
# NAVAL POSTGRADUATE SCHOOL

**MONTEREY, CALIFORNIA**

# THESIS

**AI-BASED UXO DETECTION USING sUAS EQUIPPED WITH A SINGLE- OR MULTI-SPECTRUM EO SENSOR**

by

Seungwan Cho

March 2021

Thesis Advisor: Oleg A. Yakimenko
Second Reader: Fotis A. Papoulias

THIS PAGE INTENTIONALLY LEFT BLANK

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



| **1. AGENCY USE ONLY** *(Leave blank)* | **2. REPORT DATE** March 2021 | **3. REPORT TYPE AND DATES COVERED** Master's thesis |
|---|---|---|

| **4. TITLE AND SUBTITLE** AI-BASED UXO DETECTION USING sUAS EQUIPPED WITH A SINGLE- OR MULTI-SPECTRUM EO SENSOR | **5. FUNDING NUMBERS** |
|---|---|
| **6. AUTHOR(S)** Seungwan Cho | |
| **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)**

Unexploded ordnance (UXO) poses a threat to soldiers operating in mission areas, but current UXO detection systems do not provide the required safety and efficiency to protect soldiers from this hazard. Recent technological advancements in artificial intelligence (AI) and small unmanned aerial systems (sUAS) present an opportunity to explore a novel concept for a UXO detection system. The system proposed in this study integrates a sUAS with an onboard single- or multiple-spectrum (MS) electro-optical (EO) sensor. The major contributions of this thesis include the development of an AI-based algorithm for reliable UXO detection using a Deep Learning Convolutional Neural Network, execution of experiments to validate the proposed system's performance, and analysis of the proposed system's feasibility. To that end, the thesis describes the development of the UXO detector for a single-spectrum sensor, followed by the development and integration of five UXO detectors for the MS sensor. The field experiment conducted using a commercial-off-the-shelf (COTS) sUAS equipped with a standard EO sensor is also described. This thesis concludes that AI-based UXO detection using a single-spectrum or MS sensor flown on a COTS sUAS is a feasible solution. The thesis also proposes the steps for further enhancement and improvement of the developed system and lays out additional test and evaluation strategies to fully test the developed capability.

| **14. SUBJECT TERMS** unexploded ordnance, UXO, artificial intelligence, AI, small unmanned aerial systems, sUAS, object detection, deep learning, DL, Convolutional Neural Network, CNN, multiple-spectrum, MS, electro-optical, EO sensor, commercial-off-the-shelf, COTS | **15. NUMBER OF PAGES** 117 |
|---|---|
| | **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.**

# AI-BASED UXO DETECTION USING sUAS EQUIPPED WITH A SINGLE- OR MULTI-SPECTRUM EO SENSOR

Seungwan Cho
Captain, Republic of Korea Army
BE, Korea Military Academy, 2011

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN ENGINEERING SYSTEMS**

from the

**NAVAL POSTGRADUATE SCHOOL**
**March 2021**

Approved by: Oleg A. Yakimenko
Advisor

Fotis A. Papoulias
Second Reader

Ronald E. Giachetti
Chair, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

# ABSTRACT

Unexploded ordnance (UXO) poses a threat to soldiers operating in mission areas, but current UXO detection systems do not provide the required safety and efficiency to protect soldiers from this hazard. Recent technological advancements in artificial intelligence (AI) and small unmanned aerial systems (sUAS) present an opportunity to explore a novel concept for a UXO detection system. The system proposed in this study integrates a sUAS with an onboard single- or multiple-spectrum (MS) electro-optical (EO) sensor. The major contributions of this thesis include the development of an AI-based algorithm for reliable UXO detection using a Deep Learning Convolutional Neural Network, execution of experiments to validate the proposed system's performance, and analysis of the proposed system's feasibility. To that end, the thesis describes the development of the UXO detector for a single-spectrum sensor, followed by the development and integration of five UXO detectors for the MS sensor. The field experiment conducted using a commercial-off-the-shelf (COTS) sUAS equipped with a standard EO sensor is also described. This thesis concludes that AI-based UXO detection using a single-spectrum or MS sensor flown on a COTS sUAS is a feasible solution. The thesis also proposes the steps for further enhancement and improvement of the developed system and lays out additional test and evaluation strategies to fully test the developed capability.

THIS PAGE INTENTIONALLY LEFT BLANK

# TABLE OF CONTENTS


I. INTRODUCTION....1
A. BACKGROUND....1
1. Overview of UXO....1
2. Current UXO Detection Systems....2
3. Threats of UXO in Mission Areas....3
4. Technological Opportunities....4
B. PROBLEM FORMULATION AND THESIS ORGANIZATION....6

II. CONCEPT OF OPERATIONS OF THE sUAS-BASED UXO DETECTION SYSTEM AND LITERATURE REVIEW....9
A. CONCEPT OF OPERATIONS....9
B. LITERATURE REVIEW....10
1. Potential Capability of sUAS Equipped with an EO Sensor....11
2. Proposed sUAS-based UXO Detection System....11
3. Feasibility of the DLCNN Application with sUAS....12
4. UXO Detection Using Multispectral Imagery....12

III. DEVELOPMENT OF THE DETECTION ALGORITHM....13
A. IMPLEMENTING THE DETECTION ALGORITHM IN MATLAB....13
B. DLCNN-BASED OBJECT DETECTION ALGORITHM....13
1. Object Detection Using DLCNN with YOLOv2 Network....13
2. CNN Model Setup for UXO Detector....17
3. Anchor Boxes Estimation for the UXO Detector Network....17
C. MULTISPECTRAL DETECTION ALGORITHM....20
D. MEASURE OF EFFECTIVENESS....22

IV. DATA COLLECTION AND PROCESSING PROCEDURE....25
A. AVAILABLE DATA SOURCES....25
B. TOOLS USED IN DATA COLLECTION AND PROCESSING....26
1. Single-spectrum EO Sensor....26
2. Multi-spectrum Sensor....27
3. Computation Platform....29
C. DATA COLLECTION....30
D. DATA CURATION....33
1. Image Resizing....33

2. Image Labeling to Ground Truth Data ....34
3. Dividing Labeled Data into Training/Validation/Test Subsets....35
4. Data Augmentation....36

V. ASSESSMENT OF A SINGLE-SPECTRUM UXO DETECTION....37
A. DATA PROCESSING....37
B. TRAINING PROGRESS....38
C. DETECTION DEMONSTRATION....40
D. EVALUATION OF TRAINED DETECTOR....42

VI. ASSESSMENT OF A MULTI-SPECTRUM UXO DETECTION....45
A. MULTI-SPECTRUM DATA CHARACTERIZATION AND PROCESS FLOW....45
B. TRAINING AND DEMONSTRATION OF INDIVIDUAL-SPECTRUM DETECTORS....47
C. EVALUATION OF RESPECTIVE DETECTORS....49
D. THE TWO-STEP MULTI-SPECTRUM INTEGRATION PROCESS....50
1. Simple Merging....50
2. Non-maximal Suppression....50
E. BENEFITS OF INTEGRATED UXO DETECTION USING AN MS SENSOR....51
F. EVALUATION ON MULTI-SPECTRUM DETECTOR....53

VII. ASSESSMENT OF A SINGLE-SPECTRUM SMALL FIREARM DETECTOR IN A REAL FLIGHT TEST....55
A. INTEGRATION WITH SUAS....55
B. SYSTEM IMPLEMENTATION....55
1. sUAS....55
2. EO Sensor....56
C. FLIGHT TEST EXECUTION....57
1. Detection Object....57
2. Test Scenario....57
3. Data Processing and CNN Training for Small Firearms Detector....60
4. Detection Demonstration....61
D. EVALUATION ON SMALL FIREARMS DETECTION....63

VIII. CONCLUSIONS AND RECOMMENDATIONS....65
A. SUMMARY OF RESEARCH FINDINGS....65

**B. RECOMMENDATIONS FOR FUTURE WORK.....66**

**APPENDIX A. MATLAB CODE FOR UXO DETECTOR TRAINING / EVALUATION .....69**

**APPENDIX B. MATLAB CODE FOR INTEGRATION OF MULTISPECTRAL DETECTION / EVALUATION .....77**

**LIST OF REFERENCES.....85**

**INITIAL DISTRIBUTION LIST .....91**

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF FIGURES


Figure 1. UXO in Situ at Fort Ord, CA. Source: Fort Ord Cleanup, personal communication (September 16, 2020). ..... 2

Figure 2. Example of Current Ground-Based UXO Detection. Source: S. Lee (2018). ..... 3

Figure 3. UN Peacekeeper Marks UXO in Mission Area. Source: Lukunka (2014). ..... 4

Figure 4. The 199th Infantry Brigade Experimental Force Soldiers Conduct Operational Testing of the RPUAS at Fort Benning, GA. Source: Leonel (2020). ..... 5

Figure 5. Concept of sUAS-based UXO Detection System ..... 9

Figure 6. Comparison of Three DL Tasks in CV Area ..... 14

Figure 7. Learning Speed Comparison of Three CNN Models. Source: Garg, Chowdhury, and More (2019) ..... 16

Figure 8. YOLOv2 Object Detector Network Setup ..... 18

Figure 9. IoU Definition ..... 18

Figure 10. Number of Anchors vs. Mean IoU from the Training Data for the UXO Detector ..... 19

Figure 11. Training Detectors for MS Data ..... 20

Figure 12. Two-Step Process of Multispectral Detection Integration ..... 22

Figure 13. Confusion Matrix for UXO Detector Evaluation. Adapted from Stehman (1997). ..... 22

Figure 14. Ground Condition of Camp Roberts ..... 26

Figure 15. Sony Alpha 6000. Source: "Sony A6000 E-Mount Camera with APS-C Sensor" (2020). ..... 27

Figure 16. MicaSense RedEdge. Source: MicaSense (2020). ..... 28

Figure 17. Spectral Bands of Micasense RedEdge. Adapted from MicaSense (2018). ..... 29

Figure 18. HP Pavilion Laptop. Source: HP (n.d.).....30
Figure 19. Nine UXO Used in Data Collection.....31
Figure 20. Data Collection Using a Tripod.....32
Figure 21. Randomly Placed UXO.....32
Figure 22. Data Collection Result for Single- and MS Sensors.....33
Figure 23. Image Resizing to 416×416.....34
Figure 24. Data Labeling using MATLAB Image Labeler.....35
Figure 25. Data Augmentation for UXO Detector Training.....36
Figure 26. Data Process Flow for EO Sensor.....38
Figure 27. Training Options Setup.....39
Figure 28. Training Completion and Training Loss.....40
Figure 29. Demonstration of Detecting UXO in a Series of Individual Frames.....41
Figure 30. Demonstration of Detecting UXO in Videos.....42
Figure 31. Precision-Recall Curve for Visional UXO Detector.....43
Figure 32. UXO Pictures from Each Spectrum and Related Alignment Problems.....46
Figure 33. Data Process Flow for MS Data.....46
Figure 34. Training Loss for Red Spectrum Detector.....47
Figure 35. MS Detection Demonstration.....48
Figure 36. Precision-Recall Curve for Respective Spectrum Detectors.....49
Figure 37. Simple Merging Step.....51
Figure 38. Final Detection Result and Application of NMS Method.....52
Figure 39. Example of Complementary MS Detection.....52
Figure 40. Precision-Recall Curve for MS Detector.....53
Figure 41. DJI Inspire 1 Pro System.....56
Figure 42. Zenmuse X5 camera. Source: DJI (2020).....56

Figure 43. Four Small Firearms Used in the Flight Test....58
Figure 44. Field Test Environment....58
Figure 45. Flight Test Overview....59
Figure 46. Video Labeling with Automation....60
Figure 47. Anchor Boxes Estimation for Gun Detector....61
Figure 48. Training Loss for Gun Detector Training....62
Figure 49. Demonstration of Small Firearms Detection....62
Figure 50. Precision-Recall Curve for Gun Detector....63

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF TABLES

Table 1. Sony Alpha 6000 Digital Camera Specification. Source: Sony (n.d.).....27

Table 2. Micasense RedEdge Specification. Source: Micasense (n.d.). .....28

Table 3. HP Pavilion Specification. Source: HP (n.d.). .....30

Table 4. DJI Inspire 1 Pro Specifications. Source: DJI (n.d.).....56

Table 5. Zenmuse X5 Specifications. Source: DJI (n.d.).....57

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF ACRONYMS AND ABBREVIATIONS

| | |
|---|---|
| AI | artificial intelligence |
| AP | average precision |
| CNN | Convolutional Neural Network |
| COTS | commercial-off-the-shelf |
| CV | Computer Vision |
| DL | Deep Learning |
| DLCNN | Deep Learning Convolutional Neural Network |
| DMZ | demilitarized zone |
| DOD | Department of Defense |
| EO | electro-optical |
| FAA | Federal Aviation Administration |
| FIR | Far-infrared |
| FOD | foreign object debris |
| IoU | intersection over union |
| IR | infrared |
| mAP | mean average precision |
| ML | Machine Learning |
| MS | multi-spectrum |
| NIR | near infrared |
| NMS | non-maximal suppression |
| NPS | Naval Postgraduate School |
| PKO | peacekeeping operation |
| PR | Precision-Recall |
| ReLU | Recertified Linear Unit |
| RPUAS | rucksack portable unmanned aircraft system |
| RGB | Red-Green-Blue |
| sUAS | small unmanned aircraft system |
| TEM | transient electromagnetic |
| UXO | unexploded ordnance |
| YOLO | You Only Look Once |

THIS PAGE INTENTIONALLY LEFT BLANK

# EXECUTIVE SUMMARY

Unexploded ordinance (UXO), a legacy of war, exists worldwide in weapons-testing territories, former troop-training properties, and current military mission areas (Etter and Delaney 2003). In particular, many small-group soldiers face UXO threats in their mission areas such as the demilitarized zone in the Republic of Korea and United Nation peacekeeping mission areas. The current ground-based UXO detection system, which forces the soldiers to walk the hazardous area, is dangerous and inefficient. Recent technological advancements in artificial intelligence (AI) and small unmanned aerial systems (sUAS) and the Department of Defense's (DOD) trend in deploying sUAS in many aspects of applications have brought an opportunity to explore a novel concept of a UXO detection system (U.S. Department of Defense 2015).

The core concept of the proposed UXO detection system combines sUAS equipped with a single spectrum or multi-spectrum electro-optical (EO) sensor and pretrained AI-based software. To assess the feasibility of the proposed system, this thesis aims at training and evaluating an AI-based UXO detector with EO data. For the AI algorithm, this thesis chooses Object Detection using a Deep Learning Convolutional Neural Network (DLCNN), specifically, the YOLOv2 CNN model. By putting YOLOv2 CNN layers together on ResNet-50, this thesis sets the UXO detector CNN model. The algorithm of multispectral (MS) detection relies on training respective spectrum detectors individually and then combining the detection results. Integration assumes two steps: 1) simply merging the detection results from each detector at a single space, and then 2) leaving the strongest result while suppressing other weak results using the non-maximal suppression (NMS) method (Takumi et al. 2017). To evaluate the effectiveness of the trained detectors, this thesis sets the average precision (AP) as an evaluation metric in consideration of both Precision and Recall. Precision estimates how correctly the trained CNN performs detection based on the predicted detection results and Recall estimates how many actual UXO the trained CNN misses in a given set of test data, indicating the area under the Precision-Recall curve plotted by the detection results (Everingham et al. 2010).

Data collection for the nine different UXO used in this study was conducted at the California National Guard post in Camp Roberts, CA, by carrying a sensor around manually or setting it on a tripod (because of the prohibiting restrictions on sUAS flying). Sensors used were the Sony Alpha 6000, as a single-spectrum EO sensor, and the five-spectrum MicaSense RedEdge, as a multi-spectrum (MS) sensor. In all, 1,225 pictures and 59 videos (each ten seconds long) were collected using the EO sensor, and 4,075 photos—815 photos for the respective spectrums—were collected using the MS sensor. These data were processed using the follows four steps: 1) resizing the collected data to 416pix×416pix samples, which happens to be the best image size for YOLOv2 CNN model performance, 2) labeling the resized images to be ground truth data, 3) randomly dividing the labeled data into three groups—75% for training, 15% for validation, and 15% for testing data, and 4) augmenting the data by transforming geometry and tweaking the color space of the images in order to make up for the shortages of training data.

In the training of the single-spectrum UXO detector, the 857 preprocessed training EO data and 184 validation EO data were used. For the training options, this thesis set nine Anchor Boxes, 20 maximum epochs, and eight mini-batch sizes. The training of the single-spectrum UXO detector took about two hours, with no significant training loss. In computer simulations, the trained detector detected UXO successfully in both pictures and videos. The trained detector AP was evaluated as 0.774. In other words, the detector detected UXO in a given set of test data fairly well with high Precision, and at the same time, had few misses of UXO with high Recall considering that the AP of 1 means perfectly accurate detection without missing any objects.

In the MS detection assessment, the 4,075 data samples collected in the five spectrums—Red, Green, Blue, Near-Infrared, and Red Edge—were processed in a manner similar to that of single-spectrum detection. The five spectrum detectors were trained for 20 minutes, respectively, with the training options of nine anchor boxes, eight mini-batch size, and five maximum epochs. In computer simulations, the detection results of the respective detectors were different from each other but happened to be complementary to each other; in some cases, some spectrum detectors failed to detect UXO, but the others succeeded. To take advantage of complementarity, the research

integrated the detection results with the aforementioned two steps—simple merging and NMS. The evaluation results of respective detectors are that the Blue spectrum detector AP is 0.433, Green spectrum detector AP is 0.583, Red spectrum detector AP is 0.453, Near-Infrared spectrum detector AP is 0.484, and Red Edge spectrum detector AP is 0.592, all less than for a single-spectrum sensor. When integrated, however, the AP metric improved to as high as 0.871 AP, exceeding the AP value for the single sensor.

In order to assess the feasibility of the proposed system within the operational environment, a flight test was conducted using the DJI Inspire 1 Pro for sUAS equipped with a Zenmuse X5 EO sensor. The objects to be detected were four different small firearms including pistols and rifles. Covering a 4,700 square foot area with randomly placed small firearms, the sUAS flew at a speed of 3 mph and 10 foot altitude above the ground in a serpentine flight pattern. As a result of the flight test, a total of 18 videos (each 20–30 seconds long) were recorded. Twelve of them, equivalent to 1,812 photos, were used for the detector training purposes; three, equivalent to 389 photos, for validation while training; and the remaining three videos were used for testing. The data process and the training process followed the same manner as that of a single-spectrum detector, and the training took an hour and ten minutes. In computer simulations, the trained detector successfully detected all small firearms in the three test videos. AP for the trained gun detector performed as high as 0.966, or almost 100% reliability!

The overall assessment of this research is that an AI-based UXO detection system using sUAS equipped with a single or MS sensor offers a safe and feasible solution exhibiting an amazing result. This finding suggests the necessity to extend this study and assess the suitability of the proposed approach with respect to real-time implementation, specific environments, and different weather conditions.

## References

Delores, Etter, and Delaney Bill. 2003. *Report of the Defense Science Board Task Force on Unexploded Ordnance*. Defense Science Board Task Force. Washington, DC: Office of the Under Secretary of Defense (Acquisition, Technology & Logistics). https://dsb.cto.mil/reports/2000s/ADA419970.pdf.

Everingham, Mark, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. 2010. “The Pascal Visual Object Classes (VOC) Challenge.” *International Journal of Computer Vision* 88 (2): 303–38. https://doi.org/10.1007/s11263-009-0275-4.

Takumi, Karasawa, Kohei Watanabe, Qishen Ha, Antonio Tejero-De-Pablos, Yoshitaka Ushiku, and Tatsuya Harada. 2017. “Multispectral Object Detection for Autonomous Vehicles.” In *Proceedings of the on Thematic Workshops of ACM Multimedia 2017—Thematic Workshops ‘17*, 35–43. Mountain View, California, USA: ACM Press. https://doi.org/10.1145/3126686.3126727.

U.S. Department of Defense. 2015. *Unmanned Aircraft Systems Roadmap 2005–2030*. Washington, DC: Office of the Secretary of Defense.

# ACKNOWLEDGMENTS

I would like to thank my country, The Republic of Korea (ROK), and ROK Army for giving me this golden opportunity to conduct my study in the United States and learn from the Naval Postgraduate School (NPS) professors. I also want to say thank you to all NPS members for allowing and enabling my research in an extremely difficult pandemic environment. I would like to express my special thanks to my thesis advisor, Professor Oleg Yakimenko. I had only a general idea about my thesis topic when I started my research journey, and he gave me a clear direction. He also contributed a lot with novel ideas and technical support. I could not have finished my thesis journey without him. I would also like to thank Assistant Professor Mara Orescanin for lending an MS sensor used in this research; the personnel of Camp Roberts and the manager of the NPS Field Laboratory at McMillan Airfield; Mr. Greg Arenas, for helping to arrange for the collection of UXO photo samples; and Base Police LT Supervisor Edward Macias, who helped me a lot with conducting flight tests at Fort Ord. Lastly, I would like to express my deepest gratitude to my family, Jinhee and Eunseo, for their infinite support and devoted love.

THIS PAGE INTENTIONALLY LEFT BLANK

# I. INTRODUCTION

Unexploded ordnance (UXO) is an obvious threat to warfighters in military mission areas. Moreover, current ground-based UXO detection systems require the warfighters using them to walk the dangerous area. Recent technological advancements, however, have brought an opportunity for a new concept of UXO detection systems based on artificial intelligence (AI). This thesis explores the feasibility of such a UXO detection system that uses a small unmanned aerial system (sUAS) equipped with a single-spectrum or multi-spectrum electro-optical (EO) sensor.

## A. BACKGROUND

The risks posed by UXO and inherent in the current UXO detection systems as well as the opportunity presented by advanced AI technology and the availability of sUAS inspired the idea for an AI-based UXO detection system using sUAS, which this thesis explores.

