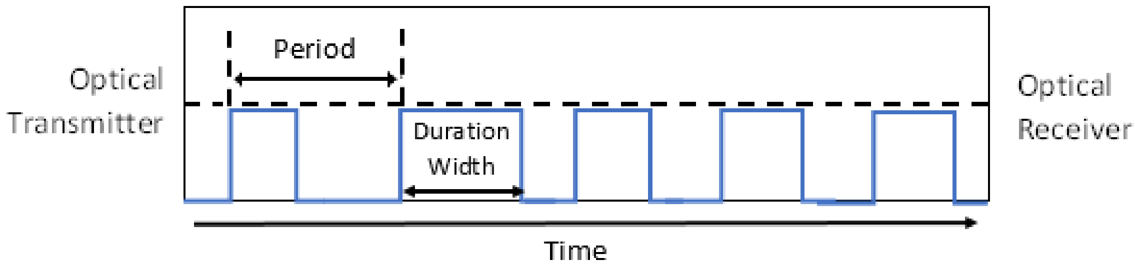


Figure 3. PWM laser modulation spaced over time.

Pulse Width Modulation techniques used in FSO communications offer better power management compared to continuous wave communication systems. According to a San Diego State University study, PWM communication systems provide significant transmit power savings that are beneficial for a longer battery lifecycle in mobile devices [7]. The University study included testing an “asynchronous point-to-point pulse-mode communication system via a unidirectional, terrestrial FSO link, utilizing a tunable, mid-infrared red quantum cascade laser” [7]. The study shows PWM techniques along with FSO system can provide exceptional power-savings capabilities for mobile devices.

6. Light Amplification as a Communication

Light Amplification by Stimulated Emission of Radiation (Laser) do not occur in nature and are artificially created. Laser describes the mechanism used for the generation of laser radiation. Lasers can produce highly collimated beams with narrow spectral beams. The laser’s narrow concentrated beams of energy allow the laser beams to travel long distances.

According to Eichler (2018), “Today, more than 10,000 different laser transitions are known generating radiation in the wavelength range from below 10 nm to over 1 mm, thus covering the X-ray, ultraviolet, visible and infrared spectral region” [6]. Lasers are commonly classified by solid-state lasers, liquid lasers, gas lasers, and semiconductor lasers. Semiconductor lasers are commonly used in state-of-the-art optical communication systems.

7. Multiple Input Multiple Output (MIMO)

Implementation of a multiple input multiple output (MIMO) wireless communication systems provides multiple research and implementation challenges, but offers benefits in diversity gains, increased coverage area, and are capable of sending and receiving multiple signals supporting synchronous communication. A major advantage the MIMO method has over the SISO method is its ability to perform at much higher data rates. A MIMO system can consist of multiple SISO systems working together.

B. ELECTRONIC WARFARE

In recent years, the military use of technology in the warfare environment has imposed intensive bandwidth and demanding data transfer requirements on mobile deployed networks. FSO technologies are being tested in air, ground, sea, and space environments to meet this demand for higher data rates and capacity. The overall goal of EW involves the use of EM energy and directed energy to control the EMS or to attack the enemy (jp3.pdf). U.S. adversaries, such as China and Russia, have developed rival electronic warfare systems to counter U.S. capabilities. When U.S. adversaries exploit, jam or sabotage the electromagnetic spectrum as a means to disable American capabilities, FSO communications can be used a communications protocol in the resulting RF-denied environments.

1. Jamming

The military faces increased threats of adversaries' electronic warfare capabilities, such as jamming. "Moderate jammers and SIGINT receivers are serious threats to communication networks. In addition, the survivability of the entire platform can be compromised if the enemy uses SIGINT receivers to detect and locate air platforms" [8]. Consider an example scenario where an adversary was performing an EW jamming attack on a U.S. military vessel. Traditional RF communications are susceptible to this type of attack. Instead, a vessel could continue communications and sharing of a common operating picture (COP) with FSO communication systems in an RF-denied environment.

2. LPI/LPD/LPE

Because of the way lasers are generated, FSO systems offer a level of protection in the low probability of intercept and low probability of detection qualities. FSO communications can be narrowed to align in a specific direction or be constrained to very small divergence angles. LPI includes signal characteristics or measures taken to avoid detection and prevent an adversary from incepting your transmission. LPD includes signal characteristics or measures taken to prevent the adversary from detecting a transmission. FSO communications produce LPI signals that are difficult to detect and read. Because FSO communications maintain a smaller footprint, they are inherently more secure than traditional RF communications. FSO systems can provide LPI, LPD and LPE communication paths to critical tactical units.

3. Emissions Control (EMCON)

In addition to higher LPI/LPD/LPE qualities, FSO communications yield another significant advantage: They do not emit energy within in the RF spectrum. FSO can be used in an emissions controlled (EMCON) environment, such as to limit ship detection in hostile environments or ammunition handling operations. EMCON conditions are set to prevent the detection, identification, and location of friendly forces by an enemy. In addition, EMCON can be used to minimize electromagnetic interference among friendly systems.

The Naval Explosive Ordnance Disposal Technology (NAVEODTECH) Division field-tested an FSO link for a counter improvised explosive device (IED) robot. The FSO surpassed traditional RF communications to successfully performed in the presence of RF jammers and displayed continued communications with the robot without interference [9].

C. HANDHELD MOBILE DEVICES FOR MILITARY USE

The use of mobile smart devices has grown exponentially in the last decade. Mobile devices are an emergent platform for military communications and computing. The world has become more connected and the need for mobile smart devices has dramatically increased. Mobile smart devices are changing the way the military communicates and conducts operations. In the future, tactical warfighters will use handheld mobile devices to direct the warfront, such as direct autonomous drones, call an airstrike, plot adversary actives, and communicate within the theater. These handheld mobile devices will need to be self-contained devices that can be operated without additional hardware and provide capable processing power and benefits such as a smaller size and weight.

Research and development demonstrations have been performed on FSO technologies in space-to-ground, airborne, and terrestrial environments.

> The challenges of developing tactically-relevant. FSO terminals encompass a diverse set of requirements inclusive of more than link performance alone, such as operating environment, platform integration, automation and user interface, network architecture, and the balance of these requirements with size, weight, power and cost [10]

FSO are not widely used in daily military operations. This is attributed to military dependence on mobile platforms and current FSO communications systems with limited mobility capability.

Mobile device agility allow for interoperability along multiple platforms and are useful where the physical connections are impractical due to terrain, hostile environment, or costs. Lockheed Martin developed "an Internet protocol based gateway product designed to provide interoperability between tactical radio systems and other communication devices including cellular telephones, landline telephones, commercial VoIP phones, and Smart Phones" [11].

1. Power Consumption in Mobile Devices

Mobile devices, such as smartphones, are designed to be energy efficient; however, there are computing and power consumption concerns that need be addressed. Among the main limitations of mobile devices are the processing power and energy consumption. Processing power limitations exist due to their battery dependency for system mobility. Power consumption and battery requirements must be understood when developing systems or software that use mobile devices. The development of mobile systems must take into account the limited resources of these small mobile devices.

2. 2D and Quick Response Codes

The technical specifications of QR codes are identified in the International Organization for Standardization, ISO-18004. "A QR code consists of an array of modules arranged in an overall square pattern, including a unique finder pattern located at three corners of the symbol and intended to assist in easy location of its position, size and inclination" [12].

Forty standard QR code version sizes can vary in allowable modules and data capacity. QR codes are capable of handling numeric and alphabetic characters, Kanji, and binary data types [12]. The largest QR code version is version forty. It consists of 177x177 modules in size and can hold 7,089 characters of numeric data using a low level of error correction. The smallest, version 1, consists of 21x21 modules in size and can hold twenty-five characters of alphanumeric data.

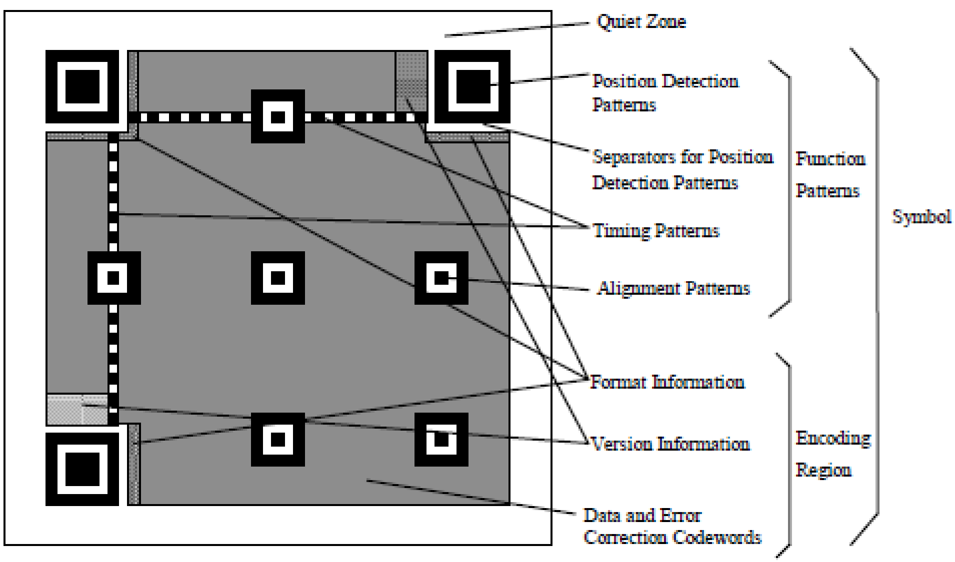


Figure 4. Structure of a QR code symbol. Source: [12]

QR codes use three positional detection indicators along the upper left, upper right, and lower left corners of the symbol to provide location and orientation. Lasers can be positioned in a similar configuration to provided rapid identification and Omni direction reading for FSO communications. Quick response codes can store data in both the horizontal and vertical direction. A laser design in a similar configuration would be ideal for FSO system to generate and transmit communications quickly.

QR codes use error correction to generate a series of error correction code words that are added to the end of the data sequence in order to enable the symbol to recover from the data loss [12]. This setup can be applied to FSO systems to provide built-in fault tolerant optical capability.

D. REAL-TIME COMPUTER VISION

This thesis will explore alternatives to using photodiodes and point-to-point for receiving optical transmissions, such as real-time computer vision and image recognition methods. Real-time computer vision and image recognition methods enable receiving devices to collect FSO communications that are projected on surfaces. These methods allow one transmitting device to project optical communications to multiple receiving devices. Receiving devices are able to process transmission projections from multiple angles. Techniques for recognizing angle position will be discussed later under 2D and quick response codes.

Image recognition is the process of object classification and detection. Image recognition methods view arrays of numerical values to define useable patterns. Images are classified using training data to teach the image recognition method. This thesis will test mobile device cameras and image recognition methods using mathematical algorithms for matching data and partitioning the points in higher-dimensional space belonging to the same class.

1. TensorFlow Lite

TensorFlow Lite is an image recognition application that can run machine learning techniques on mobile smart devices without the reliance on cloud machine learning. TensorFlow Lite is a machine learning platform that executes computations on the mobile device and eliminates the problems of transferring data between devices and external data systems or reliance on cloud environments. It is a machine learning platform optimized to run on a heterogeneous system used for accuracy, low latency, small model size, and portability to mobile and embedded devices. We will use TensorFlow Lite to develop deep learning models by training neural networks to classify images of laser projections to transmit information across mobile devices. A trained model uses labeled image sets to learn good values and accurately recognize particular patterns. A machine learning algorithm calculates the initial weights and adjusts the weights to create a model with increased prediction accuracy and lower total loss.

After the training model is generated, TensorFlow Lite converts the model to a .tflite file extension suitable for mobile devices [13]. The implementation of TensorFlow Lite deep learning process on an Android device is explained in Figure 5.

