

**EVALUATING GENERALIZABILITY OF DEEP LEARNING
NETWORKS FOR WEED IDENTIFICATION AND THEIR
OPTIMIZATION FOR DEPLOYING ON EDGE DEVICE**

by

Varun Aggarwal

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science in Agricultural and Biological Engineering



School of Agricultural and Biological Engineering

West Lafayette, Indiana

May 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Dharmendra Saraswat, Chair

School of Agricultural and Biological Engineering

Dr. Aly El Gamal

Elmore Family School of Electrical and Computer Engineering

Dr. Bryan G Young

Department of Botany and Plant Pathology

Dr. Roger L Tormoehlen

School of Agricultural and Biological Engineering

Approved by:

Dr. Nathan Mosier

Dedicated to graduate students around the world

ACKNOWLEDGMENTS

I would like to acknowledge Dr. Saraswat for his continued guidance and support throughout the program which was invaluable at various phases of research and writing. I would also like to thank my committee members, Dr. Aly El Gamal, Dr. Bryan G Young, and Dr. Roger L Tormoehlen for providing their comments and suggestions which helped in shaping this thesis.

A special thanks to Julie Young and Brent Mansfield for helping me in identifying field trials and conducting UAS flights to acquire aerial image data. I would also like to acknowledge Purdue University's ABE department for allowing me the opportunity to conduct this research and giving me a platform to engage with outstanding faculty members, graduate students, and staff. Finally, I want to express my gratitude to Ben Hancock and Aanis Ahmad for assisting me in overcoming numerous research barriers.

TABLE OF CONTENTS

LIST OF TABLES	8
LIST OF FIGURES	9
LIST OF ABBREVIATIONS	11
LIST OF SYMBOLS	13
ABSTRACT	14
1. INTRODUCTION	15
1.1 Background	15
1.2 Problem Statement	21
1.3 Research Objectives	22
1.4 Impact Statement	22
2. REVIEW OF LITERATURE	23
2.1 Global yield loss and site-specific weed control	24
2.1.1 Why is early season weed control important?	24
2.1.2 Why is it necessary to reduce herbicide use in fields?	25
2.2 Sensing techniques for weed identification	25
2.2.1 What types of imaging sensors are being used for weed identification?	25
2.2.2 Why hasn't image processing been successful in achieving near-human accuracy in weed identification?	26
2.3 Machine Learning/Deep Learning in weed identification	28
2.3.1 What Machine Learning (ML) methods have been used for weed identification? ...	28
2.3.2 What deep learning networks are being used for weed identification?	29
2.4 Effect of operational parameters on weed management	31
2.4.1 How does the height of UAS flight affect the accuracy of plant and weed identification?	31
2.4.2 Does the growth stage affect the accuracy of a DLN for weed identification?	32
2.5 Next generation of weed management-edge computers in agriculture	33
2.5.1 How are edge computers proving useful in IoT?	33
2.5.2 How can edge computers be integrated with robots?	34
2.5.3 Why do DLNs have to be optimized for edge devices?	35

3. METHODOLOGY	37
3.1 Dataset Preparation	37
3.1.1 Data collection site	37
3.1.2 Flight details	38
3.1.3 Image data.....	39
3.1.4 Image Data Annotation.....	40
3.1.5 Data Augmentation.....	45
3.2 Object Detection Architectures.....	45
3.2.1 Faster R-CNN	46
3.2.2 YOLOv4	47
3.2.2.1 Data Augmentation in YOLOv4	48
3.2.3 Object detection network training	50
3.3 Image Segmentation.....	51
3.3.1 UNET.....	51
3.3.2 DeeplabV3+	52
3.3.3 Image segmentation network training	53
3.4 Evaluation Matrices	54
3.4.1 Intersection over Union score (IoU score).....	54
3.4.2 Mean Average Precision (mAP score)	56
3.5 Network deployment on edge devices	60
3.5.1 Network Optimizations - TensorRT™	60
3.5.2 Network Optimizations – tkDNN.....	61
3.5.3 Optimization methods for NVIDIA Jetson Nano and NVIDIA Xavier NX	62
3.5.3.1 Object detection optimization process	62
3.5.3.2 Image Segmentation optimization process.....	63
4. RESULTS	66
4.1 Results for Object Detection for weed identification	66
4.1.1 Faster R-CNN - Two-stage object detection	66
4.1.2 YOLOv4 – Single-stage object detection	70
4.1.3 Comparison of Faster RCNN and YOLOv4 for weed identification	74
4.2 Results for Image Segmentation	76

4.2.1	UNET segmentation results	76
4.2.2	DeeplabV3+ segmentation results	80
4.2.3	Comparison of UNET and DeeplabV3+ segmentation networks.....	84
4.3	Results for Network Deployment on Edge devices	87
4.3.1	Object detection on Edge devices.....	87
4.3.2	Image Segmentation on Edge devices	89
4.4	Comparison of results with other studies.....	91
5.	CONCLUSION.....	94
	REFERENCES	97

LIST OF TABLES

Table 3.1: Flight parameters for data collection	39
Table 3.2: Image Data Collection details.....	40
Table 3.3: Growth stage of Soybean and average weed height for Palmer amaranth on Data Collection dates.....	40
Table 3.4: YOLOv4 training parameters	49
Table 4.1: Growth stage of Soybean and average weed height for Palmer amaranth on Data Collection dates for Object Detection.....	66
Table 4.2: Generalization mAP score of Faster RCNN networks	67
Table 4.3: Generalization mAP score of YOLOv4 networks	71
Table 4.4: Growth stage of Soybean and average weed height for Palmer amaranth on Data Collection dates for Object Detection.....	77
Table 4.5: UNET results for generalization tests.....	80
Table 4.6: DeepLabV3+ results for generalization tests.....	83

LIST OF FIGURES

Figure 3.1: Soybean field boundaries at Winamac, Indiana - 76.2 x 15.2 m2.....	38
Figure 3.2: YOLO Mark software (left), Generated *.txt file for labels (right-top), YOLO labeling format (right-bottom)	41
Figure 3.3: Representation of YOLO labelling format for Soybean label.....	42
Figure 3.4: Label Studio annotation example a) pixels marked with brush tool b) creation of outline around the leaf with the polygon tool	43
Figure 3.5: Segmentation Annotation. RED - Palmer Amaranth, BLUE - Soybean, Grey – Background	43
Figure 3.6: (a) point of mouse click for the magic wand tool (b-f) extent of selection for different tolerance values.....	45
Figure 3.7: YOLOv4 Architecture [Source: Bochkovskiy et al. (2020)]	47
Figure 3.8: Mosaic Data Augmentation (left), Cutmix Data Augmentation (right)	50
Figure 3.9: UNET Architecture (Ronneberger et al., 2015a)	51
Figure 3.10: Atrous Spatial Pyramid Pooling as used in Deeplab V3+.....	53
Figure 3.11: DeeplabV3+ Architecture (L. C. Chen et al., 2018).....	53
Figure 3.12: IoU score calculation.....	55
Figure 3.13: IoU score calculation.....	56
Figure 3.14: True Positive (TP), False Positive (FP), and False Negative (FN) calculation based on IoU threshold	57
Figure 3.15: Illustrative example of True Positive, False Positive, and False Negative	58
Figure 3.16: Flowchart depicting the process of mean average precision calculation for Object Detection	59
Figure 3.17: TensorRT™ Workflow (NVIDIA, 2021f)	61
Figure 3.18: YOLOv4 optimization flowchart	64
Figure 3.19: DeeplabV3+ optimization flowchart	65
Figure 4.1: Object Detection testing mAP score for Faster R-CNN.....	67
Figure 4.2: Faster R-CNN generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer amaranth. The RED bounding box is the ground truth, while the YELLOW bounding box is the prediction.	69
Figure 4.3: Generalization mAP score for Faster RCNN	70

Figure 4.4: Object Detection testing mAP score for YOLOv4.....	71
Figure 4.5: YOLOv4 generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer amaranth. The RED bounding box is the ground truth, while the YELLOW bounding box is the prediction.	73
Figure 4.6: Generalization mAP score for YOLOv4	74
Figure 4.7: Comparison of testing mAP score of Faster RCNN and YOLOv4.....	75
Figure 4.8: YOLOv4 detection results in hard situations. RED bounding box represents Soybean while YELLOW bounding box represent Palmer amaranth.....	76
Figure 4.9: IoU score for trained UNET models	78
Figure 4.10: UNET's generalization IoU score for Palmer amaranth.....	79
Figure 4.11: IoU score results for trained DeeplabV3+.....	81
Figure 4.12: DeeplabV3+'s generalization IoU score for Palmer amaranth	83
Figure 4.13: UNET generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer amaranth. The RED color in represents Palmer amaranth, and the black color represents the background	85
Figure 4.14: DeeplabV3+ generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer Amaranth. The RED color in represents Palmer amaranth, and the black color represents the background.....	86
Figure 4.15: Optimization results for YOLOv4 on NVIDIA Nano. The green bar represents the highest FPS	88
Figure 4.16: Optimization results for YOLOv4 on NVIDIA Xavier NX. The green bar represents the highest FPS	89
Figure 4.17: Optimization results for DeeplabV3+ on NVIDIA Nano. The green bar represents the highest FPS	90
Figure 4.18: Optimization results for DeeplabV3+ on NVIDIA Xavier NX. The green bar represents the highest FPS	91

LIST OF ABBREVIATIONS

AUC	Area Under Curve
ANN	Artificial Neural Network
AI	Artificial Intelligence
ASPP	Atrous Spatial Pyramid Pooling
AP	Average Precision
AH	Average Height
AWH	Average Weed Height
BoF	Bag-of-Freebies
BoS	Bag-of-Specials
CMOS	Complementary Metal Oxide Semiconductor
CUDA	Compute Unified Device Architecture
CT	Confidence Threshold
CNN	Convolution Neural Network
CTWR	Critical Timing of Weed Removal
cuDNN	CUDA® Deep Neural Network library
DLN	Deep Learning Network
ExG	Excess Green Index
FN	False Negative
FP	False Positive
Faster R-CNN	Faster Region-based Convolutional Neural Network
FPGA	Field programmable Gated Arrays
FP16	Floating Point 16
FP32	Floating Point 32
fps	frames per second
FCN	Fully Connected Network
GPU	Graphics Processing Unit
GCP	Ground Control Point
INT8	Integer 8
IoU	Intersection over Union
IT	IoU threshold
LiDAR	Light Detection and Radar
ML	Machine Learning
mAP	mean Average Precision
MAC	Multiply and Accumulate Operation
NDVI	Normalized Difference Vegetation Index
NGRDI	Normalized Green Red Difference Index
ONNX	Open Neural Network Exchange

PAN	Path Aggregation Network
PR-curve	Precision-Recall Curve
RF	Random Forest Approach
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Networks
RGB	Red, Green Blue
RPN	Region Proposal Network
R-CNN	Regions with CNN features
RW	Robot Weeders
SIFT	Scale Invariant Feature Transform
SPP	Spatial Pyramid Pooling Layer
SURF	Speeded Up Robust Features
SVM	Support Vector Machine
TPU	Tensor Processing Unit
TP	True Positive
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
YOLOv4	You Only Look Once version 4

LIST OF SYMBOLS

cm	centimeter
m	meter
m/s	meter per second

ABSTRACT

Recent studies have indicated the need for agriculture automation to reduce the excessive use of herbicides, its detrimental effects on the environment, and the rise of herbicide-resistant weeds. In addition, the predicted increase in population to 9.7 billion by 2050 requires an increase in food production that cannot be met without proper weed control. At the current forefront of automation, deep learning has achieved high accuracy for weed classification and localization. Limited efforts have gone into developing and evaluating the generalizability of deep learning networks (DLN) over crop growth stages and weed heights. Further, even less work has been done for optimizing these computation-hungry DLN so that they could be deployed on lightweight edge devices for potential integration with an Unmanned Aerial System (UAS). Hence, in this research, the generalizability and deployment ability of four DLNs were evaluated for two computer vision tasks, i.e., object detection and image segmentation. For each task, the best performing network was optimized on two edge devices, namely, NVIDIA® Jetson Nano™ and NVIDIA® Xavier NX™. Finally, studies were conducted to determine the edge devices' frame rate for weed identification. For image segmentation, neither DeeplabV3+ nor UNet could generalize accurately for early season weed identification. For object detection, the YOLOv4 network trained on the V1 growth stage of soybean and 7.62 cm average weed height (AWH) of Palmer amaranth generalized the best with generalizability mean average precision score of 70.33 %. When optimized using tensorRT (floating-point precision of 16) on the edge devices, YOLOv4 resulted in 4.6 fps on Jetson Nano and 27.8 fps on Xavier NX, resulting in the highest fps achieved on an edge device for weed identification from UAS images. This research has resulted in developing foundational data, identifying promising deep learning-based algorithms, and evaluating edge devices that could lead to designing a real-time weed identification and UAS-based smart weed management system.

1. INTRODUCTION

1.1 Background

An increase in global population to 9.7 billion by 2050 (United Nations, 2019) would require an estimated 25 – 70 % increase in food production to meet the rising demand (Hunter et al., 2017). A common consensus among multiple studies has been to increase the crop yields rather than acquiring more arable land to simultaneously achieve sustainable agricultural and environmental targets (Grassini et al., 2013; Hunter et al., 2017; Jaggard et al., 2010; Silva, 2018). One way to increase the crop yields is by controlling one of the dominant causes of yield loss i.e. weeds. In US corn belt alone, weeds are reported to be responsible for yield losses amounting to almost 56.6 % in corn (Soltani et al., 2016) and 51% in Soybean (Soltani et al., 2017) on farms without any weed control strategy.

Conventionally, weed control is accomplished by three methods i.e., manual, mechanical, and chemical. Manual weed control is the oldest and considered most effective method (Abbas et al., 2018). However, it is time consuming and used in locations with abundant and cheap labor availability (Young et al., 2014). Mechanical weed control is economical compared to manual methods but is effective only for crops sown in straight rows and at wide enough row spacing to allow use of tillage implements like weeders, harrows and cultivators (Abbas et al., 2018). However, mechanical weed control is not an effective method because of several factors such as requiring repeated operations thereby increasing energy demand, inability to remove weeds from within crop rows, sometimes partial uprooting of weeds that lead to their regeneration etc. Chemical weed control is more efficient and effective compared to manual and mechanical methods of weed control. However, the use of herbicides is highly detrimental to the environment and its application under the assumption of uniform weed distribution has led to evolution of herbicide resistant weeds (Gaines et al., 2020). An advanced technique of herbicide spraying, termed precision spraying has been reported to reduce herbicide requirement by almost 95% (Tona et al., 2018). Thus, it has potential to offset disadvantages associated with broadcast applications. However, precision spraying still relies on human labor that drives up the cost and reduces efficiency of this method. Furthermore, continued shortage of labor, increased labor cost

(Fennimore & Cutulle, 2019a), realization to reduce environmental damages (Hunter et al., 2017) and an increasing trend for growing food organically (USDA ERS, 2021) has made it imperative to utilize advances in robotics, machine vision, artificial intelligence etc. to revolutionize weed management by 2050 (Westwood et al., 2018).

On current automation forefront, Robotic Weeders (RW), Artificial Intelligence (AI) (Jha et al., 2019) and Unmanned Aerial Systems (UAS) are being increasingly researched (Sylvester, 2018a). A number of new solutions have been launched such as See & Spray (Blue River Technologies, Sunnyvale, CA, USA), Avo (ecoRobotix, Yverdon -les-Bains, Switzerland), Trimble® WeedSeeker® (NTech Industries Inc., Ukiah, CA, USA), a normalized difference vegetation index (NDVI) based green biomass differentiator for spot spraying (Trimble, n.d.) and Deepfield Robotics (Farming revolution GmbH, Ludwigsburg, Germany), an autonomous robot powered mechanical weeding service. However, the applicability of current solutions in different crops, under varying row spacing and varying soil conditions is still a standing challenge (Fennimore & Cutulle, 2019b). A successful automation effort for weed control will be the one that can (i) identify weeds under field conditions (ii) determine the best method of elimination, and (iii) actuate with most suitable weeding tool whether chemical or mechanical (Young, 2018).