### 1. Overview of UXO

UXO is defined as "military ammunition or explosive ordnance which has failed to function as intended" (Martin et al. 2019). Usually, UXO are armed and remain unexploded, which poses a hazard (Naval Explosive Ordnance Disposal Technology Division 1996). These UXO are a significant threat to warfighters and civilians who are serving and living in places that used to be battlefields, such as the demilitarized zone (DMZ) between South Korea and North Korea (Talmadge 2017). In Southeast Asia, research estimates that more than 20% of the land in Cambodia, Laos, and Vietnam is crowded with UXO, and these devices still injure and kill hundreds of people each year (Martin et al. 2019). The United States also has a UXO problem caused by weapon system testing or training activities (Naval Explosive Ordnance Disposal Technology Division 1996). According to the Naval Explosive Ordnance Disposal Technology Division (1996), all shapes, sizes, and types of UXO have been used and the following types of UXO are those most likely to be encountered:

- Small arms munitions
- Hand grenades
- Rockets
- Guided missiles
- Projectiles
- Mortars
- Projected grenades
- Rifle grenades
- Submunitions
- Bombs.

Figure 1 on the left presents UXO in situ and actual UXO from the site in Fort Ord, a training facility for infantry troops from 1947 to 1994, is shown at right.

Figure 1. UXO in Situ at Fort Ord, CA.
Source: Fort Ord Cleanup, personal communication (September 16, 2020).

## 2. Current UXO Detection Systems

Various relevant technologies are used to detect UXO of different types and at various depths. Most UXO are located on the land surface and can be detected visually or photographically, but the current UXO detection systems are most often equipped with subsurface UXO targeting sensors such as magnetometers or electromagnetic sensors (Bertrand et al. 2004).

The current UXO detection systems pose several problems in terms of safety and efficiency. Typically designed for hand-held operation, these systems require soldiers to walk the hazardous areas in-person to detect UXO. In addition to the potential casualties

presented by this detection method, it takes a lot of time to survey the suspicious area. Figure 2 presents an example of a UXO detecting operation near Yeongpyeong shooting range in Pocheon, South Korea, where the ROK Army 5th Engineer Brigade and the U.S. Army 2nd Infantry Division are using a ground-based electromagnetic system.

Figure 2. Example of Current Ground-Based UXO Detection. Source: S. Lee (2018).

## 3. Threats of UXO in Mission Areas

In many cases, small-group infantry soldiers need to cross a suspected UXO area without any support while they are conducting military operations such as DMZ operations and peacekeeping operations (PKO). In reality, the soldiers do not need to detect all UXO out in the mission area, but only to secure a temporary route to cross the area. In those cases, near-surface and surface UXO on the ground are actual threats rather than underground UXO. Figure 3 represents an example of the threat of near-surface UXO in the PKO mission area, which shows a UN peacekeeper using branches to mark the location of UXO on the surface in a mission area in Mellit, North Darfur.

Figure 3. UN Peacekeeper Marks UXO in Mission Area.
Source: Lukunka (2014).

### 4. Technological Opportunities

Recently, advanced technologies such as unmanned systems have arisen to increase work-efficiency and to replace human operators in dangerous situations. The development of Machine Learning (ML) and Deep Learning (DL) has resulted in further developed AI technology. These innovations have improved not only industrial systems but also the relevant military systems.

#### *a. Small Unmanned Aircraft System*

As technology has improved, various aircraft systems have become available for many different purposes. According to 14 C.F.R. § 107.3 (2021), the Federal Aviation Administration (FAA) definition of "small unmanned aircraft means an unmanned aircraft weighing less than 55 pounds on takeoff, including everything that is on board or otherwise attached to the aircraft." The same section of the Code of Federal Regulation also defines that "small unmanned aircraft system (small UAS) means a small unmanned aircraft and its associated elements (including communication links and the components that control the small unmanned aircraft) that are required for the safe and efficient operation of the small unmanned aircraft in the national airspace system." According to

the U.S. Department of Transportation (2013), the U.S. Army has 6,200 sUAS, representing approximately 55 percent of the Army's aircraft. The U.S. Department of Transportation also expects that this number will increase to 10,000 sUAS by 2035, and that would be more than 75 percent of U.S. Army's aircraft assets (U.S. Department of Transportation 2013). Additionally, in the U.S. Department of Defense's (DOD) UAS roadmap, the DOD will deploy diverse types of sUAS in many aspects, especially with commercial sUAS (U.S. Department of Defense 2015). In this regard, the U.S. Army is actually looking to acquire a next-generation sUAS, the rucksack-portable UAS (RPUAS), which is an "inexpensive, rucksack portable, vertical take-off and landing drone that provides the soldier on the ground with a rapidly deployable scouting capability to gain situational awareness" for the Short Range Reconnaissance program (PEO Aviation Press Release 2019). Figure 4 shows a soldier testing the next-generation sUAS at Fort Benning, GA, in August 2020.

Figure 4. The 199th Infantry Brigade Experimental Force Soldiers Conduct Operational Testing of the RPUAS at Fort Benning, GA. Source: Leonel (2020).

Depending on their purpose, these sUAS can be equipped with diverse assets, but they are most often equipped with cameras to capture various imagery including pictures, thermal images, and multispectral images (Unmanned Systems Technology, n.d.). Colonel John Knightstep, who is Aviation Test Directorate Director, said in a 2019 interview that "new Army technology like RPUAS gives the company commander the

ability to conduct short range reconnaissance from a safe distance without putting soldiers in harm's way" (Judson 2019). Accordingly, to improve soldiers' safety, this thesis makes use of sUAS equipped with cameras for detecting UXO instead of the ground-based detectors.

### *b. AI Technology to Detect Objects*

These days, people can come across many AI-related technologies in various industries. According to Viswanathan (2018), AI generally indicates the intelligence demonstrated by computers. Viswanathan also explains that ML is one type of AI technology, and the ML algorithm enables machines with intelligence to teach themselves using only data, without any human being's programmatic help (2018). DL is a subordinate type of ML consisting of artificial neural networks that mimic the human brain in order to process data (Viswanathan 2018). Similarly, Computer Vision (CV) is "a field of study that seeks to develop techniques to help computers 'see' and understand the content of digital images such as photographs and videos" (Brownlee 2019a). Object detection refers to the specific CV task of "locat[ing] the presence of objects with a bounding box and types or classes of the located objects in an image" (Brownlee 2019b). Lately, DL has improved the performance of object detection through the use of a trained Convolutional Neural Network (CNN), creating a new burst in the unmanned system industry (H. Liu and Lang 2019). With all these considerations, this thesis attempts to combine the AI relevant technology into a sUAS for UXO detection.

## B. PROBLEM FORMULATION AND THESIS ORGANIZATION

The objective of this thesis is to assess the feasibility of AI-based UXO detection using a sUAS equipped with a single-spectrum or MS digital EO sensor. In particular, the thesis aims to answer the following research questions:

- Is it feasible for an AI-trained sUAS consisting of a single-spectrum or a multi-spectrum EO sensor to detect UXO on the earth's surface effectively?

- Is the aforementioned UXO detection system feasible in terms of practicality within the flying operational environment?

In trying to answer the research questions, this thesis mainly addresses how to make use of the sUAS's onboard EO sensors by implementing a UXO-detection model trained with AI technology; specifically, object detection using DL. The result of this research is envisioned to have multiple practical applications because of the following contributions: 1) the usage of sUAS in UXO detection; 2) exploration of object detection performance of the single-spectrum and the multi-spectrum EO sensors; and 3) the application of AI technology in military search operations.

The remainder of this thesis is structured as follows. Chapter II presents the concept of a sUAS-based UXO detection system and reviews the relevant literature. Chapter III explains the development of the system algorithm, including the AI algorithms of the DLCNN and the multispectral detection algorithm, and describes the effectiveness evaluation strategy. Chapter IV describes the data collection setup with the assets used and the data process flow. Chapter V explains the assessment of the single-spectrum UXO detection method including training progress, and detection demonstration and evaluation. Chapter VI explains the assessment of the multi-spectrum UXO detection system including MS data characteristics, the training of respective MS detectors, the MS detection results integration process, the benefits of integrated UXO detection results from MS detectors, and evaluation. Chapter VII focuses on sUAS integration and the flight testing campaign that uses small firearms as UXO substitute objects. Finally, Chapter VIII summarizes the thesis findings and concludes with recommendations for future research.

THIS PAGE INTENTIONALLY LEFT BLANK

# II. CONCEPT OF OPERATIONS OF THE SUAS-BASED UXO DETECTION SYSTEM AND LITERATURE REVIEW

This chapter describes how this research envisions a small tactical unit would use the model sUAS-based UXO detection system and explains the proposed assessment strategy of the devised system. Then, this chapter reviews the relevant literature.

## A. CONCEPT OF OPERATIONS

The proposed system consists of a sUAS-based UXO detection system utilizing an onboard single-spectrum or multispectral EO sensor (Figure 5). The research envisions the use of the proposed system in the following scenario. The commander defines the search area and commands launching one or several sUAS to comb it. These low-flying sUAS use a standard pattern for detecting and identifying UXOs in order to secure a mission area before a small tactical unit makes a maneuver. The sUAS captures footage of the mission area with onboard single-spectrum or multispectral EO sensor and transmits the footage to the ground control station where the operator is located. The trained sUAS marks the suspected UXO with bounding boxes, which are automatically displayed on the operator's screen. The operator assesses the UXO threats based on the detection results and makes an educated decision.

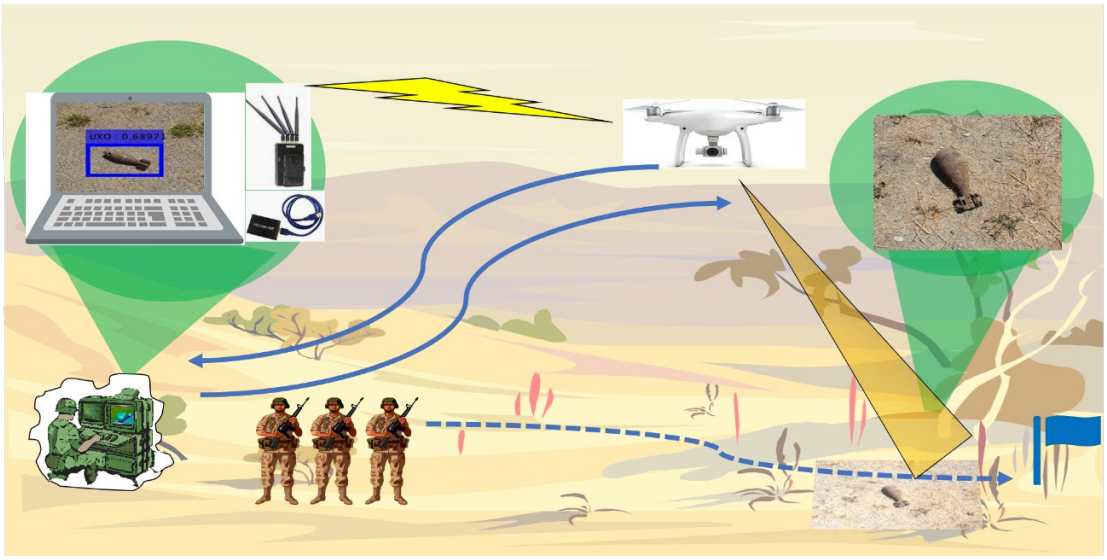

Figure 5. Concept of sUAS-based UXO Detection System

This thesis focuses on training the CNN-based detector models with EO data and assessing these trained models. Towards this goal, this thesis first explores and develops a system algorithm for UXO detection using a single or multispectrum sensor trained by DLCNN and performing object detection with a YOLOv2 network. Next, this research describes how the EO data was collected using a single-spectrum sensor and multi-spectrum (MS) camera, respectively. The single-spectrum EO sensor denotes a standard Red Green Blue (RGB) camera, and the MS sensor refers to a camera capable of capturing multispectral imagery simultaneously (this study uses a five-spectrum camera). Lastly but most importantly, this research shows how to train the CNN with the collected EO data and makes an assessment of the trained UXO detectors.

The criteria used to assess the effectiveness of the proposed UXO detection system considers both Precision and Recall: Precision estimates how correctly the trained CNN performs detection based on the predicted detection results and Recall estimates how many actual UXO the trained CNN misses in a given set of test data. Further, Average Precision (AP) computed by the area under the Precision-Recall curve is used as criteria metric (Everingham et al. 2010). This thesis evaluates system performance in two development scenarios and one operational experiment as follows:

- assessment of the effectiveness of the single-spectrum based detection system,
- assessment of the effectiveness of the MS-based detection system, and
- evaluation of the entire system in the field test campaign.

## B. LITERATURE REVIEW

This section reviews research about sUAS with an EO sensor, proposed sUAS - based UXO detection system, sUAS with DLCNN, and object detection using multispectral images.

### 1. Potential Capability of sUAS Equipped with an EO Sensor

A study by Lee in 2018 considers the concept of a system to conduct automated foreign object debris (FOD) detection using a swarm of sUAS. Lee, in the same paper, developed several image-processing algorithms to implement the system, and he gathered EO sensor data in conducting experiments over a runway (2018). He concludes that the FOD management system comprising the swarm with an onboard EO sensor is feasible and meets the presented requirements (Lee 2018). The aforementioned paper proves the potential capability and promising usability of the sUAS equipped with an EO sensor, and the developed algorithms of the image-processing inspires the concept of the sUAS-based UXO detection system for this present study.

### 2. Proposed sUAS-based UXO Detection System

DeSmet et al. (2018) designed a rapid mine detection system using sUAS equipped with a thermal sensor in order to detect a plastic mine, PFM-1, that may not be detected by the current electromagnetic detection system. The authors asserted that "available low-cost commercial UAV platforms equipped with thermal cameras allow accurate assessment of minefield presence, orientation, and potential minefield overlap" (deSmet et al. 2018). Their experiment's result proves the feasibility of using the sUAS with an onboard thermal camera to detect PFM-1 in various conditions such as different temperatures, moisture content, and burial depths (deSmet et al. 2018). Similarly, Qi et al. attempted to detect underground UXO using the sUAS-based transient electromagnetic (TEM) system in 2020. Qi et al. attached the TEM system to the sUAS and conducted a UXO detection experiment (2020). In their paper, the results of the experiment proved the safety and efficiency of the sUAS-based UXO detection system (Qi et al. 2020). The same study also asserted that sUAS-based UXO detection system is safer and less expensive in comparison with the existing helicopter-based UXO-detection TEM system (2020). These two aforementioned papers present the advantages of sUAS-based UXO detection systems in terms of their safety and cost, but the researchers made no attempt to use single-spectrum or MS EO sensors to detect UXO, even though they are the most popular onboard sensors in the sUAS industry.

### 3. Feasibility of the DLCNN Application with sUAS

A study by Dorafshan et al. (2018) researched DLCNN training with concrete deck pictures taken by a sUAS onboard camera and by a handheld digital camera for structural inspection. The same paper concluded that "it is feasible to apply DLCNNs in autonomous civil structural inspections with comparable results to human inspectors when using off-the-shelf sUAS and training datasets collected with point-and-shoot handheld cameras" (Dorafshan et al. 2018). Considering the result of the aforementioned paper, this thesis presumes that datasets collected with a handheld digital camera substituting for an EO sensor can train the desired detector model adequately. Hence, this thesis conducted data collection in a similar manner using a handheld digital camera because the military base where the data collection was conducted prohibits flying sUAS to collect data on the base for security reasons.

### 4. UXO Detection Using Multispectral Imagery

Regarding the MS data related to UXO detection, many studies have suggested the possibility of future UXO detection work making use of an MS sensor. According to a report by the Defense Science Board Task Force on UXO detection in 2003, infrared spectrum data has potential value for surface and very shallow UXO (Etter and Delaney 2003). Johnson et al. suggested the possibility of the distinguishability between UXO and natural materials by using a spectral signature (1996). In a 2017 paper, Tokyo University studied multispectral object detection for autonomous vehicles. The authors proposed multispectral object detection using data from Red-Green-Blue (RGB), Near-Infrared (NIR), Mid-infrared (MIR), and Far-infrared (FIR) images. The spectral images (FIR, MIR, and NIR) have different features from the RGB image. In the 2017 paper, the trained model with multispectral images showed 13% higher mean average precision (mAP) compared to the RGB object detection (Takumi et al. 2017). Hence, this thesis builds on the availability of the multispectral camera for sUAS, the potential to detect UXO using multispectral imagery, and the methodology of object detection using multispectral imagery in combination to implement a multispectral UXO detection system.

# III. DEVELOPMENT OF THE DETECTION ALGORITHM

This chapter describes the AI technology-based algorithm that uses the most appropriate neural network for UXO detection. This chapter also develops the multispectral detection algorithm used for the system proposed in this thesis and presents the evaluation metric used to assess the system's effectiveness.

## A. IMPLEMENTING THE DETECTION ALGORITHM IN MATLAB

To train the AI-based model that detects UXO, the thesis makes use of MATLAB along with the Computer Vision Toolbox, ML Toolbox, and the DL Toolbox. MATLAB is a programming tool that integrates computation, visualization, and programming in a user-friendly computing environment (Yakimenko 2011). Without profound knowledge of CNN, users can leverage the MATLAB Toolbox to apply the technology for any objective. To implement the system's desired capability, this research adapts object detection from DLCNN MATLAB code examples.

## B. DLCNN-BASED OBJECT DETECTION ALGORITHM

This section explores object detection tasks and appropriate DLCNN models for UXO detection followed by designing the UXO detector model with the YOLO v2 network and estimating the required anchor boxes for the designed UXO detector CNN model.

### 1. Object Detection Using DLCNN with YOLOv2 Network

There are three common tasks for detecting objects in the CV area: Image Classification, Object Detection, and Semantic Segmentation (Sharma 2019). Image classification simply determines what the object is in the input image. Object detection uses bounding boxes to mark the predicted object in the input images (Zhao et al. 2019). The advantage of object detection is that it provides detailed detection results with the location of the object. Semantic segmentation gives the predicted object as pixel information. This last approach is proper for large-scale pictures such as data from a

satellite. Figure 6 shows how the three tasks look visually when they are applied in the UXO detection system.

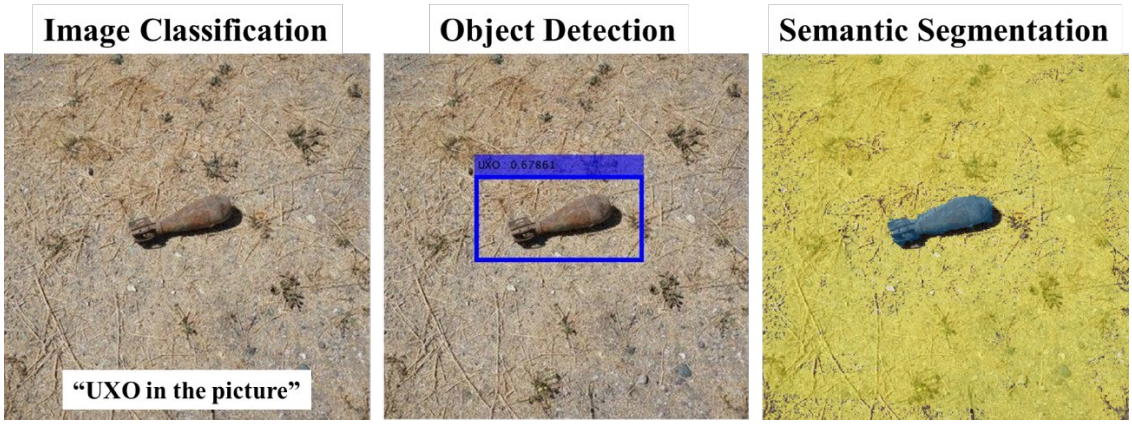


Figure 6. Comparison of Three DL Tasks in CV Area

The desired UXO system shall give the user enough information about the where UXO is located in the image and indicate how confident the system is about the detection result. In this regard, the research uses the object detection methodology. According to Zhao et al. (2019), object detection consists of two major tasks—object localization to give information about where the targeted objects are located by marking the detected objects with the bounding boxes, and object classification to give information about what the detected objects are by showing the category to which the detected objects belong. In a UXO detection system, the purpose of the system is to warn users about approximate location of the hazard; and classification of the object is not necessary because the users do not need to know what kind of ordinance the hazard is.

Deep Learning (DL)—also known as Deep Structured Learning—has improved the performance of object detection by introducing a trained CNN. Shen et al. explained that "deep learning allows computational models to learn the representations of visual data with multiple layers of abstraction" (2018). Specifically, the DL method based on the artificial multi-layer is called Deep Learning Convolutional Neural Network, or DLCNN, and the DLCNN is the most representative model of DL (LeCun, Bengio, and Hinton 2015). During the development of the proposed system, this thesis tried several ML-based detectors such as AGF and HOG, but finally applied DLCNN to the object

detection task for UXO detection to take advantage of increasing capability from its deeper network architecture.

The thesis explored three available popular CNN models for object detection tasks in the MATLAB environment:

- Faster R-CNN
- SSD
- YOLOv2

Faster R-CNN, proposed in 2016 by Ren et al., is an improved version of Fast R-CNN by Girshick in 2015. Ren et al. introduced the "Region Proposal Network (RPN) that shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals" (2016). Their method makes it possible to detect objects in real-time frame rates by integrating the RPN and Fast R-CNN (Ren et al. 2016).

SSD, which stands for Single Shot MultiBox Detector, was introduced by Liu et al. in 2016. With SDD, the authors introduced "a method for detecting objects in images using a single deep neural network" (2016). They also explained that a key feature of SSD is "the use of multi-scale convolutional bounding box outputs attached to multiple feature maps at the top of the network" (Liu et al. 2016). In the same paper, Liu et al. explained that "SSD is simple relative to methods that require object proposals because it completely eliminates proposal generation and subsequent pixel or feature resampling stages and encapsulates all computation in a single network" (2016).