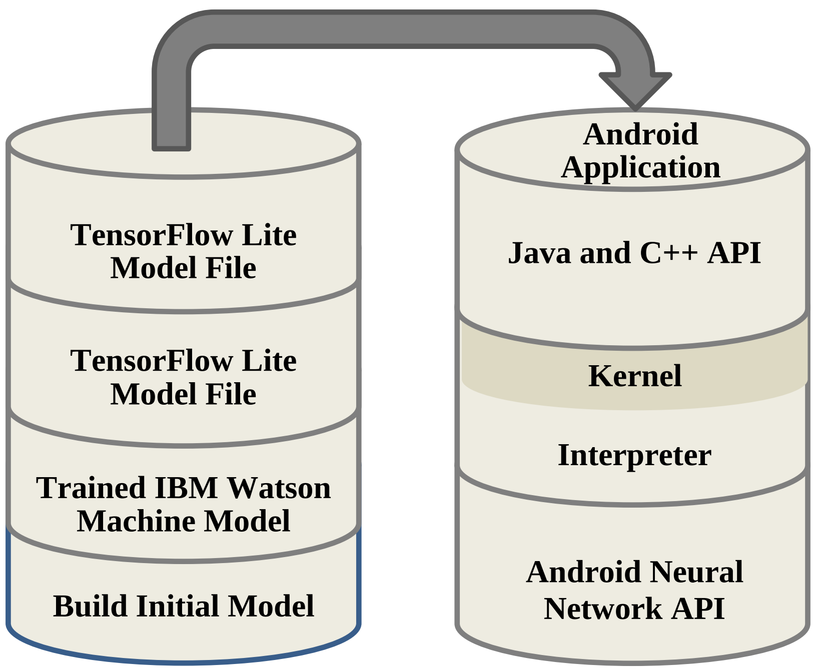


Figure 5. Flow diagram of a TensorFlow-Lite deep learning implementation model on an Android mobile device. Source: [13].

An android device is any device using the Android OS. Android is a development platform that includes the operating system, core applications and middleware for mobile devices. Android is the leading mobile operating system worldwide, controlling the mobile

OSS market with 74 percent in the market share. Developing machine learning communication systems for the mobile application using an Android platform maximizes usability among the mobile devices.

TensorFlow detects objects in real time from a camera feed using pre-trained and optimized models, such as the IBM Watson method that is tested in the thesis. TensorFlow can be used to identify classes of objects, including letters, animals, people, activities, places, and plants. The application allows image recognition with a mobile device without the use of large computational resources or having to be tethered to a cloud service. Each image is processed on the mobile device. This setup increases the device mobility and greatly reduces the processing time required for FSO communications, therefore improving throughput.

2. IBM Watson Machine Learning

Mobile devices can learn from data that is both unstructured and unlabeled to become smarter with the use of machine learning. IBM Watson Machine Learning is a cloud-based computing platform to train analytical models and build neural networks to use in multiple applications. IBM Watson Machine Learning method can be used to build analytical models and neural networks to deploy on mobile devices. These analytical models can be trained using images of MIMO laser communications to identify patterns. IBM Watson Machine Learning method trains the models with the data supplied and applies estimators and optimizers to the data.

The term “artificial intelligence” is used to define all methodologies to simulating intelligence. Artificial intelligence can be divided into two subsets: machine learning and deep learning. Machine learning is a subset of artificial intelligence that uses algorithms whose performance improves as it is exposed to more data over time. IBM Watson Learning method is included in this subset. Deep learning is a subset of machine learning concerned with algorithms in which multilayered neural networks are used to simulate the learning process.

3. Open Source Computer Vision

Open source computer vision library (OpenCV) is an open-source computer vision library created to provide a common resource for diverse computer vision applications. The machine learning and open-source computer vision software library consist of more than 2,500 computer vision algorithms. The OpenCV consists of a modular structure that includes libraries used for the use of image processing, object detection and identification, tracking, deep neural networking, machine learning, and image stitching applications. The library provides a common infrastructure for computer vision applications.

OpenCV can be implemented within an FSO communications system for the detection and identification of laser patterns. The system camera can capture the laser patterns as images for the application to process. OpenCV can be used to train an artificial intelligence algorithm to detect the difference in light patterns to decipher data send over free space. OpenCV can be coded in C++ and Python to support real-time computer vision applications on Android mobile devices. Most of the classification and regression algorithms used by OpenCV applications are implemented as C++ classes.

E. SUMMARY

FSO communication systems provide a promising solution for the increased bandwidth demands for mobile communications. FSO communications operate outside of the traditional RF boundaries. Compared with traditional radio waves, FSO possess multiple advantages for wireless communications FSO communications can provide enormous bandwidth, license-free spectrum and electronic warfare capabilities, such as jamming resistance, LPI, LPD, and LPE.

The primary benefit of this study is to implement an operational FSO model using handheld mobile smart devices. The model will consist of a transmitting device and a receiving device. A secondary benefit is the improvement of protocols and methods of existing two-way communication within FSO systems. The system will use the QR code concept for transmitting FSO signals from the transmitter to receiving devices. The use of a similar design as QR codes allows for the transmission of multiple characters of information at a time and provides signal location and orientation to receiving devices.

A goal is to determine how efficient real-time computer vision and image recognition methods are at FSO communications using handheld mobile devices. The system will process optical communications by collecting real-time images of the communications using a mobile device camera. The images will be detecting and classifying communications in real-time.

This chapter discussed aspects of FSO and its relation in the EM spectrum, advantages and challenges, environmental effects, and optical components. Also, the chapter reviewed electronic warfare and the use of handheld mobile devices within the military. Growth in FSO technologies makes FSO a viable means for reliable and capable communications in RF-denied environments. In addition, we discussed how neural networks and real-time computer vision methods are used for image recognition and detection in mobile FSO communication systems.

III. EXPERIMENTAL ARCHITECTURES

This chapter introduces the methodology and experimental architectures for mobile FSO communications systems. The experimental concepts in this chapter demonstrate the feasibility of FSO communications using mobile devices by identifying system capabilities and limitations. This chapter outlines the hardware and software needed to implement FSO communications on a mobile device. In addition, this chapter identifies the architecture and processes needed to achieve fully operational performance in real-world scenarios. Experimental architectures were created to demonstrate the methodology and proof-of-concept for mobile FSO communications using low-cost COTS components.

There are several challenges to consider when implementing an FSO channel versus a traditional RF system. We present a method for establishing a bi-directional communication channel between mobile devices utilizing the custom LED panels and the built-in camera on mobile phones. We demonstrate a directional line-of-sight (LoS) communication channel with potential speeds superior to RF communication methods.

The primary deliverables from this study are a system architecture to support optical communications between mobile devices and associated algorithms and heuristics. This study sought to build a theory in answer to the following research questions:

1. How might we create a system architecture to support optical communications between mobile devices within a non-RF environment?

2. How do we develop an architecture using COTS components to support mobile optical communications?

A. MOBILE MIMO COMMUNICATIONS USING TF-LITE WITH IBM ML

The MIMO proof-of-concept is to demonstrate a method for multiplying the capacity of an optical link using multiple lasers for transmission and multiple optical receivers to exploit multipath propagation. Multiple receivers at different locations within LoS are able to receive the same MIMO signal from originating transmitter. MIMO offers the potential to transmit data in parallel between a source and multiple receivers. This system demonstrates the capabilities and limitations of FSO communications within an RF denied environment, especially with regards to the use of low-power, COTS handheld mobile devices. The system leverages machine learning software to recognize, receive, and decode encrypted laser patterns.

1. Free Space Optics Transmit Process

For the proof-of-concept system, user input is entered via a smartphone application with a text-based user interface. This data is encoded and a sequence of patterns are modulated onto eight lasers configured in a grid pattern, using on-off keying (OOK). A ninth laser is also included for timing purposes, making it a 3x3 array. These patterns are projected across free space onto a smooth surface in LoS of the receiving device.

The OOK of each laser operates independently from the other lasers. The combined modulation from the nine laser OOK creates a light pattern that can be interpreted by multiple receivers. User input is provided on a smartphone by typing alphabetical characters into the user interface. The alphabetical characters are then translated into 8-bit binary ASCII characters. These ASCII characters are transmitted over free space by modulating a laser array to a surface for the receiving mobile device to interpret.

The receiver smartphone detects incoming data patterns using the on-board camera. The machine learning based image-recognition software, running in the phone, associates the patterns using the channel library. The decoded data is then passed to a display module that outputs text within the smartphone application. A schematic is provided in Figure 6.

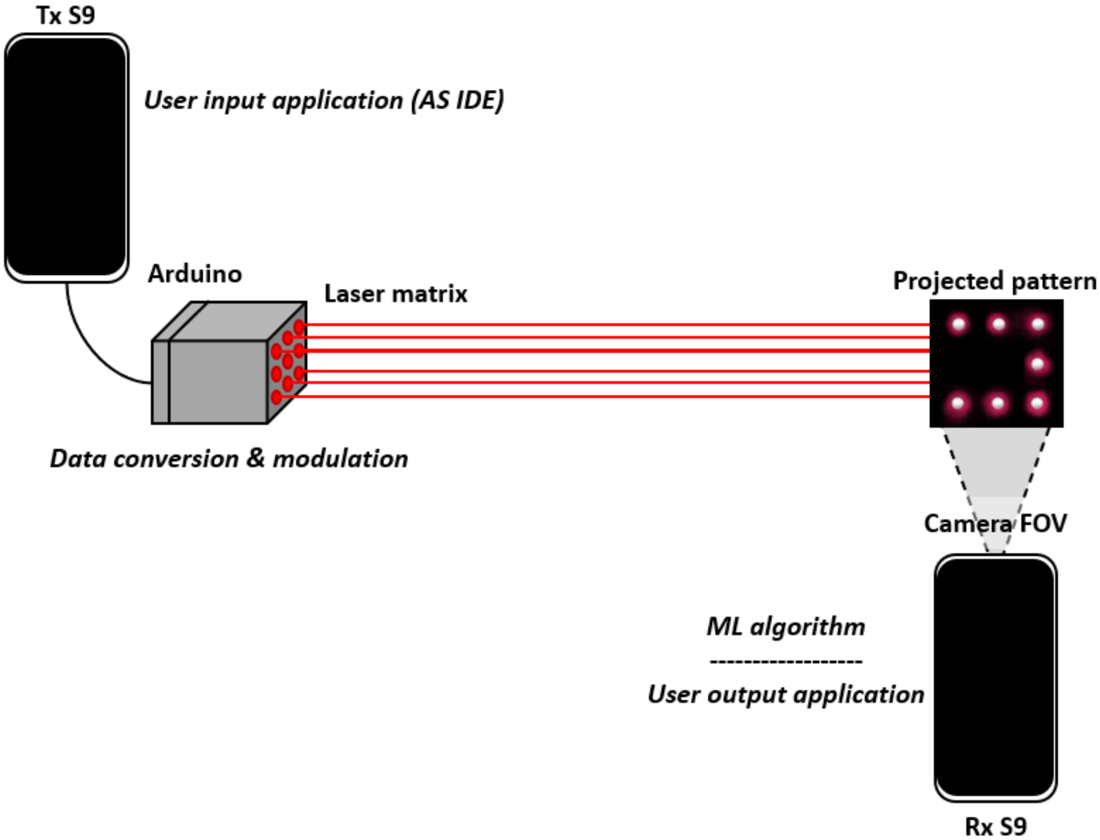


Figure 6. Proof-of-concept system diagram

a. Light Pattern Model Development

The IBM Watson Machine Learning method uses custom-trained models to detect objects. The custom-model used photographs and 3D designs of MIMO laser configurations. The images were uploaded to the IBM Watson Cloud to be trained as an SSD model. Before training a model, each image is annotated by drawing bounded boxes over each pattern in an image. An example of image types used to train an object detection model is presented in Figure 7. The image on the left in Figure 7 is a 3D model representation of light patterns for the letters ‘o’ and ‘l’. The image on the right is a picture of light patterns representing the letters ‘e’ and ‘h’.