Weed identification under field conditions has been pursued based on either spectral or spatial properties of plants. Difference in reflectance spectra of leaves has been used for differentiating between weeds and crops (Barrero & Perdomo, 2018a; Y. Huang et al., 2016; Reddy et al., 2014; Sanders et al., 2019). Other efforts have used a combination of multi-spectral, infra-red and red-green-blue (RGB) images for weed identification based on vegetative indices like NDVI, normalized green red difference index (NGRDI) and excess green index (ExG) (Barrero & Perdomo, 2018a; López-Granados et al., 2016a; Lottes et al., 2017; Sa, Chen, et al., 2018). However, spectral properties of plants could vary based on stress, cloud cover, sunlight, shadow and dust (Louargant et al., 2018) thus, limiting applicability of spectral based methods alone in Midwest USA with variable environmental conditions. To avoid these challenges, other studies have tried using the spatial properties of plant leaves such as shape and texture for distinguishing between crops and weeds (Bakhshipour & Jafari, 2018b; Hung et al., 2014). Some have also utilized geometrical features within an image to identify crop rows and then isolate plants outside

them for inter-row weed identification (Al Mansoori et al., 2018; M. Bah et al., 2018; M. D. Bah et al., 2017; Lottes et al., 2017). Such techniques can work well with row crops but would fail in situations with heavy occlusion due to crop canopy or for the purpose of intra-row detection.

Neither spectral nor spatial properties can solely be used in every situation of weed identification. This is especially a problem during early season crop growth as early crops and weeds share similar spectral and spatial properties (López-Granados et al., 2016b). Though use of Unmanned Aerial Vehicle (UAVs) (also referred to as UAS in this chapter) carrying high resolution cameras could help overcome limitation with spatial resolution but high spatial resolution comes at the cost of low spectral resolution (Fernández-Quintanilla et al., 2018). Further, accurate identification of early season weeds requires achieving a minimum ground sampling distance of 0.6 cm (A. de Castro et al., 2018). This requirement restricts the UAV to fly closer to the ground which in-turn reduces the amount of land covered on a full battery charge (Torres-Sánchez et al., 2013).

UAVs are increasingly being used for aerial imaging in agriculture. One estimate by the Federal Aviation Administration suggested UAV sales increased to approximately 6.4% between 2018 to 2019 (Federal Aviation Administration, 2020). For a typical farm less than five hectares, it is costlier to use satellite images and manned aircrafts for the purpose of precision agriculture (Matese et al., 2015; Xiang & Tian, 2011). UAVs on the other hand are small, comparatively inexpensive and can capture quality data with high temporal frequency. They can be retrofitted with a wide range of sensors like RGB cameras, Thermal Infrared Cameras, Multispectral and Hyperspectral cameras depending on the requirements. While they are being increasingly researched in the community, they provide their own sets of challenges which are needed to be overcome i.e. limited payload capacity, limited flight time and limited real time onboard processing (Deng et al., 2020a; Y. He & Weng, 2018).

Multispectral sensors have been used by a number of research groups for weed identification (Amziane et al., 2020; Osorio et al., 2020; Pantazi et al., 2017). These sensors have a limited number of bands, usually between 3 to 10. This necessitates prior knowledge about selecting bands to differentiate reflectance spectra between weeds and crops. And because each field has a unique set of weeds and crops, it is not possible to devise a universal multispectral sensor that can be used

in all the fields and under all variable environmental conditions. On the other hand, with hyperspectral sensors, a wider spectrum with hundreds or thousands of bands is captured at once with each band's width ranging between 5 – 20 nm (Adão et al., 2017). In fact, during the analysis, having higher number of spectral bands can prove to be more beneficial for weed identification (Farooq et al., 2018). Moreover, hyperspectral sensors can be used to determine the significant spectral regions to differentiate between weeds leading to higher confidence in identification (Y. Li et al., 2021; Shirzadifar et al., 2020). The spectral regions are only 10-20 nm wide therefore, it's not possible to differentiate between weeds and crops (Scherrer, 2019) or differentiate between non-resistant weeds (Shirzadifar et al., 2020) using a multispectral sensor alone due to their wider band width. Despite the promise of hyperspectral sensors, they are several times more expensive compared to RGB and multispectral sensors (Adão et al., 2017). Another challenge of using multispectral or hyperspectral sensors in UAVs, is the need of radiometric calibration. This is a tedious task and requires a lot of additional time and expertise to accomplish. An accurate and cost-effective solution for performing radiometric calibration of UAS acquired images remains an active area of research (Iqbal et al., 2018). Nonetheless, irrespective of the sensor type, the sensor data is analyzed using state-of-the-art computer algorithms for identifying weeds.

Conventionally, digital image processing methods were used for this purpose. The digital image processing methods involved extracting shape, texture and color features from the images and feeding them into a feature classifier for classification (Kounalakis et al., 2016, 2018; Lottes et al., 2017). Some studies also utilized features like Fourier components, Wavelet features, Scale Invariant Feature Transform (SIFT) (Lowe, 2004) and Speeded Up Robust Features (SURF) (Bay et al., 2006) for training the classifiers (Suh et al., 2018; Wilf et al., 2016). By using machine learning classifiers such as Support Vector Machines (SVM) and Random Forest Approach (RF), these studies were able to pave the way for using computer vision for weed identification. However, image processing algorithms used for weed identification (Ahmad et al., 2018; López-Granados et al., 2016b; Louargant et al., 2018; Sapkota et al., 2020) could not result in achieving accuracy levels higher than 95% needed for developing a robust weed sensor (Westwood et al., 2018). Above all, for successful classification of weeds with image processing, the features had to be manually selected to maximize classification accuracy. This task is not only tedious and time-consuming but also suffers from the issue that the features set suitable for a differentiating between

certain plant set would not be the ideal ones for a different plant set (Cruz et al., 2017). Moreover, classification methods are incapable of identifying the presence of multiple weeds in a single image. This limits their applicability in field conditions because it is not uncommon to see multiple weeds growing in vicinity in the field (Fernández-Quintanilla et al., 2018; A. Wang et al., 2019a).

Recently, with the rise of DLN, the system of manual feature selection is becoming outdated. These networks are capable of selecting the best features directly from the input images without human intervention. Another benefit from DLN is the ability to transfer learning from a system trained in one setting to another setting, thereby reducing efforts required for training another system (Espejo-Garcia et al., 2020; Suh et al., 2018). But compared to conventional machine learning algorithms like SVM and K-Nearest Neighbor, DLNs take longer time to train. Although, at the time of inference, they run significantly faster (Y. Chen et al., 2014).

DLNs can be trained for a wide array of applications. They can not only be used to determine the species of a crop/weed in an image (Ahmad et al., 2018) but also for localizing the said crop/weed in the image (Buddha et al., 2019; Champ et al., 2020). Such networks which localize and classify objects in an image are called Object detection networks. They are capable of localizing multiple instances of an object in an image which is particularly important when using UAVs for scouting the fields for weeds (Sivakumar et al., 2020a). Most of the object detection networks can be classified into two types i.e. two stage detectors and single stage detectors. Two stage detectors use a Region Proposal Network (RPN) to first find the regions within an image which could detect potential objects followed by second stage which classifies the objects into classes and determines the size of bounding box for the classified object. Some examples of two-stage detectors are Mask Region based Convolution Neural Networks (Mask RCNN) (K. He et al., 2017) and Faster Region based Convolution Neural Networks (Faster RCNN) (Ren et al., 2016). On the other hand, single stage detector like Single Shot Detector (SSD) (W. Liu et al., 2016) and You Only Look Once (YOLO) (Redmon et al., 2016) directly determines the class and bounding boxes for all the objects in the image by using the concept of anchors (Soviany & Ionescu, 2018). Recently, some anchor-free single stage detectors are also being developed for object detection, like CornerNet (Law & Deng, 2019). Single stage detectors are comparatively less accurate compared to the two-stage detectors but at the same time are much faster. This has led to the popularity of YOLO networks

for object detection. YOLO has been used by multiple studies for object detection in plant images (Espinoza et al., 2020; Gao et al., 2020; Yu et al., 2020). Recently, the fourth version of YOLO called YOLOv4 was published in April 2020, with accuracies comparable to other state-of-the-art object detection network and real-time frame rates of around 65 fps (Bochkovskiy et al., 2020b). YOLOv4 has not been used for weed identification and localization studies to date, therefore, it is important to assess its performance vis-a-vis two-stage detectors to make informed decisions.

Some DLNs can also be used for pixel-wise image semantic segmentation. Each pixel of an image is assigned a class and a cluster of pixels belonging to same class become an object. The benefit of semantic segmentation is that it is one of the most effective technique is tackling the lingering problem of overlapping and occlusion in weed identification (A. Wang et al., 2020). Another advantage of using image segmentation over object detection is the ability to determine the weed density in the field images directly from the output. This is because, in output, every pixel is labelled and by simply dividing the count of weed pixels with total pixels in the field, density can be estimated (Fawakherji, Potena, et al., 2019). Although, the problem of using segmentation in early season crops is the time-consuming process of labelling the images. Especially because the size of weed is very small in UAV images and to label them pixel by pixel is difficult (Zou et al., 2021). In 2020, Khan et al. proposed a new Deep Neural Network for semantic segmentation particularly for weed identification (Khan et al., 2020). It was called CED-Net, short for cascaded encoder-decoder network. The results showed that it was able to outperform then state of the art semantic segmentation algorithm i.e. DeepLabV3 (L. C. Chen et al., 2017) while being three times smaller in terms of number of parameters. This meant that such networks could now run faster and with higher accuracy on edge devices.

With edge computing, the vision of real-time weed identification can be realized. Edge computers are devices which do the computation close to data source rather than on cloud servers (W. Shi et al., 2016). This reduces the time delay between data collection and data processing. Granted, cloud servers have more resources and are more powerful but the rising challenge is to develop fast and accurate deep learning algorithms which do inference in resource limited situations (O’Grady et al., 2019). Industries are pushing the limit of hardware for edge computing with big companies like Google and NVIDIA releasing new and more powerful hardware every year. They are

equipped with specialized integrated circuits, called Tensor Processing Units (TPU) (Jouppi et al., 2017), which are specifically designed to accelerate the performance of DLNs. Some research groups have been successful in integrating such edge devices with commercially available UAVs for weed identification (Deng et al., 2020a; Fawakherji, Potena, et al., 2019). Both of these studies use NVIDIA TX2 module for weed identification, but they were unable to achieve real-time frame rates. In November 2019, NVIDIA announced the smallest and the most powerful AI accelerator, Jetson Xavier NX (NVIDIA Newsroom, 2019). This device is about 3.5 times faster and 3.5 times lighter than NVIDIA TX2 module. This would open the door to deploy faster DLNs on UAV and hence its effectiveness for real-time weed identification needs to be evaluated.

1.2 Problem Statement

Weed identification is a challenging task because of the variable conditions the weeds grow in. Early season, both crop and weeds have a very similar spatial and spectral characteristics. This poses a big challenge for computer vision algorithms to differentiate between the two. The visual features are limited largely due to the resolution of imaging sensor. When it comes to aerial imaging, there is a scale to balance. On one hand, we can get high resolution images by having camera sensors very close to the ground but it would take a lot of time to cover the whole field. On the other hand, we can choose to employ aerial systems, like UAS, to cover the whole field faster but because the images are captured from a higher elevation, pixel resolution will suffer thus reducing the minute details which are necessary to differentiate weeds early in the season.

To add to this complexity, when capturing images, lighting is a key factor in getting accurate colors and details. It is not uncommon for clouds to change lighting conditions in a field even on a sunny day. This drastically affects the quality of images. Though this effect is more pronounced in RGB images than multispectral or hyperspectral images.

Hence there is a need for a robust and generalizable deep learning algorithm, which can effectively differentiate between the crop and weed varieties with an accuracy upwards of 95% (Westwood et al., 2018) from the images captured by commercially available RGB cameras. Additionally, there is a need for the development of algorithms that can run in resource limited edge computers.

1.3 Research Objectives

- Evaluate and fine tune single stage object detection networks and compare their performance with two stage object detection networks for classifying and localizing weeds in RGB images
- Determine the effect of growth stage on identification and generalization of image segmentation deep neural networks for weed identification
- Deploy the best performing network from objective 1 and 2 on memory and energy limited minicomputers i.e. Jetson Xavier NX and Jetson Nano and evaluate their performance for in-field weed identification

1.4 Impact Statement

Assessing performance of Deep Learning algorithms that help identify weeds under field conditions will help in developing smart machines for site-specifically treating weeds. Deployment of optimized deep learning algorithms on lightweight, low powered, and resource limited computing devices such as NVIDIA Nano and Xavier NX will help in developing smart UAS that can automate site-spraying operations. This research is aimed at developing foundational data, identifying promising deep learning-based algorithms, and evaluating edge devices that could lead to designing a real-time weed identification and UAS-based smart weed management system

2. REVIEW OF LITERATURE

In this chapter, multiple studies are evaluated for site-specific weed management, automation in precision weed control, sensors in weed identification, application of deep learning in weed identification, operational parameters of weed management, and use of edge computers in agriculture. The chapter contains five sections, with each section organized in the form of answers to important research questions:

- Section 1: Global yield loss and site-specific weed control
 - Why is early season weed control important?
 - Why is it necessary to reduce herbicide use in fields?
- Section 2: Sensing techniques for weed identification
 - What types of imaging sensors are being used for weed identification?
 - Why hasn't image processing been successful in achieving near-human accuracy in weed identification?
- Section 3: Machine Learning/Deep Learning in weed identification
 - What Machine Learning (ML) methods have been used for weed identification?
 - What deep learning networks are being used for weed identification?
- Section 4: Effect of operational parameters on weed management
 - How does the height of UAS flight affect the accuracy of plant and weed identification?
 - Does the growth stage affect the accuracy of a Deep Learning Network (DLN) for weed identification?
- Section 5: Next generation of weed management-edge computers in agriculture
 - How are edge computers proving useful in IoT?
 - How can edge computers be integrated with robots?
 - Why do DLNs have to be optimized for edge devices?

2.1 Global yield loss and site-specific weed control

2.1.1 Why is early season weed control important?

One of the most significant challenges in site-specific weed management (SSWM) is early-season weed detection (A. I. de Castro et al., 2018). Early in the season, weeds are very competitive with the crop and fight for nutrients and space to the detriment of crop growth (Tursun et al., 2015). Barnyardgrass, a weed responsible for corn yield loss in Greece, reduced the grain yield by 24 to 34% when emergence happened before the V4 growth stage of corn (Travlos et al., 2011). But, following the V4 growth stage, corn had a competitive advantage and outgrew the effect of barnyard grass. Only 9% yield loss was reported in corn when the barnyardgrass emergence happened after the V4 growth stage. A similar study for corn was conducted in 2003 to assess the timing effect of postemergence glyphosate application on weed control in corn. The study was conducted in eight states in the northern-central USA. The study concluded that optimal timing for initial glyphosate application in glyphosate-resistant corn was when weeds were below 10 cm in height, within 23 days of corn planting, and before the V4 growth stage (Gower et al., 2003).

For soybean, a study was conducted in southern Wisconsin between 2008 and 2009. It was reported that controlling weeds before they are 23 cm in height or soybean had reached the V4 stage would reduce the yield loss (Fickett et al., 2013). Additionally, the yield would be reduced even further if weeds were controlled before 15 cm in height. Another study looked at the effect of row spacing on the critical timing of weed removal (CTWR) in soybean. It was observed that CTWR occurred earlier when row spacing was higher. In 76 cm soybean rows, weeds had to be removed on or before the V1 soybean growth stage. In 38 cm rows, CTWR coincided with V2 growth stage of soybean. Finally, for 19 cm soybean rows, CTWR occurred during the V3 growth stage of soybean. In summary, weeds must be targeted between the soybean's V1 – V3 growth stage (Knezevic et al., 2003). More studies conducted for soybean have all observed that minimal yield loss would occur if weed is controlled between V2-V4 growth stage for soybean or when weed height is below 10-15 cm (Coulter & Nafziger, 2007; Dalley et al., 2004; Eyherabide & Cendoya, 2002; Soltani et al., 2019). Hence, based on all the recommendations, early-season weed detection and removal become important to minimize yield loss in soybean.

2.1.2 Why is it necessary to reduce herbicide use in fields?

Chemical control of weeds in fields has been ever prevalent since the introduction of 2,4-D (Peterson, 1967). While weed control is important to ensure desired yield during a crop production season, chemical weed control poses a serious threat in the future unless done sustainably. The threats include an increase in herbicide-resistant weeds and environmental degradation due to the overuse of herbicides (Weed Science, 2021). The growth in herbicide-resistant weeds is further fueled by the fact that the discovery of new herbicides is limited. Moreover, repeated use of the same herbicide lead to development of herbicide-resistant weeds (Westwood et al., 2018).

Residue from broadcast application of herbicide can be washed away by intense rainfall and surface waterflow into nearby water bodies which could lead to water pollution (Krutz et al., 2005). Additionally, herbicides can kill non-targeted plants even while not impacting the overall yield of the crop. A study conducted in France showed that the use of herbicide led to the killing of rare plant species while not affecting the wheat yield (Gaba et al., 2016). Herbicides have also been shown to negatively affect honeybee colonies (Jumarie et al., 2017). Worst of all, many of the chemical herbicides are determinantal to human health (Caiati et al., 2019; Tyohemba et al., 2021). Given the serious concerns associated with herbicide use, demand for organic food is growing (Shafie & Rennie, 2012). Keeping these factors in mind, new sustainable weed control methods have to be developed (Westwood et al., 2018).