Finally, YOLOv2 is an improved version of the You Only Look Once network (YOLO) (Redmon and Farhadi 2017). Redmon et al. (2016) said that they "frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities" and presented the YOLO network. They also explained that "a single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation" (Redmon et al. 2016). In the same paper, they said that "since the whole detection pipeline is a single network, it can be optimized end-to-end directly on

detection performance" (Redmon et al. 2016). Thus, YOLOv2 adopted several strategies to improve the YOLO by using batch normalization, high resolution classifier, anchor boxes, and dimension clusters (Redmon and Farhadi 2017).

The three most common CNN models are available in the MATLAB environment, and any one of them is reasonable for use in the object detection system. In a Garg et al.'s 2019 study of traffic sign recognition using the three CNN models, the SSD model lags behind Faster R-CNN and YOLOv2 in terms of both accuracy and speed. YOLOv2 is 3.5% more accurate than Faster R-CNN. On top of that, the learning rate of YOLOv2 is 68% faster than Faster RCNN and 16% speedier than SSD. However, the accuracy of YOLOv2 can be lower than Faster RCNN when detecting very small objects (Garg, Chowdhury, and More 2019). Figure 7 shows how quickly each network processed the data in frames per second in their study.

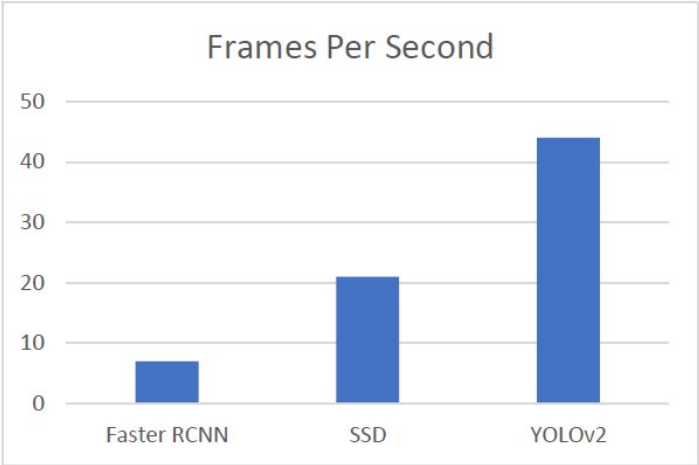


Figure 7. Learning Speed Comparison of Three CNN Models. Source: Garg, Chowdhury, and More (2019).

This thesis chooses the YOLOv2 CNN model because it is the fastest model and the UXO are considered big objects under the low-altitude sUAS flight.

### 2. CNN Model Setup for UXO Detector

To develop the UXO detection CNN model, this research chooses the ResNet50 as a backbone network. ResNet50 refers to the Residual Network with 50 layers based on residual learning. Deep convolutional neural networks have a characteristic that a deeper network solves more complex tasks with improvement of recognition accuracy. Nonetheless, as the network goes deeper, the more difficult it becomes to train the learning features of its last layers. Instead of learning features, a residual network is learning residual input from those layers, making it easier to train with a simple deep convolutional neural network (He et al. 2015).

This research chooses the Activation 40 Recertified Linear Unit (ReLU) layer to for feature extraction. At this layer, the model extracts UXO features from the collected UXO data. With this approach, the layers succeeding the feature layers are removed from the ResNet-50. The detection subnetwork along with the YOLOv2 transform and YOLOv2 output layers are added to the feature layer of the base network. Figure 8 shows the network layer structure.

### 3. Anchor Boxes Estimation for the UXO Detector Network

In developing their Faster R-CNN, Ren et al. introduced Anchor Boxes, which are predefined boxes "that serve as references at multiple scales and aspect ratios" (2016). While performing object detection, the set of anchor boxes tile across the image and predict bounding boxes. Redmon and Farhadi applied the anchor boxes to YOLOv2 to improve on the accuracy of the YOLO network (2017). The MATLAB CV Toolbox provides the anchor boxes estimation function using a k-means clustering method with the intersection over union (IoU) distance metric (Math Works, Inc. 2020). IoU indicates the overlap between two boundaries from the anchor box and the ground truth, and IoU can be expressed as Figure 9.

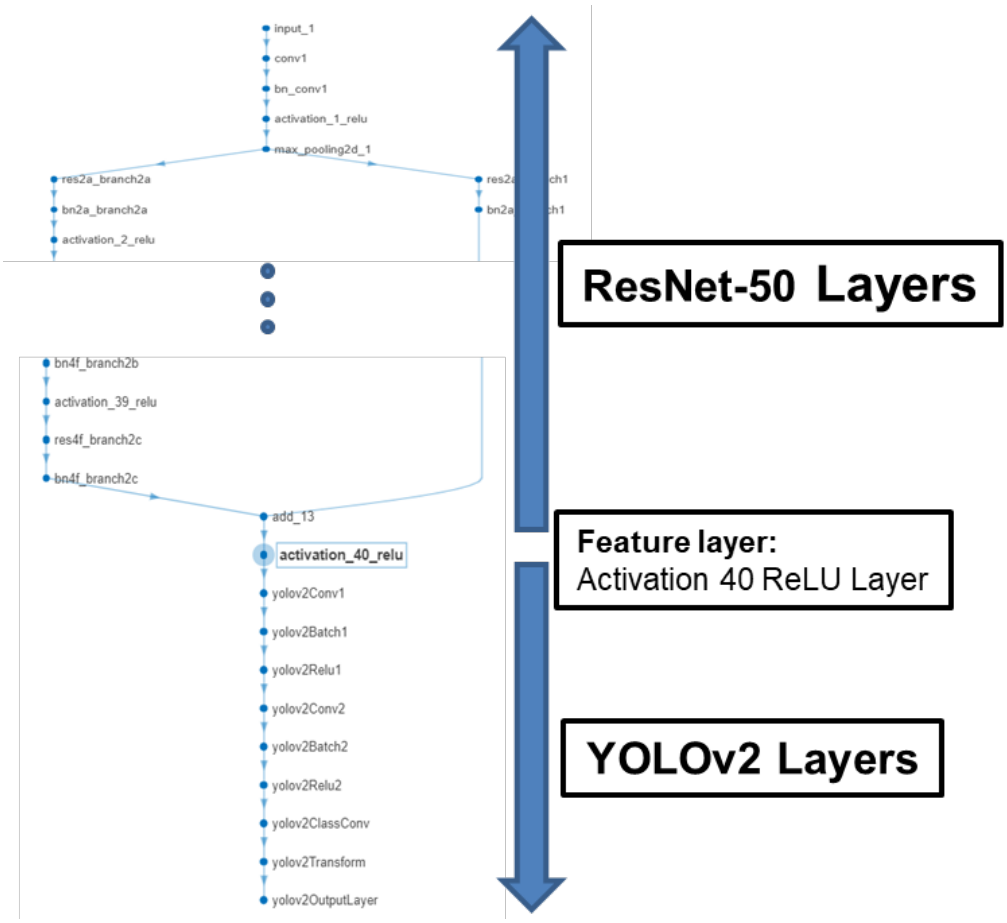


Figure 8. YOLOv2 Object Detector Network Setup

$$\text{IoU} = \frac{\text{area of overlap}}{\text{area of union}} =$$

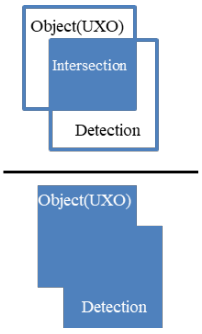


Figure 9. IoU Definition

The number of anchor boxes for training is an important parameter for YOLOv2 and because the number of Anchor Boxes impacts the efficiency and accuracy of the trained detectors (Redmon and Farhadi 2017). The more anchor boxes the training model uses, the higher the mean IoU the model achieves; however, using more anchor boxes can increase the training time and lead to overfitting—and worse detecting performance (Math Works, Inc. 2020). This thesis uses about a thousand UXO photos of the nine different kinds of UXO as training data, so this research presumed that the training network probably requires as many anchor boxes. From the training data for the UXO detector, the provided MATLAB code estimates the relationship between the number of anchor boxes and the mean IoU. Figure 10 plots the mean IoU versus the number of anchor boxes to measure the trade-off between them.

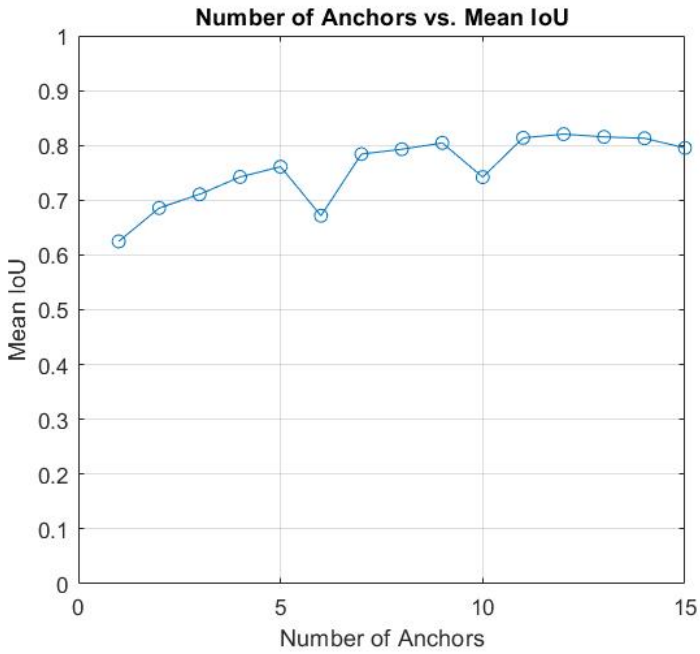


Figure 10. Number of Anchors vs. Mean IoU from the Training Data for the UXO Detector

Using nine anchor boxes results in around a 0.8 value of mean IoU. Six and ten anchors show an unexpectedly low mean IoU due to the UXO shape differences. Given these results, the research chooses nine anchor boxes to train the UXO detector with the mean IoU for the nine anchor boxes as 0.8045.

## C. MULTISPECTRAL DETECTION ALGORITHM

The multispectral camera generates multiple images of the different spectrums at the same moment. In the aforementioned Tokyo University study of multispectral object detection for autonomous vehicles, Takumi et al. presumed that "each spectral image has different characteristics and hence separately detecting objects in each spectral image will help to exploit the feature separately in each spectrum," and proposed an ensemble detection method for exploiting MS data to conduct object detection separately in each spectral detector and integrate the results (Takumi et al. 2017). This thesis applies the concept of the ensemble method in UXO imagery as the revised manner.

First, each spectrum of training data is used for each spectrum detector, respectively. The MS sensor used in this thesis generates five spectrum data: Blue, Green, Red, Near Infrared, and Red Edge. With the five sets of the different spectrum data, five detectors are trained; that is, Detector 1 for the Blue spectrum UXO image, Detector 2 for the Green spectrum UXO image, and so on. Figure 11 shows the procedure of training detectors graphically.

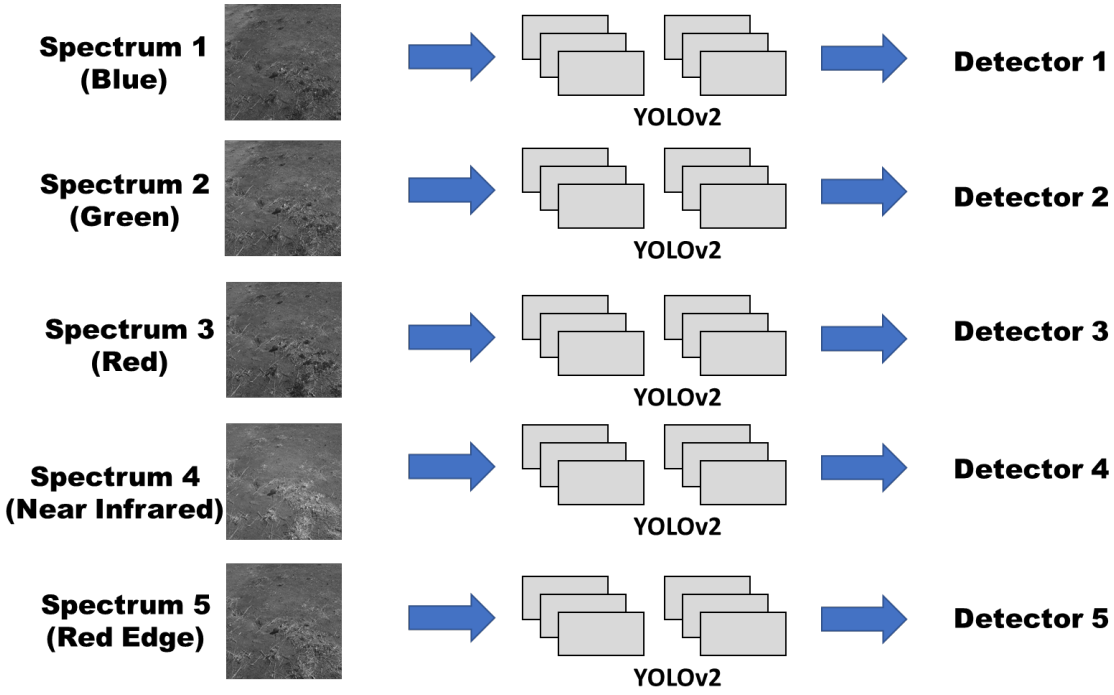


Figure 11. Training Detectors for MS Data

In detecting UXO on given MS data, the given MS data go into the respective detectors; that is, the Red spectrum image goes into Detector 3, the Blue spectrum image goes into Detector 1, and so on. As a result, each detector detects UXO in its spectrum area and has the different detection results with those of the other spectrum detectors.

Using the detection results, this thesis takes the following two steps to integrate the five different types of spectral detection results into one.

1. Simple Merging
2. Non-Maximal Suppression (NMS)

Simple merging puts the detection results from the five detectors into a single space. For instance, if Detectors 1 and 2 detect UXO but Detectors 3, 4, and 5 do not, the simple merging result will be two bounding boxes in an UXO image. This application of this method enables one spectral detector to complement the detection results from each of the other detectors.

The next step is suppressing weak detection results and leaving the best detection result by using the NMS method. From the results of simple merging, candidates for the whole detection result, and NMS chooses the best candidate from the detection box with the highest score while suppressing all the other overlapped detection boxes above a certain threshold of IoU (Bodla et al. 2017). Although the IoU threshold is usually set as 0.3, the IoU threshold is set as 0.1 in this thesis. Hence, all the other candidates are suppressed for which IoU is higher than 0.1, with the best detection result. The reason for setting the IoU threshold as 0.1 relates to the MS sensor characteristic; the MS sensor used in this thesis has five physically different lenses for the five spectrums, and the different physical locations of the lenses cause misaligned MS data. Even though each spectral detector successfully detects UXO, the simply merged bounding boxes are misaligned and this with a high IoU threshold makes the system think they detected different objects. To prevent the actual overlapped detection result from being recognized as not overlapped, the IoU threshold is set lower as 0.1. Hence, the redundant detection results can be effectively suppressed except for the best detection result. The two-step integration process of MS detection is visually explained in Figure 12.

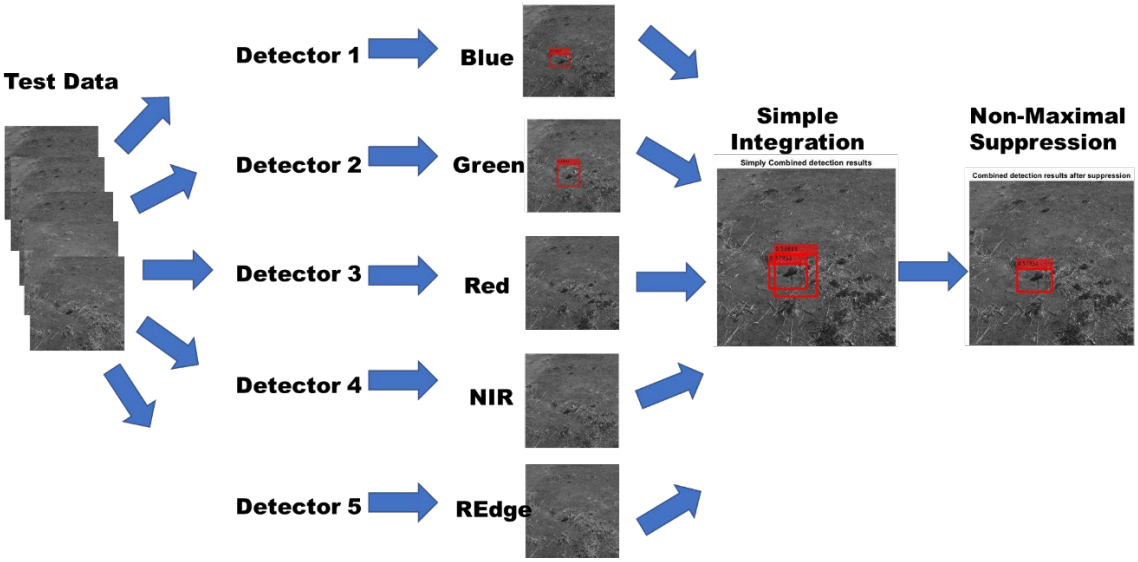


Figure 12. Two-Step Process of Multispectral Detection Integration

## D. MEASURE OF EFFECTIVENESS

After training the UXO detector model, this thesis evaluates the trained detector in terms of performance and feasibility. This thesis uses average precision (AP), an evaluation metric in popular object detection competitions, to measure the accuracy of the trained object detectors. In the ML area, to compute AP, a confusion matrix is used to visualize the performance of an algorithm, in which each row indicates a test result and each column indicates an actual state (Powers 2008). Figure 13 shows the confusion matrix for the trained UXO detector evaluation.

| Confusion Matrix | | Actual state | | |
|---|---|---|---|---|
| | | True | False | |
| Test result | Positive Detection | True Positive | False Positive | Total Positives |
| | Negative Detection | False Negative | True Negative | Total Negatives |
| | | Total True | Total False | |

Figure 13. Confusion Matrix for UXO Detector Evaluation. Adapted from Stehman (1997).

Positive or Negative is determined by whether the UXO detector detects UXO in a given image. True or False is determined by whether the value of the IoU is larger than a certain IoU threshold, indicating how much the detection results and the actual UXO location overlap. Precision measures how accurate the UXO detector's predictions are based on the ground truth data—in this case, actual UXO location. So,

- if the IoU between the prediction and the ground truth is more than or equal to a IoU threshold, the detection is classified as True Positive;
- if IoU is less than a IoU threshold, the detection is classified as False Positive; and
- if a ground truth is present in the picture and the detector failed to detect the object, it is classified as False Negative.

This concept of UXO detector precision is represented in an equation as:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}} = \frac{\text{True Postive (Correct Detection)}}{\text{Total Positive (All Detections)}} \quad (1)$$

Recall measures how well the detector finds all positives based on the ground truth data. The recall of the UXO detector evaluation is mathematically defined as:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negative}} = \frac{\text{True Positive (Correct Detection)}}{\text{Total True (Actual UXO)}} \quad (2)$$

This research sets the IoU threshold as 0.5 for the evaluation of single-spectrum UXO detection and the IoU threshold as 0.4 for the evaluation of multi-spectrum UXO detection. The reason the MS detector has a lower threshold is the MS data are not aligned with each other as discussed in the MS detection algorithm. The misalignment of data can cause the phenomenon in which a True Positive—correct detection result—is recognized as a False Positive—wrong detection result. Setting a lower threshold of 0.1 for MS detection can prevent the actual True Positive from being evaluated as False Positive.

After the trained UXO detector detects UXO in a given set of test data, this thesis ranks the detection results in descending order of the predicted confidence level. To visualize the Precision and Recall values of the detection results, this thesis plots the Precision-Recall (PR) curve, which indicates the precision of the trained detector at different recalls (Math Work, Inc. 2020). AP computes the average Precision value for the Recall value and finds the area under the PR curve. In the PASCAL Visual Object Classes (VOC) Challenge in 2010, the shape of the PR curve is calculated as the mean Precision at a set of 11 equally spaced recall levels [0,0.1,…,1] (Everingham et al. 2010). The 11-point interpolated AP is calculated as an equation.

$$\mathrm{AP} = \int_0^1 P(r) \approx \frac{1}{11} \sum_{r \in \{0, 0.1, \ldots, 1\}} P_{\mathrm{interp}}(r) \tag{3}$$

Calculated AP would thus be between 0 and 1, and this thesis presumes that the higher the AP, the better the detector.

# IV. DATA COLLECTION AND PROCESSING PROCEDURE

This chapter describes the data sources found within this research, the data collection setup, and data processing procedure.

## A. AVAILABLE DATA SOURCES

For training the designed sUAS-based UXO detector model, this research needed to acquire lots of UXO imagery, especially images taken of the ground. To obtain sufficient and useful data from UXO imagery in situ, this researcher tried to get access to UXO samples at three locations: Fort Ord, CA; Camp Roberts, CA; and Yuma Proving Ground (YPG), AZ. UXO samples that Ford Ord has are only available on the stand whiteboards for an educational purpose, so it was limited to UXO imagery data on the ground. YPG has enough UXO samples but the author could not travel to collect that data because of the restriction of movement due to COVID-19. Eventually, the data used for this thesis, which consist of pictures and videos of UXO on the ground surface, were collected at Camp Roberts in San Miguel, CA. The camp is a National Guard base and the Naval Postgraduate School performs the experimentation of unmanned systems at this base. The single-spectrum EO sensor data were collected on July 31, 2020, and the MS data were collected on November 13, 2020. There was no precipitation on those days, and the pictures were taken from 10:00 to 13:00 on both dates. The ground condition of the environment is a chalky soil with little vegetation. Figure 13 shows the appearance of the land at Camp Roberts where the data collection was executed.