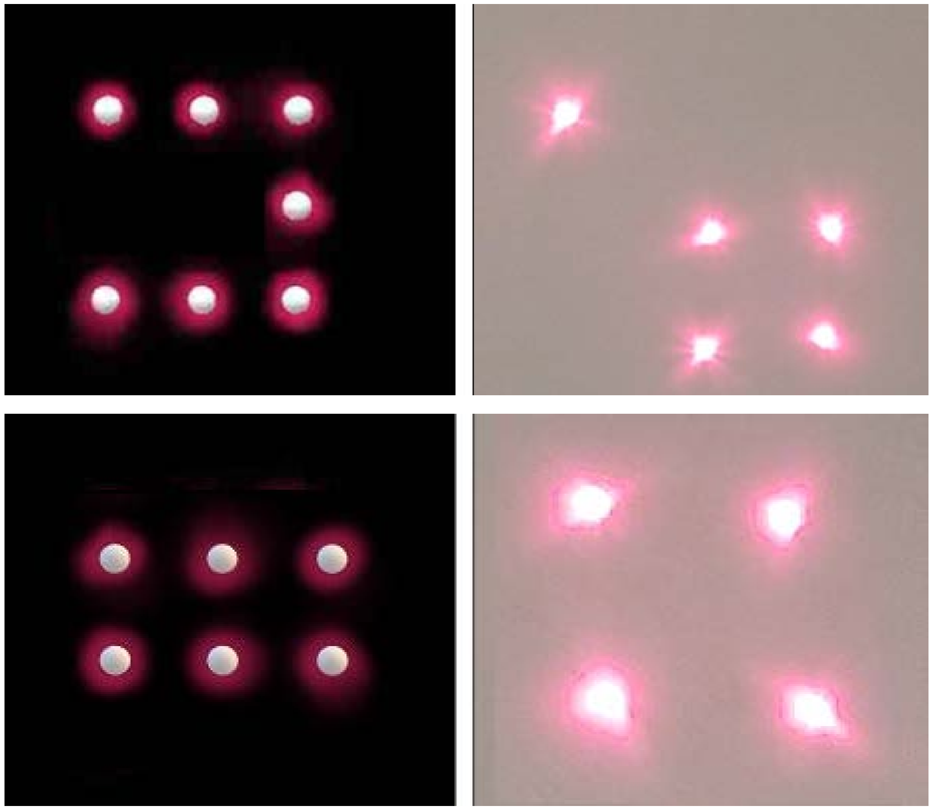


Figure 7. Example 3D and photographic image types used to train the object detection model of light patterns.

IBM Watson ML cloud should be provided at least 50 training images per label for a model to successfully identify an object. Model identification and classification results improve by increasing the size of the training dataset. The model used in this experiment consisted of 200 Joint Photographic Experts Group (.jpg) images for each alphabetical letter used in the experiment. All .jpg consisted of 300x300 pixel images representing light patterns that correlated with an 8-bit American Standard Code for Information Interchange (ASCII) character sets.

An ASCII character set of eight-bit characters allow for 128 printable and control characters. The ASCII scheme in the FSO mobile communications system is used to translate the transmission light patterns to human text. The ASCII based experimental architecture contains letters from A to Z, upper and lower case, the numbers from 0 to 9. While the experiment limited light patterns to characters, the same method can be used to portray control characters or multiple bytes based on a different pattern scheme. Control characters are designed to control hardware devices, issue instructional commands, and provide error correction.

The laser patterns developed used in this system store less data than traditional QR codes in the same area of space. A version 40 QR having 177 rows and 177 columns can store 23,648 bits or 4,296 alphanumeric characters [14]. While the data capacity for traditional QR code patterns is higher, they are not capable of continuous high-speed data transfer. Traditional QR codes are not suitable for continuous or bidirectional high-speed data transfers. This is because they were created for one direction single transfer of data where latency is not important.

b. Methods of Object Detection

The training process of the machine learning application is done by the IBM Watson Machine Learning cloud to identify and recognize light patterns as alphabetical characters. The trained machine learning model is then loaded to an Android smart device. Three methods of object detection are discussed in this thesis: the faster region-based convolutional neural networks (R-CNNs), YOLO, and SSD. Each method uses deep machine learning-based objection detection. We considered accuracy and speed to determine which object detection methods to be used. R-CNN object detection method is computationally more expensive and has an average frame rate of 5 fps [15]. The YOLO object detection method can process between 45-155 fps depending on the graphics processing unit [16]. The YOLO training method results in a faster indentation algorithm than R-CNN and SSD, but YOLO is less accurate and has a higher error rate.

IBM Watson Machine Learning architecture was implemented using an SSD training method. SSD models work well with mobile devices due to their small file size and can execute quickly with medium accuracy. The diagram in Figure 8 demonstrates the SSD MobileNet model process. MobileNet is a term used to classify efficient models of convolutional neural networks run on mobile devices [17]. "MobileNets are based on a streamlined architecture that uses depth-wise separable convolutions to build lightweight deep neural networks" [17]. This streamlined architecture improves the mobile speed at the cost of object detection accuracy. This SSD method uses 300 by 300-pixel light pattern images and assigns an importance based on learnable weights and biases to the light pattern in the image. The images were assigned weights and biases to allow the FSO receiver to differentiate patterns.

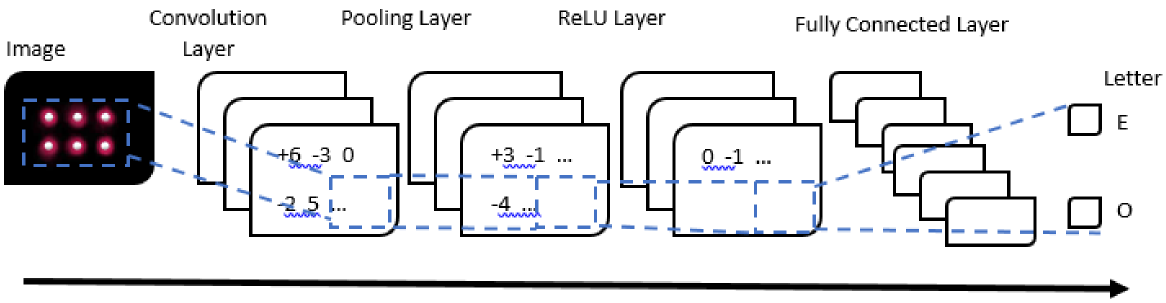


Figure 8. Diagram of the SSD MobileNet model process.

The light pattern model described in this chapter was trained on IBM Watson ML cloud method. Then loaded onto the mobile application used by the receiver. The model we trained is optimized for photographs of objects in the real world. This model works well with our mobile MIMO communications receiving system that uses a mobile phone's built-in camera. Training images were taken at multiple viewing angles. In addition, image file resolution, background, and lighting settings were controlled during the experiment.

c. *Smart Device and Android Software*

This experiment used a smartphone device (Samsung S9) connected via a USB on-the-go (OTG) cable to establish a communication link between the transmitting device. The USB OTG supports 480 Megabits per second (Mbps) in high-speed mode between the smartphone and transmit device. The USB OTG allowed the smartphone and transmitter to switch back and forth between the roles of host and device. The mobile phone can send data to the transmitting device and receive acknowledgments upon receipt of data.

A custom-made Android application was written in the Android Studio integrated development environment (IDE) for both the transmit and receive devices. The application allows user interaction with a text-based chat interface. The application can sense the transmitter device via USB OTG, establish the connection, and provides a user interface to send information. The transmitter application interface chat application contains a text window. The text window operates in a manner of a typical chat application for sending and receiving strings of information. User inputs include data entry and transmission control. The system is only designed for point-to-point communications, and there are currently no networking protocols implemented. Application data is passed to the Arduino board and a small sketch is used to modulate the ASCII characters onto the laser array.

d. Arduino Board and Software

Arduino is an open-source language consisting of functions used for building electronics projects. It consists of a physical programmable circuit board and IDE software. The Arduino board is capable of reading inputs from the USB OTG and outputting 1s and 0s via its 14 I/O pins. The Arduino IDE uses the Arduino programming language. This programming language is a simple hardware programming language, similar to the C language. The code was written in the Arduino IDE, and then uploaded on the Arduino board for execution.

In software, it is important to maximize program effectiveness by increasing execution speed and limiting latencies. A programing language like C can provide a significantly improved execution speed compared to other languages such as python, java, Visual Basic. Arduino language is a set of C like functions that can be called from the Arduino IDE software that are passed directly to a C compiler.

The Arduino software used a data dictionary to translate the text characters to an 8-bit binary ASCII representation. A ninth bit was added to each group of eight bits to be used for timing and error correction in future experiments. An Arduino architecture with an eight-bit laser array to transmit data has a maximum transmission rate is 115.2 Kilobytes per second (KBps).

e. Laser Transmitting Device

The laser transmitting is responsible for transferring the data into free space where one or more receiving device(s) can collect and interpret the data. The transmitting prototype consists of nine Adadfruit Class 3R 5 milliwatt laser diodes. The laser diodes are placed in a three by three pattern. The eight parameter diodes light up in assigned patterns to represent 8-bit binary ASCII characters. The characters are then received by a smart device optical lens. The power of the laser beam is a major factor in its ability to transmit over long distances. Due to safety considerations, this experiment's laser transmitter was limited to an eye safe class 3 laser diode.

The transmitter box houses nine individual lasers. The benefit of MIMO communications is the increase of capacity in terms of very high spectral efficiencies and an increased reliability. Reliability can be improved by system error correction encoding and MIMO pattern synchronization to identify duplication of data at the receiving system. The nine lasers are aligned in three rows with three lasers in each row. The system communication distance is constrained to less than 100 feet because the beam output power of laser is limited.

2. Free Space Optics Receive Process

The FSO receiver for this experiment consists of a smart device and associated software. The software used was a tailor Android software application, TensorFlow Lite library, and a model created using the IBM Watson ML cloud. The TensorFlow Lite application detects and captures an image of the 8-bit binary ASCII data projections. TensorFlow Lite library and ML model are used to identify the image as alphabetical characters. The characters would then be displayed to the user via the smart device chat screen.

a. Smart Device and Android Software for the Receiver

The Android S9 device can capture a maximum of 960 frames per second (fps). However, it can only maintain 960 fps for two seconds. After that, the camera maintains 30 fps. These speeds are acceptable for this proof-of-concept design, but are not adequate for fully operational system. Typical commercially sold FSO systems have transmit rate of 10 Gbps. Experimental FSO systems can compete with fiber optics systems and have demonstrated a successful transmission rates of 120 Gbps over an FSO link [18]. An optimal optical receiver will need capture and process data at these high rates. The receiver software was built using an Android IDE.

Figure 9 demonstrates the transmission process from the transmitter to receiver. The flowchart in Figure 9 identifies the method by which input is received via a mobile application and sent to a connection transmitter. The mobile device is connected to the Arduino using an OTG cable to establish the communications link with the transmitting device. The transmitter displays laser light patterns on a flat surface using a 3x3 laser array. The mobile built-in camera is used to capture the light patterns and use the object detection application and an SSD MobileNet model for identification. The information is then displayed to the user via the mobile device. In Figure 9, the left image displays the system properly identifying a character from light pattern projected on a wall.

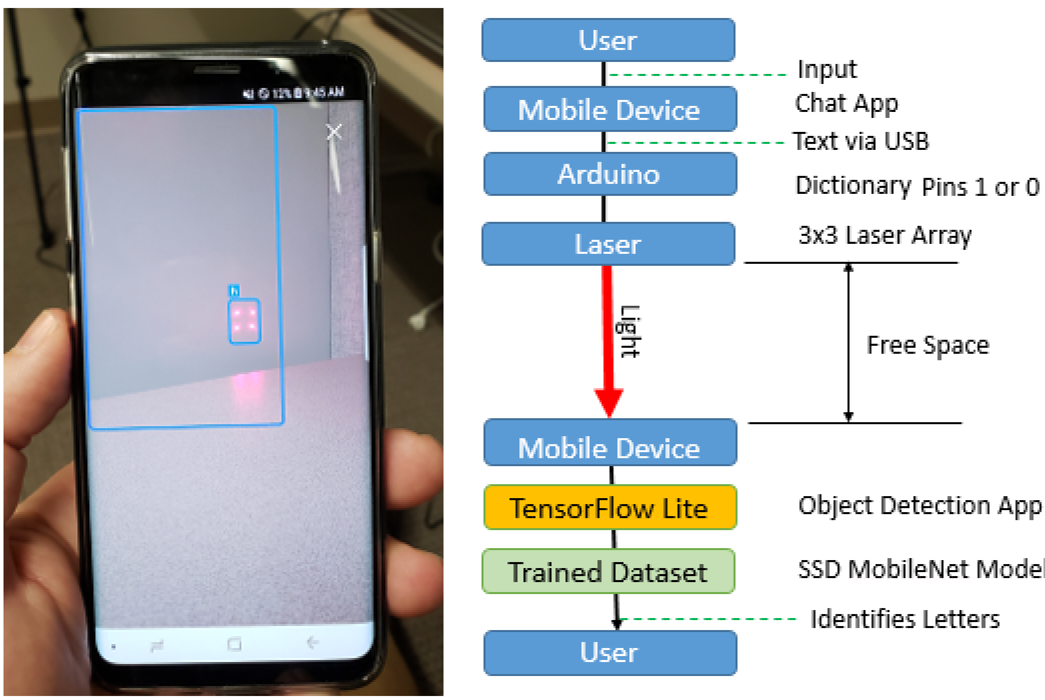


Figure 9. Transmission process from the user to user and character identification snapshot.