2.2 Sensing techniques for weed identification

2.2.1 What types of imaging sensors are being used for weed identification?

Sensor technologies have enabled ground-based weed management with companies like WeedSeeker (Trimble Inc., California, USA), enabling precision weed sprayers that have saved almost 45-67% herbicide use (Christensen et al., 2003). Advancements in UAS have led to increased payload capacity, increased flight duration, and increased stability. This has enabled research exploration for faster and more robust weed identification in an agricultural field by combining state-of-the-art sensor technology (Sylvester, 2018b). Sensors such as Red, Green, Blue (RGB) cameras, multispectral and hyperspectral cameras, and thermal infrared cameras are utilized.

Out of the four sensors mentioned in the preceding paragraph, RGB is the most commonly used for weed identification (Esposito et al., 2021). In comparison, multiple studies have utilized multispectral sensors due to the availability of radiometric bands that can specifically target plants (Gibson et al., 2004; Louargant et al., 2017; Sa, Popović, et al., 2018; Slaughter et al., 2008). However, the pre-requisite of radiometric calibration (Micasense, 2019) and lower resolution than RGB sensors (Barrero & Perdomo, 2018b) make them less than ideal for weed identification. Similarly, while hyperspectral cameras contain hundreds of bands that can enable even more granular separation of vegetation (Che'ya et al., 2021), their much higher cost and heavyweight (Taghizadeh et al., 2011) make them difficult to be mounted on a UAS. Thermal cameras can be useful for agricultural applications due to their ability to detect plant stress by measuring the surface temperature change (Calderón et al., 2013; Gago et al., 2015), however, their limited spatial resolution and frequent need for calibration (Messina & Modica, 2020), has not resulted in their wide-spread use.

Given the ready availability of RGB sensors, their higher resolution compared to other imaging sensors, and their low cost, more efforts have been invested in using the RGB cameras for weed identification research.

2.2.2 Why hasn't image processing been successful in achieving near-human accuracy in weed identification?

Image processing has proven to be an accurate tool in precise weed identification (Taghaddomi-Saberi et al., 2015). The traditional image processing workflow has four steps: 1) image pre-processing, 2) vegetation segmentation, 3) feature extraction from vegetation, and 4) feature classification.

Image pre-processing is performed with techniques like normalization (Tang et al., 2018), color space transforms (H. D. Cheng et al., 2001), and denoising (H. Liu et al., 2014). Image pre-processing is meant to transform the input image such that it becomes easier to perform vegetation segmentation.

Vegetation segmentation is performed to isolate the background from vegetation. While there are multiple methods for segmentation (Hamuda et al., 2016), studies have reported success with the use of thresholding with green indices from RGB images (Guijarro et al., 2011) or by using the

NDVI to extract vegetative parts from RGB+NIR images (Potena et al., 2017). Although, a precursor to successful vegetation segmentation is accurate sensor calibration. Lighting changes very often in the field, especially in the US Midwest (the region of interest for this research), making it even more important to do proper sensor calibration (Sujaritha et al., 2017). Without good calibration, the potential of image processing would either be lacking, or it would not work well in field conditions (Elstone et al., 2020).

The third step of the image processing workflow is feature extraction from segmented vegetation. Features are extracted from vegetation so that the features could be used to distinguish between crop and weed. Features include visual characteristics like leaf shape (Swain et al., 2011), leaf texture (Slaughter, 2014), or spectral characteristics like color (B. Cheng & Matson, 2015). But the problem with feature extraction is the overlapping of plants. In field, leaves of various plants overlap (Fernández-Quintanilla et al., 2018). In the case of row crops, if the row widths are small or during the late season when the canopy starts to close, the overlap increases even further (H. Liu et al., 2014). Additionally, the presence of weeds inside crop rows also induces overlap and occlusion. Both overlapping and occlusion affect feature extraction. For example, if the feature to be extracted is leaf area, overlapping would lead to calculating the leaf area of multiple leaves instead of one.

Subsequently, the final image processing step (i.e., feature classification) depends on the features extracted from vegetation. Feature classification utilizes classifiers like SVM (Bakhshipour & Jafari, 2018a) and clustering (Behmann et al., 2015) for differentiating between plant and weed features. But finding features that enable these classifiers to differentiate well requires feature selection by the domain expert (A. Wang et al., 2019b). Moreover, due to weeds and crops' spectral and spatial similarity in the early season, an optimal set of features that classifiers can use for differentiation is hard to find (LÓPEZ-GRANADOS, 2011; Pérez-Ortiz et al., 2016). Not to mention, a set of features that work well for one set of plants might not be useful for distinguishing between another set of plants. This would not only prevent having a generalizable system for weed identification but also add overhead for curating a separate set of optimal features for different field locations (A. Wang et al., 2019b).

Overall, traditional image processing workflow will always face issues to achieve near-human accuracy in weed identification; hence, deep learning algorithms need to identify weeds as they address the limitations of traditional image processing (Olsen et al., 2019).

2.3 Machine Learning/Deep Learning in weed identification

2.3.1 What Machine Learning (ML) methods have been used for weed identification?

ML algorithms are a class of computer algorithms capable of automatically learning a given task based on experience and performance feedback (Mitchell, 1997). When programming ML algorithms, experience is essentially the data provided for training. At the same time, performance feedback measures performance in a task that is used to inform and incentivize ML algorithms to learn better. Some of the ML models used for application in weed identification include regression, clustering, and DLNs.

Regression is used to predict an output variable based on a set of input variables. Examples of regression algorithms include linear and logistic regression (Cox, 1959), ordinary least square regression (Craven & Islam, 2011), multivariate adaptive regression splines (Friedman, 2007), and cubist regression (Quinlan, 1992). Particularly for weed identification, regression has been used in conjunction with feature extractors such that the input for regression becomes the weed features while the output becomes the weed class. Conventionally, the input features have been handcrafted (Andújar et al., 2012; Hestir et al., 2008; W. Zhang et al., 2018), although, in recent years, features have also been extracted automatically based on deep learning (Espejo-Garcia et al., 2020; W. Zhang et al., 2018).

Clustering is a technique used to find a natural grouping of data (Tryon, 1957). These grouping of data are called clusters. Multiple algorithms of clustering exist like k-means clustering (Lloyd, 1982), expectation-maximization algorithm (Dempster et al., 1977), and hierarchical clustering (Johnson, 1967). For weed identification, clustering has been utilized by creating a multi-dimensional data space followed by creating clusters of data. Based on the separation of the clusters, various weed species have been identified (Meyer et al., 2004; Weis & Gerhards, 2008; S. Zhang et al., 2019, 2021).

DLNs have been popularized in the recent decade, especially after the success of AlexNet Network (Krizhevsky et al., 2017) in the LSVRC-2010 challenge (Russakovsky et al., 2015). Unlike other machine learning models like regression and clustering, DLNs do not require manual feature extraction prior to weed identification. Instead, the model learns through iterations to identify the best features for weed identification and differentiation. The process of learning for DLNs is also called training. Learning and training would be used interchangeably in the context of DLNs in this thesis. More details on Deep Learning and its applications for weed identification are given in section 2.3.2.

2.3.2 What deep learning networks are being used for weed identification?

DLNs are a type of Machine learning model which utilizes Artificial Neural Networks (ANN) to learn tasks. Most of these models utilize an ANN, which contains multiple convolutional layers inside them. Such networks are also called Convolutional Neural Networks (CNNs). Convolution is a mathematical operation performed on two functions. In the context of images, the two functions are the input image and a filter. Hence, convolution for images is sliding a filter across the input image from left to right and top to bottom. The filters extract multiple features from an image like the edges, corners, texture, color tonality, etc. Although, compared to traditional image processing, the filters do not have to be hand-picked. Instead, the power of deep learning lies in the fact that networks govern the best filters during training with the sole objective of improving accuracy (Krizhevsky et al., 2012).

Specifically, three different types of DLNs are used for computer vision: Image Classification, Object Detection, and Image Segmentation.

Image Classification is the task of classifying the whole image into predefined categories called classes. In the context of weed identification, these classes could be of weed subclasses, i.e., monocot and dicot, weed family, e.g., Pigweed Family or Umbel Family, and weed name, e.g., Palmer amaranth. Moreover, there could be an additional class of a specific crop to differentiate between crops and weeds. Multiple research groups have shown the advantage of DLN over traditional ML models. When classifying Para grass, Nutsedge, and Parthenium, CNN outperformed SVM, one of the most powerful machine learning classifiers. Out of 125 weed

images, SVM misclassified 21 images, while CNN only misclassified four images (T. et al., 2019). For detecting dicot weeds in pasture, CNN proved to be more accurate in weed classification with an accuracy of 96.88 % compared to SVM's accuracy of 89.40 % (W. Zhang et al., 2018). In the soybean production system, CNN achieved a weed classification precision of 99.1 % compared to SVM's precision of 97.1 % (dos Santos Ferreira et al., 2017). In this study, although the performance of SVM is comparable to CNN, the authors argue that CNN has an inherent benefit of automatic feature selection, which eliminates the need for a domain expert needed for SVM's good performance.

Object detection is a computer vision task where the location is also determined in addition to the classification of the object determined. Typically, the location of an object in the image is given by a bounding box. The bounding box is drawn such that it encapsulates the entirety of the object within its bounds. For example, for weed identification, a good bounding box would be one that would contain all the leaves of a weed within its bounds. Compared to classification, where the entire image is processed for a single image label, object detection treats the image area of interest as a potential object. This enables the identification of multiple objects within an image, each of which could be of a different type.

Image segmentation is a computer vision task of designating a class to each pixel of the image. For example, given an image, the image segmentation model would aim to identify each pixel of the image as either a weed, crop, or soil. Hence, compared to Image classification, image segmentation provides richer information. For weed identification, Fully Connected Network (FCN) was used to segment the weeds and crops from the background with an F1 score of 92.4 % (Lottes et al., 2018). Similarly, VGG-16 based SegNet was used to identify crop (sugarbeet), weed, and background (Sa, Popović, et al., 2018). The objective of the study was to create a weed cover map from images collected by a UAS. The results looked very promising, with Area under the Curve (AUC) scores of 83.9%, 86.3 %, and 78.2 % for background, crop, and weed, respectively. Other research groups have also explored the application of deep learning-based image segmentation in other crop production systems for weed identification. Modified U-NET image segmentation was utilized for carrot fields with weed infestation with an average dice similarity coefficient of 83.44 % (Brilhador et al., 2019). A custom image segmentation network was used

for detecting *Colchicum Autumnale* flowers in grasslands. 88.6 % of the flowers were detected successfully using a custom image segmentation network (Petrich et al., 2020). Image segmentation was used in the rice production system to determine the weed cover map. The accuracy of the network was 91.96%, and it additionally led to herbicide saving varying between 58.3 % to 70.8 % (H. Huang et al., 2018). The accuracy of weed segmentation in cornfields using cNET resulted in an accuracy of 92.08 % (Andrea et al., 2018).

Given the advantage of deep learning networks, such as their ability to process UAS images and creation of weed cover map for site-specific weed management (H. Huang et al., 2018; Sa, Popović, et al., 2018), its application in soybean production system becomes important to explore.

2.4 Effect of operational parameters on weed management

2.4.1 How does the height of UAS flight affect the accuracy of plant and weed identification?

Deep learning networks generally perform better when more distinguishing features can be extracted from a given image. Therefore, it is advantageous to capture higher resolution images from a UAS for plant and weed identification. The higher resolution of images can be achieved in two ways when acquiring them via a camera mounted on a UAS, a) by increasing the resolution of the imaging camera, and b) by flying closer to the ground. Increasing the camera resolution is not an ideal solution primarily because it would increase the cost of the system. On the other hand, flying closer to the ground has multiple disadvantages. These include a large number of images captured in low altitude flights to cover a small area, the added time required to process the additional images, the effect of wind draft from UAS propeller in capturing blur-free images, and risk of the UAS colliding with objects or with the human operator (Hassanein & El-Sheimy, 2018).

Given these challenges, there is a need to identify the optimal UAS flight height and determine the effect of flight height on the accuracy of weed identification. A study conducted for the *Solanum rostratum* Dunal plant to evaluate the impact of UAS flight height on segmentation of plants from the background concluded that flying higher than 10 m reduced both the precision and recall of plant segmentation (Q. Wang et al., 2021). Additionally, out of 2.5m, 5m, 10m, and 15m flights,

5m height gave the highest precision and recall score for plant segmentation showing the benefit of higher resolution images for deep learning. Another similar study conducted for corn/maize looked at the effect of UAS flight height on estimating the plant distance (J. Zhang et al., 2018). It was reported that the relative error increased when the flight height increased from 2m to 5m. The increase in error was more pronounced when the planting distance was small, i.e., the maize was planted closer to each other.

The effect of UAS height on DLN accuracy is not an objective pursued in the current research, yet it remains an active area of research.

2.4.2 Does the growth stage affect the accuracy of a DLN for weed identification?

The growth stage is an important factor in determining the timing and type of herbicide application in the field (Mahmoodi & Rahimi, 2009) and for crop yield prediction (Tahir Ata-Ul-Karim et al., 2016). Currently, the growth stage is determined primarily by scouting, and the scope for automatic growth stage estimation is limited (Rasti et al., 2021). Through studies conducted in determining the growth stage of plants and weeds, an inference can be drawn about the success of DLN at various stages.

A study conducted on 18 weed species counted leaves to determine the growth stage (Teimouri et al., 2018). The best performing DLN was reported to achieve an accuracy of 70%. It was additionally observed that it was easier for the network to identify early-season weeds than later-season weeds. This was a counterintuitive result as typically early season weeds would be more challenging to identify due to lack of significant morphological features. In the study, counterintuitive results were attributed to the overlapping of weeds, which could have caused the DLN system to miscount.

A separate study was conducted for twelve wheat growth stages and eleven barely growth stages. While the classification accuracy was not reported for individual growth stages of the crops, the overall classification of growth stages was around 90 % (Rasti et al., 2021).

For sugar beets, the performance of DLN was compared on early-season and late-season images. The late-season images were acquired two weeks following the early-season images. The precision increased by 3 % on late-season images (Milioto et al., 2017). A different study tried to determine the accuracy of DLN on the four-week advance stage compared to the stage on which it was trained. A reduction in accuracy of 37.6 % was reported (Potena et al., 2017).

While there aren't a lot of studies that evaluate the effect of growth stages on DLN performance, there are even fewer which try to determine the generalizability accuracy of DLN across different growth stages. Hence both of the areas remain a gap in the literature.

2.5 Next generation of weed management-edge computers in agriculture

2.5.1 How are edge computers proving useful in IoT?

The increase in data generation in the past few years has had a tremendous effect on the growth of industries. This has led to the development of countless connected devices and equipment. A growing number of IoT devices has been a topic of concern as it has caused an increase in network bandwidth required for transferring data and an increase in data storage required for the transmitted data. One estimate by Cisco Global Cloud Index (Cisco, 2020) states that amount of data generated by any device would reach 847 zettabytes per year compared to only 218 zettabytes in 2016. This has shown to be a major bottleneck in the cloud computing paradigm (W. Shi & Dustdar, 2016). Cisco Global Cloud Index estimated that by 2021, IoT connections will reach 13.7 billion (Cisco, 2020). That is almost twice the number of humans in the world currently, and this number will only increase in the future. Edge computing is shown to be a potential solution to the problem of the bottleneck (W. Shi & Dustdar, 2016). By processing data closer to the site of generation, load on the network can be reduced.

Moreover, processing data at the edge has additional benefits; this includes reduced response time (Satyanarayanan et al., 2009), reduced energy consumption by reducing the load on cloud computers (Chun et al., 2011), and reduced privacy concerns of data (Cook et al., 2018). In agriculture, edge devices are being used for monitoring irrigation systems (Mateos Matilla et al.,

2021), real-time poultry monitoring (Debauche et al., 2020), automated fleet management (Dautov & Song, 2019), and plant disease detection (Ale et al., 2019).

For agricultural applications, it is not uncommon to encounter fields with a spotty data connection, therefore, solely relying on cloud computing is questionable (Shepherd et al., 2020). Moreover, farmers are hesitant to send their farm data to cloud servers (Gupta et al., 2020). Since edge devices solve both these problems, it becomes an ideal option for application with agricultural robots.

2.5.2 How can edge computers be integrated with robots?

Given the benefits of edge computers in solving bottlenecks for data processing (outlined in Section 2.5.1), their integration with the robotic system is rising. Robotic systems are equipped with sensors like RGB cameras, multispectral, hyperspectral, Light Detection and Ranging (LiDAR). These sensors help in robot navigation as well as vision-based tasks like object detection. In agriculture, vision-based robots are being developed for harvesting (Zhao et al., 2016), corn counting (Z. Zhang et al., 2020), weed classification (Jasiński et al., 2018), and crop quality monitoring (Saha et al., 2018).