Figure 14. Ground Condition of Camp Roberts

## B. TOOLS USED IN DATA COLLECTION AND PROCESSING

This research made use of commercial-off-the-shelf (COTS) assets available at the NPS to collect data and process the collected data.

### 1. Single-spectrum EO Sensor

In this research, single-spectrum EO Sensor refers to a standard RGB camera. Many commercial sUAS are equipped with an EO sensor, and users can take a view from the camera in the air through the EO sensor. Recently, the performance of the drone camera has become almost as good as a high-resolution handheld camera. The single-spectrum EO sensor is able to take pictures and record videos just like a digital handheld camera. This research uses the Sony Alpha 6000 camera, shown in Figure 15, as a substitute for the sUAS onboard EO sensor. DJI drones integrated with EO/IR sensors are currently banned at Camp Roberts, so all tests were conducted manually, by the researcher carrying a sensor around in his hands. The brief summary of the camera's specifications is provided in Table 1.


Figure 15. Sony Alpha 6000.
Source: "Sony A6000 E-Mount Camera with APS-C Sensor" (2020).

Table 1. Sony Alpha 6000 Digital Camera Specification.
Source: Sony (n.d.).

| | |
|---|---|
| **Brand** | Sony a6000 Digital Camera |
| **Spectrum** | Single Spectrum (RGB) |
| **Size** | 4.72×2.64×1.77 inches |
| **Weight** | 0.76 lbs. |
| **Angle of View** | 83° to 32° |
| **Shutter Speed** | 1/4000 sec |
| **Capture Rate** | 60 fps (Video Recording) |
| **Resolution** | 6000×4000 (24 MP) |

### 2. Multi-spectrum Sensor

The MS sensor refers to a camera that can take a picture within specific wavelength ranges across the multiple spectra. The MS sensor is well-known in the commercial sUAS industry for professional purposes, and users make use of the captured MS data for a variety of objectives. For instance, the rescue industry uses MS sensors to rescue people and agriculture industry uses the MS to analyze vegetation.

Within the military sector, a DOD study explained that UXO detection based on MS data including infrared-spectrum data can be effective in detecting surface or near-surface UXO on the ground (Etter and Delaney 2003). A project focused on developing a strategy for UXO sensing also asserted that the structure within the spectral signature of the UXO material may provide features with which to distinguish between UXO and natural materials (Johnson et al. 1996). Additionally, Takumi et al. from Tokyo

University found that MS-based object detection had a higher average precision rate than single-spectrum RGB object detection in their project (Takumi et al. 2017).

In this regard, this thesis tries to take advantage of the sUAS onboard MS sensor and the multispectral data from UXO. This research uses a MicaSense RedEdge camera, shown in Figure 16, to collect multispectral data. The multispectral camera has five lenses on it and each lens operates for a specific spectrum: Red, Green, Blue, Near-infrared, or Red Edge spectrum. Additionally, the camera simultaneously captures five discrete spectral bands. The specifications of the camera are listed in Table 2. As with the digital camera, this research collects the multispectral data by hand, capturing images without flying a sUAS.


Figure 16. MicaSense RedEdge. Source: MicaSense (2020).

Table 2. Micasense RedEdge Specification. Source: Micasense (n.d.).

| **Brand** | MicaSense RedEdge |
|---|---|
| **Spectrum** | Five Spectrum (R G B IR RedEdge) |
| **Size** | 4.8×1.8×2.6 inches |
| **Weight** | 0.33 lbs. |
| **Angle of View** | 48° |
| **Shutter Speed** | 1/500 sec |
| **Capture Rate** | 1 fps |
| **Resolution** | 1280×960 (1.2 MP per EO band) |

The MicaSense RedEdge is designed to provide multiband data for agricultural remote sensing. The five spectrums capture narrow bands of blue (centered on 475 nm), green (560 nm), red (668 nm), near infrared (NIR, 840 nm), and red edge (717 nm), allowing for accurate detection of vegetation health and discernment of species based on their spectral signatures as shown in Figure 17. This research assumes that UXO imagery has different features in the different bands of the spectrum.

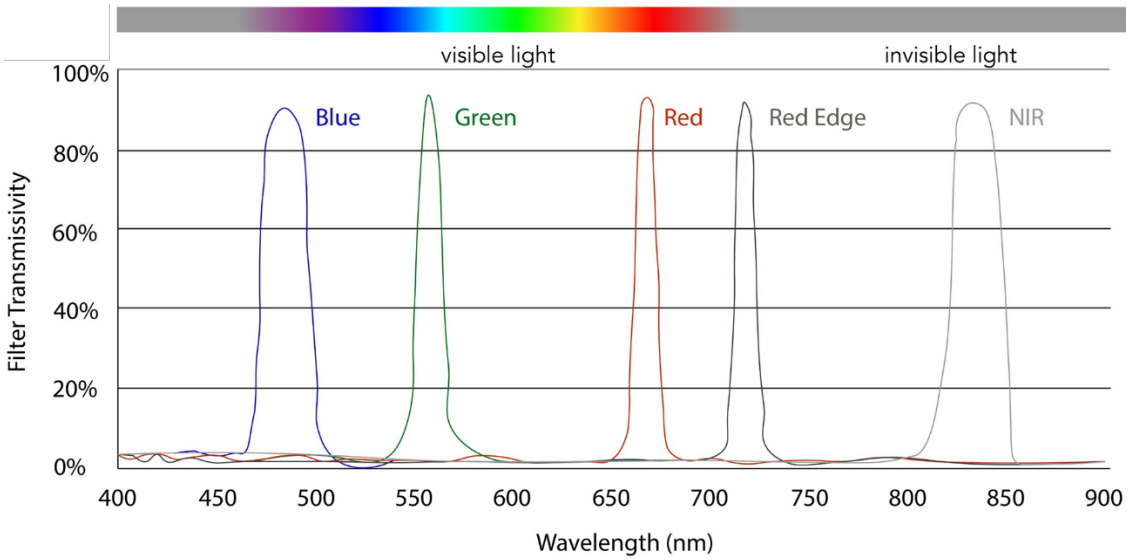


Figure 17. Spectral Bands of Micasense RedEdge. Adapted from MicaSense (2018).

### 3. Computation Platform

Computations were conducted on an HP Pavilion 15, shown in Figure 18, equipped with a I7 Core and NVIDIA GeForce MX250 video card. The performance of the laptop is lower-end dedicated and may not be an optimal choice for DL applications. As discussed in Chapter III, MATLAB software was run on this laptop to implement the system algorithm for training and to analyze the results. The specific performance of the laptop used in this research is shown in Table 3.

Figure 18. HP Pavilion Laptop. Source: HP (n.d.).

Table 3. HP Pavilion Specification. Source: HP (n.d.).

| | |
|---|---|
| **Laptop name** | HP Pavilion—15-cs3073cl |
| **Microprocessor** | Intel® Core™ i7-1065G7 (1.3 GHz base frequency, up to 3.9 GHz with Intel® Turbo Boost Technology, 8 MB cache, 4 cores) |
| **Memory** | 16 GB DDR4-2666 SDRAM (2×8 GB) |
| **Hard drive** | 1 TB 5400 rpm SATA |
| **Graphic card** | NVIDIA GeForce MX250 |

## C. DATA COLLECTION

For UXO imagery data, the thesis chooses nine different types of UXO. The UXO belong to Camp Roberts and are used for educational purposes. They include mortar projectile, hand-grenade, bullets, etc., and all of them have a different shape, color, and size. This thesis does not specifically classify the kind of UXO data. Based on the assumption that different UXO probably have some commonalities in their appearance, the designed UXO detector does not classify what kind of UXO but only marks the location of the suspected UXO using a bounding box with a score. Figure 19 shows the nine different UXO types that are used in this research.

Figure 19. Nine UXO Used in Data Collection

Camp Roberts is basically a security-sensitive military base, so they regulate flying sUAS, especially commercial DJI brand drones made in China, at the base. That is, the collecting of UXO imagery data by flying sUAS with onboard sensors was prohibited. As mentioned previously in the literature review, the trained detector model using a training dataset from the point-and-shoot handheld camera shows acceptable detection results while on flying sUAS (Dorafshan et al. 2018). In consideration of the limitation and the reviewed literature, the data collection activities were conducted manually, by the author carrying sensors around in his hands or using a tripod. So, the data collection was conducted at a fixed five-foot altitude. Figure 20 shows how the data collection was conducted using a tripod.

Figure 20. Data Collection Using a Tripod

To obtain diverse data and an actual operational challenge, UXO were randomly placed on the ground. Some were placed under the bushes to implement an operational environment. Figure 21 shows examples of the UXO imagery taken.

Figure 21. Randomly Placed UXO

With the aforementioned conditions, EO sensor and MS sensor data were collected under the same environment conditions. In all, 1,225 photos of EO sensor data—single-spectrum RGB imagery—were taken using the Sony Alpha 6000 and 59 videos, each about ten to 20 seconds in length, were taken as well. For MS data, a total of 4,075 photos—815 photos for each spectrum—were collected using the MicaSense RedEdge. Figure 22 graphically shows the data collection result.

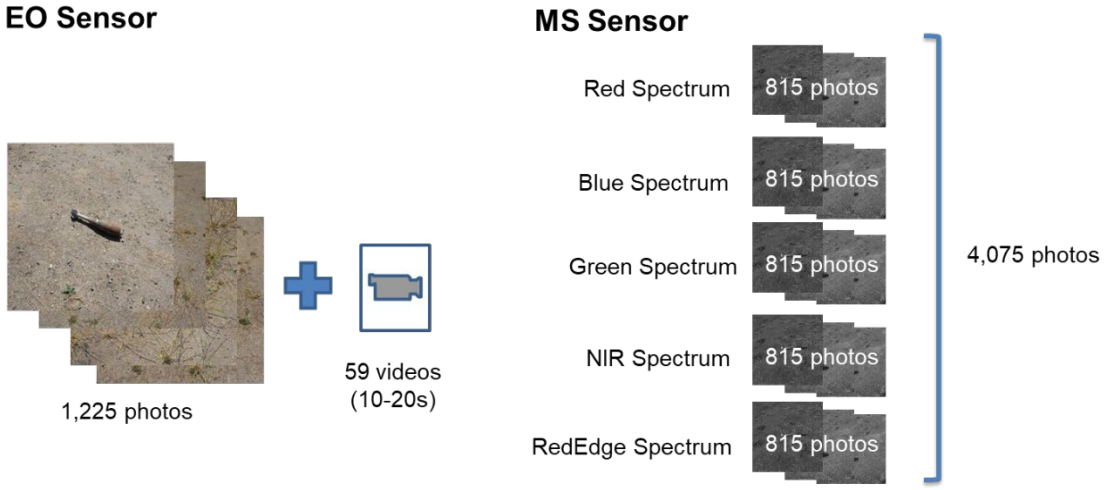


Figure 22. Data Collection Result for Single- and MS Sensors

## D. DATA CURATION

This section explains the data process workflow to manipulate the collected raw data to it was proper for training the UXO detector model. The process consists of four steps: image resizing, image labeling, dividing data, and augmenting data. The data process was conducted in the MATLAB environment and the specific code to implement the process is presented in Appendix A. The detailed data process for each sensor is discussed in Chapters V and VI, respectively.

### 1. Image Resizing

This research chose the most effective way to process data. Although the selected CNN model of YOLOv2 can train a model with pictures of any size, using this capability would make training take longer than using images of a specified size. Hence, the

collected raw images were resized as 416×416 because YOLOv2 shows optimal performance with images of that size. The image resolution of the raw data collected by the Sony Alpha A6000 was 6000×4000, and the raw data was resized and saved to 416×416 through MATLAB computation as shown in Figure 23.

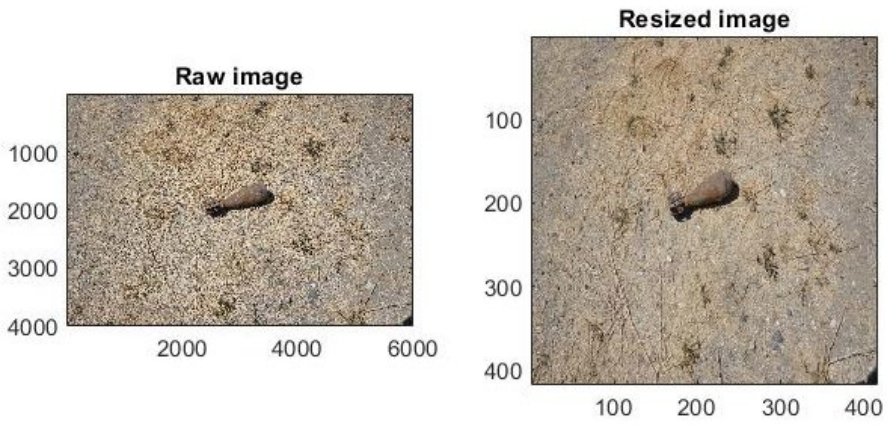


Figure 23. Image Resizing to 416×416

### 2. Image Labeling to Ground Truth Data

Object detection requires image labeling to identify and locate objects in images. This research made use of the Image Labeler app provided by MATLAB. This app has a user-friendly interface as shown in Figure 24, so users can easily label the images using a graphical approach.

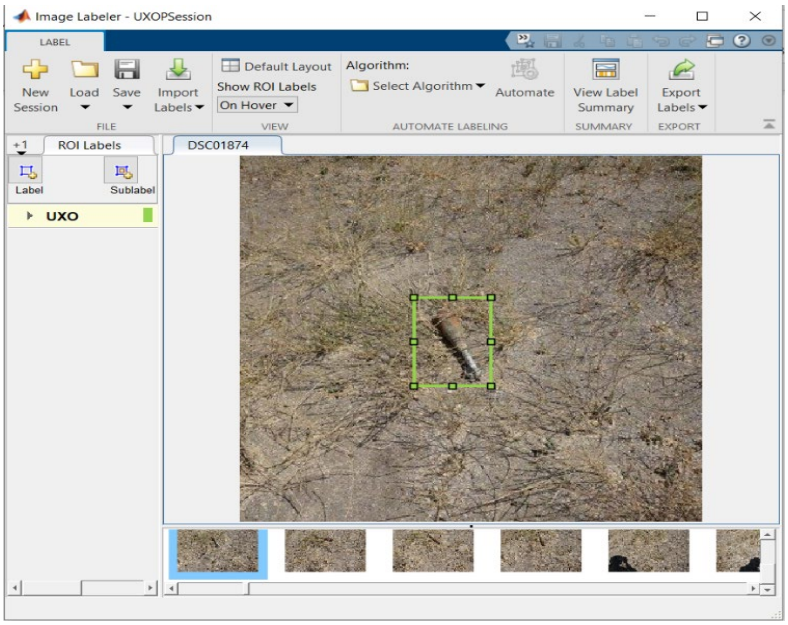


Figure 24. Data Labeling using MATLAB Image Labeler

This research classified all data as one type of labeling, UXO. By doing so, the trained UXO detector would not classify types of UXO but would notify users of the presence of any suspected UXO. This method converts the resized pictures to labeled ground truth data.

### 3. Dividing Labeled Data into Training/Validation/Test Subsets

With the collected data, this research tries to effectively train UXO detectors with a relatively small amount of data and evaluates the trained detector performance from the perspective of feasibility. In this regard, this thesis divides the labeled data into three sets randomly: a set of training data, a set of validation data, and a set of test data. Specifically, this thesis shuffles and divides the collected and labeled data, respectively, as 70% for training, 15% for validation, and 15% for test and evaluation. Training data is literally for training the UXO detector. Validation data enables the UXO detector to correct itself at regular intervals during training, and this improves the detection performance and accuracy effectively. Test data is for evaluation of the trained detector.

## 4. Data Augmentation

Data augmentation is a strategy to increase the amount of data for training by tweaking existing data without collecting additional data (Ho et al. 2019). Shorten and Khoshgoftaar explained that "Data Augmentation can improve the performance of their models and expand limited datasets to take advantage of the capabilities of big data" (2019). Even though hundreds of pictures of the UXO on the ground surface are purposely taken for this research, the number of images is still not sufficient to train the CNN-based UXO detector model. Lack of training data causes overfitting because of the high variance and this results in the bad performance of the trained detector model (Shorten and Khoshgoftaar 2019). This thesis applies data augmentation, including geometric transformations and color space augmentations, in the collected UXO image data. Figure 25 shows the augmented UXO images with the label by rotating, changing brightness, and changing the contrast.

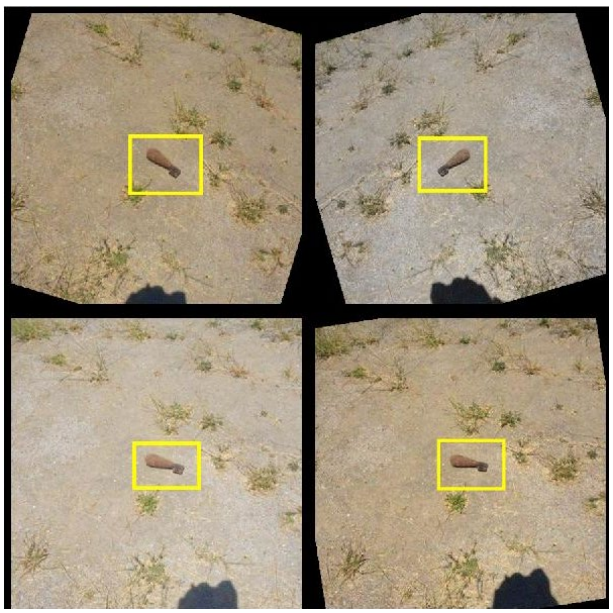

Figure 25. Data Augmentation for UXO Detector Training

# V. ASSESSMENT OF A SINGLE-SPECTRUM UXO DETECTION

This chapter describes the process of single-spectrum UXO detection. First, this research processes the collected single-spectrum RGB UXO data by resizing, labeling, dividing, and augmenting that data. Then, this research executes training a UXO detector model along with proper training options. Next, the research visually demonstrates the detection results for a given set of test data using bounding boxes. Lastly, the research evaluates the detector according to AP evaluation metrics using an IoU threshold of 0.5. The MATLAB code for training and assessing the UXO detector for a single spectrum is presented in Appendix A.

## A. DATA PROCESSING

For the vision data, 1,225 UXO pictures were collected using a Sony Alpha A6000 camera as a single-spectrum EO sensor substitute. The data process flow for the single-spectrum data follows, as discussed in Chapter IV, section D. First, all 1,225 raw images were resized as 416×416, which is the optimal image size for YOLOv2. Second, the resized UXO pictures were labeled with the localization and classification information using Image Labeler in MATLAB. Next, the labeled data are shuffled and divided into the three groups randomly: 70% of 1,225 data (that is, 857) is designated as training data, and 184, which is 15% of the data, is assigned a set of validation data. The remaining 15%, 184 test data, is set to test and evaluate the trained detector. Lastly, the assigned training data are augmented by rotating, changing brightness, and changing the contrast. In this thesis, the training data are augmented to four times of the training data, that is, the 857 data become 3,428 data. Figure 26 visualizes the data process flow for single-spectrum data.

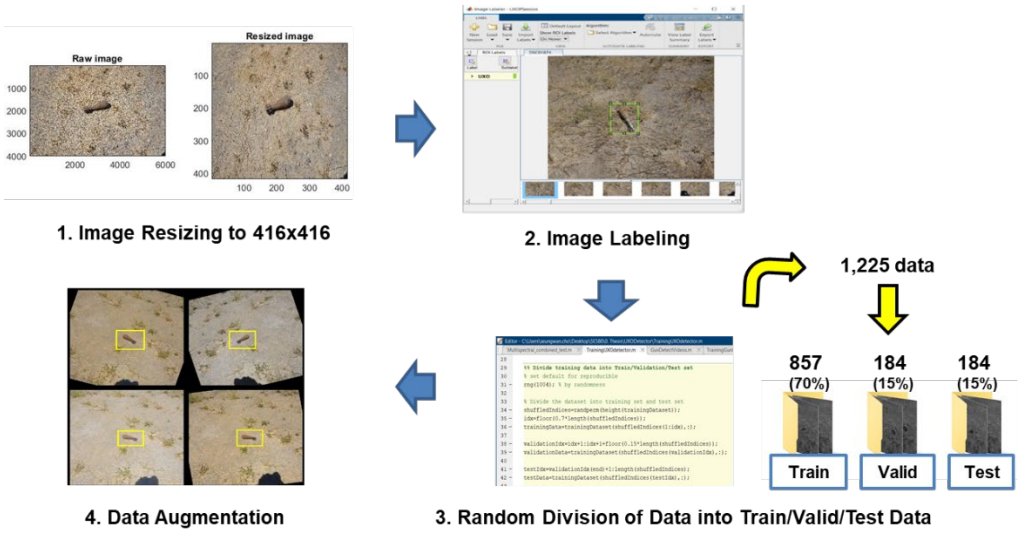


Figure 26. Data Process Flow for EO Sensor

## B. TRAINING PROGRESS

For training the UXO detector, this research uses a ResNet50-based YOLOv2 neural network as suggested in Chapter III. This research attempts to apply optimal training options for the UXO detector. For the optimizer, this research applies the Adam method, which is "an algorithm for first-order gradient-based optimization of stochastic objective functions based on adaptive estimate of lower-order moments" (Kingma and Ba 2017). This algorithm uses a subset of training data, which is called a mini-batch. According to Siddiqua et al., "The full pass of the training algorithm over the entire training set using mini-batches is one epoch" (2019). This research sets 20 maximum epochs and eight mini-batch sizes. With these options, the detector is trained with augmented data for training and 184 labeled data for validation, as divided previously. The number of training data, 857, is divided into eight mini-batches of 107. That is, 107 iterations complete an epoch, so total 2,140 iterations are taken for 20 epochs. Figure 27 explains the setup of the training options graphically.