As previously discussed, the transmitting system uses a custom-built chat application to receive user input. Figure 10 shows the application's user interface. The user types "Hello World!" and presses the enter arrow. The text is then translated and displayed in the 3 by 3 optical array box found in the top on Figure 10.

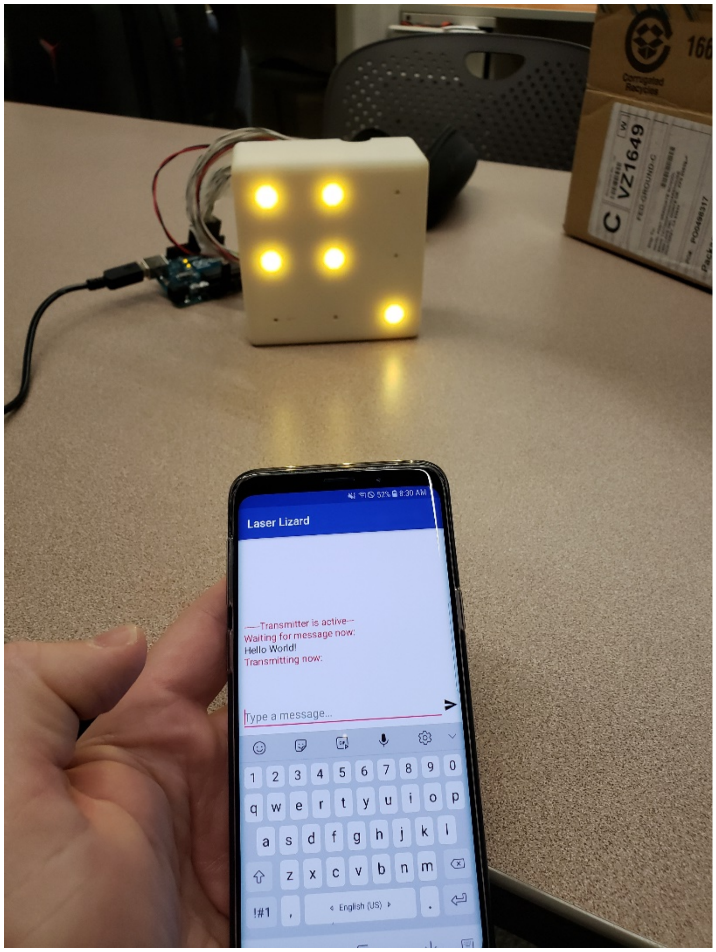


Figure 10. Demonstration of the custom-built chat application.

b. TensorFlow Lite library

A limiting factor for mobile devices is storage and processing the images. While TensorFlow Lite optimizes the computationally demanding operations used in deep neural networks, the process still requires significant processing time compared to traditional methods, such as using a photodiode receiver to receive FSO communication. TensorFlow Lite is an open source software library for numerical computation using data flow graphs. As a machine learning tool, TensorFlow Lite uses trained machine learning models on mobile and embedded devices while maintaining proficient runtimes and accurate neural reasoning with minimum latency [19].

The Android device's neural network application programming interface interfaces with TensorFlow Lite accelerate model operations. TensorFlow Lite provides a lower latency, higher throughput, and light weight framework for trained machine learning models. The models are then compressed and uploaded to mobile devices [19]. TensorFlow Lite receives the camera input using the functions defined in the file CameraActivity.java. The CameraActivity.java file depends on Android Studio AndroidManifest.xml to define the camera settings.

c. Data Transfer Environment

The FSO communications system can operate in environment where the RF-spectrum is constrained and/or denied. FSO communications must operate regardless of light conditions, outside element, and noise. The proof-of-concept architecture was designed to test ability and determine a protocol for mobile-to-mobile FSO communications and not test the systems capabilities under hostile environmental factors.

The system components were run and tested indoors. The system was run in a dark environment to illuminate the laser patterns for the camera to focus. In addition, the set conditions limit the objects detected and identified by the machine learning application. Ideally, the system will transfer data under harsh environmental conditions. In addition, the system would not require the receiver or transmitter to be stationary. This system would continue to transfer data between a singular transmitter to multiple receivers while receiver or transmitter is mobile. The system would require the communicating mobile devices to remain within line-of-sight.

B. MIMO COMMUNICATIONS USING OPENCV WITH ML

This section identifies the design and implementation of the MIMO FSO system using a machine learning architecture. It addresses a conceptual design for synchronous communication and an experimental design to demonstrate the design feasibility. The purpose of the conceptual and experimental designs is to demonstrate how machine learning algorithms can be used with FSO communications to detect and identify optical transmissions. This chapter identifies the transmission process, hardware and software requirements, the method of object detection, the receiver process for optical communications, and coding method to maximize error-free data communications.

a. FSO Transmit Process

The transmitter hardware consists of the same transmit hardware used in the mobile MIMO communications using TF-Lite with IBM machine learning architecture. The mobile device can receive user input via a chat application, processing the input, and send the input as 1s and 0s to an Arduino connected to an array of lasers. Users provide input by typing alphabetical characters into the user interface. The alphabetical characters are then translated into bits and transmitted one byte per pattern.

The Arduino is used control the ON/OFF position of the laser array based transmit the byte pattern. The laser array consists of a box containing 9 lasers in 3 by 3 pattern. Similar to the receiver described in Chapter III, Section A, the ON/OFF laser array modulation creates a light pattern that can be interpreted by multiple receivers. The byte pattern is then transmitted over free space to a receiving device(s) for processing and interpretation.

b. FSO Receive Process and Object Detection

In this section, we discuss how the FSO system receives FSO transmissions and how objects are detected and processed. The software programs used are Python, ImageAI, and OpenCV. ImageAI is a python library that is used by OpenCV to build a system with self-contained deep learning and computer vision capabilities. The deep learning algorithm must be trained before the FSO receiver can detection and process transmissions. The machine learning algorithm is trained to detect and identify light patterns as bytes. By this process, we have an end-to-end FSO system to transmit and receive bytes using a machine learning technique.

(1) YOLOv3 Network Architecture

The experimental design used you-only-look-once version 3 (YOLOv3). A key feature that differentiates YOLOv3 from previous versions is its ability to make object detections at three different scales. The detection is done by applying detection kernels on images of three different sizes at three different layers in the convolutional neural network. Each image is twice the resolution of the previous image. For example, if the first image size was 6 by 6 by 50 pixels, the next image would be 12 by 12 by 50 pixels, and the last image would be 24 by 24 by 50 pixels.

The YOLOv3 CNN architecture consists of multiple CNN layers for multi-scale object detection. YOLOv3 requires darknet53. Darknet53 is a CNN made up of 53 convolutional layers trained via ImageNet. The image in Figure 11 is a schematic of the YOLOv3 network architecture. The Figure 11 schematic highlights the key sections of the YOLOv3 CNN: the backbone, known as the darknet53, and the YOLO detection layers. The 53 YOLO layers plus the 53 darknet53 convolutional layers makes up the total 106 convolutional neutral network layers used for modeling. The additional networks improve the YOLO's latest version detection capability, but cause it to run slower than previous versions that ran using only 53 network layers.

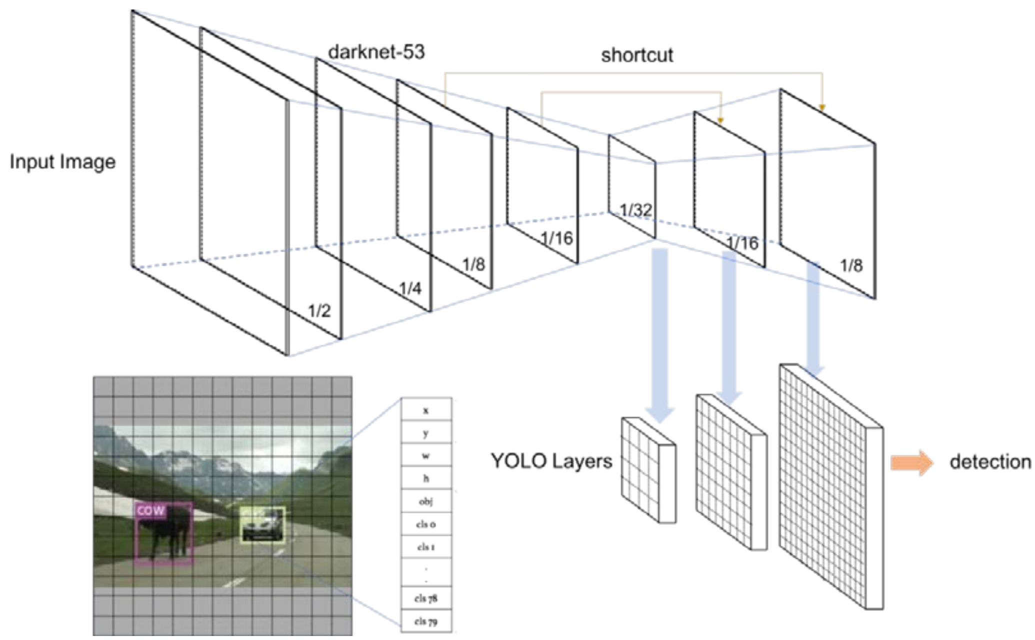


Figure 11. YOLOv3 network architecture schematic. Source: [20].

The first layer extracts the input from the image and breaks the data into residual blocks [20]. Each residual block is made up of paired convolutional layers. The paired convolutional layers have shortcut connection to each other [20]. Each pair have a 3x3 and a 1x1 layer dimension. The spatial resolution of last darknet53 layer 3.125% the size of the original image.

Following the initial 53 CNN layers, YOLOv3 makes prediction at three layers. The first prediction is made at the 82nd layer where the input image is down sampled by thirty-two. In Figure 11, the first prediction is done at 1/32 grid resolution of the input image. The second prediction is made at the 94th layer where the input image is down sampled by sixteen. In Figure 11, the second prediction is done at the one sixth grid resolution of the input image. The third prediction is made at the 106th layer where the input image is down sampled by eight. Likewise, the last prediction is run at one eighth grid resolution of the input image identified in Figure 11.

(2) ML Model Development

The machine learning model development process involved the creation of a custom object detection model. The development process of a custom detection model includes training and validating the model before model deployment. The system utilizes the Detection Model training class found in the ImageAI detection library. The Detection Model training class utilizes YOLOv3 to train object detection models from image datasets.

Currently, YOLOv3 is the only model type that can be trained using ImageAI. The training process generates a JSON file that maps objects labels and detection anchors in from the image dataset. In addition, a new model is created at the end of each cycle. In theory, the model refines itself as it progresses and every new model is better than its predecessor.

Before the machine learning algorithm can use a training model, the training data must be sorted and annotated. Proficient machine learning algorithms require a minimum of 200 images per object. Multiple objects can be annotated on one image. For the development of the models used for this experiment, the images were annotated by defining bounding boxes around the objects utilizing LabelImg software application. LabelImg used PASCAL visual object class (PASCAL VOC) to annotate the objects within the images. PASCAL VOC provides standardized image data sets for object class recognition in an XML format.

(3) Laser Array Design

For the proof-of-concept, user input is entered via a smartphone text-based interface. This data is encoded, and a sequence of patterns are modulated onto eight lasers configured in a grid pattern using OOK. A ninth laser is also included for timing purposes, making a 3x3 array. These patterns are projected across free space onto a smooth surface in LoS of the receiving device.

The transmission device consists of nine Adafruit Class 3R 5mW laser diodes in a small fabricated box. The system communicates using a 3x3 array if laser diodes. The machine learning model can be trained to identify individual laser diodes as a 1 or 0. The Arduino is used control OOK of the laser array based on whether the bit is a 1 or 0.