While deep learning provides promising results in vision-based tasks, its integration with edge devices is a challenge, primarily due to the computational requirements of deep learning (Thompson et al., 2020). Graphical Processing Units (GPU) are a necessity for efficient deep learning training and inference. This is because GPU can perform multiple calculations simultaneously owing to the presence of multiple processing units. Each of the processing units is called a core. Alternatives to GPUs are neural processing units and tensor processing units, although their basic functioning is similar to a GPU.

NVIDIA provides multiple libraries which are targeted toward deep learning applications. Examples include Compute Unified Device Architecture (CUDA®) (NVIDIA, 2021c), NVIDIA CUDA® Deep Neural Network library (cuDNN) (NVIDIA, 2021d), and TensorRT (NVIDIA, 2021f). While CUDA and cuDNN are libraries to train deep learning networks, TensorRT is meant to infer deep learning networks, especially on NVIDIA GPU. Edge devices need to be equipped with GPUs for efficient deep learning inference. NVIDIA has released multiple edge devices

which come equipped with GPU. These include Jetson TX2, Jetson Nano, Xavier AGX, and Xavier NX (NVIDIA, 2021b).

These edge devices are being integrated with UAS and ground robots for vision applications in agriculture. NVIDIA Jetson TX2 has been integrated with UAS for application in Fruit Tree Pests Identification (C. J. Chen et al., 2021), weed mapping (Deng et al., 2020b), and spraying applications (L. Wang et al., 2019). Additionally, Xavier NX and Jetson Nano have been used for real-time apple detection (Mazzia et al., 2020). As highlighted in the studies above, the need for real-time deep learning algorithms on agricultural robots (Hu et al., 2021) shows that edge devices equipped with GPUs would prove to develop intelligent machines in agriculture.

2.5.3 Why do DLNs have to be optimized for edge devices?

For vision-based tasks, deep learning networks have to be optimized to achieve real-time performance. The real-time performance of deep learning networks is measured in frames per second (Wan & Goudos, 2020). The definition of real-time varies based on the application, but it is defined in terms of the camera frame rate for computer vision. A real-time deep learning network would be one that could achieve frame rates equal to or greater than the frame rate of the imaging camera (J. Chen & Ran, 2019). Typically, this number varies between 30 – 60 frames per second. Usually, deep learning networks like Faster R-CNN (Ren et al., 2017) and YOLOv4 (Bochkovskiy et al., 2020b) target real-time applications. In fact, YOLOv4 promises a frame rate of 65 fps. But such frame rates are only achieved on server-grade GPUs like NVIDIA V100. While deep learning networks are designed for application on mobile and edge devices, for example, MobileNet, tiny YOLOv4, they lack accuracy compared to deeper networks like YOLOv4.

Edge devices cannot contain server-grade GPUs primarily because of power and weight constraints. Moreover, it is not ideal for implementing mobile deep learning networks because of their lack of accuracy. Therefore, performing real-time identification on edge devices with large deep learning networks is not a trivial task. Fortunately, with the advancement in computer technology and the development of new hardware, it is possible to optimize deep learning networks such that their computation requirements during inference are reduced. It is important to note that

optimizations are only meant to be applied to a deep learning network after it has been trained. Optimization does not affect the training time for the networks.

Optimizations can be performed using TensorRT (NVIDIA, 2021f) on a trained deep learning network. Several optimizations are performed by TensorRT to reduce inference time, including Reduce Mixed Precision, Layer and Tensor Fusion, Kernel Auto Tuning, Dynamic Tensor Memory. The details about these optimizations are presented in Section 3.5.1.

Optimized deep learning networks are currently being used for application in autonomous navigation in vineyards. One study reported that optimizations increased control frequency from 21.95 Hz to 47.15 Hz (Aghi et al., 2020). Another study showed that when optimizations were applied to the image classification model Resnet50, the frame rate increased from 5.55 to 18.7 (Olsen et al., 2019). A very recent study used tensorRT optimizations for semantic segmentation of weeds from UAS images. They reported an increase in fps from just 14.33 fps to 45.05 fps for the MobileNetV2-UNet network (Lan et al., 2021).

These studies indicate that tensorRT can be viably applied for deep learning deployment on edge devices like Jetson Nano and Xavier NX. TensorRT is an actively developing library (NVIDIA, 2021f) and opens up a lot of potential for developing the next generation of agricultural robots.

3. METHODOLOGY

3.1 Dataset Preparation

Correct training data is a critical prerequisite in ensuring that deep learning networks adequately identify objects of interest. The training data should be representative of the field conditions to avoid poor network performance. For example, if occlusion is prevalent after a particular crop growth stage, then training the deep learning network with examples of occlusion will avoid poor detection outcomes. Additionally, consistency of training data annotation or data labeling is essential to ensure proper training of networks. For instance, if a weed is mismarked as a crop during the annotation process, the network would learn to misclassify weed when presented with an unseen image. The following sub-sections describe the data preparation pipeline starting with the acquisition of data to the training of deep learning networks.

3.1.1 Data collection site

The UAS was used to acquire data from a field (41°06'56.1" N, 86°41'20.9" W) located close to the city of Winamac in Indiana, USA. The primary crop and the primary weed in the field were Soybean and Palmer amaranth (*Amaranthus Palmeri*), respectively. Soybean was planted on May 22. The size of the experimental field was 76.2 m x 15.2 m. It helped demarcate experimental field boundaries and create an orthomosaic image by placing four Ground control points (GCP) at the corners of the field and one GCP at the center of the field (Figure 3.1). An orthomosaic image combines multiple orthophotos, i.e., image with geometric distortions removed, free of lens distortions, and corrected for topographic relief.



Figure 3.1: Soybean field boundaries at Winamac, Indiana - 76.2 x 15.2 m²

3.1.2 Flight details

Four flights were conducted between June 11-18, 2020, to acquire experimental field images. The flights were conducted between 10 AM and 3 PM to stay close to the solar noon. It helped in minimizing the impact of shadows on the images and image noise due to limited light.

The flight path was planned using DJI™ Ground Station Pro (SZ DJI Technology Co., Ltd., Shenzhen, China) application (hereafter referred to as "app") installed on an iPad. The position of the four GCPs was given as input to the app in addition to other parameters such as side overlap and front overlap. A complete list of input parameters is given in Table 3.1. A total of 10 flight lines were created parallel to the longest side of the field. The imagery data was acquired using Mavic Pro (SZ DJI Technology Co., Ltd., Shenzhen, China), an UAS from DJI flown at a speed of 1 m/s. It took approximately 12 minutes to cover the experimental field.

Table 3.1: Flight parameters for data collection

Flight Parameter	Parameter Value
Front overlap	80 %
Side overlap	80 %
Flight lines	10
Flight Speed	1 m/s
Flight Height	5 m
Total Flight time	12 minutes

3.1.3 Image data

DJI Mavic Pro (SZ DJI Technology Co., Ltd., Shenzhen, China) was equipped with a RGB Complementary Metal Oxide Semiconductor (CMOS) sensor that allowed capturing images at a resolution of 4000 x 3000 pixels. The camera was stabilized with an in-built 3-axis gimbal.

The images were collected from June 11 (i.e., 27 days after soybean planting) to June 18 (i.e., 34 days after planting). A test flight was conducted on June 5 (21 days after planting), though weed emergence was limited on that day. The duration of each flight was 12 min and resulted in 152 full-resolution images. Each full-resolution image was split into a tile size of 250 x 250 pixels before using as an input for training deep learning networks. The total number of tiles obtained per full-resolution image was 192. After splitting all full-resolution images, a total of 29,184 image tiles were obtained per day. Hereafter, the term "image" refers to image tiles discussed in this section. Details about image data are summarized in Table 3.2 and details about crop growth stage and the average weed height are given in Table 3.3

Table 3.2: Image Data Collection details

Data Collection Dates	Test Run June 5, 2020	Data Used for study June 11 to June 18, 2020
Full Resolution Images collected each day	152	
Full Resolution Image size	4000 x 3000 pixels	
Image Tile size	250 x 250 pixels	
Total number of image tiles per day	$152 \times \frac{4000 \times 3000}{250 \times 250} = 29184$	

Table 3.3: Growth stage of soybean and average weed height for Palmer amaranth on data collection dates

Date	Soybean Growth Stage	Palmer Amaranth Average Weed Height
June 5, 2020	Cotyledon Stage	Less than 2.54 cm
June 11, 2020	Unifoliate Stage	3.81 cm
June 12, 2020	Unifoliate Stage	3.81 cm
June 13, 2020	Unifoliate Stage	5.08 cm
June 14, 2020	V1 stage	6.35 cm
June 15, 2020	V1 stage	6.35 cm
June 16, 2020	V1 stage	7.62 cm
June 17, 2020	V1 stage	7.62 cm
June 18, 2020	V2 stage	8.89 cm

3.1.4 Image Data Annotation

Annotation is the process of categorizing and labeling objects or pixels in an image to train a deep learning network. The images were manually annotated for carrying out both object detection and image segmentation studies. In object detection studies, a bounding box was used to describe the location of the object of interest. Image segmentation, on the other hand, required pixel-level image annotation. Pixel-level annotation being a more detailed process, took approximately thrice the time compared to bounding box annotation. On average, each day's image took approximately 30 hours for bounding box annotation and 80 hours for pixel-wise annotation.

Bounding box annotation was manually carried out using YOLO Label Software (Kwon, 2018). The software was installed on a laptop with Intel™ Core® i7-7700HQ CPU @ 2.8GHz running Microsoft™ Windows 10 Pro operating system (OS). The software had a point-and-drag interface. The interface of the YOLO label is shown in Figure 3.2.

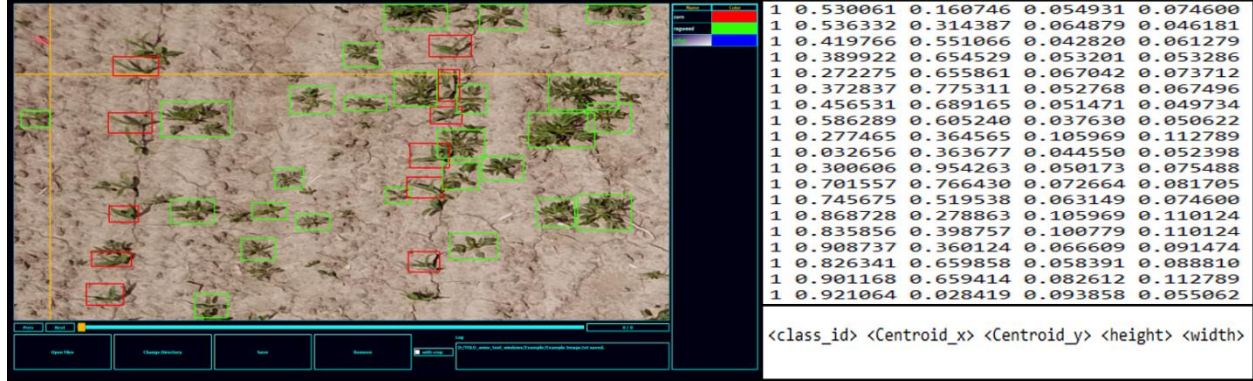


Figure 3.2: YOLO Mark software (left), Generated *.txt file for labels (right-top), YOLO labeling format (right-bottom)

Two classes on each image that were subjected to annotation were Soybean plants and Palmer amaranth weeds. After creating bounding boxes for the two classes of interest, the annotations were saved in YOLO format (Figure 3.2). Each line in the annotation represents one bounding box (Equation (1)).

$$< class\ id > < \frac{Xo}{X} > < \frac{Yo}{Y} > < \frac{W}{X} > < \frac{H}{Y} > \quad (1)$$

where,

class id = label index of the class (Soybean [0] or Palmer amaranth [1])

Xo = X coordinate of the bounding box's center (pixels)

Yo = Y coordinate of the bounding box's center (pixels)

W = Width of the bounding box (pixels)

H = Height of the bounding box (pixels)

X = Width of the image (pixels)

Y = Height of the image (pixels)

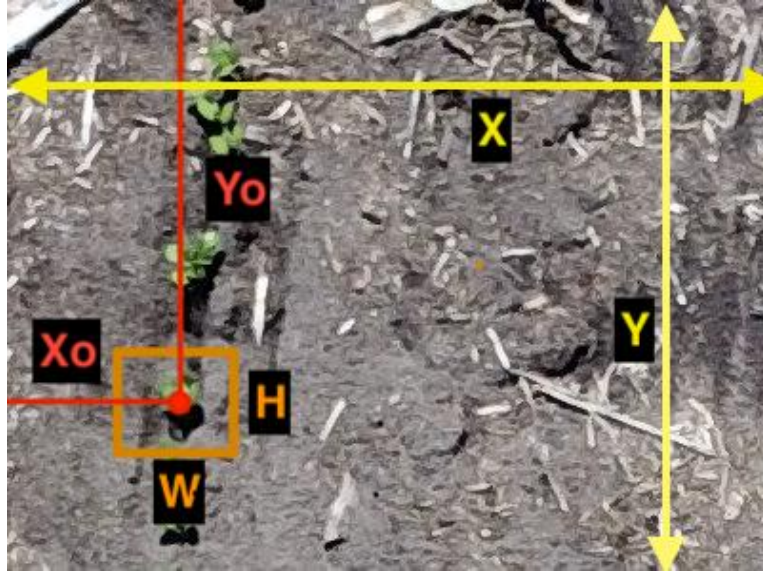


Figure 3.3: Representation of YOLO labelling format for Soybean label

Once all objects belonging to two classes of interests were marked in the images, a corresponding text file with the same name as the image file but a different extension, i.e., *.txt, was saved for storing the annotations. Both the text file and images were used for training the networks used in this study.

Pixel-wise annotation was performed for three classes, namely, Soybean, Palmer amaranth, and background. The annotation was completed using two different software, each with its advantages and disadvantages. The first software used was Label Studio (Tkachenko et al., 2020), an open-source software. The UI for label studio was accessed using a web browser like Google Chrome (Google LLC, California, USA) or Mozilla Firefox (Mozilla Foundation, California, USA). The interface of the software was customizable for different needs. For pixel-wise annotation, the interface was set to include two primary tools for annotation, the brush tool and the polygon tool (Figure 3.4). The brush tool was used to paint over areas of the image by clicking and dragging the mouse pointer. The thickness of the brush was adjusted based on the size of the object being annotated. For example, a thicker brush size was used around the center of the plant, while a thinner brush was used around the edges where precise annotation was required to avoid marking areas outside the plant. The range of brush size varied from 1 to 50. The polygon tool was used to draw a multi vertex polygon around the object boundary. All the pixels contained inside the polygon

were assigned the same class. An example of a pixel-wise annotated image for an image tile is given in Figure 3.5.

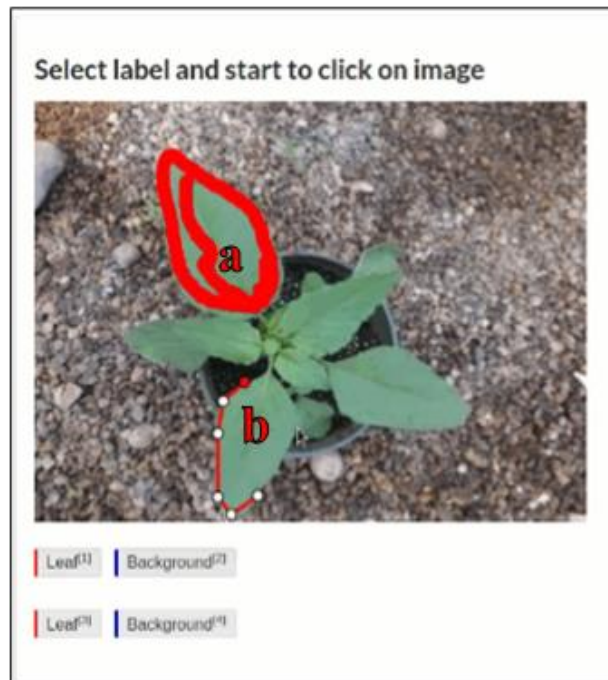


Figure 3.4: Label Studio annotation example a) pixels marked with brush tool b) creation of outline around the leaf with the polygon tool



Figure 3.5: Segmentation Annotation. RED - Palmer Amaranth, BLUE - Soybean, Grey – Background

The second software used for annotation was Adobe Photoshop (Adobe Inc., San Jose, California, US). It is commercial software for image editing developed by Adobe. The advantage of using Photoshop was the availability of the Magic Wand tool. It is a selection tool that allows selecting pixels based on the tone and color of the selection. The selection began by clicking in the object's center (Soybean plant, Palmer amaranth, or background). A selection of pixels was created such that all the pixels, similar to the pixels at the location of the mouse click, were part of the selection. The similarity of the pixels was based on the tolerance values (Figure 3.6). The tolerance was adjusted on a case-by-case basis using values of 5, 10, 15, 20, and 32 to ensure that most object pixels were selected and pixels belonging to background or other undesired classes were not a part of the selection. Once the single weed or crop pixels were selected with the magic wand tool, a paint bucket tool was used to annotate the pixels into respective classes by coloring the pixel with predefined class color (Figure 3.6-f). Sometimes, the selection was inaccurate, especially around the edges. During such instances, a paintbrush tool, similar to the brush tool used in the label studio, had to be used to refine the edges.