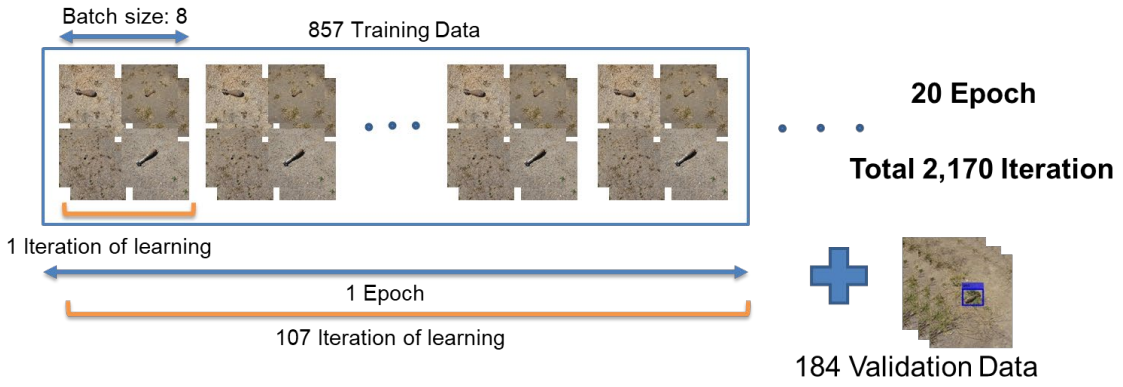


Figure 27. Training Options Setup

These conditions made the training take an hour and 58 minutes to complete. During training, the repeating process causes training loss and the YOLOv2 network optimizes the training loss. The training loss is a mean squared error (MSE), which is the summation of localization error, confidence error, and classification loss (Math Works, Inc. 2020). At each iteration of training, MATLAB provides the training loss information to the user, and the training loss is presumed as a metric to evaluate the training progress. Training loss for the first iteration is 439.1298, but it is lower than 20 at the sixth iteration. At the end of the 2,140th iteration, the training loss is near zero. Figure 28 shows the loss information provided by MATLAB while training, and the training loss plot representing the training loss that occurred for all iterations. There is little training loss at the end, so this thesis concludes that the training progress was fairly decent.

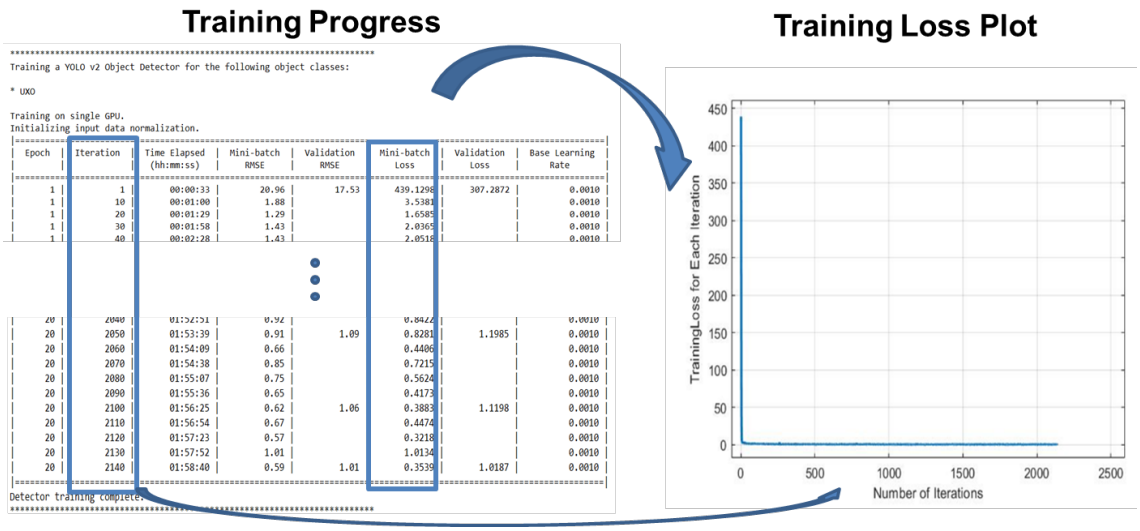


Figure 28. Training Completion and Training Loss

## C. DETECTION DEMONSTRATION

The trained UXO detector can detect UXO in any given pictures and videos. In this section, the research demonstrates how the detector detects and marks UXO visually for the users.

To demonstrate how the detector works in the pictures, the trained detector detects UXO in the several random test data from among 184 data. The UXO detector marks the bounding box on the predicted UXO in the picture along with detection confidence scores. The detection threshold is 0.5, which means the detections that have scores less than 0.5 are removed. Figure 29 shows detection examples of three bounding boxes of detected UXO pictures with each score and one picture of undetected UXO.


Figure 29. Demonstration of Detecting UXO in a Series of Individual Frames

In addition to the collection of pictures, this thesis recorded 59 UXO videos for the purpose of demonstration. The videos are between 10 and 20 seconds long and were made under the same conditions as the picture data. Four UXOs are randomly placed on the ground, and the author recorded videos from the five-foot altitude while walking slowly, which serves as the sUAS flying. Among the recorded videos, three videos are randomly chosen, and the trained detector detects UXO in the videos. The detection demonstrations in the videos are shown in Figure 30.

Figure 30. Demonstration of Detecting UXO in Videos

There are four UXO for each video, so a total of 12 UXO are on the three videos. Each video consists of thousands of frames, and hence, the detector has a lot more chances to detect UXO rather than it does from a given single picture. This research assumes that the success of the detection is cogitable bounding boxes for the UXO. As a result, the trained model detected 11 of the 12 UXO and marked them successfully.

## D. EVALUATION OF TRAINED DETECTOR

This research uses AP as a metric to evaluate the effectiveness of the trained UXO detector model. Precision and Recall are computed based on the test data detection result. By plotting the PR curve, the AP can be computed. MATLAB CV toolbox provides a handy code to compute the AP—for the specific codes used in this system, refer to Appendix A.

Conducting UXO detection on 184 test data, the detector detected 193 UXO with the values of Precision and Recall based on the ground truth data. The IoU threshold is set as 0.5. With those values, this thesis plotted the PR curve as shown in Figure 31. In the PR curve, the ideal precision is 1 at all recall levels. The AP, which is the area under the PR curve, is commutated using the 11-interpolation method as used in the PASCAL

VOC2008 Object Detection Challenge. The AP of the single-spectrum UXO detector is computed as 0.774.

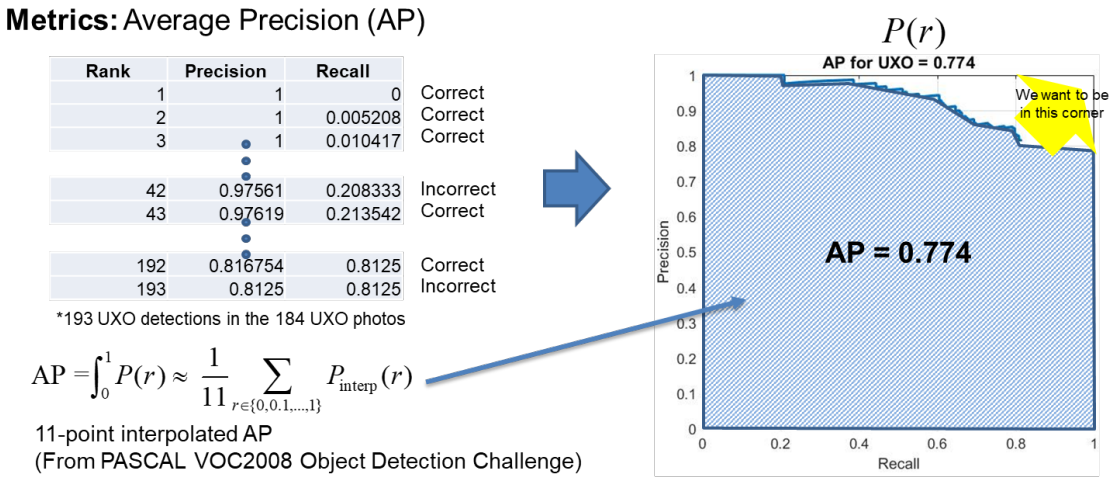


Figure 31. Precision-Recall Curve for Visional UXO Detector

THIS PAGE INTENTIONALLY LEFT BLANK

# VI. ASSESSMENT OF A MULTI-SPECTRUM UXO DETECTION

This chapter describes multi-spectrum UXO detection using an MS dataset. The data process workflow is the same as the one for the single-spectrum UXO detection assessment described in the previous chapter. The UXO detector for the five different spectrums was trained respectively. After conducting UXO detection in test data by respective UXO detector, the detection results are integrated with two steps—simple merging and non-maximal suppression (NMS). The first step is simply merging the detection results from the respective detector, and the second step is suppressing detection results other than the best, using the NMS method. Considering the MS data characteristic, which is the misalignment of different spectrum data, the IoU threshold of 0.1 is applied when suppressing anything other than the strongest detection results. Finally, this research evaluates the multispectral detection result using the AP evaluation metric with an IoU threshold of 0.4. The MATLAB code for the integration of MS UXO detection is presented in Appendix B.

## A. MULTI-SPECTRUM DATA CHARACTERIZATION AND PROCESS FLOW

For MS data, the MicaSense RedEdge camera equipped with the five spectrum lenses took 4,075 pictures, and this means 815 pictures were obtained for each spectrum. Because the series of pictures were taken at the same time but with lenses located in different spots, the pictures taken are slightly misaligned. That is, the UXO is located in a slightly different location from the other spectrum data. This difference is not immediately recognizable, but this may cause an overlap insufficiency problem when integrating and suppressing the detection results. As discussed in Chapter III, this phenomenon affects the MS algorithm, requiring a lower IoU than the single-spectrum detection algorithm in order to prevent the actual overlapped detection results from being recognized as not overlapped. Figure 32 shows an example of one series of UXO pictures.

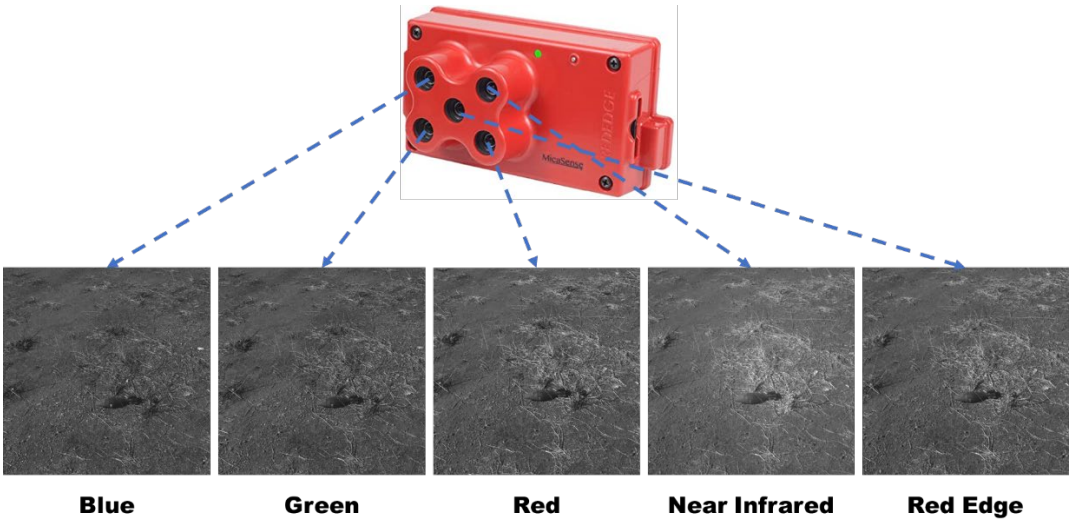


Figure 32. UXO Pictures from Each Spectrum and Related Alignment Problems

This research follows the same process as the single-spectrum UXO detector data process flow. First, all 4,075 images are resized to 416×416. Then, all UXO in the images are labeled with bounding boxes. Next, the data are randomly organized for each spectrum; 815 data for each spectrum are divided into 571 data for training, 123 data for validation, and 122 data for testing. To complement the data shortage, data augmentation is applied as well. Figure 33 shows the data process flow for the MS sensor.

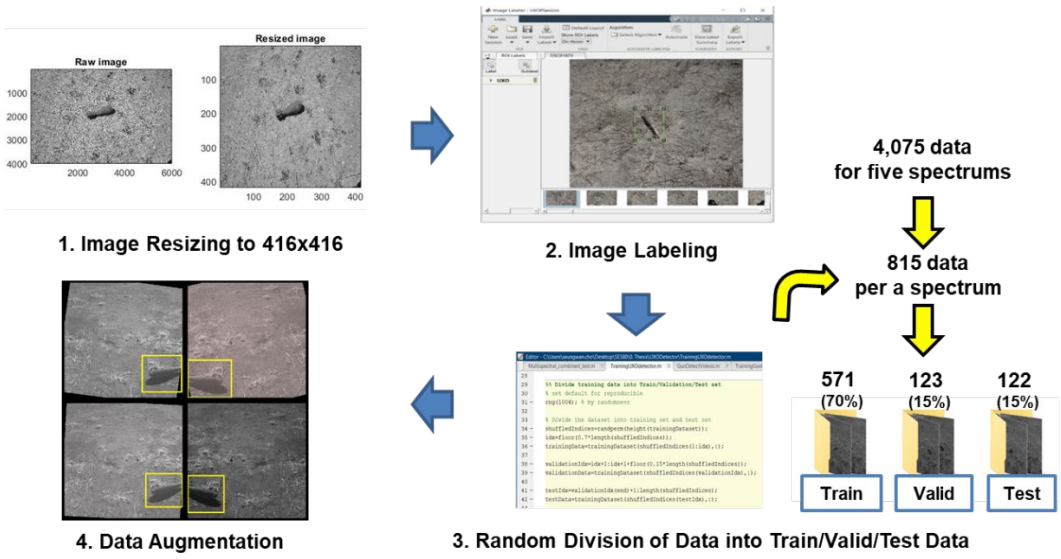


Figure 33. Data Process Flow for MS Data

## B. TRAINING AND DEMONSTRATION OF INDIVIDUAL-SPECTRUM DETECTORS

The training process for each detector is also executed in the same manner as it was for the single-spectrum detection training. The mini batch size is selected as eight and the maximum number of epochs is selected as five to reduce the training time. With the 571 data for each spectrum detector, the training takes about 20 minutes for each detector. While training, all five single-spectrum detectors show only insignificant training loss. Figure 34 shows the training loss of the Red spectrum detector as an example and the rest of the four detectors show similar training loss plots.

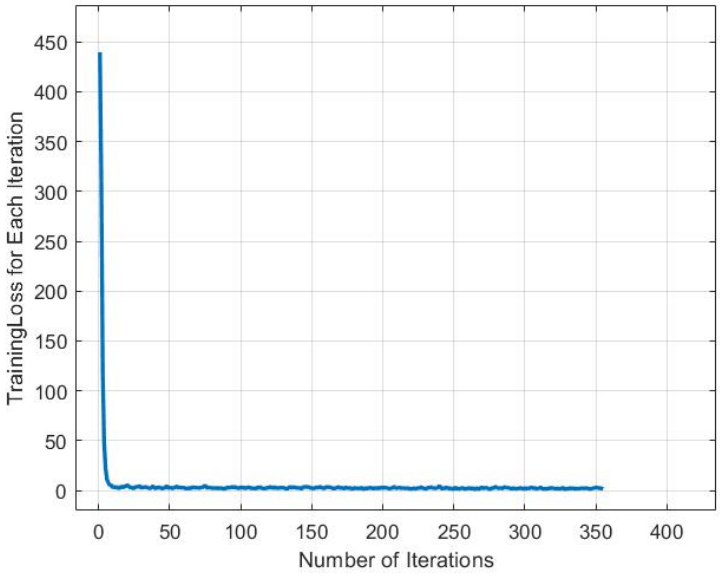


Figure 34. Training Loss for Red Spectrum Detector

With the five trained UXO detectors, this thesis conducted UXO detection for each spectrum, respectively. The workflow is the same as it was for single-spectrum UXO detection. Figure 35 is an example of the detection results from two different detectors for a given set of test data. In this instance, the Blue detector detects UXO using the smaller, precise bounding box that has a score of 0.57829. The NIR detector detects UXO in a slightly dislocated, bigger bounding box that has a score of 0.56028. In the same picture, the two detectors show only a little different detection result.

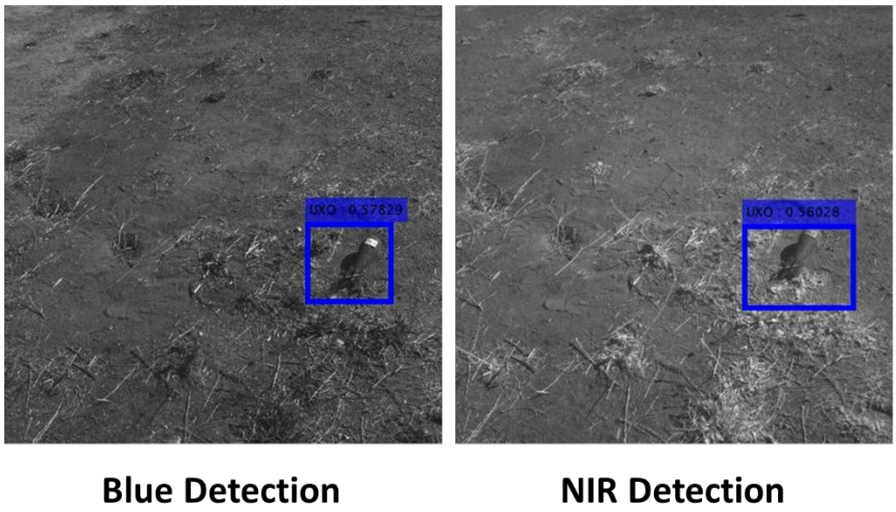


Figure 35. MS Detection Demonstration

Based on the comparison of these detection differences, this research found two interesting phenomena:

- Some single-spectrum detectors can detect UXO in their spectrum data, but other spectrum detectors cannot detect UXO in their spectrum data. For instance, in some cases Green and Blue detectors detect UXO, but the other detectors cannot. In some other cases, only the NIR detector can detect UXO while the other detectors cannot.

- When all detectors can detect UXO in their own spectrum, all the bounding boxes differ in size. That is, some spectrum detectors can detect UXO more precisely than the others. For instance, in some cases Green and Blue detectors detect UXO more precisely than the other detectors. In some other cases, the Red and NIR detectors detect UXO more precisely than the others.

These two findings reflect that respective spectrum detectors are complementary. In this regard, it is presumed that integrated detection results would be improved when their detection results are integrated.

## C. EVALUATION OF RESPECTIVE DETECTORS

Evaluation of each spectrum detector for 122 test data was executed. This research sets the IoU threshold for the evaluation as 0.5 as the evaluation metric for the single-spectrum detector. AP for the respective detectors is computed as between 0.484 and 0.592:

- Blue spectrum detector AP: 0.433
- Green spectrum detector AP: 0.583
- Red spectrum detector AP: 0.453
- Near infrared spectrum detector AP: 0.484
- Red edge spectrum detector AP: 0.592

The respective spectrum detectors show a lower AP than the single-spectrum EO detector, which has a 0.774 AP. This is because the EO sensor data, the RGB photos, include wide wavelength spectrums with more information. The specific PR curves for the respective spectrum detection results are shown in Figure 36.

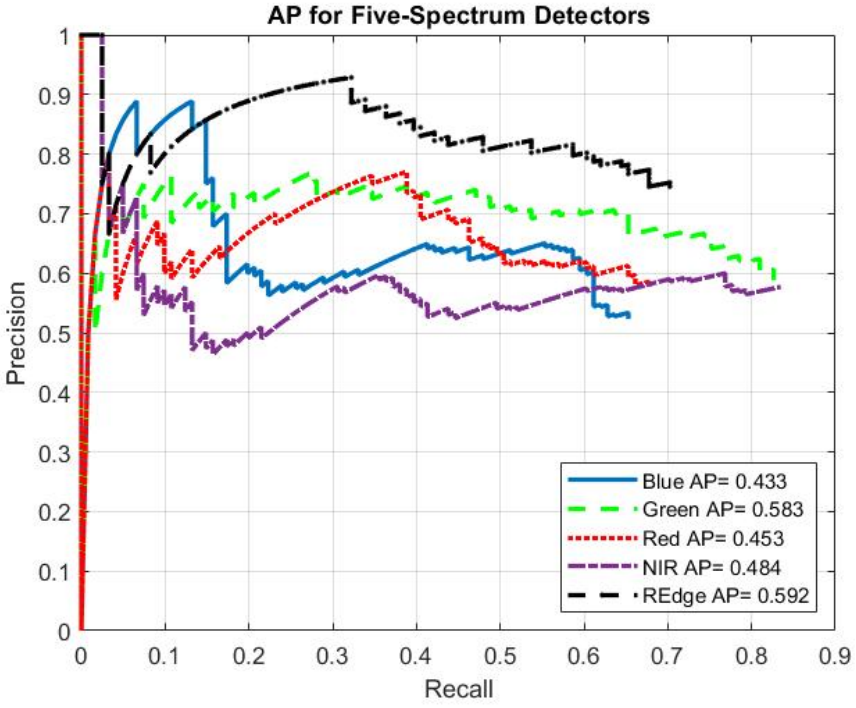


Figure 36. Precision-Recall Curve for Respective Spectrum Detectors

## D. THE TWO-STEP MULTI-SPECTRUM INTEGRATION PROCESS

The multispectral detection algorithm is based on the presumption that each spectral image from the UXO would help to exploit the feature in each spectrum. To make the most use of this advantage, the detection results from the respective UXO detectors are integrated. The integration process consists of two steps. First, the detection results of each detector are simply merged into a single space. Then, the system chooses the best detection results from the simple merging by using the NMS method. The MATLAB code for the integration process is presented in Appendix B.