By grouping eight bits together in a light pattern, the system can transmit 1 byte per cycle. In general, 1 bit can represent two states or patterns. Each addition bit doubles the number of patterns. The system uses eight bits for data transmission; therefore, the system is capable of 256 patterns. The eight-bit pattern is large enough to transmit one typed character, e.g. 'a' or 'B' or '#'. The system uses the ASCII character encoding standard before transmission and each character is transmitted in a byte.

(4) Receiver Output

The FSO receiver used in this MIMO communications experiment using OpenCV and machine learning consists of the same hardware as the first experiment using TensorFlow-Lite and IBM machine learning algorithm. Both experiment receivers consisted of an Android S9 device smart device. The receiver software was built using an Android IDE. The Android software application detect the one byte per cycle laser projection and transform the signal to character representation using ASCII code. After the characters have been translated, they will be displayed on Android application display screen. Therefore, without an error-correction encoding method, 100 bytes received is presented in 100 characters in the receiver chat application.

Like the previous experiment, there are hardware limitations that affect the system performance. The Android S9 smart phone maximum frame capture rate is 960 fps at a limit of two seconds. Afterward, the maximum frame capture rate is reduced to 30 fps. Viable FSO receivers will need to capture data at higher frame rates.

c. Maximum Error-Free Data Communications

Mobile system communications generally bottleneck at the transmitter due to bandwidth capability. A mobile MIMO communications system must maximize its data rate to be a viable tool for mobile device communications in a hostile wartime environment. The communications need is for maximizing given bandwidth. The Shannon-Hartley capacity theorem can be used to determine the maximum data rate that can be transmitted over the MIMO communication system without errors at a specified bandwidth and average noise power. The noise power is the total unwanted disturbance energy within the communications.

From the Figure 12, it is observed the capacity of the channel is dependent on the number of MIMO channels, bandwidth, transmission power and the noise the signal ratio. To raise the capacity of the MIMO communications system, we can increase bandwidth and transmit power. In addition, we can increase the number of channels used the MIMO architecture and lower the channel noise. In a real-world scenario, these variables may be fixed. The Shannon-Hartley capacity theorem provides a way to maximize given variables. The theorem is used to transmit binary digits at higher speeds with a randomly low amount of errors. The theorem's coding method is used retrieve the greatest average speed with a randomly low amount of errors [16].

$$C = M * B * log_2\left(1 + \frac{S}{N}\right)$$

C	Capacity of the channel in bits per second (bps)
M	MIMO number of channels
B	Bandwidth (Hertz)
S	Average transmit power (Watts)
N	Noise on channel (Watts)
S/N	Signal to noise ratio

Figure 12. Shannon-Hartley capacity theorem and descriptions. Source: [16].

C. SUMMARY

This chapter described our methodologies for transmitting and receiving MIMO FSO communications on mobile devices. The systems design solutions relied on readily available COTS components and low power devices to operate. We further described the application of machine learning methods running different software implementations. The software used in the experimental architectures included TensorFlow-Lite, SSD built using IBM Watson suite, OpenCV, and YOLOv3. For both experiments, we used the TensorFlow Lite library to process our image captures.

Our experiment architectures provided methods and details for the implementation of a functional FSO communications system with intergraded machine learning capabilities for laser pattern projection detection and identification. Key ideas in machine learning implementation on mobile devices and FSO mobile communications identified in this chapter are tested and analyzed in the next chapter.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. TESTING AND ANALYSIS

The purpose of this thesis was to examine how FSO communication systems would perform between mobile devices. It further examined whether FSO communications could be improved with machine learning algorithms. We tested and evaluated mobile FSO communication methods using COTS components and low power devices to operate. In this chapter, we present our proof-of-concept system design and implementation, and testing results. Specifically, we examined system performance on time and memory requirements as well as its accuracy on detection of anomalous or malicious activities. We provided success rates of our system against simulated benign and anomalous data, as well as an investigation of our system's theoretical success using extrapolation of our initial results.

A. MACHINE LEARNING EXPERIMENT USING TF-LITE AND IBM WATSON

The TensorFlow-Lite with IBM machine learning experiment measured the success rate in identifying data sent by fast burst images using the mobile MIMO communications architecture identified in Chapter III, Section A. This section examined the machine learning technique used in real-time computer vision and image recognition. In this experiment we transferred ASCII characters from a transmitter to a receiver running a machine language algorithm and analyze the system's performance.

The training model labeled data sets to determine positive values for all the deep neural network (DNN) connected weights. This reinforcement learning process was run for 500 epochs. If the cycle result was negative, the method would adjust the weights in attempt to make a different decision on the following cycle. The loss function using cross entropy is minimized with respect to the model weights.

As seen in Figure 13, our TF-Lite IBM ML experiment resulted in a 1.82 total loss over 500 training cycles. Notice in Figure 13, there are high disparities between the test loss and training loss lines during training cycles. The gray line indicates the loss in training model that used the test dataset. The blue line represents the loss in the training model that used a training dataset. In a better predictive model, display test loss would more closely aligned with the training loss dataset. The graph shows a steady decrease in the total loss resulting in a score of 1.82 over 500 training cycles.

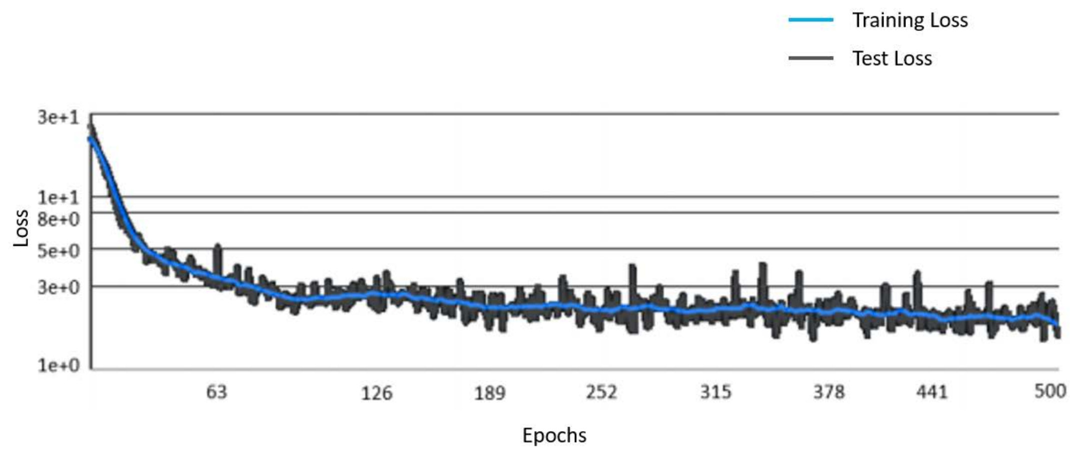


Figure 13. Loss during SSD MobileNet model training.

There are several parameters which can be considered to improve DNN performance. The amount of training data processed is a key parameter for DNN performance to properly identify objects. A DNN requires multiple images to develop virtual neuron mappings and assigns random numerical weights between these connections. There were 50 images with predefine labels for each letter used in this experiment. The experiment could benefit by using additional labeled training set images.

Typical DNNs can consist of hundreds to thousands of images with predefined labels. This experiment identified shortcoming in the training method using the IBM Watson ML, the experiment's training sets were limited to 50 predefined images with labels for each alphabetic letter used. The limitation in training sets negatively impacted our training and identification results.

B. MACHINE LEARNING EXPERIMENT USING OPENCV AND IMAGEAI

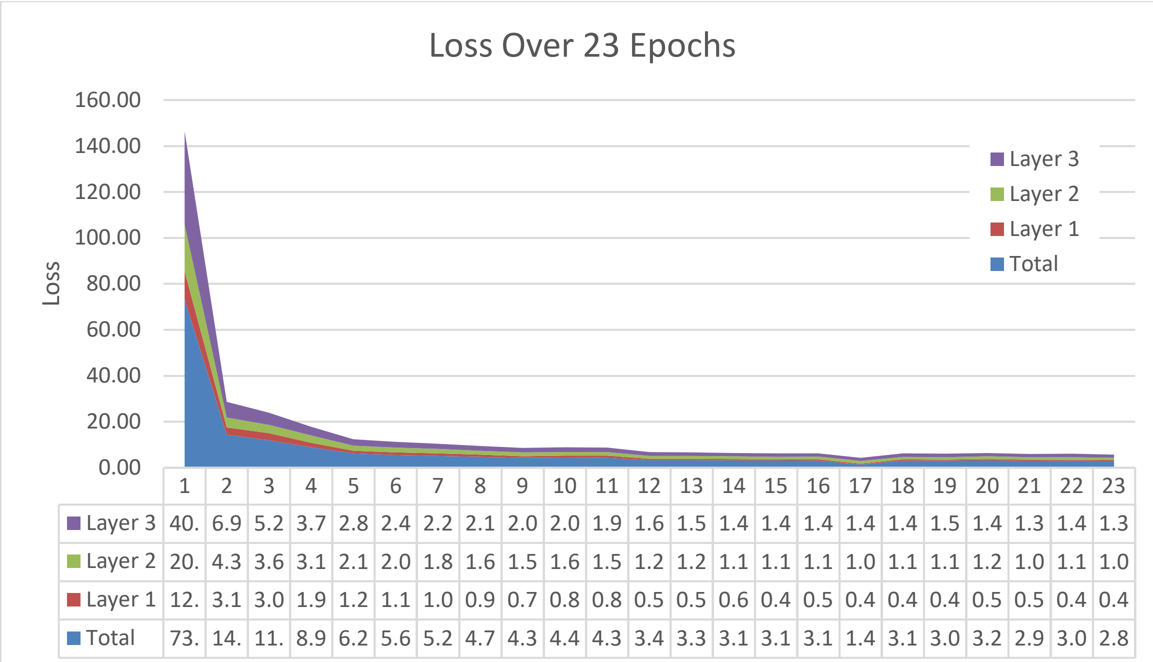
This experiment was written in Python with Jupyter Notebook and required OpenCV, Anaconda Keras version 2.3.1, TensorFlow version 1.15.0, TensorFlow-GPU version 1.15.0, and ImageAI version 2.1.5, and YOLO applications. This experiment was built from the experimental architecture discussed in Chapter III, Section B. The object detection algorithm used for this experiment was YOLOv3. This section begins with a detailed presentation of the experiment design and the machine learning model used in the testing and analysis. This section then presents a detailed analysis of the system's machine learning ability to detect and identify laser patterns captured by a mobile device. We conclude with performance analysis of object detection loss, precision probability, model training weights, and model detection speed measurements. The experiment tested an artificial intelligence model for the detection of FSO light patterns from captured images.

The experimental design used two laser patterns to transmit. Both laser patterns were optical representations of eight bits. The first set of bits were 11111111 and the second set of bits transmitted were 10010011. The dataset consisted of 250 images of each pattern and was used for training and testing a machine learning model to detect light patterns. Most machine learning algorithms use pre-trained models. Pre-trained models can save process time and increase system predictability results. However, the light patterns used in this experiment were unique and pre-trained model were not available. Therefore, transfer learning was not used and the model training started with three warmup experiment cycles.

The process of creating a trained machine learning model was computationally expensive. The training could be not run on a typical laptop and required additional processing power over a prolong period of time. As a result, model training was done using Google Colab, a free to use research tool for machine learning, using a Tesla K80 GPU. The initial plan was to run the training model through 200 training cycles, epochs, for all 256 different patterns. This method can take several days to complete.

Due to computational processing and server time limitations, the experimental model used two light patterns with 360 images for testing. An addition 40 validation images were used after the model training completed. Typically, 90% of images are used for training and 10% for validation. Before training, the dataset was reviewed manually for discrepancies in object labels. We used images collected at multiple angles and distances to allow the machine learning algorithm to identify light patterns from different vantage points.