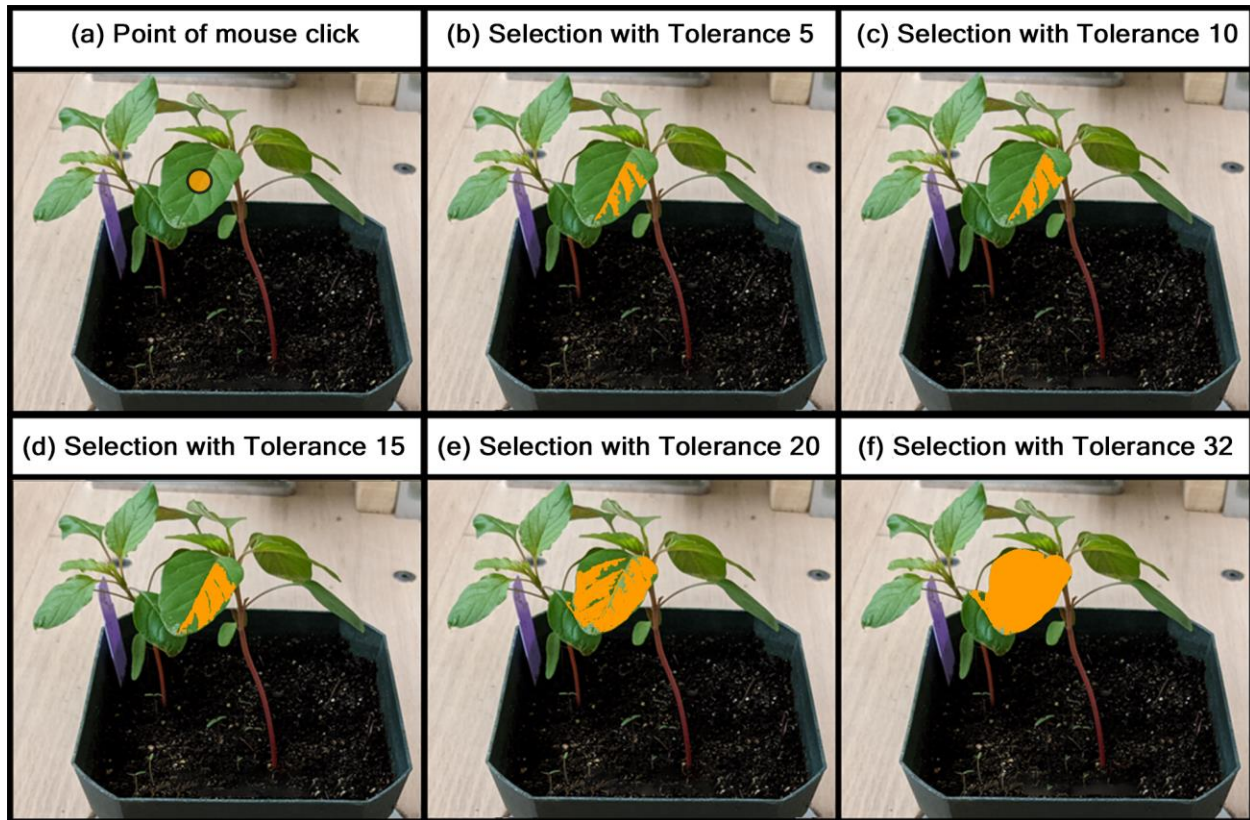


Figure 3.6: (a) point of mouse click for the magic wand tool (b-f) extent of selection for different tolerance values

3.1.5 Data Augmentation

Data augmentation is a technique used for training deep learning networks to increase/augment the existing training dataset. The existing dataset is augmented by applying linear and non-linear transformations. Data augmentation has been shown to increase testing accuracy and generalizability of the network (i.e., testing accuracy on a different dataset), especially for image identification tasks (S. Wu et al., 2020). Some conventional examples of data augmentation include brightness change, hue change, rotation, scaling, and vertical and horizontal flips.

3.2 Object Detection Architectures

Deep learning networks for object detection were used for detecting multiple instances of crops and weeds in each image. These networks belong to two classes of convolutional networks, i.e., two-stage detectors and single-stage detectors. As the name suggests, there are two stages of

detection in the former. Stage 1 is called Region Proposal Network (RPN), and Stage 2 is called object detection. On the other hand, using anchors, single-stage detectors combine region proposal and object detection into a single stage. The two-stage detector of choice for this research was Faster R-CNN. In contrast, the single-stage detector of choice was YOLOv4.

3.2.1 Faster R-CNN

Faster R-CNN is a part of the "Regions with CNN features" (R-CNN) family of networks (Ren et al., 2016). It is one of the most widely used networks among the family of R-CNN networks. It finds application in fields of marine exploration (P. Shi et al., 2021), human medicine (Kim & Huh, 2020), agriculture (Wan & Goudos, 2020) and autonomous vehicles (Yang et al., 2020). Faster R-CNN is a two-stage network. The architecture consists of

Stage 1: Region Proposal Algorithm

Stage 2: Object Detection Algorithm

Region proposal algorithm takes the training image as an input and outputs potential regions within the image where a crop or a weed could be present. Such proposals are generated by using a CNN called RPN. Initially, RPN generates a feature map of the input image. Feature map contains distinguishing features of the image, which RPN uses to generate proposed regions. The pruning layer processes the regions, eliminating overlapping proposals and prunes the total proposal to a set limit. By default, the limit is set to 300. Increasing the total proposed region limit leads to better detection chances and an increase in computation cost. On the other hand, reducing the limit leads to reduced computational load but could also cause object detection networks to miss some instances of crop and weed in the image.

The proposals are passed on to a second CNN, i.e., Fast-RCNN. Fast-RCNN shares the feature map of the input image, which RPN generated. Sharing the feature map between the two stages of the Faster R-CNN enables the network to learn shared features, which increases the overall detection accuracy. Moreover, it saves on computation cost as the feature map does not have to be separately generated for the two stages. Essentially, Fast-CNN can be thought of as having two

sets of input, the first being the shared feature map and the second being the proposals. The output of Faster R-CNN is thus the class and location of weed and crop in the input image.

3.2.2 YOLOv4

YOLOv4 is the newest successor in the YOLO series of object detection networks (Bochkovskiy et al., 2020b). It can deliver frame rates close to 65 frames per second (fps) (on NVIDIA Tesla V100) on the MS COCO dataset with 80 classes with an AP_{50} Score of 67.5%. Two major contributions compared to the previous version of YOLO (YOLOv3) are referred to by authors of YOLOv4 as Bag-of-Freebies (BoF) and Bag-of-Specials (BoS).

BoF consisted of changes made to training which directly improved the inference accuracy. Since these changes were made at the time of training, inference speed was not affected. It includes new data augmentation techniques like Cutmix (Yun et al., 2019) and Mosaic (Bochkovskiy et al., 2020b). BoS consists of changes that increase the inference accuracy at the expense of a slight decrease in inference time. YOLOv4 consists of three parts: backbone, neck, and head (Figure 3.7).

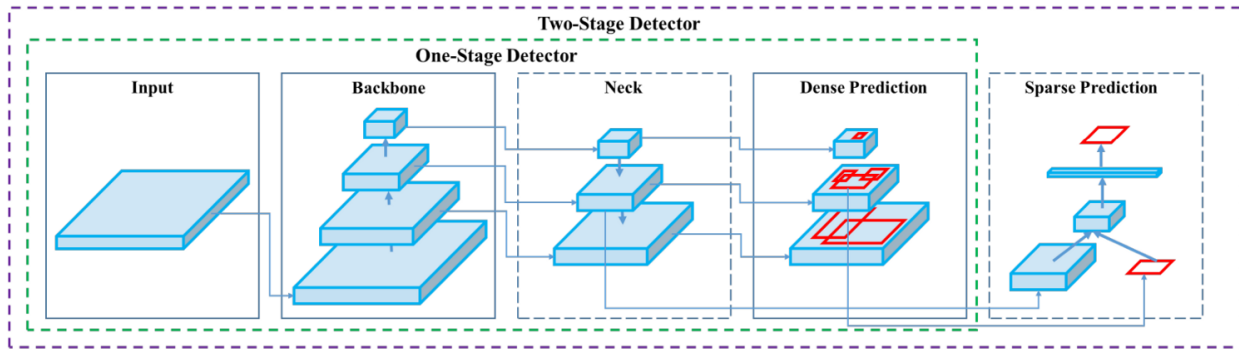


Figure 3.7: YOLOv4 Architecture [Source: Bochkovskiy et al. (2020)]

The purpose of the backbone is to extract features from the input image. CSPDarknet53 is used in YOLOv4 for the backbone compared to YOLOv3, which had Darknet53 as the backbone. The major difference between the two is the inclusion of Cross-Stage Partial Connections (CSP connections). CSP connections reduce the computation of the model by dividing the input feature of dense into two parts. The first part is sent through the dense and transition layers, while the

second part is transmitted to the next stage. Because only a part of the feature map is utilized for computation, the memory requirement of the network is also reduced.

The head of the network is responsible for actually predicting the bounding box location and object class. Although, before the features from the backbone are passed on to the head, the neck of the network enhances them. In YOLOv4, the neck includes Spatial Pyramid Pooling Layer (SPP) (K. He et al., 2015) and Path Aggregation network (PAN) (S. Liu et al., 2018).

The complete list of BoF and BoS applied to the network are as follows [Also listed in the original publication (Bochkovskiy et al., 2020b)]:

- BoF for backbone includes techniques for Data Augmentation like cutmix and mosaic and additional techniques like Class label smoothing and dropblock regularization
- BoS for the backbone of the network include new activation function, called Mish activation and network modifications including multi-input weighted residual connections and Cross-stage partial connections
- For the detector, the BoF includes techniques for data augmentation similar to one used in backbone, i.e. mosaic data augmentation. In addition, it includes extensive hyperparameter optimization, a new loss function: CIoU-loss and other techniques like CmBN, DropBlock regularization, self-adversarial training, random training shapes, multiple anchors for single ground truth, cosine annealing scheduler
- Finally, the BoS for the detector includes, activation function similar to backbone i.e. Mish activation, bounding box suppression with DIoU-NMS and new network blocks like SPP-block, SAM-block, PAN path-aggregation block.

3.2.2.1 Data Augmentation in YOLOv4

The mosaic data augmentation method was developed by Glenn Jocher, Founder and CEO of Ultralytics (Los Angeles, CA, USA). In this method, four training images are combined into one (Figure 3.8-left). The portion of the four images that make up the final augmented image and the ratio of the area occupied in the final augmented image is randomly determined for each new augmented image. Because only a part of the original image is visible in the final augmented image,

the YOLOv4 network is trained to identify plants in the image based on fewer distinguishing features. Additionally, mosaic data augmentation allows the YOLOv4 network to identify plants on a different scale. This is because when four images are used to fill up the space of a single image, the resulting size of each of the four images is reduced by at least one-fourth.

In Cutmix data augmentation, random patches from one image are pasted over a random area of a second image (Figure 3.8-right). The random patches are of variable size. Due to Cutmix augmentation, a specific portion of the image is covered/occluded by the patch. Thereby, the YOLOv4 network is trained to focus on image areas that would usually be ignored for distinguishing between plants. Hence, the localization ability of the network in the presence of occlusion goes up with the use of Cutmix Augmentation.

The network parameters for the YOLOv4 network and their significance are given in Table 3.4.

Table 3.4: YOLOv4 training parameters

Parameter	Significance	Parameter value
Network Size	Signifies the input size for the network.	512 x 384
Training Iterations	The total number of training iterations the network would be trained for.	6000
Batch Size	Signifies the number of images that are used for each training iteration.	64
Subdivisions	Signifies the number of sets a batch of images would be divided into. It reduces the amount of memory required to train the network.	8
Learning Rate	It is a parameter used to determine the amount of change in weights after every iteration.	0.0013



Figure 3.8: Mosaic Data Augmentation (left), Cutmix Data Augmentation (right)

3.2.3 Object detection network training

One of the objectives of this research was to determine the generalization ability of the networks in weed identification. Generalization essentially measures the accuracy of a trained network in identifying weeds when presented with a new and unseen dataset. The new dataset must differ from the training dataset in certain aspects such as field location, data collection date, and/or type of image sensor. In this research, four datasets were used, each differing from the other based on the data collection date. Both the object detection networks, i.e., Faster R-CNN and YOLOv4, were trained on UAS data. Four dates selected for assessing the generalization ability of the networks were 11th, 14th, 16th, and 18th June 2020. The YOLOv4 and Faster R-CNN networks were trained on-field images on only one date out of the four and tested on the remaining three dates. The generalization accuracy is reported separately for each of the three dates. For example, a YOLOv4 network trained on images from June 11 would be tested on images from June 14, June 16, and June 18, resulting in three separate generalizations accuracies, one for each of the three days. The network accuracy for object detection was determined by the Mean Average Precision score (mAP). A higher mAP produces better results and hence is favored. Further details about evaluation measures, including mAP, are available in Section 3.4.

3.3 Image Segmentation

Image segmentation is fundamentally different in comparison to Object detection. In image segmentation, instead of detecting the presence of plants in the image by drawing a bounding box around it, each pixel is assigned an object class. In addition to the two classes used in object detection (Soybean and Palmer amaranth), a new class was added in image segmentation. This class was called background. Background class included pixels corresponding to the soil, old dead remains of plants, and any other plant which was neither a crop or weed of interest. Because each image pixel was marked into one of the three classes, the deep learning-based segmentation networks took more time to train and produce interferences. For this research, two networks were selected, namely UNET and DeeplabV3+, and their performance was compared.

3.3.1 UNET

UNET was developed by Olaf Ronneberger for biomedical segmentation applications (Ronneberger et al., 2015a). UNET has an encoder-decoder architecture. It comprises two main components, which are called paths. The first of the paths is called the contracting path (encoder). It contains convolution layers, an activation function (Rectified Linear Unit – ReLU), a pooling layer, and finally downsampling layer. The purpose of the contracting path is to capture the context of the pixels, i.e., determine the essential features of a plant that can be used to determine its type. Although, by repeated downsampling, the location of the plant is lost.

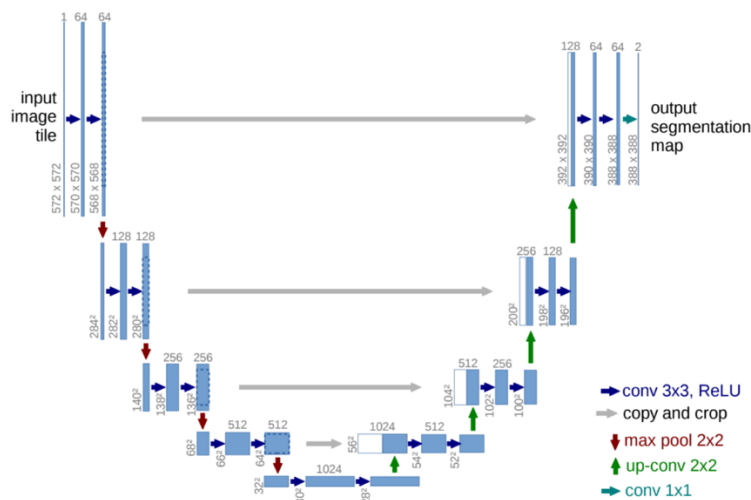


Figure 3.9: UNET Architecture (Ronneberger et al., 2015a)

The second path is called the expansive path (decoder). It is symmetric to the contracting path in terms of the layers. It begins with up-sampling and up-convolution. At this step, the features from the corresponding contracting path are concatenated before sending the features onto the following convolution layers. The purpose of the expansive path is to localize the pixel precisely. It could be thought of as tracking the location of the leaf pixels of a plant as the network up samples the original image at each step while borrowing the contextual information from the contraction path. Because both the paths are symmetric, the network architecture looks like a U, hence the name UNET (Figure 3.9).