### 1. Simple Merging

In MATLAB, the respective detection results are saved as a table format. In the table cells, bounding boxes are a four-element vector of the form [x y width height], and confidence scores are a vector. These values are put together as matrix values by using the loop code in MATLAB. The combined bounding boxes and scores can be plotted in any spectrum image. Figure 37 shows how the respective detection results are integrated into a single space. This study chooses the NIR picture as the integration space.

### 2. Non-maximal Suppression

NMS, a popular method in conventional object detection algorithms, selects the strongest bounding box from overlapping clusters. NMS treats the overlapped multiple bounding boxes as the same object when their IoU exceeds a certain threshold.

As discussed in relation to the MS data characteristic, the simple merged detection results from respective detectors are not aligned. Usually, most object detection methods set the threshold of the IoU in NMS as 0.3. In the multispectral object detection study at Tokyo University (2017), researchers set the threshold to 0.1 to correct the displacement. This thesis uses the same IoU threshold value of 0.1 because of the misalignment of bounding boxes from the five-spectrum detectors (Takumi et al. 2017). In other words, this way of setting a lower IoU threshold of 0.1 can prevent the actual redundant UXO detection result from being not suppressed.

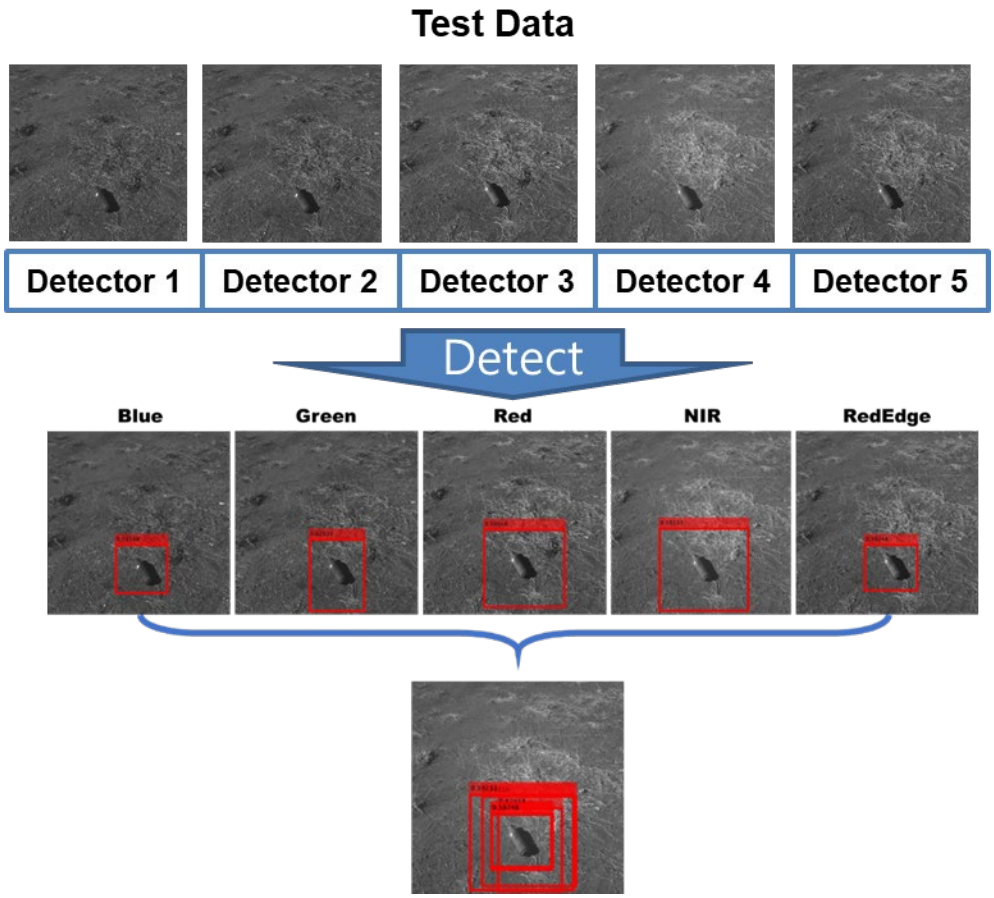


Figure 37. Simple Merging Step

By using "selectStorongestBbox" code in MATLAB, this research implements the NMS method in the simple integration of UXO detection. Figure 38 is an example of the NMS method application in UXO detections. The five candidates of UXO detection results are simply merged, and when NMS is applied, the four detection results are suppressed other than the one best detection result with the highest score of 0.67918. The NMS chooses one bounding box with the highest score and suppresses the other bounding boxes.

## E. BENEFITS OF INTEGRATED UXO DETECTION USING AN MS SENSOR

In some cases, only a few detectors can detect UXO, while the other detectors cannot. Figure 39 shows a case in which only the Blue and Green spectrum detectors

detect UXO. The Blue detector detects UXO with a score of 0.57933, the Green detector detects UXO with a score of 0.53613, and the others do not detect any UXO.

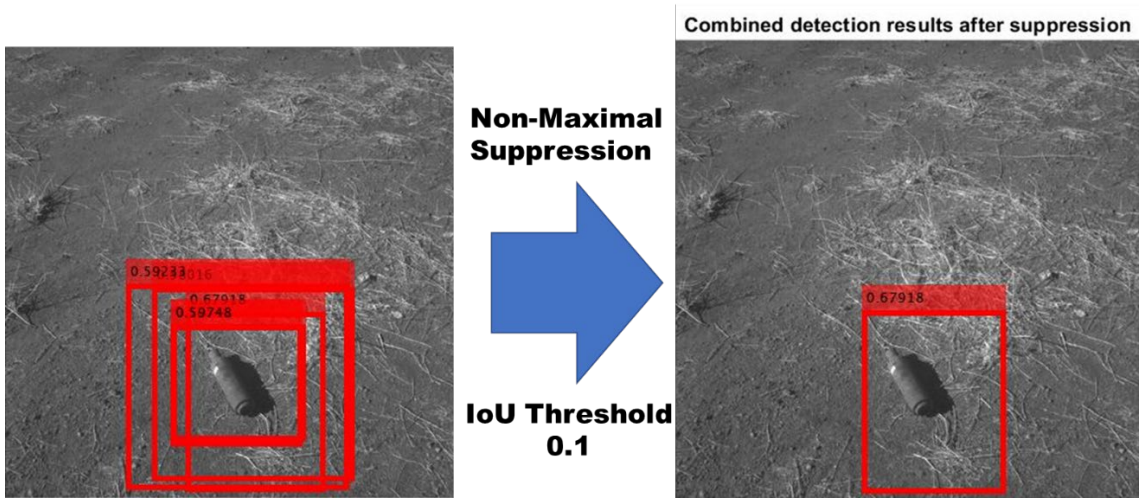


Figure 38. Final Detection Result and Application of NMS Method

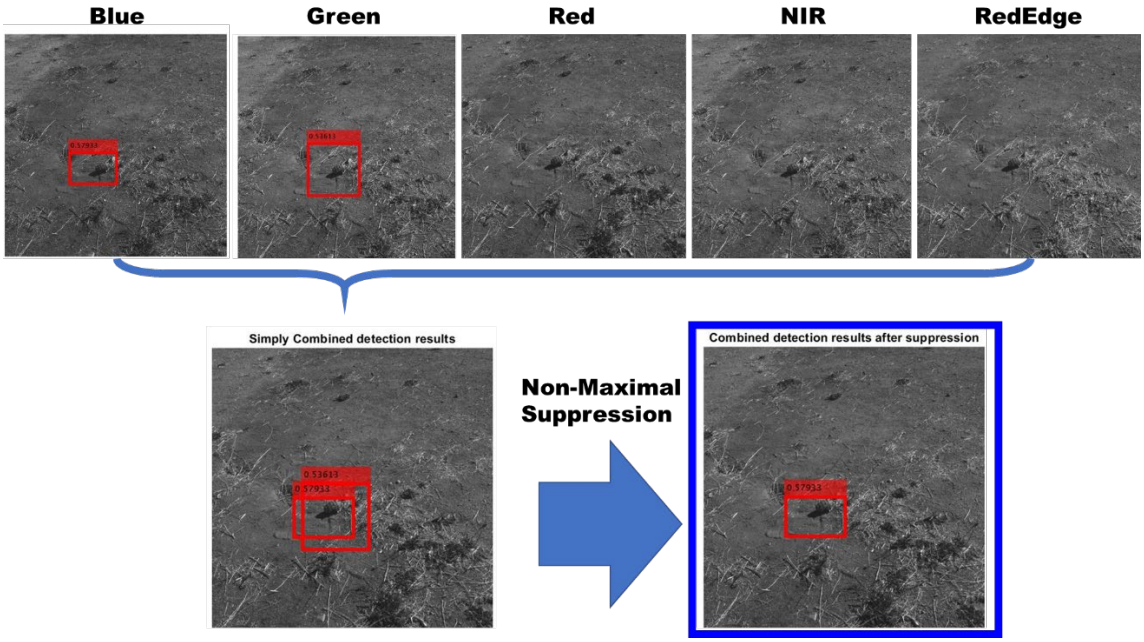


Figure 39. Example of Complementary MS Detection

With the integration of the detection results from the respective spectrum detectors, the improvement of UXO detection performance can be implied. Using this method, the MS detection algorithm can integrate a greater number of spectrums.

## F. EVALUATION ON MULTI-SPECTRUM DETECTOR

To evaluate the integrated MS detection results, this research chooses the Red spectrum test data as the evaluation criteria. Specifically, the integrated detection results are evaluated on the basis of Red spectrum data. For the evaluation, this research sets the IoU threshold as 0.4, not 0.5, for the same reason as the NMS method IoU threshold. That is, the final selected detection result might not be aligned with the Red spectrum test data, because the best detection results can be from the other spectrum detection results. To prevent the actual True Positive detection result from being evaluated as a False Positive detection because of the misalignment, an IoU threshold lower than 0.5 is required during evaluation. Finally, the evaluating value of AP for the integrated MS detection result is computed as 0.871. Compared to the 0.774 AP of the single-spectrum EO detector, the MS sensor detection has improved. Figure 40 shows the PR curve for the MS detector.

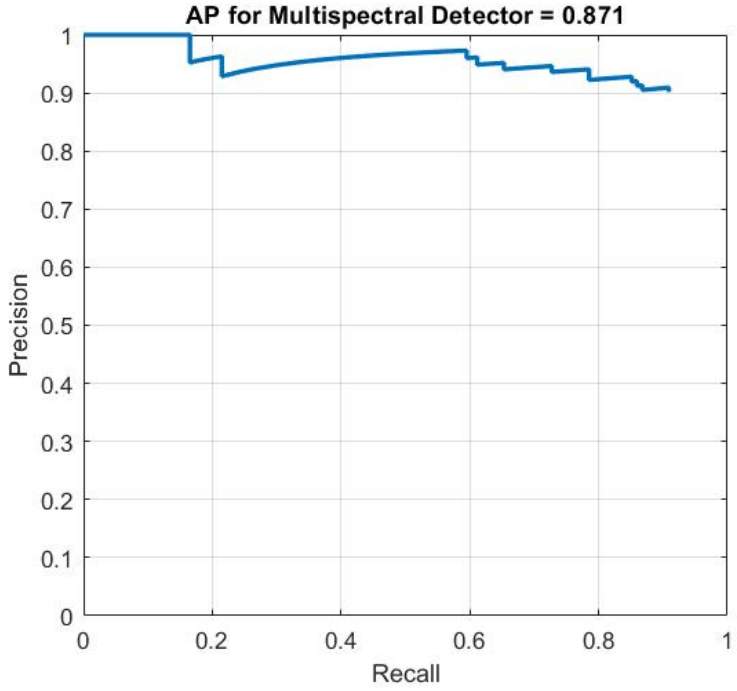


Figure 40. Precision-Recall Curve for MS Detector

THIS PAGE INTENTIONALLY LEFT BLANK

# VII. ASSESSMENT OF A SINGLE-SPECTRUM SMALL FIREARM DETECTOR IN A REAL FLIGHT TEST

This chapter implements the system integration of the trained EO sensor with the sUAS. The main purpose of the implementation is to test the feasibility of the trained detector within the operational environment—on a flying sUAS. The system integration is implemented by COTS assets, and the detection object is substituted by small firearms in the field flight test. With the integrated system, a field flight test was executed, and the detection result was assessed.

## A. INTEGRATION WITH SUAS

Although the assessments of the single-spectrum and MS sensors show them to be reasonably feasible, there is still a question of their practicality: Is the system algorithm still feasible under the actual flying circumstance? Taking footage by flying an sUAS onboard sensor probably generates a different quality of data compared to taking footage with a handheld camera because the sUAS onboard sensor in the air is affected by wind, sunlight, flying debris, and other factors. In this regard, this research found it necessary to integrate the trained EO sensor with sUAS and the field flight test. Ultimately, the purpose of the field test with the integrated system is to prove the feasibility of the trained detector within the actual operational flying environment. Additionally, this implementation explores the potential to expand the UXO detection system for other articles.

## B. SYSTEM IMPLEMENTATION

For the flight test, the integrated system of sUAS with an EO sensor is implemented using COTS.

### 1. sUAS

DJI Inspire 1 Pro, which is for professional aerial filmmaking, is used as the sUAS. Figure 41 shows the DJI Inspire 1 Pro aircraft and the remote controller, and Table 4 briefly describes the DJI Inspire 1 Pro specifications.

Figure 41. DJI Inspire 1 Pro System

Table 4. DJI Inspire 1 Pro Specifications. Source: DJI (n.d.).

| | |
|---|---|
| **Max Speed** | 18 m/s |
| **Max Wind Speed Resistance** | 10 m/s |
| **Max Flight Time** | 15 minutes |
| **Max Transmitting Distance** | 5 km (3.1 miles) |

### 2. EO Sensor

To serve as an EO sensor, a Zenmuse X5 gimbal onboard camera is integrated into the sUAS. Figure 42 shows the camera's appearance and Table 5 describes the specifications of the sensor.

Figure 42. Zenmuse X5 camera. Source: DJI (2020).

Table 5. Zenmuse X5 Specifications. Source: DJI (n.d.)

| | |
|---|---|
| **Lens** | Equivalent 30 mm |
| **Shutter Speed** | 8~1/8000 sec |
| **Video Encoder** | MPEG4/AVC/H.264 |
| **Max Video Resolution** | 4096×2160 |
| **Max-Pixels** | 16.0 M |
| **Style** | Standard Red Green Blue (sRGB) |

## C. FLIGHT TEST EXECUTION

This section describes the flight test execution of using substituted detection objects, test execution process, data process, CNN training for small firearms detection, and detection demonstration.

### 1. Detection Object

UXO samples from Camp Roberts cannot be carried out in the field test area; for that reason, this test uses four small firearms—two rifles and two pistols—as the detection objects. The upside of this substitution is that the test can explore the potential for the UXO detector algorithm to be applied for other objects. Figure 43 shows the small firearms used in this field flight test.

### 2. Test Scenario

With the described system and the four small firearms, the flight field test was executed on Calera Canyon Road, in Salinas, CA, on October 28, 2020, between 11:00 and 13:00. The operational environment was the common ground type in California, with no precipitation. It was typical weather for the fall season, with no strong wind. Figure 44 represents the area where the field flight test was executed.

Figure 43. Four Small Firearms Used in the Flight Test

Figure 44. Field Test Environment

The following describes the operational scenario for the field flight test:

- UAS Speed: 3 mph
- Flight Altitude: 10 feet
- Mission Area: 4,700 square feet (a basketball court size)
- Flight path: Serpentine route
- Detection Objects: Two pistols and two rifles
- The positions of the small firearms kept being changed arbitrarily during the sUAS flight and data collection.

The operational activity under the scenario is illustrated in Figure 45.

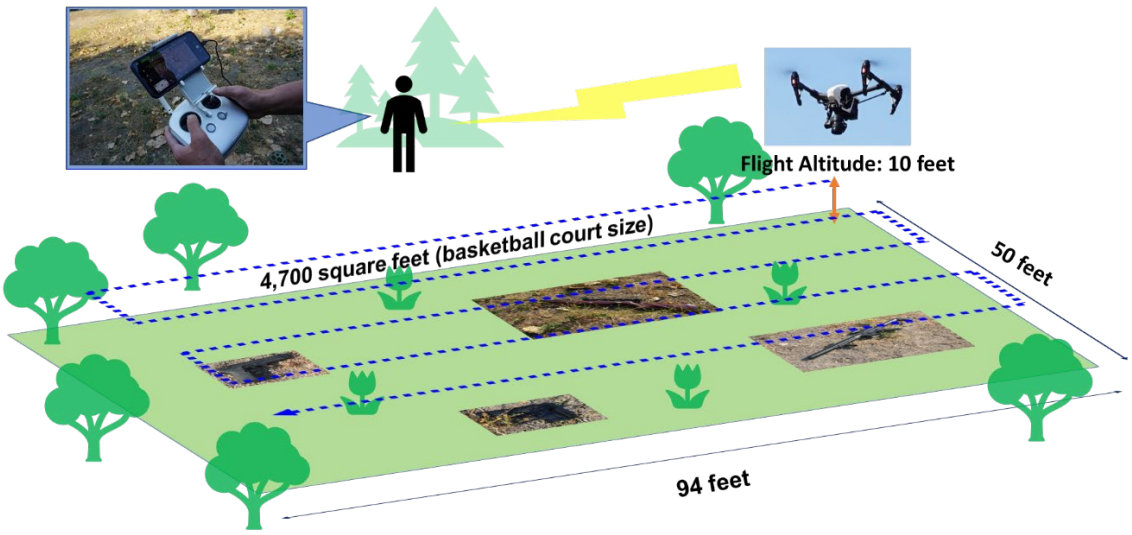


Figure 45. Flight Test Overview

As a result of the field flight, the sUAS recorded a total of 18 videos of approximately 30 seconds in length, equivalent to 2,590 photos. In all, there were 12 videos equivalent to 1,812 photos for training the detector, the three videos equivalent to 389 photos for validation while training, and the remaining three videos equivalent to 389 photos for testing and evaluation of the trained detector.

## 3. Data Processing and CNN Training for Small Firearms Detector

The workflow of the data processing and training basically follows the same procedure as the single-spectrum and MS detector training. This section, however, focuses on how this procedure differs from the previous procedure.

In consideration of the data type, the collected data were labeled by Video Labeler in MATLAB. The Video Labeler app provides automated labeling using a temporal interpolator algorithm. The four small firearms are not classified by their respective types but classified as one classification, gun. The labeling automation work in this test is presented in Figure 46.

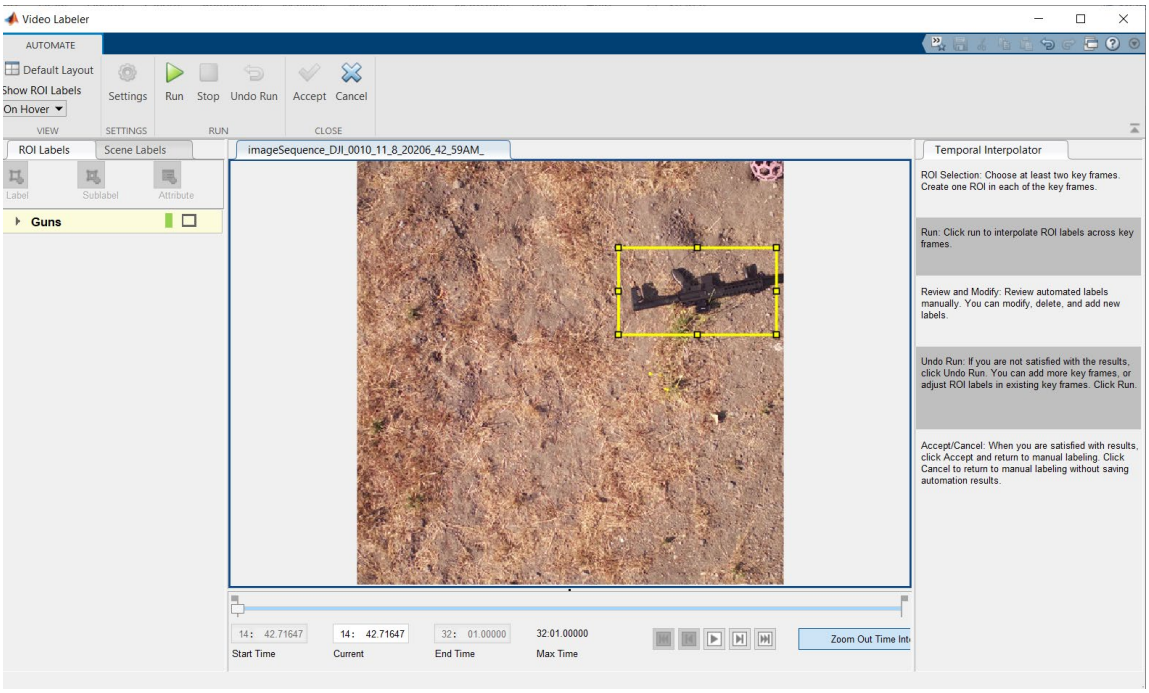


Figure 46. Video Labeling with Automation

As in the cases with the single-spectrum and MS detection assessments, anchor boxes were estimated for the small firearms detector to train the detector efficiently and achieve decent accuracy. Figure 47 shows the relation of the mean IoU and the number of anchors in small firearms detection. Similar to the previous UXO detectors, the value of IoU increases steeply for the first five anchor boxes. This research chooses eight anchor

boxes and sets the mean IoU value at 0.7998 in consideration of the different sizes of rifles and pistols.