As seen in Figure 14, the object loss was calculated after each training cycle. The model ran twenty-three epochs and required five hours to train. The results indicated the more training cycles we iterate, the lower our object loss got and the better our model became of object detection. Eventually the returns gained from training cycles diminished and continued training may not have improved the model's object detection ability. The three layers depicted in Figure 14 were defined in Chapter III under the YOLOv3 Network Architecture subsection.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Layer 3	40.	6.9	5.2	3.7	2.8	2.4	2.2	2.1	2.0	2.0	1.9	1.6	1.5	1.4	1.4	1.4	1.4	1.4	1.5	1.4	1.3	1.4	1.3
Layer 2	20.	4.3	3.6	3.1	2.1	2.0	1.8	1.6	1.5	1.6	1.5	1.2	1.2	1.1	1.1	1.1	1.0	1.1	1.1	1.2	1.0	1.1	1.0
Layer 1	12.	3.1	3.0	1.9	1.2	1.1	1.0	0.9	0.7	0.8	0.8	0.5	0.5	0.6	0.4	0.5	0.4	0.4	0.4	0.5	0.5	0.4	0.4
Total	73.	14.	11.	8.9	6.2	5.6	5.2	4.7	4.3	4.4	4.3	3.4	3.3	3.1	3.1	3.1	1.4	3.1	3.0	3.2	2.9	3.0	2.8

Figure 14. Loss during model training displayed over 23 cycles.

The YOLOv3 model was trained by calculating the object loss. The loss function training score relates to the YOLOv3 detection performance. Figure 15 displays the object loss function used by YOLOv3 to make its predictions. The loss function is comprised of the localization loss, confidence loss, and classification loss. The localization loss is determined by the errors in the predicted boundary box locations and sizes of boundary boxes. The confidence loss is dependent on whether the object was detected or not within the boundary box. The classification loss is measured for each cell in the image.

Localization Loss

$$\lambda_{\mathbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

$$+ \lambda_{\mathbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

where

$1_{ij}^{obj} = 1$ if the j th boundary box in cell i is responsible for detecting the object, otherwise 0.

λ_{coord} increase the weight for the loss in the boundary box coordinates.

Confidence Loss

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2$$

where

$\hat{C}_i$ is the box confidence score of the box j in cell i.

$1_{ij}^{obj} = 1$ if the j th boundary box in cell i is responsible for detecting the object, otherwise 0.

$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2$$

where

1_{ij}^{noobj} is the complement of 1_{ij}^{obj}.

$\hat{C}_i$ is the box confidence score of the box j in cell i.

λ_{noobj} weights down the loss when detecting background.

Classification Loss

$$+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2 \qquad (3)$$

Figure 15. Object loss function is the sum of localization, confidence, and classification loss. Source: [21].

There are multiple settings to account for when training an ML model. Figure 16 displays the experimental settings and their results. Intersection over union (IOU), object, and NMS thresholds can be modified to improve object detection probability and reduce loss. The IOU threshold for this experiment was set at 0.5. The IOU calculation is our desired minimum Intersection over Union value for the mean average precision (mAP) computation. The IoU is a measurement of the overlap between two boundaries, predicted boundary and real object boundary. This measurement is then used to determine how much our predicted boundary overlaps with the real object boundary. It can be set to values between 0 to 1.

The object threshold was set a 0.3. therefore, any detection with a confidence level less than the minimum of 0.3 was removed. The object threshold can be lowered to increase objective detection sensitivity. The Non-maximum suppression setting at 0.5 is used to set the minimum confidence level when multiple bounding boxes are detected for the mAP evaluation.

The evaluation results depicted in Figure 16 are from a model that iterated through 23 training cycles. We received an average precision of 0.9298, or roughly 93%, for the byte sequence 10010011. We received an average precision of 0.9677, or roughly 97%, for the byte sequence 11111111. The experiment had 0.9488, or 95% mAP. YOLOv3 used a mAP metric to measure the accuracy of object detection. The mAP metric is an average over all precision metrics used.

```
Starting Model evaluation....
Model File:  bytes\models\detection_model-ex-001--loss-0004.901.h5

Using IoU :  0.5
Using Object Threshold :  0.3
Using Non-Maximum Suppression :  0.5
10010011: 0.9298
11111111: 0.9677
mAP: 0.9488
===============================
--------------------------------------------------------
Iteration for 11111111: 0.9676923076923079
Iteration for 10010011: 0.9298185482718098
--------------------------------------------------------
```

Figure 16. Model evaluation settings and mAP results.

Of the tests given, the average speed for YOLO object detection and identification was 0.096298 seconds. That is roughly 10 fps and is significantly lower than earlier YOLO version, which advertised 45 fps. The additional detection layer and complexity of the YOLOv3, the processor used during the experiments can impact the result. The experiments were run using an Intel i7-9750H CPU. On an Android device operating at a frame capture rate of 30 fps, the YOLO application would create a point of congestion in the FSO communication system due to its average detection rate 10 fps.

Our experiment used the two core libraries, OpenCV and ImageAI, to create a custom detection model for identifying FSO light pattern projections over flat surfaces. Table 2 list the probability results of seventeen captured images using our trained model. The captured images consist of FSO light pattern projections. Half the dataset patterns consisted of the 11111111-byte pattern. The second pattern was made up of a 10100011-byte pattern that the model was not trained to detection. The test results showed an average probability of 83% that the predictions were correct for 11111111. The system identified the unknown patterns with an average 42% probability as 10010011. While incorrect, it did label the new pattern with the closest known trained class. By setting the object threshold weight to 50%, we can eliminate five of the six false positives.

Table 2. Probability and pattern box points of captured objects.

Object Count	Byte Detected	Probability Percentage	Pattern Box Points
1	10010011	40.1960969	[295, 375, 467, 472]
2	11111111	80.97246289	[293, 369, 492, 478]
3	10010011	39.79818821	[297, 374, 466, 474]
4	11111111	83.07163119	[294, 369, 492, 479]
5	10010011	42.86958873	[297, 373, 466, 477]
6	11111111	83.86185765	[294, 368, 492, 481]
7	10010011	49.32615459	[297, 371, 466, 483]
8	11111111	85.37793756	[292, 369, 494, 485]
9	10010011	52.54725218	[298, 372, 465, 486]
10	11111111	82.8428328	[291, 368, 494, 490]
11	10010011	44.39910352	[301, 374, 463, 488]
12	11111111	85.71779132	[293, 367, 495, 494]
13	10010011	33.39224458	[305, 372, 498, 494]
14	11111111	88.48442435	[305, 372, 498, 494]
15	10010011	32.93569982	[313, 375, 499, 495]
16	11111111	88.52020502	[313, 375, 499, 495]
17	11111111	95.8409667	[285, 333, 371, 398]

After the ML model was trained, its detection capabilities were tested using multiple images. We are able to use our trained model and make inference from the test images. For inference we use the pre-defined weights and a set object threshold. Figures 17-20 display the system's ability to identify object location and determine object probability. Overall, our model did a sufficient job properly detecting FSO patterns with only 23 cycle iterations. Figure 17 shows the byte pattern reflected from FSO transmission. The pattern was properly identified as 10010011 with a probability percentage of 98.392.

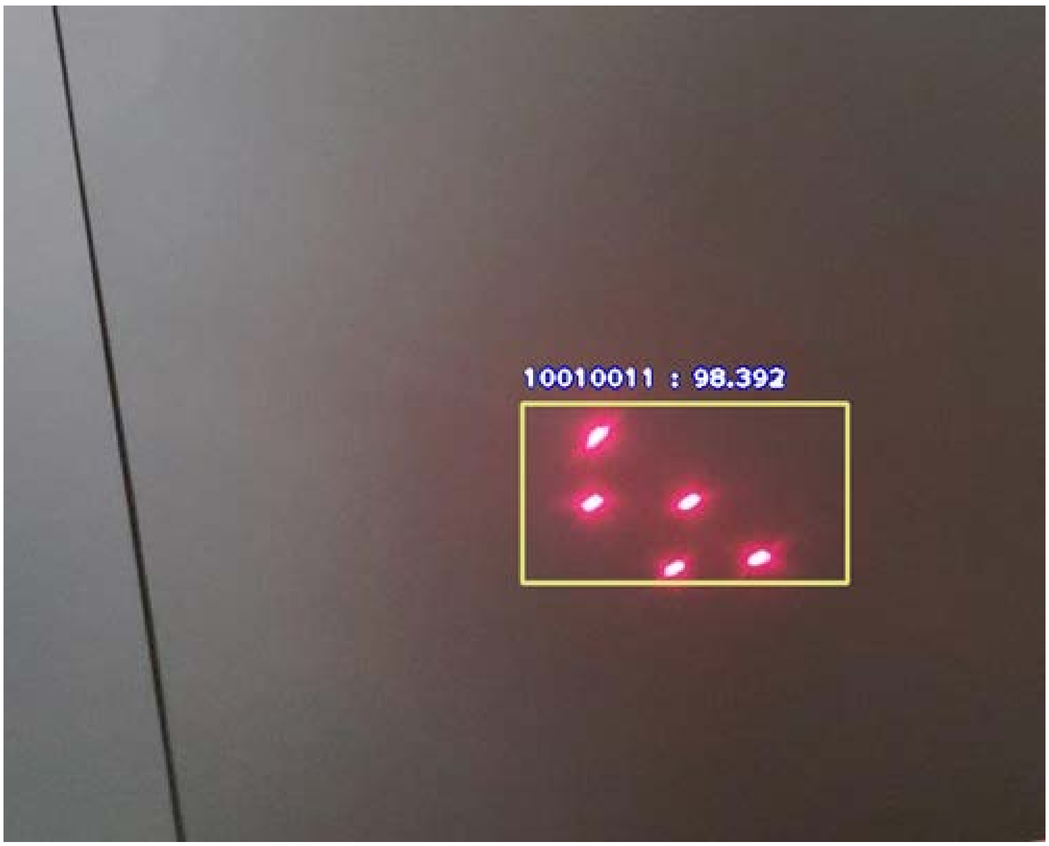


Figure 17. Byte pattern reflected from FSO transmission with a 98% probability.

In Figure 18, the byte pattern 10010011 was reflected from a flat surface using a 3x3 laser array. The transmitter was placed six feet from the reflecting surface and the receiver, Android device, collected the image a four foot distant. The pattern was properly identified as 10010011 with a probability percentage of 96.498.

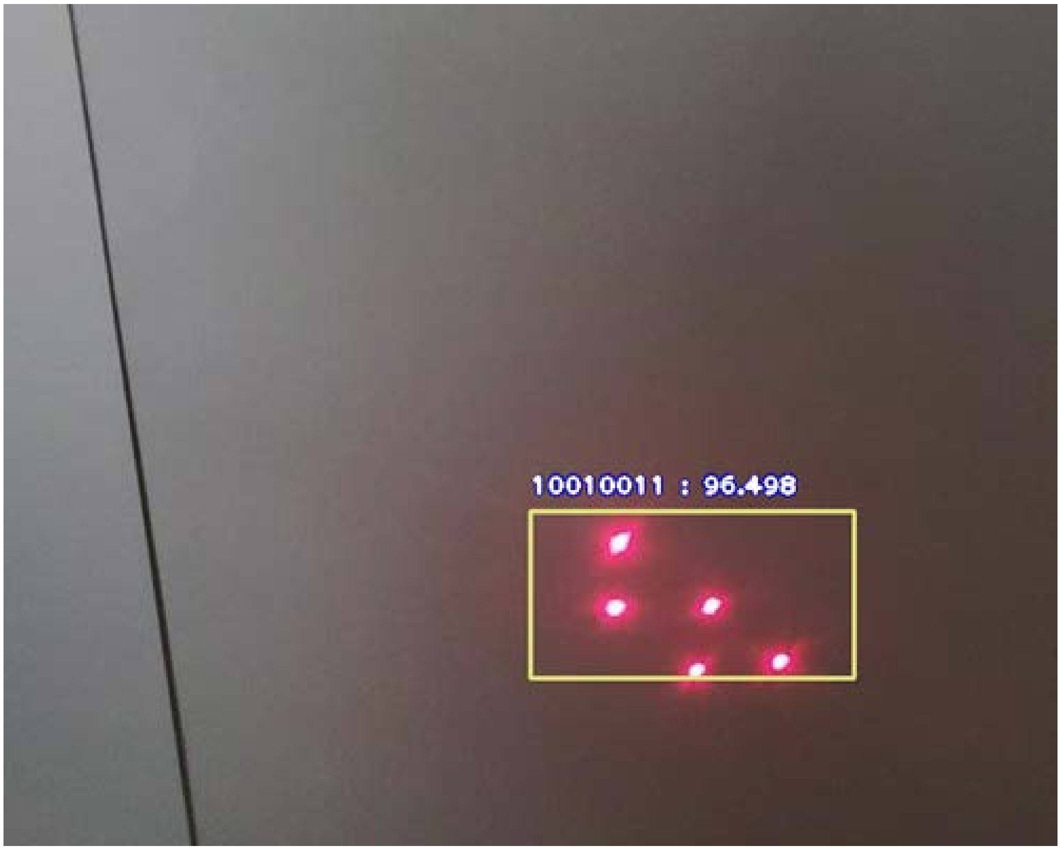


Figure 18. Byte pattern reflected from FSO transmission with a 96% probability.