3.3.2 DeeplabV3+

DeeplabV3+ was developed by Google and is an improvement over DeeplabV3 (L.-C. Chen et al., 2018). Similar to UNET, it is an encoder-decoder network. In addition, it comprises Atrous Spatial Pyramid Pooling (ASPP) to encode multi-scale context (Figure 3.10). This would be particularly useful in weed identification using UAS because the size of the weed changes from image to image because of the stage of the weed (

Table 3.3) and the variation in flight altitude. And because multi-scale information could be encoded, detecting smaller weeds in early growth stages could prove helpful. The network architecture is shown in Figure 3.11 .

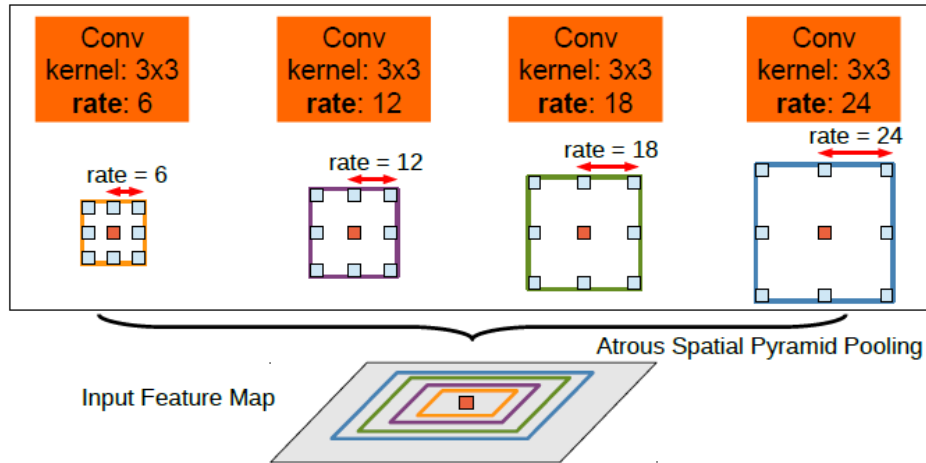


Figure 3.10: Atrous Spatial Pyramid Pooling as used in Deeplab V3+

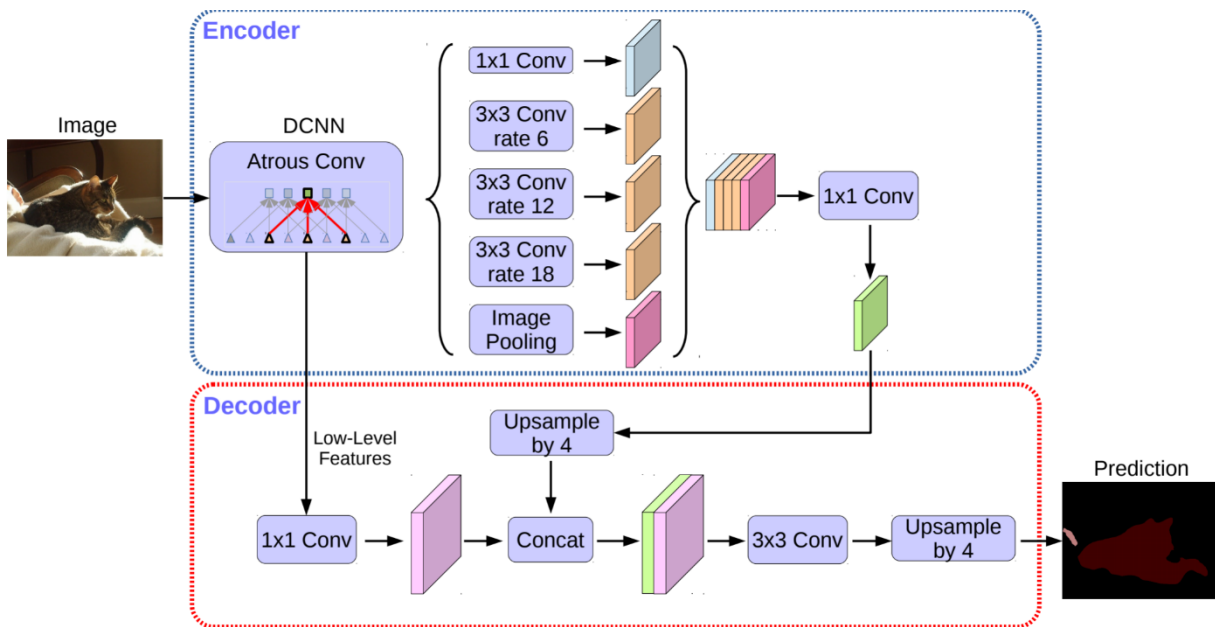


Figure 3.11: DeeplabV3+ Architecture (L. C. Chen et al., 2018)

3.3.3 Image segmentation network training

The two image segmentation networks were trained on UAS images starting from June 11 to June 18. This led to eight trained networks corresponding to each day. This was done to determine the baseline performance of the networks on each day. The models were evaluated on Intersection

over Union (IoU) score (Refer to Section 3.4 Evaluation Matrices) for each of the three classes, i.e., Soybean, Palmer amaranth, and background. The results were plotted on a line graph with training data on the x-axis and the IoU score of Soybean and Palmer amaranth on the y-axis. The IoU score for the background was reported separately in a table because it was not a significant evaluation matrix for the model.

Additionally, out of eight networks, four were selected for the generalization study. The four chosen dates were the same as those chosen for object detection networks, i.e., 11th, 14th, 16th, and 18th June 2020. The networks trained on one date were tested against the images from the remaining three dates. The results were presented in a table for the IoU score for each of the models. The model performance was compared to determine the best image segmentation model for weed identification among UNET and DeeplabV3+.

3.4 Evaluation Matrices

The performance of the two-image segmentation networks (UNET and DeeplabV3+) was evaluated using Intersection over Union (IoU) score. In contrast, the two object detection networks (YOLOv4 and Faster R-CNN) was evaluated using mAP. Both IoU and mAP will be defined in this section.

3.4.1 Intersection over Union score (IoU score)

IoU score is used to quantify the amount of overlap between the prediction and the ground truth. IoU can be calculated both for object detection and image segmentation. For simplicity of figures, in this section, the IoU score would be explained concerning bounding boxes. Nonetheless, the concept would be translated for image segmentation wherein instead of bounding boxes, the collection of pixels for different weeds and plants would be used to calculate the intersection and union of the area between the ground truth and prediction.

IoU simply is the ratio of intersection of the area of overlap and area of the union of the ground truth and prediction (Figure 3.12). The higher the match between the ground truth and prediction, the higher the IoU score. IoU score is calculated as a percentage. A complete overlap of ground truth and prediction is indicated by a score of 100 %, while a wholly separated ground truth and prediction would result in an IoU score of 0 %.

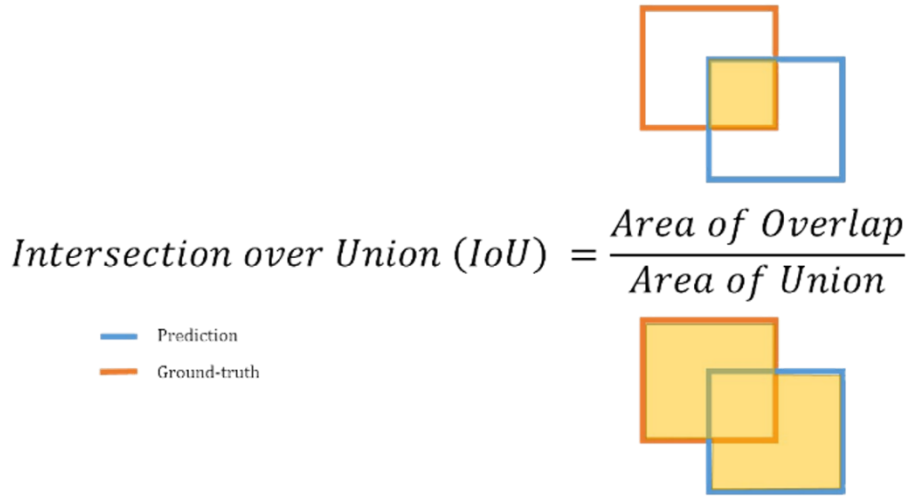


Figure 3.12: IoU score calculation

Consider an example of a plant that occupies an area of 1 cm² (Figure 3.13a). In the figure, the bounding box for the plant was subdivided into 144 cells. Each cell has a size of 1/12 cm x 1/12 cm. If the prediction box (blue in Figure 3.13) completely overlaps the ground truth box (brown in Figure 3.13), the IoU score would be:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \times 100 = \frac{121 \text{ overlapping cells}}{121 \text{ union cells}} = \frac{1}{1} \times 100 = 100 \%$$

If the prediction box (blue in Figure 3.13) is displaced by 1/12 cm in both x and y-axis, the IoU score would be:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \times 100 = \frac{121 \text{ overlapping cells}}{167 \text{ union cells}} = \frac{0.84}{1.16} \times 100 = 72.4 \%$$

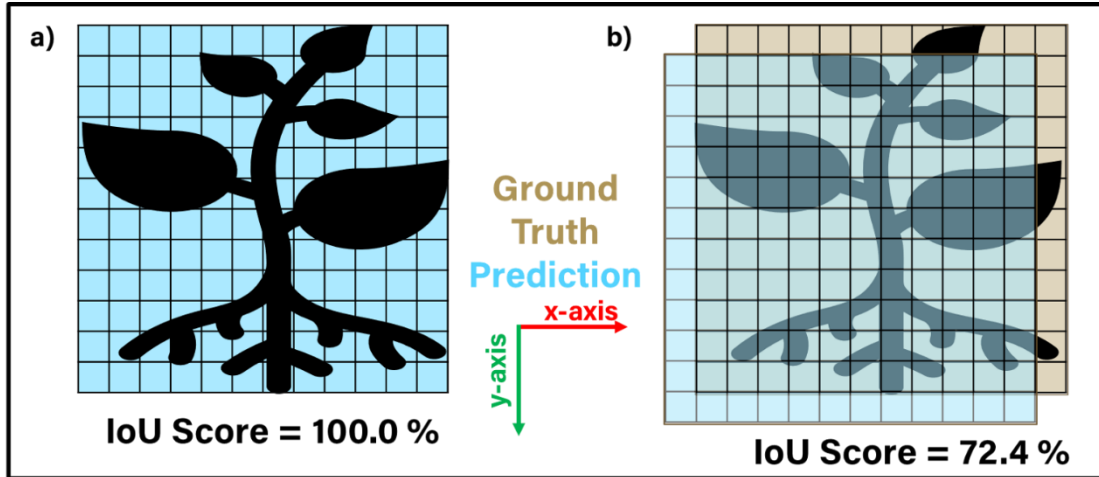


Figure 3.13: IoU score calculation

3.4.2 Mean Average Precision (mAP score)

mAP score was used to evaluate the performance of object detection algorithms in this research.

Whenever an image input is given to a trained object detection network, the output is a list of prediction boxes and the corresponding confidence score for each prediction box. The confidence score measures how strongly the network determines a prediction box belongs to a particular class (weed or plant). The value of confidence varies between 0 % and 100 %.

The process of mAP calculation begins with looping over prediction boxes for each of the classes separately. For each class, an average precision (AP) score is calculated. In the end, the average of AP scores for all the classes results in mAP score.

AP is an average of precision values corresponding to 11 recall values [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0] from Precision-Recall Curve (PR Curve).

PR curve is generated by varying the confidence threshold from 0 to 100. When the confidence threshold (CT) is varied, it affects the True Positive (TP), False Positive (FP), and False Negative (FN) detections, which in turn affects the precision and recall values. This is because CT is used as a filter to remove detection boxes with unacceptable confidence values (Confidence score <

CT). The remaining prediction boxes, called the pruned prediction boxes, are used to calculate TP, FP, and FN values.

TP, FP, and FN are calculated based on the IoU score between prediction boxes and ground truth boxes. Firstly, several ground-truth boxes with missed predictions are counted towards FN detections. In other words, if a plant is present in the image, but the network cannot predict its presence, then it is an instance of False-negative detection. Out of all the prediction boxes, any prediction box with IoU score higher than a predefined IoU threshold (IT) is counted towards a true positive detection. Additionally, any prediction box with IoU scores less than IT is counted towards False positive. TP, FP, and FN are diagrammatically represented in Figure 3.14. Additionally, the difference between TP and FP can be understood based on a simplistic example as given in Figure 3.15.

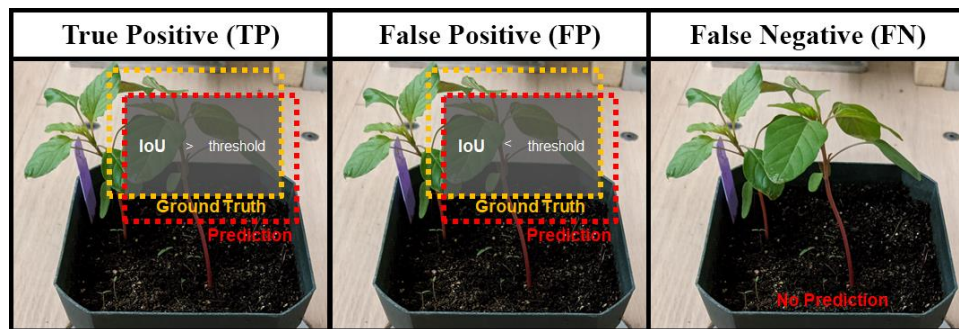


Figure 3.14: True Positive (TP), False Positive (FP), and False Negative (FN) calculation based on IoU threshold

Based on the value of TP, FP, and FN, the precision and recall are calculated. Precision is the percentage of successful detection of plants out of a total number of positive detection of plants in the image. On the other hand, recall is the percentage of positive detections out of the image's total number of ground truth boxes. Equation (2) and Equation (3) show the calculation for Precision and Recall, respectively.




True Positive (TP)	False Positive (FP)	False Negative (FN)
 <p>This is corn</p>	 <p>This is corn</p>	 <p>There is no corn</p>

Figure 3.15: Illustrative example of True Positive, False Positive, and False Negative

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

The complete process of mAP calculation as described in this section is represented with a flowchart in Figure 3.16..

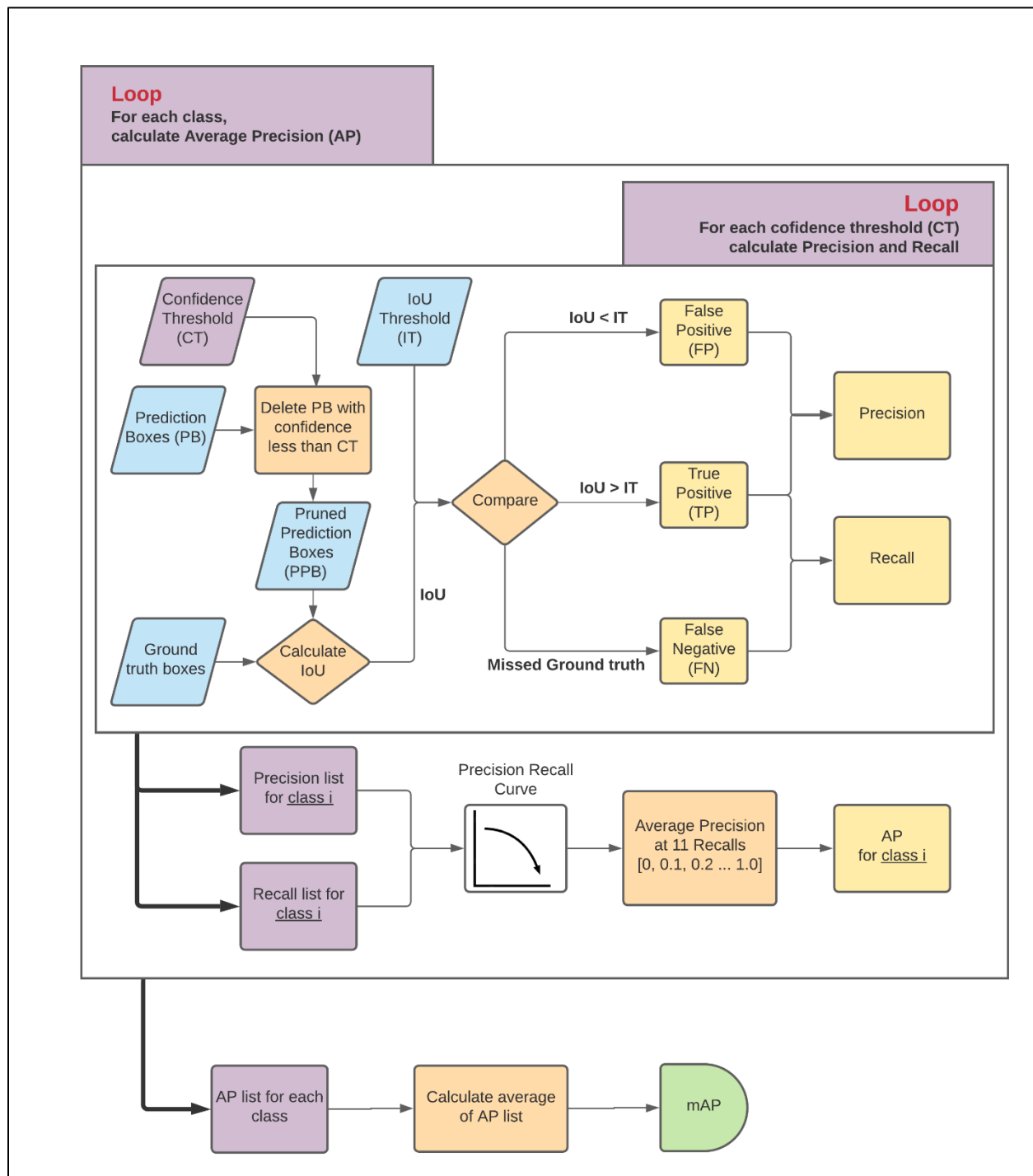


Figure 3.16: Flowchart depicting the process of mean average precision calculation for Object Detection

3.5 Network deployment on edge devices

For deep learning algorithms to be used in fields, they should run on UAS and ground robots. A significant barrier to doing so is the large number of computational resources required to run deep learning models. Research groups have reported using mini-laptops on ground vehicles (Bhong et al., 2020) and tractors (Machleb et al., 2020) for deploying additional computing resources in the field. While mini laptops can work on ground vehicles with a bit of relaxed restriction on weight and power, they won't be feasible in the air on a UAS. Hence, lighter-weight edge devices are required that use limited power from UAS and also do not add a lot to the payload.