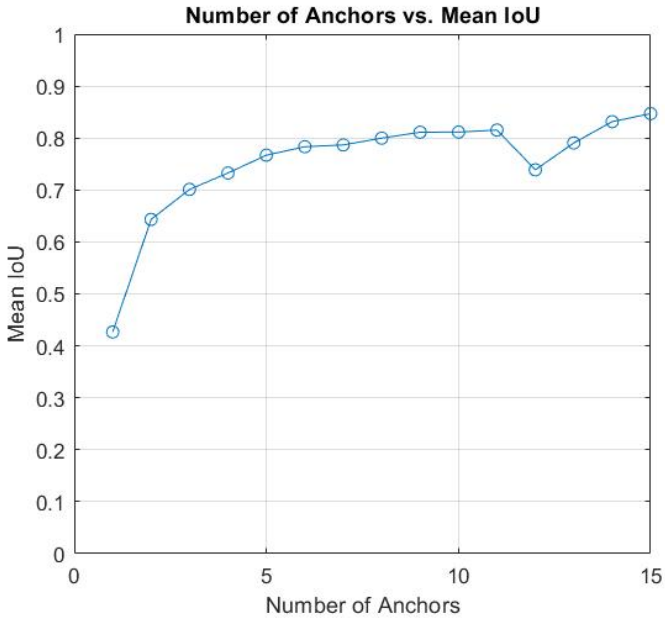


Figure 47. Anchor Boxes Estimation for Gun Detector

The network training options are similar to those for the single-spectrum and MS detector training. To decrease the training time, this research chooses a mini-batch size of eight, and sets the maximum number of epochs as five. The 12 videos equivalent to 1,812 photos were used for training, and the three videos equivalent to 389 photos were used for validation while training. The training took an hour and ten minutes. As shown in Figure 48, the training loss for each iteration was nearly zero for other than the first few iterations. This represents that the small firearms detector training with the processed data had no issue while training.

### 4. Detection Demonstration

For the small firearms detection demonstration, this research used the three videos for testing. The trained detector read the three test videos and detected suspected guns. In all three videos, the detector detected the four small firearms successfully as shown in Figure 49.

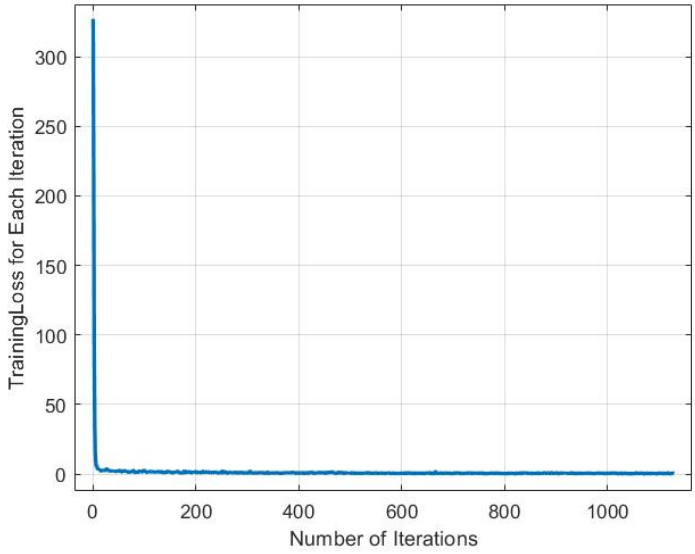


Figure 48. Training Loss for Gun Detector Training

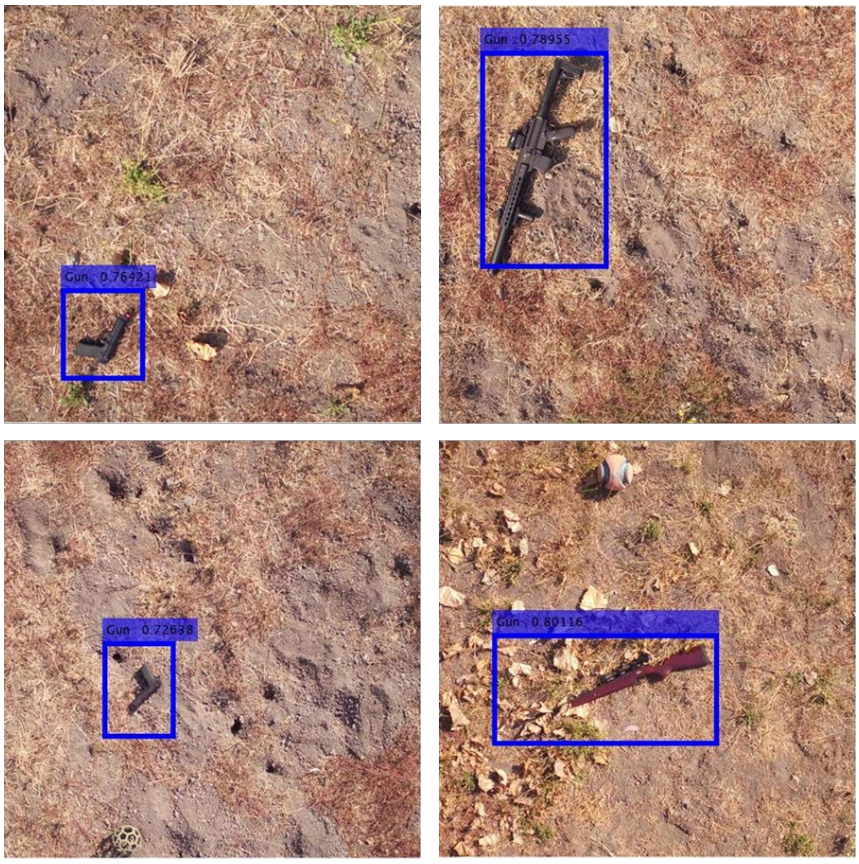


Figure 49. Demonstration of Small Firearms Detection

## D. EVALUATION ON SMALL FIREARMS DETECTION

To evaluate the small firearms detector, the three videos for testing were converted into 389 pictures with labeling. Then, the trained detector conducted detection over the test pictures. This research evaluates the detection results with the same evaluation metric of AP, and the AP was computed as 0.966. The trained detector detects the small firearms in the test data quite precisely. This is fairly high performance compared to UXO detection. The PR curve of the detection result is plotted in Figure 50.

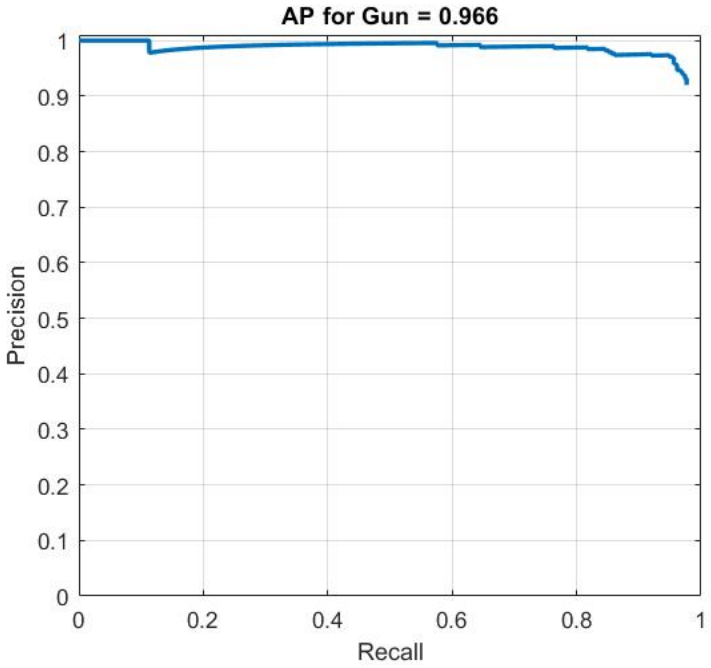


Figure 50. Precision-Recall Curve for Gun Detector

THIS PAGE INTENTIONALLY LEFT BLANK

# VIII. CONCLUSIONS AND RECOMMENDATIONS

This chapter summarizes the findings of the research presented and proposes recommendations for future work that would advance the proposed UXO detection system.

## A. SUMMARY OF RESEARCH FINDINGS

This thesis aimed at answering two research questions:

- Is it feasible for the AI-trained sUAS including a single-spectrum or a multi-spectrum EO sensor to detect UXO on the earth's surface effectively?
- Is the aforementioned UXO detection system feasible in terms of practicality within the flying operational environment?

The research described in this thesis answered these questions as follows:

- For a single-spectrum EO sensor, the most popular onboard sensor for sUAS, the trained detector was able to detect UXO successfully in both types of data: pictures and videos. The evaluation of the trained detector showed 0.774 AP against the test data, and this is quite a good performance for the object detection area. Because the training data was not sufficient, the AP of the detector has the potential to improve with more data.
- As anticipated, the respective spectrum UXO images for a five-spectrum sensor have different characteristics, and so the detection results from the respective detectors were different. It was found, however, that the different spectrum detection results were complementary. By a two-step integration process, the detection results can be effectively integrated, ensuring much better detection results. It was also found that while the respective spectrum detectors had only about 0.5 AP, the integrated MS

detection featured near 0.9 AP. Compared with the single-spectrum EO detector AP value of 0.774 AP, the multispectral detector AP value of 0.871 shows about a 0.13 improvement under the same conditions.

- While UXOs do not necessarily feature a distinctive geometry, other objects that the sUAS system maybe looking for could. Based on the projected operational scenario, the sUAS equipped with the EO sensor conducted a representative field operation of collecting (lost/misplaced) small firearms data. Using exactly the same process of CNN training as before, the single-spectrum trained detector demonstrated an even higher result of 0.966 AP UXO (about 0.25 improvement over UXOs under the same condition). The result also ensures that the proposed approach based on using AI systems onboard sUAS can be expanded to detect objects other than UXOs.

## B. RECOMMENDATIONS FOR FUTURE WORK

Future research related to this thesis may be extended in the following manner:

- UXO detection system demonstration using actual sUAS: In Chapter VII, this thesis demonstrates the UXO detection system using the sUAS, where small firearms substituted for UXO for security reasons. While demonstrating the detection system, this research found that the collected pictures and videos differ from the pictures and videos taken by a handheld camera due to flying debris, the shadow of the sUAS, and the angle of the camera. In spite of that the field flight test for small firearms detection proved the system's feasibility; hence, it is worth implementing an actual UXO detection system with flying sUAS to finalize the system feasibility analysis.

- A real-time system implementation: YOLOv2 is basically designed for a real-time object detection system (Redmon and Farhadi 2017). MATLAB also provides enabler tools such as a USB webcam support package. With

a few pieces of equipment, such as a USB capture board and HDMI transmitter/receiver, it is feasible to implement a real-time UXO detection system, which would enable UXO to be detected while the sUAS is flying.

- UXO nighttime detection test using different sUAS onboard sensors such as thermal imagery from an Electro-Optical/Infra-Red (EO/IR) sensor: This thesis executed the experiments and the demonstration during the daytime in order to use the vision sensor and the multispectral sensor. Yet, several diverse EO/IR sensors that provide thermal imagery are also popular onboard sensors for sUAS. To expand the UXO detection system's ability, future researchers should study thermal-sensor UXO detection during nighttime.
- Optimization of flight altitude for the UXO detecting sUAS: This thesis assumed a low-altitude flight due to experimental limits. If the sUAS flies higher, it can cover a bigger mission area quickly. On the other hand, it is hard to capture UXO of smaller sizes this way. The size of UXO varies, but it can be presumed that there is a certain interval of the size average for most ammunitions. This thesis recommends, therefore, further studies for the optimal flight altitude by comparing detection results with varied flight altitude.

THIS PAGE INTENTIONALLY LEFT BLANK

# APPENDIX A. MATLAB CODE FOR UXO DETECTOR TRAINING / EVALUATION

```
% This code is adapted from MATLAB 2020b example of
<trainYOLOv2ObjectDetector>

%% Prepare data with Groundtruth labeling

% addpath('TrainingData');
% trainingData =
objectDetectorTrainingData(gTruth,'WriteLocation','Trai
ningData');
% imageLabeler('UXOPSession.mat')

load UXOPgTruth.mat


%% Create table data format from ground truth data
trainingDataset=gTruth.LabelData;
trainingDataset.files=gTruth.DataSource.Source;
trainingDataset=trainingDataset(:,[2,1]);

%% Visualize the training images

index=500;
%colorized
Iout=imread(trainingDataset.files{index});
for k=1:width(trainingDataset)-1
 bboxes = table2array(trainingDataset(index,k+1));
 if ~isempty(bboxes{1})

Iout=insertObjectAnnotation(Iout,'rectangle',bboxes{1},
...

trainingDataset.Properties.VariableNames{k+1},'Color','
blue','fontsize',12,'linewidth',5);
 end
end
figure, imshow(Iout);

%% Divide training data into Train/Validation/Test set
% set default for reproducible
```

```
rng(1004); % by randomness

% Divide the dataset into training set and test set
shuffledIndices=randperm(height(trainingDataset));
idx=floor(0.7*length(shuffledIndices));
trainingData=trainingDataset(shuffledIndices(1:idx),:);

validationIdx=idx+1:idx+1+floor(0.15*length(shuffledInd
ices));
validationData=trainingDataset(shuffledIndices(validati
onIdx),:);

testIdx=validationIdx(end)+1:length(shuffledIndices);
testData=trainingDataset(shuffledIndices(testIdx),:);


%% use imageDatastore and boxLabelDatastore to create
datastores --later
imdsTrain=imageDatastore(trainingData{:,'files'});
bldsTrain=boxLabelDatastore(trainingData(:,2:end));

imdsValidation=imageDatastore(validationData{:,'files'}
);
bldsValidation=boxLabelDatastore(validationData(:,2:end
));

imdsTest=imageDatastore(testData{:,'files'});
bldsTest=boxLabelDatastore(testData(:,2:end));

% Combine image and box label datastores
trainingData=combine(imdsTrain,bldsTrain);
validationData=combine(imdsValidation,bldsValidation);
testData=combine(imdsTest,bldsTest);

%% Get Ready for the training - Define backbone network
basenetwork=resnet50();
numClasses=width(trainingDataset)-1;
inputSize=[416 416 3];

%% Anchor Box Estimation - Visualize the Number of
Anchor Boxes versus the Mean IoU
```

```
anchorTraining =
boxLabelDatastore(trainingDataset(:,2:end));
maxNumAnchors=15;
meanIoU=zeros([maxNumAnchors,1]);
anchorBoxes=cell(maxNumAnchors,1);

doTrain= true;
if doTrain
 for k = 1:maxNumAnchors

[anchorBoxes{k},meanIoU(k)]=estimateAnchorBoxes(anchorT
raining,k);
 end
 save AnchorUXO anchorBoxes meanIoU
else
 load AnchorUXO.mat
end
figure
plot(1:maxNumAnchors,meanIoU,'-o')
ylabel("Mean IoU")
xlabel("Number of Anchors")
title("Number of Anchors vs. Mean IoU")
grid on

%% Choose the number of anchor box
numAnchors=9; % from the anchor box estimation
allBoxes=round(cell2mat(reshape(table2array(trainingDat
aset(:,2:end)),[],1)));
scale=inputSize(1:2)./size(imread(trainingDataset.files
{1}),[1 2]);
anchorBoxes = round(anchorBoxes{numAnchors}.*scale)
meanIoU_chosen = meanIoU(numAnchors)

%% Choose Network to Train
nettotrain = 1;
% 1 = YOLOv2, 2 = SSD, 3 = FasterRCNN

%% Create Network (YOLOv2 or SSD or FasterRCNN)
if nettotrain ==1
 featureLayer='activation_40_relu';

lgraph=yolov2Layers(inputSize,numClasses,anchorBoxes,ba
senetwork,featureLayer);
```

```
elseif nettotrain==2
 lgraph=ssdLayers(inputSize, numClasses, 'resnet50');
else
 lgraph=fasterRCNNLayers(inputSize, numClasses,
anchorBoxes, 'resnet50');
end

%% Data Augementation --- function later
augmentedTrainingData=transform(trainingData,@augmentDa
ta);

augmentedData = cell(4,1);
for k = 1:4
 data=read(augmentedTrainingData);

augmentedData{k}=insertShape(data{1},'Rectangle',data{2
},'linewidth',5);
 reset(augmentedTrainingData);
end
figure
montage(augmentedData,'BorderSize',10,'Size',[2 2])

%% Preprocess Training Data --- later
preprocessedTrainingData=transform(augmentedTrainingDat
a,@(data)preprocessData(data,inputSize,nettotrain));
preprocessedValidationData=transform(validationData,@(d
ata)preprocessData(data,inputSize,nettotrain));
%% Configure the network training options
options = trainingOptions('adam',...
 'InitialLearnRate',0.001,...
 'Verbose',true,...
 'MiniBatchSize',8,... % Change!
 'MaxEpochs',20,... % Change!
 'Shuffle','never',...
 'VerboseFrequency',10,...
 'ValidationData',preprocessedValidationData);

%% Train the network

doTrain=true; %false
if doTrain
 if nettotrain == 1
```

```
[detector,info]=trainYOLOv2ObjectDetector(preprocessedT
rainingData,lgraph,options); %preprocessed trainngData
later
 save UXO_yolo_detector
 elseif nettotrain ==2

[detector,info]=trainSSDObjectDetector(preprocessedTrai
ningData,lgraph,options);
 save UXO_SSD_detector
 else
 [detector,info] =
trainFasterRCNNObjectDetector(preprocessedTrainingData,
lgraph, options);
 save UXO_FasterRCNN_detector
 end
 detector
 figure
 plot(info.TrainingLoss)
 grid on
 xlabel('Number of Iterations')
 ylabel('TrainingLoss for Each Iteration')
else
 load UXO_yolo_detector.mat
end
%% Read a test image into the workspace
I=read(testData);
I=imresize(I{1},1);
[bboxes,scores,labels]=detect(detector,I,'threshold',0.
5);

%% Display the results.
[~,ind]=ismember(labels,detector.ClassNames);
if(~isempty(bboxes))
 I = insertObjectAnnotation(I,'rectangle', bboxes,
strcat(string(labels)', " : ",...
 string(scores)), "LineWidth," 5, "fontsize," 12,
'Color','blue');
 end
figure
clf
imshow(I)
```

```
%% Evaluate Detector Using Test Set
preprocessedTestData =
transform(testData,@(data)preprocessData(data,inputSize
,nettotrain));

% Run the detector on all the test images
detectionResults = detect(detector,
preprocessedTestData, 'threshold',
0.5,'ExecutionEnvironment','cpu'); % gpu does not work
in my environment

% Evaluate the object detector using average precision
metric.
[ap, recall, precision] =
evaluateDetectionPrecision(detectionResults,
preprocessedTestData, 0.5);


% The precision/recall (PR) curve highlights how
precise a detector is at varying levels of recall. The
ideal precision is 1 at all recall levels.
% The use of more data can help improve the average
precision but might require more training time. Plot
the PR curve.
figure
plot(recall, precision)
xlabel('Recall')
ylabel('Precision')
grid on
temp = sprintf(' = %.3f', ap);
title(sprintf(['AP for ',
trainingDataset.Properties.VariableNames{2}, temp]))


%% my functions

function data =
preprocessData(data,targetSize,nettotrain)
% Resize image and bounding boxes to the targetSize.
if nettotrain==2
 scale=targetSize(1:2)./size(data{1},[1 2]);
 data{1}=imresize(data{1},targetSize(1:2));
 data{2}=bboxresize(data{2},scale);
```

```
else
 data{1}=imresize(data{1},1);
 data{2}=bboxresize(data{2},1);
end
end

function B = augmentData(A)
B=cell(size(A));

I=A{1};
sz=size(I);
if numel(sz)==3 && sz(3) ==3
 I=jitterColorHSV(I,...
 'Contrast',0.2,...
 'Hue',0,...
 'Saturation',0.1,...
 'Brightness',0.2);
end

tform = randomAffine2d('XReflection',true,'Scale',[1
1.2], 'Rotation',[-20 20]);
rout=affineOutputView(sz,tform,'BoundsStyle','CenterOut
put');
B{1}=imwarp(I,tform,'OutputView',rout);

[B{2},indices]=bboxwarp(A{2},tform,rout,'OverlapThresho
ld',0.25);
B{3}=A{3}(indices);

if isempty(indices)
 B=A;
end
end
```