The previous two images shown in Figures 17 and 18 were captured directly in front of the target object from a range of four feet. Figure 19 displays a different reflected byte pattern captured from an upward vantage point. The pattern was properly identified as 11111111 with a probability of 98.392.15, roughly 98%.

Figure 19. Image of a byte pattern reflected from FSO transmission with a 96% probability.

In Figure 20, the YOLO object detection system identified the captured image as both 11111111 and 10010011. The 11111111-byte pattern was identified with a 78% probability while the 10010011-byte pattern was identified at 36% probability. Again, an adjustment of the object threshold weight could eliminate false positive detections that have low probability scores.

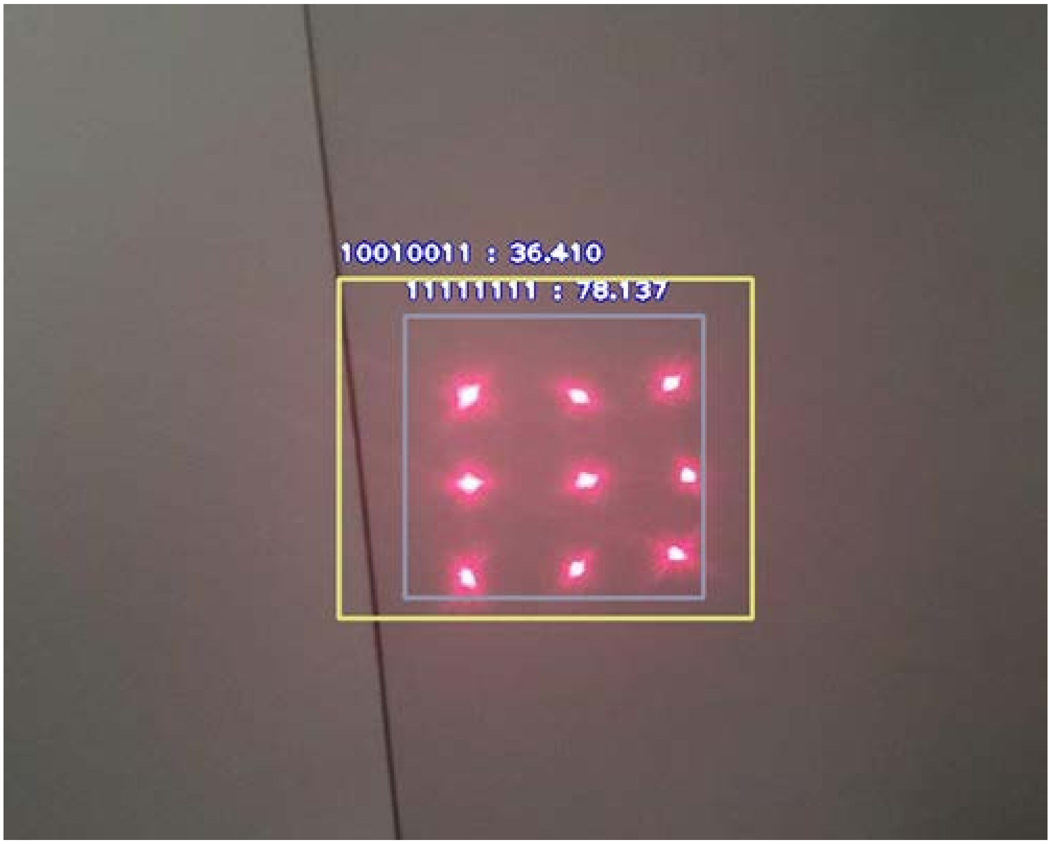


Figure 20. Example image of a false positive pattern detection.

A false positive detection may be the result of overfitting. The object detection system can product significantly more false positives if its model were overfitted during the model training process. The overfitting can happen when the model is trained on a complex pattern of data and our training error is significantly lower than our validation error.

For example, a distant outlier in the training data may result in the shift of class boundaries in the direction of the outlier. A shift in boundaries to include distant outliers can result in more false positives. In overfitting, the distant outliers are kept at the cost of having additional false positives. Greater defined classes would have dismissed most distant outliers as false negatives with a higher accuracy percentage.

C. SUMMARY OF RESULTS

In summary, this chapter discusses machine learning methods using two experimental models. We preformed testing and analysis using SSD and YOLOv3 models. The experiments demonstrated potential for ML algorithms to be used in FSO communications. It showed that communications were experiencing a bottleneck during the ML object detection process that would limit the frame transmission to 10 fps.

In addition, the experiment demonstrated high false positive rates. This may be a result of overfitting the models to the FSO light patterns. Additional model training and analysis may provide improvements in the object detection. The experiments were successful in demonstrating the ability to capture light patterns projected on a flat surface from multiple angles using a mobile device with ML technology.

V. CONCLUSIONS AND RECOMMENDATIONS

This chapter provides the conclusions for this work on developing a mobile FSO communications systems using ML models to detection light patterns. It includes pertinent observations made while testing ML systems and the mobile hardware platforms. This chapter also discusses potential future work with ML algorithms in communications and error correction.

A. DISCUSSION OF FINDINGS

The principal research question was to determine how we might create a system architecture to support optical communications between mobile devices within a non-RF environment. We examined how FSO communication systems performed using ML models to detect and receive data. Several conclusions are made from our findings, to include a comparison of traditional QR code designs and our developed diode matrix, ML processing performance outcome, and transmission and receiver benefits of our architectural design.

FSO communication using a 3x3 matrix diode transmitter displayed the capability to maintain continuous high-speed data transfer. Traditional QR codes are designed for one direction single transfer of data where latency is not important. They are not suitable for continuous or bidirectional high-speed data transfers. An increase in transmitter diodes would increase the transmission rate without an increase in design complexity or readable issues found in Traditional QR codes.

The FSO transmission tests using mobile devices and ML models demonstrated an average detection and identification rate of 0.096298 seconds per frame. The ML method required additional processing time and was only able to process 10 fps. The system performed at 33% of its hardware capability using a ML algorithm to process each frame. The ML method provided flexibility and capability to detection and identify light patterns into data from multiple angles and distances, but did impose significant performance limitations.

The performance loss can be mitigated by additional computing power or limiting heavy computational ML methods to initial detection and identification of an FSO signal. First, a heavy computational ML method can be used identify a signal's location, transmission speed, and map the signal patterns. Second, pass the parameters gained from ML method to a simplified image detection method that require less computing power and process time to complete.

A trained AI system that can detect and identify projected light patterns from multiple angles can benefit FSO communications ability to initialize and transverse data. Advantages to this approach allow optical communications to be performed beyond line-of-sight and without the receiver having to be directly in the path of transmission. In addition, the experimental architecture demonstrated the ability to have multiple receivers actively collecting data from the same transmission signal.

FSO communications channels require ML models to be trained on all patterns used to perform pattern recognition and decoding of optical patterns. Tests performed have shown ML models have difficulty in identifying patterns that can appear to be two smaller patterns. This difficulty demonstrated limitations of ML models in precisely identify similar patterns. The ML model may identify one object or pattern as multiple objects or patterns during on OOK cycle. Additional model training or introduction of different variables can reduce the identification errors and model overfitting issues.

B. RECOMMENDATIONS FOR FUTURE WORK

This thesis examined the feasibility of ML FSO communications using mobile devices. Future work can be done to improve the effectiveness machine learning algorithms and error correction methods in optic communications.

1. Initialize Communicates Using Machine Learning

Based the above results, an FSO communications system can benefit from a ML method for signal detection and identification. ML architectures provide a fast and accurate method for initializing FSO communication systems. However, the machine learning time requirement for processing images impedes traffic flow. A faster method processing signals is needed after a communications link has been made.

We recommend using a machine learning system for initial signal detection and follow-up with computer vision specific to motion analysis. Motion analysis in computer vision can be used to detect an object's movement. For example, a machine learning system identifies the position of a signal in reference to receiver's location. Then a motion analysis method would process changes in its state, such as on-off keying, faster than a trained machine learning model having to continuously reprocess the entire field of view capture by the receiving camera. A dedicated ML method for initializing communications can be optimized to recognize environmental factors based on image weights and signal loss to achieve minimal latency and maximum data rates.

2. Continued Study in Error Detection and Correction Methods

An error control technique can be used to enable dependable delivery of FSO communications over in adverse conditions, such as noise and fading. There are several error detection and correction methods that can detect communication errors and rebuild the original data. A forward error correction (FEC) method should be studied to improve the capabilities of the FSO ML communications architecture.

Shannon-Hartley and Reed Solomon encoding can be explored to provide error detection and correction ability to our system. The Shannon-Hartley capacity theorem is a theorem in forward error correction that can be used to determine the maximum data rate that can be transmitted over the MIMO communication system without errors at a specified bandwidth and average noise power. The Reed-Solomon encoder may be a viable method to easily add error correction. It adds additional bits to provide redundancy for the data sent.

C. CONCLUSION

ML technology for signal detection may be able to improve FSO communications capabilities using trained models to detect signal patterns. This thesis identified and tested two machine learning experimental methods for object detection and ability to process project light patterns as a method for communication. Both ML methods successfully demonstrated the ability to detect light patterns projected over flat surfaces.

The bandwidth capacity, an absence of spectrum license requirements, and robust security qualities of FSO make it an attractive technology for military communications. Our experiments demonstrated a method for mobile devices to communicate using FSO from multiple vantage points without direct line-of-sight. Continued work in this area is needed to improve the communication technique and increase bandwidth for a viable FSO system.