Because of the limited computational power of edge devices, deep learning models cannot directly run on them. Rather the networks have to be optimized such that the computations are reduced without a significant reduction in the accuracy of the network. For this research, two devices, i.e., NVIDIA® Xavier NX and NVIDIA® Nano (NVIDIA Corporation, Santa Clara, California, USA), were used to deploy the best network from the object detection and image segmentation group.

3.5.1 Network Optimizations - TensorRT™

NVIDIA® TensorRT™ is a software development kit used for optimizing the deep learning networks for inference (Figure 3.17). It is programmed on Compute Unified Device Architecture (CUDA®) and is compatible with a majority of the deep learning frameworks, including the popular ones like Tensorflow, Pytorch, and Matlab. The optimizer within TensorRT™ uses various strategies to reduce the computational requirements. These strategies are outlined below as well as depicted in Figure 3.17.

1. Reduce Mixed Precision – It reduces the precision of weights and biases of a network from Floating Point 32 (FP32) to Floating Point 16 (FP16) or Integer 8 (INT8). Because each number is represented by fewer bits, the processor's throughput can be increased.
2. Layer and Tensor Fusion – This combines certain layers which often occur in succession into a single layer. This includes layers succession like Convolution- Activation. By reducing the number of layers, the overall computation time can be reduced with this technique.

3. Kernel Auto Tuning – Selects the best algorithm for optimization based on the available GPU memory
4. Dynamic Tensor Memory – Optimizes the size of matrices used during inference for faster calculation
5. Multi-Stream Execution – While not applicable in this study, TensorRT Optimizer is capable of handling multiple input streams, i.e., given enough computational resources are available on the edge device, videos from multiple cameras can be processed in parallel
6. Time Fusion – Optimizes recurrent neural networks (RNN), not used in this study

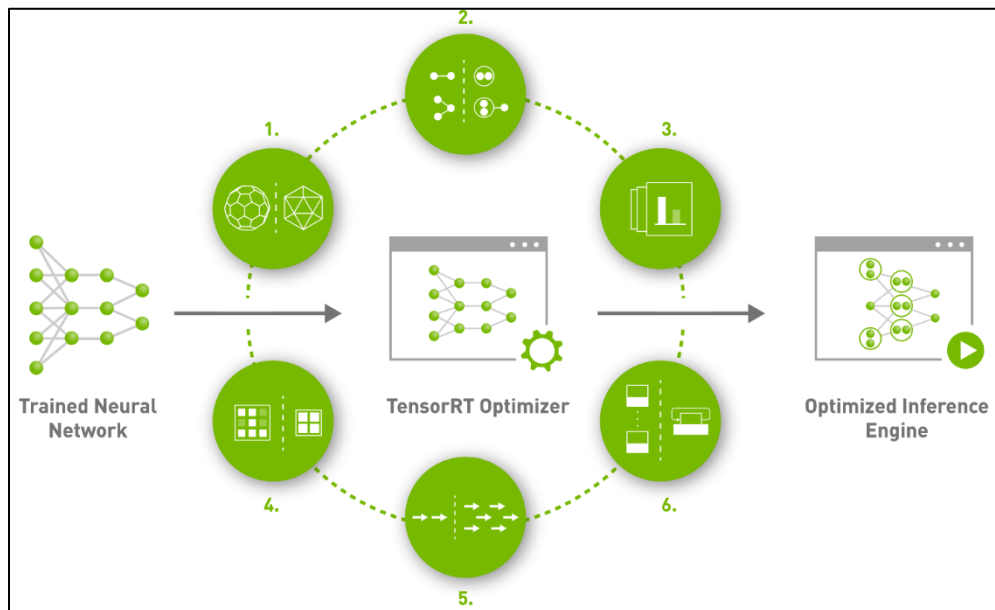


Figure 3.17: TensorRT™ Workflow (NVIDIA, 2021f)

3.5.2 Network Optimizations – tkDNN

tkDNN is a deep neural network library developed in 2020 (Verucchi et al., 2020). It was primarily designed to be used on NVIDIA Jetson™ boards. This library aimed to best utilize the Jetson™ boards to achieve the highest performance. NVIDIA CUDA® Deep Neural Network library (cuDNN) and tensorRT primitives were used to program tkDNN.

3.5.3 Optimization methods for NVIDIA Jetson Nano and NVIDIA Xavier NX

The optimization was applied for the best performing network based on the IoU score among Image segmentation models. For object detection, YOLOv4 was optimized because it is a single-stage detector that is more suitable for deployment on the edge device. The optimization methods are hardware-specific, i.e., a network optimized on Jetson Nano cannot run on Xavier NX. Hence the networks were optimized separately on NVIDIA® Xavier NX and NVIDIA® Nano.

Before either of the networks could be optimized, the networks were converted in Open Neural Network Exchange (ONNX) format. ONNX is an open standard created by a collaboration of technological industries and research groups to facilitate collaboration and interpolation of models in machine learning. An ONNX model could be deployed on an edge device for inference. But it is important to note that an ONNX model is not optimized and would give no improvement in fps compared to an unoptimized network. This is because the ONNX standard was primarily developed as a common communication medium so that other deep learning frameworks like Tensorflow, PyTorch, or Caffe can be converted between each other.

The performance of the models was compared based on fps. It is the number of static image frames that can be processed by the model every second. The higher the fps value, the better the model will be for real-time applications.

3.5.3.1 Object detection optimization process

In the case of object detection, the YOLOv4 network was optimized. First, the inference speed, measured in fps, was calculated for the native YOLOv4 network (unoptimized).

Following this, the network was optimized using tkDNN (Section 3.4.2). The optimization with tkDNN was performed with two different floating-point precision, i.e., FP 16 and FP 32. Hence, two optimized networks were obtained with tkDNN, each used to evaluate the inference speed.

To optimize the YOLOv4 network with tensorRT (Section 3.4.1), the network was first converted to the ONNX model. This was because the native YOLOv4 network cannot be directly converted

into tensorRT. Once the ONNX model for YOLOv4 is obtained, the inference speed of the model is calculated. Finally, the ONNX model was converted into tensorRT engines with two floating-point precision: FP 16 and FP 32. The inference speed for both the tensorRT engines was calculated and reported (Figure 3.18).

3.5.3.2 Image Segmentation optimization process

For image segmentation, a similar object detection strategy was adopted to obtain optimized tensorRT engines of DeeplabV3+. tkDNN optimizations were not undertaken for image segmentation because tkDNN was primarily designed for the YOLO network.

Hence, initially, the inference speed (fps) of the native DeeplabV3+ network was calculated. Next, the native model was converted into the ONNX model. The inference speed of the ONNX network was calculated and reported. Finally, the ONNX model was converted to two tensorRT engines at different floating-point precision (FP 16 and FP 32). For each of the two engines, the inference speed was reported (Figure 3.19).

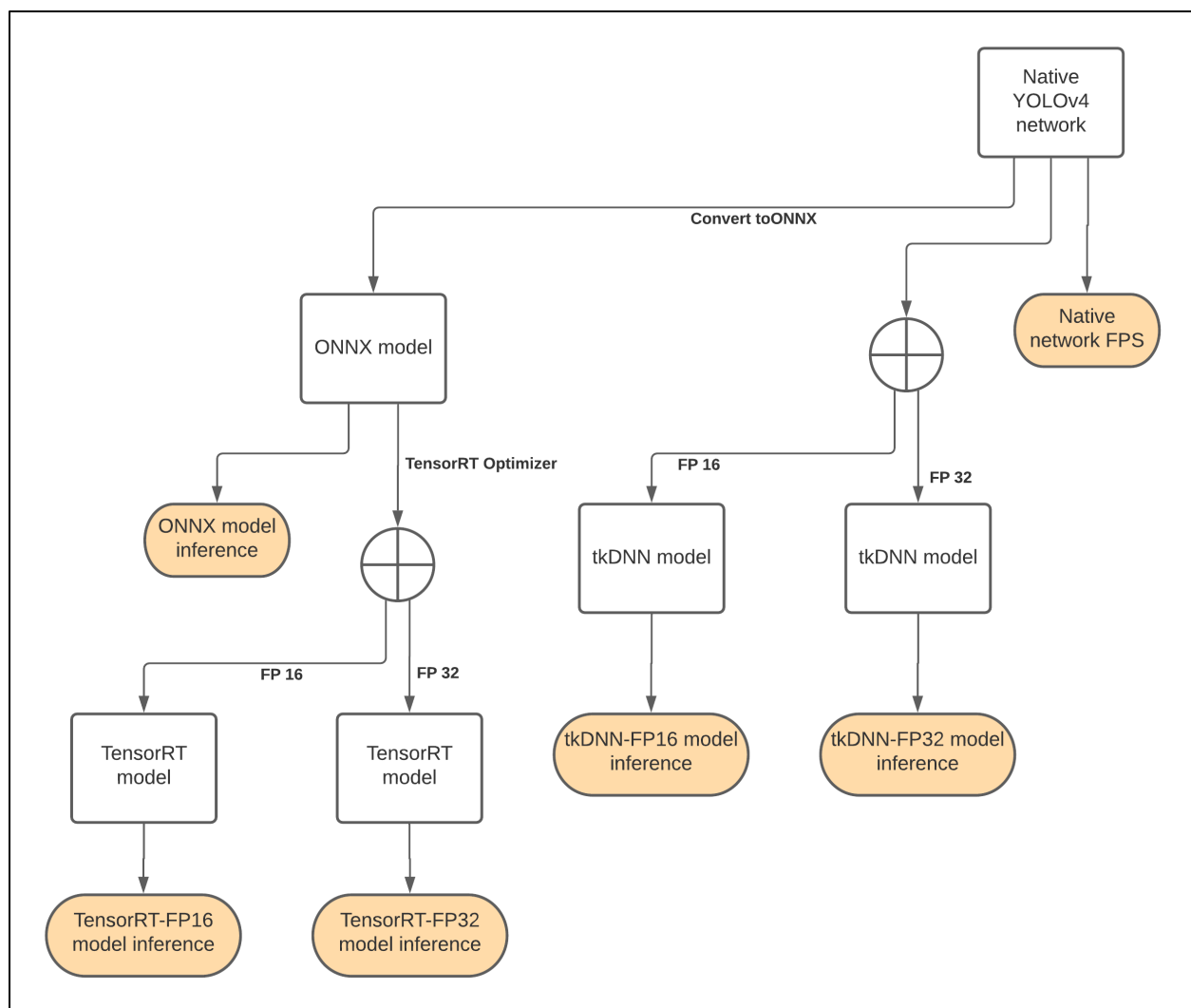


Figure 3.18: YOLOv4 optimization flowchart

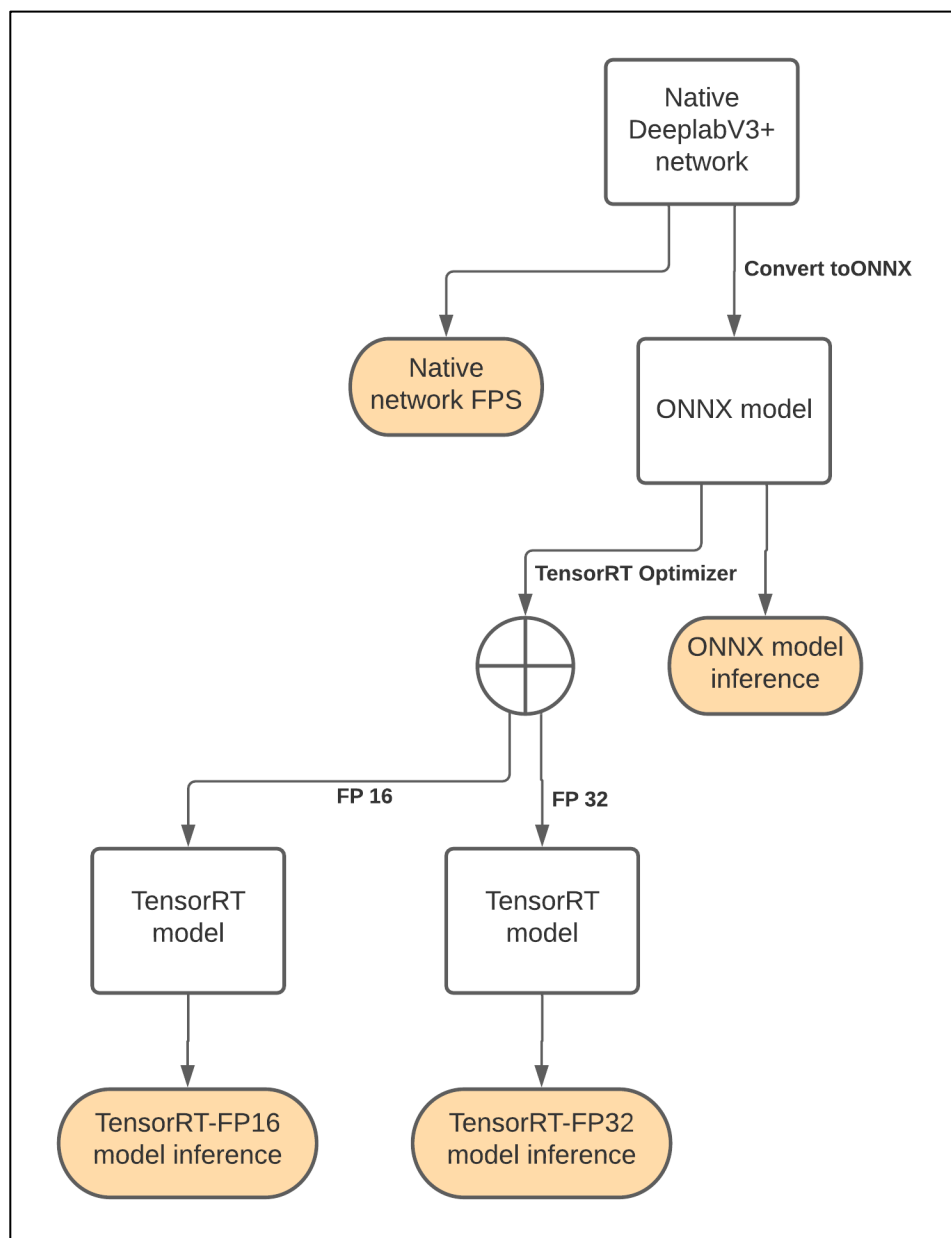


Figure 3.19: DeeplabV3+ optimization flowchart

4. RESULTS

4.1 Results for Object Detection for weed identification

4.1.1 Faster R-CNN - Two-stage object detection

The results for establishing testing accuracy of Faster R-CNN for each of the four dates (Figure 4.1) and the results for setting generalization accuracy of Faster R-CNN over different imaging dates (Table 4.2) are discussed in this subsection.

In the original paper, Faster R-CNN resulted in mAP score of 70.4 % on the PASCAL VOC 2012 image dataset (Ren et al., 2016). In comparison, the testing mAP score for Faster R-CNN for all four-dates was higher than 70.4 %. On June 11, the mAP score was 78.3 %, on June 14 and 16, mAP score was 79.1 %, and on June 18, the mAP score was 79.5 %. The highest score was obtained on 18th June. The bounding boxes of weeds was approximately four times larger on 18th June compared to bounding boxes of weeds on 11th June. The larger bounding box is the result of the larger size of the weeds and crop (Table 4.1). The average height of weeds on June 18 was approximately two times compared to June 11.