THIS PAGE INTENTIONALLY LEFT BLANK

# APPENDIX B. MATLAB CODE FOR INTEGRATION OF MULTISPECTRAL DETECTION / EVALUATION

```
%% Load Test image sets

load Data_Blue_Test.mat
load Data_Green_Test.mat
load Data_Red_Test.mat
load Data_NIR_Test.mat
load Data_REdge_Test.mat

testData1=Data_Blue_Test.LabelData;
testData1.files=Data_Blue_Test.DataSource.Source;
testData1=testData1(:,[2,1]);

testData2=Data_Green_Test.LabelData;
testData2.files=Data_Green_Test.DataSource.Source;
testData2=testData2(:,[2,1]);

testData3=Data_Red_Test.LabelData;
testData3.files=Data_Red_Test.DataSource.Source;
testData3=testData3(:,[2,1]);

testData4=Data_NIR_Test.LabelData;
testData4.files=Data_NIR_Test.DataSource.Source;
testData4=testData4(:,[2,1]);

testData5=Data_REdge_Test.LabelData;
testData5.files=Data_REdge_Test.DataSource.Source;
testData5=testData5(:,[2,1]);

imdsTest=imageDatastore(testData1{:,'files'});
bldsTest=boxLabelDatastore(testData1(:,2:end));
testData1=combine(imdsTest,bldsTest);

imdsTest=imageDatastore(testData2{:,'files'});
bldsTest=boxLabelDatastore(testData2(:,2:end));
testData2=combine(imdsTest,bldsTest);

imdsTest=imageDatastore(testData3{:,'files'});
bldsTest=boxLabelDatastore(testData3(:,2:end));
testData3=combine(imdsTest,bldsTest);
```

```
imdsTest=imageDatastore(testData4{:,'files'});
bldsTest=boxLabelDatastore(testData4(:,2:end));
testData4=combine(imdsTest,bldsTest);

imdsTest=imageDatastore(testData5{:,'files'});
bldsTest=boxLabelDatastore(testData5(:,2:end));
testData5=combine(imdsTest,bldsTest);


%% Load trained detectors

load Blue_Detector.mat
load Green_Detector.mat
load Red_Detector.mat
load NIR_Detector.mat
load REdge_Detector.mat

%% Detection Results gathering

% Run the detectors for each test image data
detectionResults_1 = detect(detector1, testData1,
'threshold', 0.5,'ExecutionEnvironment','cpu'); % gpu
does not work in my environment
detectionResults_2 = detect(detector2, testData2,
'threshold', 0.5,'ExecutionEnvironment','cpu'); % gpu
does not work in my environment
detectionResults_3 = detect(detector3, testData3,
'threshold', 0.5,'ExecutionEnvironment','cpu'); % gpu
does not work in my environment
detectionResults_4 = detect(detector4, testData4,
'threshold', 0.5,'ExecutionEnvironment','cpu'); % gpu
does not work in my environment
detectionResults_5 = detect(detector5, testData5,
'threshold', 0.5,'ExecutionEnvironment','cpu'); % gpu
does not work in my environment

save(detectionResults_1.mat,detectionResults_1);
save(detectionResults_2.mat,detectionResults_2);
save(detectionResults_3.mat,detectionResults_3);
save(detectionResults_4.mat,detectionResults_4);
save(detectionResults_5.mat,detectionResults_5);
```

```
%% Load Saved results
load detectionResults_1.mat
load detectionResults_2.mat
load detectionResults_3.mat
load detectionResults_4.mat
load detectionResults_5.mat

%% Combine data and make them ready to be used
% convert table to cell
detectionResults_1=table2cell(detectionResults_1);
detectionResults_2=table2cell(detectionResults_2);
detectionResults_3=table2cell(detectionResults_3);
detectionResults_4=table2cell(detectionResults_4);
detectionResults_5=table2cell(detectionResults_5);

% Combine detection results of bboxes
for i=1:length(detectionResults_1)

CombinedBbox{i,1}=vertcat(detectionResults_1(i,1),detec
tionResults_2(i,1),...

detectionResults_3(i,1),detectionResults_4(i,1),detecti
onResults_5(i,1));
end

for j=1:length(CombinedBbox) % Remove empty cell
 RemoveCell=find(cellfun('isempty',CombinedBbox{j,1}));
 CombinedBbox{j,1}(RemoveCell,:)=[];
end


for k=1:length(CombinedBbox) % convert cell to matrix
 CombinedBbox{k,1}=cell2mat(CombinedBbox{k,1});
end


% Combine detection results of scores
for i=1:length(detectionResults_1)

CombinedScores{i,1}=vertcat(detectionResults_1(i,2),det
ectionResults_2(i,2),...
```

```
detectionResults_3(i,2),detectionResults_4(i,2),detecti
onResults_5(i,2));

end

for j=1:length(CombinedScores)

RemoveCell=find(cellfun('isempty',CombinedScores{j,1}))
;
 CombinedScores{j,1}(RemoveCell,:)=[];
end

for k=1:length(CombinedScores)
 CombinedScores{k,1}=cell2mat(CombinedScores{k,1});
end

save CombinedBbox
save CombinedScores

%% Select strongest bounding boxes from overlapping
clusters
% Make use of the selectedStrongedBbox function based
on Non-maximal suppression
FinalBbox=num2cell(zeros(length(CombinedBbox),1));
FinalScores=num2cell(zeros(length(CombinedScores),1));



for i=1:length(CombinedBbox)
[selectedBbox,selectedScores]=selectStrongestBbox(Combi
nedBbox{i,1},CombinedScores{i,1},...
 'OverlapThreshold',0.1); %Overlapthreshold 0.1 because
of 5 spectrum pictures alignment are all different
FinalBbox{i,1}=selectedBbox;
FinalScores{i,1}=selectedScores;
end

save FinalBbox
save FinalScores

%%
load FinalBbox.mat
```

```
load FinalScores.mat
load CombinedBbox.mat
load CombinedScores.mat

% To show how they work

% convert data to table
testData1=Data_Blue_Test.LabelData;
testData1.files=Data_Blue_Test.DataSource.Source;
testData1=testData1(:,[2,1]);

testData2=Data_Green_Test.LabelData;
testData2.files=Data_Green_Test.DataSource.Source;
testData2=testData2(:,[2,1]);

testData3=Data_Red_Test.LabelData;
testData3.files=Data_Red_Test.DataSource.Source;
testData3=testData3(:,[2,1]);

testData4=Data_NIR_Test.LabelData;
testData4.files=Data_NIR_Test.DataSource.Source;
testData4=testData4(:,[2,1]);

testData5=Data_REdge_Test.LabelData;
testData5.files=Data_REdge_Test.DataSource.Source;
testData5=testData5(:,[2,1]);

% Pick any data
index=102;
I=imread(testData2.files{index});
I1 =
insertObjectAnnotation(I,'rectangle',detectionResults_2
.Boxes{index},detectionResults_2.Scores{index},'Color',
'r','fontsize',12,'linewidth',5);
I2 =
insertObjectAnnotation(I,'rectangle',CombinedBbox{index
,1},CombinedScores{index,1},'Color','r','fontsize',12,'
linewidth',5);

figure, imshow(I1); title('Combined detection results
after suppression')
figure, imshow(I2); title('Simply Combined detection
results')
```

```
I1 =
insertObjectAnnotation(I,'rectangle',FinalBbox{index,1}
,FinalScores{index,1},'Color','r','fontsize',12,'linewi
dth',5);
I2 =
insertObjectAnnotation(I,'rectangle',CombinedBbox{index
,1},CombinedScores{index,1},'Color','r','fontsize',12,'
linewidth',5);

figure, imshow(I1); title('Combined detection results
after suppression')
figure, imshow(I2); title('Simply Combined detection
results')


%% Evaluate the combined object detector using average
precision metric.

FinalResults=table(FinalBbox,FinalScores); % Convert
results to table
save FinalResults

load FinalResults.mat

% Convert data to datastore form
imdsTest=imageDatastore(testData1{:,'files'});
bldsTest=boxLabelDatastore(testData1(:,2:end));
testData1=combine(imdsTest,bldsTest);

imdsTest=imageDatastore(testData2{:,'files'});
bldsTest=boxLabelDatastore(testData2(:,2:end));
testData2=combine(imdsTest,bldsTest);

imdsTest=imageDatastore(testData3{:,'files'});
bldsTest=boxLabelDatastore(testData3(:,2:end));
testData3=combine(imdsTest,bldsTest);

imdsTest=imageDatastore(testData4{:,'files'});
bldsTest=boxLabelDatastore(testData4(:,2:end));
testData4=combine(imdsTest,bldsTest);
```

```
imdsTest=imageDatastore(testData5{:,'files'});
bldsTest=boxLabelDatastore(testData5(:,2:end));
testData5=combine(imdsTest,bldsTest);

detectionResults_1=cell2table(detectionResults_1);
detectionResults_2=cell2table(detectionResults_2);
detectionResults_3=cell2table(detectionResults_3);
detectionResults_4=cell2table(detectionResults_4);
detectionResults_5=cell2table(detectionResults_5);

[ap1, recall1,
precision1]=evaluateDetectionPrecision(detectionResults
_1, testData1, 0.5);
[ap2, recall2,
precision2]=evaluateDetectionPrecision(detectionResults
_2, testData2, 0.5);
[ap3, recall3,
precision3]=evaluateDetectionPrecision(detectionResults
_3, testData3, 0.5);
[ap4, recall4,
precision4]=evaluateDetectionPrecision(detectionResults
_4, testData4, 0.5);
[ap5, recall5,
precision5]=evaluateDetectionPrecision(detectionResults
_5, testData5, 0.5);


[ap, recall, precision] =
evaluateDetectionPrecision(FinalResults, testData3,
0.4); %Threshold is 0.4 because of multispectral
image's alignment

% The precision/recall (PR) curve highlights how
precise a detector is at varying levels of recall. The
ideal precision is 1 at all recall levels.
% The use of more data can help improve the average
precision but might require more training time. Plot
the PR curve.

% AP for each detector
figure
```

```
plot(recall1, precision1,recall2,precision2,'g--
',recall3,precision3,'r:',recall4,precision4,'-
.',recall5,precision5,'k--.','LineWidth',2)
xlabel('Recall')
ylabel('Precision')
grid on

temp1 = sprintf('Blue AP= %.3f', ap1);
temp2 = sprintf('Green AP= %.3f', ap2);
temp3 = sprintf('Red AP= %.3f', ap3);
temp4 = sprintf('NIR AP= %.3f', ap4);
temp5 = sprintf('REdge AP= %.3f', ap5);
legend(temp1,temp2,temp3,temp4,temp5,'Location','southe
ast')

title('AP for Five-Spectrum Detectors')

% AP for Multispectral detector
figure
plot(recall, precision)
xlabel('Recall')
ylabel('Precision')
grid on
temp = sprintf(' = %.3f', ap);
title(sprintf(['AP for Multispectral Detector', temp]))
```

# LIST OF REFERENCES


Bertrand, Harold E., David C. Heberlein, John T. Frasier, and Sherryl S. Zounes, et al. 2004. *Report of UXO Technology Subgroup: Overview and Technology Assessment.* IDA Document D-3007. Alexandria, VA: Institute for Defense Analyses.

Bodla, Navaneeth, Bharat Singh, Rama Chellappa, and Larry S. Davis. 2017. "Soft-NMS —Improving Object Detection with One Line of Code." *ArXiv:1704.04503 [cs.CV]*. http://arxiv.org/abs/1704.04503.

Brownlee, Jason. 2019a. "A Gentle Introduction to Computer Vision." *Machine Learning Mastery* (blog), March 18, 2019. https://machinelearningmastery.com/what-is-computer-vision/.

———. 2019b. "A Gentle Introduction to Object Recognition With Deep Learning." *Machine Learning Mastery* (blog), May 21, 2019. https://machinelearningmastery.com/object-recognition-with-deep-learning/.

DeSmet, Timothy, Alex Nikulin, William Frazer, Jasper Baur, Jacob Abramowitz, Daniel Finan, Sean Denara, Nicholas Aglietti, and Gabriel Campos. 2018. "Drones and 'Butterflies': A Low-Cost UAV System for Rapid Detection and Identification of Unconventional Minefields." *Journal of Conventional Weapons Destruction* 22 (3) (Article 10): 1–9. https://commons.lib.jmu.edu/cisr-journal/vol22/iss3/10.

DJI. n.d. "DJI - Official Website." Accessed January 18, 2021. https://www.dji.com/.

———. n.d. "Inspire 1 Pro/Raw." Digital image. Accessed December 29, 2020. https://www.dji.com.

Dorafshan, Sattar, Robert J. Thomas, Calvin Coopmans, and Marc Maguire. 2018. "Deep Learning Neural Networks for SUAS-Assisted Structural Inspections: Feasibility and Application." In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, 874–82. Dallas, TX: IEEE. https://doi.org/10.1109/ICUAS.2018.8453409.

Etter, Delores, and Bill Delaney. 2003. *Report of the Defense Science Board Task Force on Unexploded Ordnance*. Defense Science Board Task Force. Washington, D.C.: Office of the Under Secretary of Defense (Acquisition, Technology & Logistics). https://dsb.cto.mil/reports/2000s/ADA419970.pdf.

Everingham, Mark, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. 2010. "The Pascal Visual Object Classes (VOC) Challenge." *International Journal of Computer Vision* 88 (2): 303–38. https://doi.org/10.1007/s11263-009-0275-4.

Garg, Priya, Debapriyo Roy Chowdhury, and Vidya N. More. 2019. “Traffic Sign Recognition and Classification Using YOLOv2, Faster RCNN and SSD.” In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1–5. Kanpur, India: IEEE. https://doi.org/10.1109/ICCCNT45670.2019.8944491.

Girshick, Ross. 2015. “Fast R-CNN.” In *2015 IEEE International Conference on Computer Vision (ICCV)*, 1440–48. Santiago, Chile: IEEE. https://doi.org/10.1109/ICCV.2015.169.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. “Deep Residual Learning for Image Recognition.” *ArXiv:1512.03385 [cs.CV]*. http://arxiv.org/abs/1512.03385.

Ho, Daniel, Eric Liang, Ion Stoica, Pieter Abbeel, and Xi Chen. 2019. “Population Based Augmentation: Efficient Learning of Augmentation Policy Schedules.” *ArXiv:1905.05393 [cs.CV].* http://arxiv.org/abs/1905.05393.

HP. n.d. “HP® Pavilion Laptops.” Accessed December 15, 2020. https://store.hp.com/us/en/mlp/laptops/pavilion-344505--1.

Johnson, Bernadette, Charles A. Primmerman, Sepil Ayasli, Dennis J. Blejer, Check F. Lee, Thomas G. Moore, and Thomas P. Opar. 1996. *A Research and Development Strategy for Unexploded Ordnance Sensing.* ESC-TR-96-027. Lexington, MA: Lincoln Laboratory, MIT.

Judson, Jen. 2019. “US Army Takes Another Stab at ‘Rucksack-Portable’ Drone.” *Defense News*, April 30, 2019. https://www.defensenews.com/land/2019/04/29/army-takes-another-stab-at-rucksack-portable-unmanned-aircraft/.

Kingma, Diederik P., and Jimmy Ba. 2017. “Adam: A Method for Stochastic Optimization.” *ArXiv:1412.6980 [cs.CV]*. http://arxiv.org/abs/1412.6980.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. “Deep Learning.” *Nature* 521 (7553): 436–44. https://doi.org/10.1038/nature14539.

Lee, Sukjong. 2018. “South Korea and the U.S. Support the Safety of Local Residents.” *Korea Defense Daily News,* March 29, 2018. http://kookbang.dema.mil.kr/newsWeb/20180329/2/BBSMSTR_000000010023/view.do.

Lee, Wee Leong. 2018. “Assessment of Foreign Object Debris Management using Group 1 Unmanned Aerial Systems.” Master’s thesis, Naval Postgraduate School. https://calhoun.nps.edu/handle/10945/60426.

Leonel, Rubio. 2020. “Infantry Soldiers Test Short-Range Reconnaissance Unmanned Aircraft amid COVID-19.” U.S. Army. August 19, 2020. https://www.army.mil/article/238277/infantry_soldiers_test_short_range_reconnaissance_unmanned_aircraft_amid_covid_19.

Liu, Hongyu, and Bo Lang. 2019. “Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey.” *Applied Sciences* 9 (20) (October): 4396. https://doi.org/10.3390/app9204396.

Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. 2016. “SSD: Single Shot MultiBox Detector.” In *Computer Vision – ECCV 2016*, edited by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, 9905: 21–37. Lecture Notes in Computer Science. Cham, Switzerland: Springer International Publishing. https://doi.org/10.1007/978-3-319-46448-0_2.

Lukunka, Sharon. 2014. “Averting the Risks of Unexploded Ordnance.” UNAMID. Last modified April 4, 2014. https://unamid.unmissions.org/averting-risks-unexploded-ordnance.

Martin, Michael F., Ben Dolven, Andrew Feickert, and Thomas Lum. 2019. *War Legacy Issues in Southeast Asia: Unexploded Ordnance (UXO).* Report No. R45749. Washington, DC: U.S. Congressional Research Service. https://crsreports.congress.gov/product/pdf/R/R45749.

Math Works, Inc. MATLAB, version 2020b. Natick, MA: The Math Works, Inc., 2020. Accessed December 28, 2020. https://www.mathworks.com/

MicaSense. 2018. “Why Narrow Bands Matter.” *Micasense* (blog), March 20, 2018. https://blog.micasense.com/why-narrow-bands-matter-2e5e38d82fc2.

———. n.d. “Micasense Sensor Comparison.” Accessed December 22, 2020. https://micasense.com/compare-sensors/.

Naval Explosive Ordnance Disposal Technology Division. 1996. *Unexploded Ordnance (UXO): An Overview.* SFIM-AEC-PC 5179, U.S. Army Environmental Center. https://apps.dtic.mil/dtic/tr/fulltext/u2/a422504.pdf.

PEO Aviation Press Release. 2019. “U.S. Army Partners with Defense Innovation Unit to Leverage Commercial Drone Companies for Short Ran.” U.S. Army. April 29, 2019. https://www.army.mil/article/221053/u_s_army_partners_with_defense_innovation_unit_to_leverage_commercial_drone_companies_for_short_ran.

Powers, David. 2008. “Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation,” *Machine Learning Technology* (January). https://www.researchgate.net/publication/228529307_Evaluation_From_Precision_Recall_and_F-Factor_to_ROC_Informedness_Markedness_Correlation.

Qi, Zhipeng, Xiu Li, He Li, and Wentao Liu. 2020. “First Results from Drone-Based Transient Electromagnetic Survey to Map and Detect Unexploded Ordnance.” In *IEEE Geoscience and Remote Sensing Letters*, 1–5. https://doi.org/10.1109/LGRS.2019.2962754.

Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. “You Only Look Once: Unified, Real-Time Object Detection.” *ArXiv:1506.02640 [cs.CV]*. http://arxiv.org/abs/1506.02640.

Redmon, Joseph, and Ali Farhadi. 2017. “YOLO9000: Better, Faster, Stronger.” In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6517–25. Honolulu, HI: IEEE. https://doi.org/10.1109/CVPR.2017.690.

Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. 2016. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.” *ArXiv:1506.01497 [cs.CV]*. http://arxiv.org/abs/1506.01497.

Sharma, Pulkit. 2019. “Image Classification vs Object Detection vs Image Segmentation.” *Medium* (blog), August 21, 2019. https://medium.com/analytics-vidhya/image-classification-vs-object-detection-vs-image-segmentation-f36db85fe81.

Shen, Dongqing, Xin Chen, Minh Nguyen, and Wei Qi Yan. 2018. “Flame Detection Using Deep Learning.” In *2018 4th International Conference on Control, Automation and Robotics (ICCAR),* 17843715. Auckland, New Zealand: IEEE. https://doi.org/10.1109/ICCAR.2018.8384711

Shorten, Connor, and Taghi M. Khoshgoftaar. 2019. “A Survey on Image Data Augmentation for Deep Learning.” *Journal of Big Data* 6 (1) (July): 60. https://doi.org/10.1186/s40537-019-0197-0.

Siddiqua, Shahzia, Naveena C, and Sunilkumar S Manvi. 2019. “Recognition of Kannada Characters in Scene Images Using Neural Networks.” In *2019 Fifth International Conference on Image Information Processing (ICIIP)*, 146–50. Shimla, India: IEEE. https://doi.org/10.1109/ICIIP47207.2019.8985672.

Sony. n.d. “Sony A6000 E-Mount Camera with APS-C Sensor.” Accessed October 13, 2020. https://www.sony.com/electronics/interchangeable-lens-cameras/ilce-6000-body-kit.

Stehman, Stephen V. 1997. “Selecting and Interpreting Measures of Thematic Classification Accuracy.” *Remote Sensing of Environment* 62 (1) (October): 77–89. https://doi.org/10.1016/S0034-4257(97)00083-7.

Takumi, Karasawa, Kohei Watanabe, Qishen Ha, Antonio Tejero-De-Pablos, Yoshitaka Ushiku, and Tatsuya Harada. 2017. “Multispectral Object Detection for Autonomous Vehicles.” In *Proceedings of the Thematic Workshops of ACM Multimedia 2017 - Thematic Workshops ‘17*, 35–43. Mountain View, CA: ACM Press. https://doi.org/10.1145/3126686.3126727.

Talmadge, Eric. 2017. “64 Years after Korean War, North Still Digging up Bombs.” AP News. Last modified July 24, 2017. https://apnews.com/dd6256bad51e458cb2e8a1bf64b5c2b6.

Unmanned Systems Technology. n.d. “UAV Cameras | Suppliers of Cameras for UAV, Drones, UGV & Robotics.” *Unmanned Systems Technology* (blog). Accessed January 5, 2021. https://www.unmannedsystemstechnology.com/category/supplier-directory/cameras-imaging-systems/cameras/.

U.S. Department of Defense. 2015. *Unmanned Aircraft Systems Roadmap 2005 - 2030*. Washington, DC: Office of the Secretary of Defense.

U.S. Department of Transportation. 2013. *Unmanned Aircraft System (UAS) Service Demand 2015–2035: Literature Review & Projections of Future Usage*. U.S. Department of Transportation John A. Volpe National Transportation Systems Center. https://www.hsdl.org/?abstract&did=.

Viswanathan, Vibhuthi. 2018. “10 Latest Technology Used in Artificial Intelligence.” *SpringPeople* (blog), September 5, 2018. https://www.springpeople.com/blog/10-latest-technology-used-in-artificial-intelligence/.

Yakimenko, Oleg A. 2011. *Engineering Computations and Modeling in MATLAB/Simulink*. Reston, VA: American Institute of Aeronautics and Astronautics.

Zhang, Aston, Zachary C. Lipton, Mu Li, and Alexander J Smola. 2021. *Dive into Deep Learning*.

Zhao, Zhong-Qiu, Peng Zheng, Shou-Tao Xu, and Xindong Wu. 2019. “Object Detection with Deep Learning: A Review.” *IEEE Transactions on Neural Networks and Learning Systems* 30 (11): 21.

THIS PAGE INTENTIONALLY LEFT BLANK

# INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California