APPENDIX A. ARDUINO TRANSMIT CODE

```
##################################################
# Title: Arduino Transmitter Code
# The Arduino transmitter code provides the functionality between the Arduino device receiving data from an Android device and the laser array hardware.
# Written By: James D Miller
# Date: 05Jul2020
##################################################

int ledArray [60][10] = {
 {'a',0,1,1,0,0,0,0,1,1},//a
 {'b',0,1,1,0,0,0,1,0,1},//b
 {'c',0,1,1,0,0,0,1,1,1},//c
 {'d',0,1,1,0,0,1,0,0,1},//d
 {'e',0,1,1,0,0,1,0,1,1},//e
 {'f',0,1,1,0,0,1,1,0,1},//f
 {'g',0,1,1,0,0,1,1,1,1},//g
 {'h',0,1,1,0,1,0,0,0,1},//h
 {'i',0,1,1,0,1,0,0,1,1},//i
 {'j',0,1,1,0,1,0,1,0,1},//j
 {'k',0,1,1,0,1,0,1,1,1},//k
 {'l',0,1,1,0,1,1,0,0,1},//l
 {'m',0,1,1,0,1,1,0,1,1},//m
 {'n',0,1,1,0,1,1,1,0,1},//n
 {'o',0,1,1,0,1,1,1,1,1},//o
 {'p',0,1,1,1,0,0,0,0,1},//p
 {'q',0,1,1,1,0,0,0,1,1},//q
 {'r',0,1,1,1,0,0,1,0,1},//r
 {'s',0,1,1,1,0,0,1,1,1},//s
 {'t',0,1,1,1,0,1,0,0,1},//t
 {'u',0,1,1,1,0,1,0,1,1},//u
 {'v',0,1,1,1,0,1,1,0,1},//v
 {'w',0,1,1,1,0,1,1,1,1},//w
 {'x',0,1,1,1,1,0,0,0,1},//x
 {'y',0,1,1,1,1,0,0,1,1},//y
 {'z',0,1,1,1,1,0,1,0,1},//z
 {' ',1,1,0,0,0,1,0,0,1},//space
 {'A',0,1,0,0,0,0,0,1,1},//A
 {'B',0,1,0,0,0,0,1,0,1},//B
 {'C',0,1,0,0,0,0,1,1,1},//C
 {'D',0,1,0,0,0,1,0,0,1},//D
 {'E',0,1,0,0,0,1,0,1,1},//E
 {'F',0,1,0,0,0,1,1,0,1},//F
```

```
 {'G',0,1,0,0,0,1,1,1,1},//G
 {'H',0,1,0,0,1,0,0,0,1},//H
 {'I',0,1,0,0,1,0,0,1,1},//I
 {'J',0,1,0,0,1,0,1,0,1},//J
 {'K',0,1,0,0,1,0,1,1,1},//K
 {'L',0,1,0,0,1,1,0,0,1},//L
 {'M',0,1,0,0,1,1,0,1,1},//M
 {'N',0,1,0,0,1,1,1,0,1},//N
 {'O',0,1,0,0,1,1,1,1,1},//O
 {'P',0,1,0,1,0,0,0,0,1},//P
 {'Q',0,1,0,1,0,0,0,1,1},//Q
 {'R',0,1,0,1,0,0,1,0,1},//R
 {'S',0,1,0,1,0,0,1,1,1},//S
 {'T',0,1,0,1,0,1,0,0,1},//T
 {'U',0,1,0,1,0,1,0,1,1},//U
 {'V',0,1,0,1,0,1,1,0,1},//V
 {'W',0,1,0,1,0,1,1,1,1},//W
 {'X',0,1,0,1,1,0,0,0,1},//X
 {'Y',0,1,0,1,1,0,0,1,1},//Y
 {'Z',0,1,0,1,1,0,1,0,1},//Z
 {'.',0,0,1,0,1,1,1,0,1},//.
 {'@',1,1,1,1,1,1,1,1,1},//CALIBRATE ALL
 {'#',0,0,0,0,0,0,0,0,1},//CALIBRATE CENTER ONLY
 {'$',1,1,1,1,1,1,1,1,0},//CALIBRATE EDGES ONLY
 {'?',0,0,1,1,1,1,1,1,1},//?
 {',',1,1,0,0,0,1,1,0,1},//,
 {'!',0,0,0,0,0,1,1,1,1},//
 };
 double ledRate = 300;
 int lightUp[10]= {0,0,0,0,0,0,0,0,0,0};
void setup() {

 Serial.begin(115200);
 Serial.print("---");Serial.print("Transmitter is active");Serial.print("---");Serial.println();
}
void loop() {
 // put your main code here, to run repeatedly:
 pinMode(2, OUTPUT);
 pinMode(3, OUTPUT);
 pinMode(4, OUTPUT);
 pinMode(5, OUTPUT);
 pinMode(10, OUTPUT); //center light
 pinMode(6, OUTPUT);
 pinMode(7, OUTPUT);
 pinMode(8, OUTPUT);
```

```
pinMode(9, OUTPUT);

Serial.print("Waiting for message now:");Serial.println();
while(Serial.available()==0){}
String message = Serial.readString();
Serial.print("Transmitting now:");Serial.println();Serial.println();
//Message Breakdown
int messageSize=message.length()-1;
char messageBreakdown[100];
for (int i=0;i<=messageSize;i++)
{
messageBreakdown[i]=message[i];
Serial.print(messageBreakdown[i]);Serial.print(" ");
for(int j=0;j <= 59; j++){
if (messageBreakdown[i] == ledArray[j][0]){
 for(int k = 1;k<10;k++){
 lightUp[k]= ledArray[j][k];
 if (k == 9){
 if(lightUp[1] == 1){digitalWrite(2, HIGH);}else{digitalWrite(2, LOW);}
 if(lightUp[2] == 1){digitalWrite(3, HIGH);}else{digitalWrite(3, LOW);}
 if(lightUp[3] == 1){digitalWrite(4, HIGH);}else{digitalWrite(4, LOW);}
 if(lightUp[4] == 1){digitalWrite(5, HIGH);}else{digitalWrite(5, LOW);}
 if(lightUp[5] == 1){digitalWrite(7, HIGH);}else{digitalWrite(7, LOW);}
 if(lightUp[6] == 1){digitalWrite(8, HIGH);}else{digitalWrite(8, LOW);}
 if(lightUp[7] == 1){digitalWrite(9, HIGH);}else{digitalWrite(9, LOW);}
 if(lightUp[8] == 1){digitalWrite(10, HIGH);}else{digitalWrite(10, LOW);}
 if(lightUp[9] == 1){digitalWrite(6, HIGH);}else{digitalWrite(6, LOW);}
 delay(5500);
 }
 }
 break;
}
}
 digitalWrite(2, LOW);
 digitalWrite(3, LOW);
 digitalWrite(4, LOW);
 digitalWrite(5, LOW);
 digitalWrite(6, LOW);
 digitalWrite(7, LOW);
 digitalWrite(8, LOW);
 digitalWrite(9, LOW);
 digitalWrite(10, LOW);
 delay(1000);
}
Serial.println();Serial.println(); }
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. ML USING OPENCV AND IMAGEAI CODE

```
################################################
# Title: FSO Communications Using OpenCV and ImageAI with ML
# Source Code Author: Glenn Prince
# Modified by: LT James D Miller
# Date: 05Jul2020
################################################

# Initialize libraries
import cv2 as cv
import numpy as np
import requests as req
import os as os
import random
import time

from imageai.Detection import ObjectDetection as od
from imageai.Detection.Custom import DetectionModelTrainer
from imageai.Detection.Custom import CustomObjectDetection

# Display specified image when the showImage function is called.
def showImage(img):
 window_name = 'image'
 cv.imshow(window_name, img)
 cv.waitKey(0)
 cv.destroyAllWindows()

# Load your predefined model into the detector
detector = od()
detector.setModelTypeAsYOLOv3()
detector.setModelPath(<YOUR PREDEFINED MODEL HERE)
detector.loadModel()

# Trains the data. This example is trained to identify two byte combinations.
trainer = DetectionModelTrainer()
trainer.setModelTypeAsYOLOv3()
trainer.setDataDirectory(data_directory="bytes")
trainer.setTrainConfig(object_names_array=["11111111","10010011"],       batch_size=4,
num_experiments=1,train_from_pretrained_model="bytes\models\detection_model-ex-
023--loss-0002.857.h5")
trainer.trainModel()
```

```
# Calls to test and validate models created for the two byte patterns identified.
trainer.evaluateModel(model_path="bytes\models\detection_model-ex-001--loss-
0004.901.h5",
  json_path="bytes\json\detection_config.json", iou_threshold=0.5,
  object_threshold=0.3, nms_threshold=0.5)


model05 = trainer.evaluateModel(model_path="bytes\models\detection_model-ex-001--
loss-0004.901.h5",
  json_path="bytes\json\detection_config.json", iou_threshold=0.5,
  object_threshold=0.3, nms_threshold=0.5)

print('-------------------------------------------------------')
print(Cycle Number:', model05[0]['average_precision']['11111111'])
print(Cycle Number:', model05[0]['average_precision']['10010011'])
print('-------------------------------------------------------')

# Used to created models on random images to evaluate model probability.
detector = CustomObjectDetection()
detector.setModelTypeAsYOLOv3()
detector.setModelPath("bytes\models\detection_model-ex-001--loss-0004.901.h5")
detector.setJsonPath("bytes\json\detection_config.json")
detector.loadModel()


testImages2 = os.listdir("bytes/validation/images")
randomFile = testImages2[random.randint(0, len(testImages) - 1)]


start = time.time()
detectedImage, detections = detector.detectObjectsFromImage(output_type="array",
input_image="bytes/validation/images/{0}".format(randomFile),
minimum_percentage_probability=30)
end = time.time()
showImage(detectedImage)


for eachObject in detections:
 print(eachObject["name"] , " : ", eachObject["percentage_probability"], " : ",
eachObject["box_points"] )
 print("--------------------------------")
 print("[INFO] YOLO took {:.6f} seconds".format(end - start))
```

LIST OF REFERENCES

[1] A. Prokes, “Atmospheric effects on availability of free space optics systems,” *Optical Engineering*, vol. 48, no. 6, pp. 066001–0660010, Jun. 2009.

[2] I. Son and S. Mao, “A survey of free space optical networks,” *Digital Communications and Networks*, vol. 3, no. 2, pp. 67–77, 2017.

[3] L. C. Andrews, R. L. Phillips, and C. Young, *Laser Beam Scintillation with Applications*. Bellingham: SPIE, 2001.

[4] D. Messier, “DLR researchers set world record in free-space optical communications,” Parabolic Arc, November 5, 2016. [Online]. Available: http://www.parabolicarc.com/2016/11/05/dlr-researchers-set-world-record-freespace-optical-communications/

[5] National Aeronautics and Space Administration, “The electromagnetic spectrum.” Accessed August 25, 2020. [Online] Available: https://imagine.gsfc.nasa.gov/science/toolbox/emspectrum1.html/

[6] H. J. Eichler, J. Eichler, and O. Lux, *Lasers: Basics, Advances and Applications*, Switzerland: Cham Springer, 2018.

[7] E. Kozachenko and M. Anderson, “A free space optical communications system: An M-ary multi-pulse width Modulation scheme with emphasis on optimizing transmit power,” May 2011, pp. 195–201, doi: 10.1109/ICSOS.2011.5783666

[8] E. Ghashghai, *Communications Networks to Support Integrated Intelligence, Surveillance, and Reconnaissance Strike Operations*. Santa Monica, CA: Rand, 2004.

[9] W. S. Rabinovich, J. L. Murphy, M. Suite, M. F. Ferraro, R. Mahon, and P. G. Goetz. “Free-space optical link to an explosive ordnance disposal (EOD) robot,” Naval Research Lab Washington, DC, USA, Rep. 167-169, 2009. [Online]. Available: http://www.dtic.mil/docs/citations/ADA524991

[10] L. M. Thomas, C. Mann, C. I. Moore, and W. S. Rabinovich, “Free space optics for tactical environments,” in *Proceedings of SPIE*, Feb. 2018, pp. 1052402–1052402–9.

[11] Lockheed Martin, “Universal communications platform.” Accessed August 25, 2020. [Online] Available: https://www.lockheedmartin.com/en-us/products/universal-communications-platform.html/

[12] *Information Technology — Automatic Identification and Data Capture Techniques — QR Code Bar Code Symbology Specification*, ISO/IEC 18004, 2015. [Online]. Available: https://www.iso.org/standard/62021.html

[13] A. Zeroual, M. Derdour, M. Amroune and A. Bentahar, "Using a fine-tuning method for a deep authentication in mobile cloud computing based on Tensorflow Lite framework," *2019 International Conference on Networking and Advanced Systems (ICNAS)*, Annaba, Algeria, 2019, pp. 1-5, doi: 10.1109/ICNAS.2019.8807440.

[14] Dendo Wave Incorporated. Information capacity and versions of the QR code. Accessed June 19, 2020. [Online]. Available: https://www.qrcode.com/en/about/version.html

[15] S. Ren, K. He, R. Girshick, and J Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[16] V. Zorich, "Communication in the presence of noise," in *Mathematical Analysis of Problems in the Natural Sciences*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 55–75.

[17] A. G. Howard, M. Menglong, "MobileNets: efficient convolutional neural networks for mobile vision applications," arXiv, Ithaca, NY, USA, RR-1704.04861v1, Apr. 2017.

[18] M. Singh and J. Malhotra, "Performance investigation of high-speed FSO transmission system under the influence of different atmospheric conditions incorporating 3-D orthogonal modulation scheme," *Optical and quantum electronics*, vol. 51, no. 9, pp. 1–13, 2019.

[19] TensorFlow. "TensorFlow-Lite guide." Accessed August 25, 2020. [Online]. Available: https://www.tensorflow.org/lite/guide/

[20] H. Hiroto, "Reproducing training performance of YOLOv3 in PyTorch," Medium. February 1, 2019. [Online]. Available: https://medium.com/@hirotoschwert/reproducing-training-performance-of-yolov3-in-pytorch-part1-620140ad71d3#:~:text=YOLOv3%20consists%20of%20the%20backbone,detection%20layers%20called%20YOLO%20layers

[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," arXiv, Ithaca, NY, USA, RR-1506.02640v5, May. 2016.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California