Table 4.1: Growth stage of Soybean and average weed height for Palmer amaranth on Data Collection dates for Object Detection

Date	Crop Growth Stage	Average Weed Height (AWH)
June 11, 2020	Unifoliate Stage	3.81 cm
June 14, 2020	V1 stage	6.35 cm
June 16, 2020	V1 stage	7.62 cm
June 18, 2020	V2 stage	8.89 cm

Essentially, a larger bounding box enables the network to extract more features for identifying weeds (J. Li et al., 2017; Tong et al., 2020). Additionally, distinguishing weeds from the background and other plants in the image, like crops, also becomes easier if more features represent weeds.

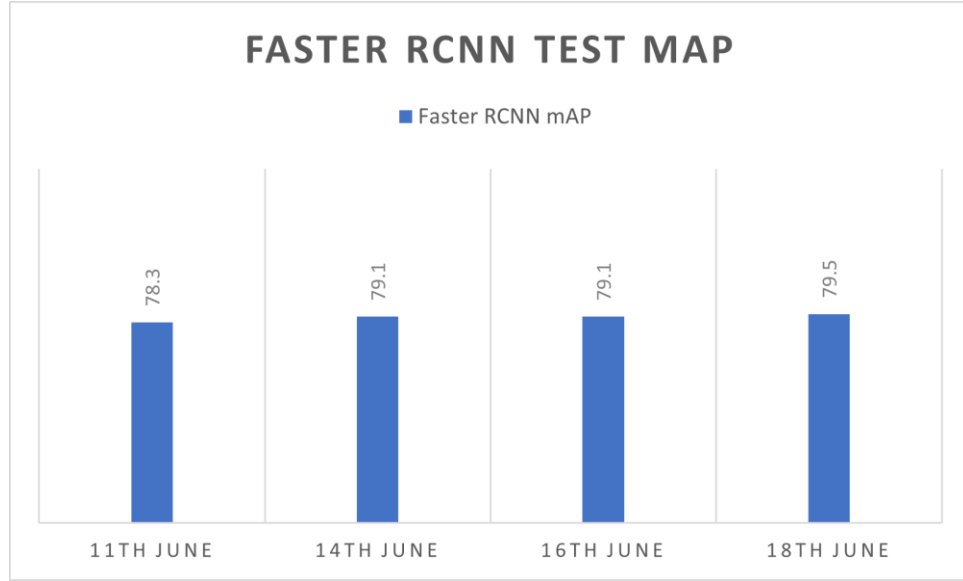


Figure 4.1: Object Detection testing mAP score for Faster R-CNN

Table 4.2: Generalization mAP score of Faster RCNN networks

Faster RCNN (mAP %)		Testing Data Date			
		11 th June	14 th June	16 th June	18 th June
Training Data Date	11 th June	-	75.3	75.2	75.2
	14 th June	75.4	-	75.3	75.2
	16 th June	75.1	75.2	-	75.2
	18 th June	75.1	75.3	75.2	-

The first Faster R-CNN network for which the generalization accuracy was accessed was trained on data from June 11. It achieved a mean mAP score of 75.23 % with a standard deviation of 0.06 % for June 14, 16, and 18. Compared to testing accuracy, the generalization accuracy for the 11th June Faster RCNN model was reduced by around 3.1 %. In an ideal case, the generalization accuracy should be equal to testing accuracy, although, in practical applications, generalization accuracy is seldom equal to testing accuracy. Yet, the target is to reduce the gap between the two (Kawaguchi et al., 2017; Neyshabur et al., 2017; L. Wu et al., 2017).

The detection results for the June 11 Faster R-CNN network on a single patch of the field with Palmer amaranth weed are shown in Figure 4.2.

The second Faster R-CNN network for the generalization experiment was trained on images from 14th June. It achieved a mean mAP score of 75.3 % with a standard deviation of 0.1 % for June 11, 16, and 18. Additionally, the difference between testing and generalization mAP score is approximately 3.8 %. The detection results for the June 14 Faster R-CNN network on a single patch of the field with Palmer amaranth weed are shown in Figure 4.2.

The third Faster R-CNN network was trained on images from June 16. It achieved a mean mAP score of 75.17 with a standard deviation of 0.06% for June 11, 14, and 18. Generalization accuracy reduced by approximately 3.9 % with June 16 Faster R-CNN network, similar to June 14 Faster R-CNN model. The detection results for the June 16 Faster R-CNN network on a single patch of the field with Palmer amaranth weed is shown in Figure 4.2.

The final Faster R-CNN network was trained on images from 18th June. The mean mAP score was 75.2 %, with a standard deviation of 0.1% for June 11, 14, and 16, respectively. For the final Faster R-CNN model, the generalization accuracy was reduced by around 4.3 %. The detection results for the June 11 Faster R-CNN network on a single patch of the field with Palmer amaranth weed is shown in Figure 4.2.

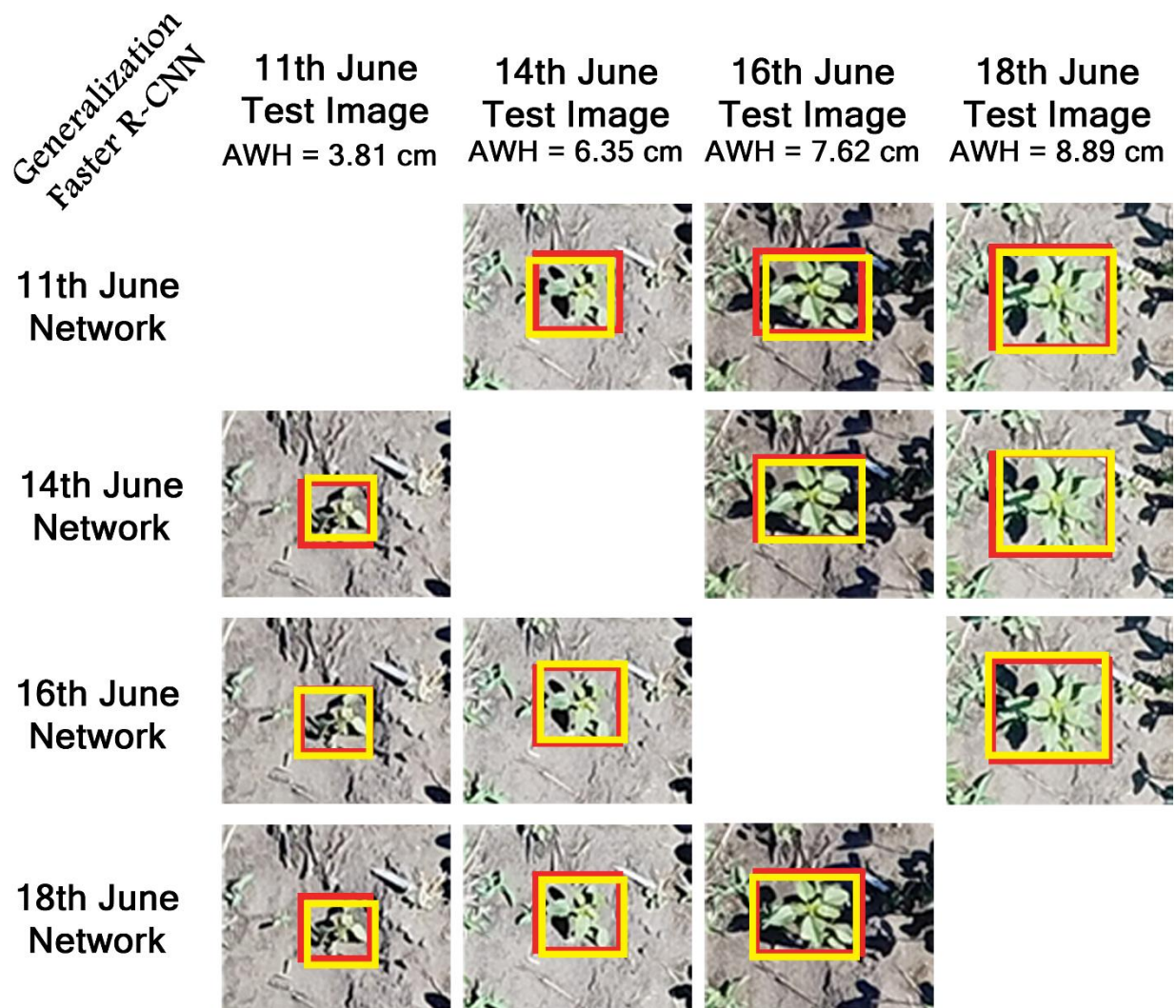


Figure 4.2: Faster R-CNN generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer amaranth. The RED bounding box is the ground truth, while the YELLOW bounding box is the prediction.

The mean and the standard deviation for the generalization mAP score of the four Faster R-CNN models can be seen in Figure 4.3. The four Faster RCNN networks, when compared amongst each other, did not show a particular trend for generalization mAP. Although, compared to testing mAP, the generalization mAP decreased in each of the four Faster R-CNN models. This could be because the Faster R-CNN did not generalize too well on the scale and height of weeds in the image (The scale of weeds in the image refers to the ratio of total bounding box pixels for a weed to the total number of pixels in the image). Between the test images, the scale of the weeds is very evident

(Figure 4.2). Despite the reduction in generalization mAP score, visually, the detection is only slightly displaced from the bounding box.

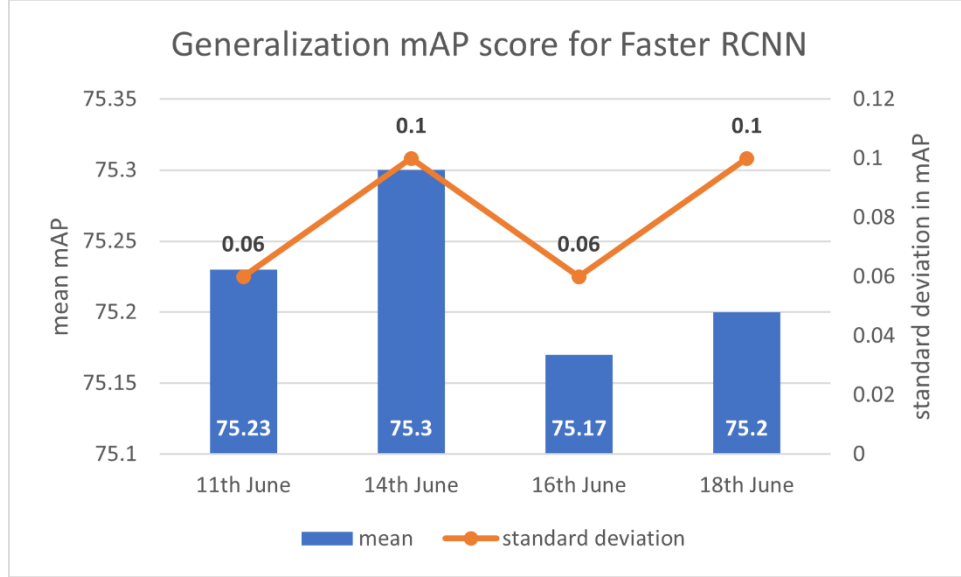


Figure 4.3: Generalization mAP score for Faster RCNN

4.1.2 YOLOv4 – Single-stage object detection

Similar to Faster R- CNN, the detection results of YOLOv4 on weed identification from UAS images are presented in this subsection. The results for establishing the testing accuracy of YOLOv4 for each of the four dates are given in Figure 4.4. The results for the second test on determining the generalization accuracy of YOLOv4 over different imaging dates are given in Table 4.3.

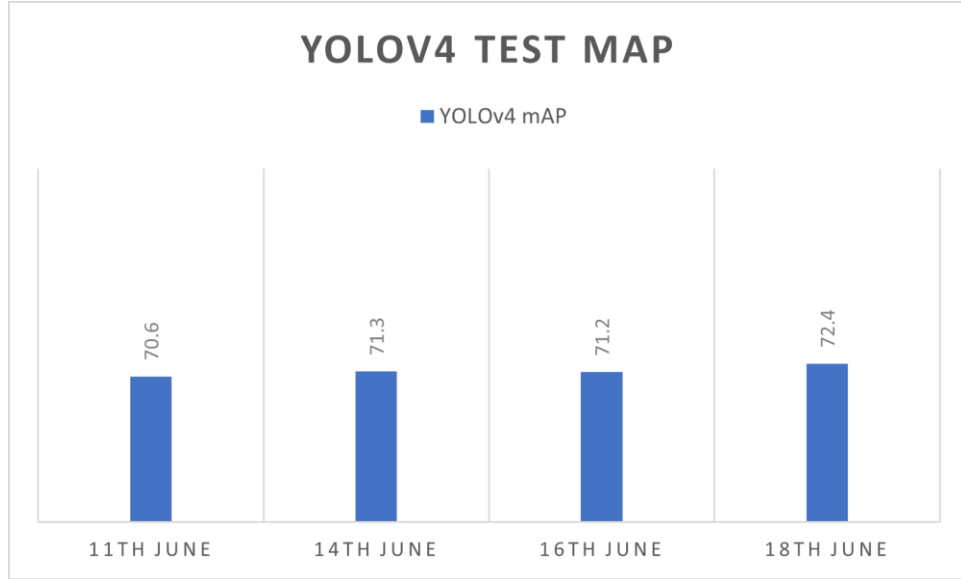


Figure 4.4: Object Detection testing mAP score for YOLOv4

Table 4.3: Generalization mAP score of YOLOv4 networks

YOLOv4 (mAP %)		Testing Data Date			
		11 th June	14 th June	16 th June	18 th June
Training Data Date	11 th June	-	69.4	69.1	68.3
	14 th June	70.5	-	69.4	69.4
	16 th June	70.2	70.4	-	70.4
	18 th June	69.8	70.5	70.3	-

The baseline for the mAP score for YOLOv4 was set based on the mAP score obtained in the original paper on the MS COCO dataset. Testing mAP score of 65.7 % was obtained in the original YOLOv4 paper (Bochkovski et al., 2020a). In the current research, the mAP score is higher than the baseline of 65.7 %. YOLOv4 network trained on June 11 resulted in a test mAP score of 70.6 %. When trained on images from June 14 June, mAP score of 71.3 % was observed. The test mAP score for the YOLOv4 network was reduced to 71.2 % on June 16 but increased to 72.4 % on

image data from June 18. In the case of YOLOv4, the difference in test mAP score between June 18 and 11 networks is 1.8 %. This would be because bounding boxes of weeds occupy around four times more pixels on June 18th June than on June 11th June. The difference in the size of the bounding box is primarily due to the difference in the height of weeds and the crop's growth stage (Table 4.1). YOLOv4 gives a higher mAP score of large objects, as it was shown in the original paper. For large objects in the MS COCO dataset, the YOLOv4 mAP score was 56.0 %, while the mAP score was only 24.3 % for small objects (Bochkovskiy et al., 2020a).

For the generalization experiments, a YOLOv4 network trained on June x, 2020 would be referred to as June x YOLOv4 network. The four networks were June 11 YOLOv4 network, 14 June YOLOv4 Network, 16 June YOLOv4 Network and 18 June YOLOv4 Network.

June 11 YOLOv4 network achieved a mean generalization mAP for June 14, 16, and 18 of 68.93 % with a standard deviation of 0.57 %. The reduction in mAP score from testing accuracy for June 11 YOLOv4 network was 1.7 %. The detection results for Palmer amaranth on the same patch of the field on June 14, 16, and 18 can be seen in Figure 4.5.

The mean generalization mAP for June 14 YOLOv4 network for June 11, 16 and, 18 was 69.77 %, with a standard deviation of 0.64 %. For the June 14 model, the mean generalization mAP was reduced by 1.5 % compared to the testing mAP score. The detection results for Palmer amaranth on the same patch of the field on June 11, 16, 18 can be seen in Figure 4.5.

The third YOLOv4 network, i.e., June 16 June YOLOv4 network was tested on June 11, 14, and 18 networks. The mean generalization mAP score of 70.33 % was observed with a standard deviation of 0.16 %. Compared to testing mAP, the generalization mAP for June 16 YOLOv4 was reduced by 0.9 %. The detection results for Palmer amaranth on the same patch of the field on June 11, 14, 18 can be seen in Figure 4.5.

The fourth and final YOLOv4 network was trained on images from June 18. When the network was tested in images from June 11, 14, and 16, a mean generalization mAP of 70.2 % was obtained with a standard deviation of 0.36 %. Additionally, the difference between testing and

generalization mAP score is approximately 2.2 %. The detection results for Palmer amaranth on the same patch of the field on June 11, 14, and 16 can be seen in Figure 4.5.

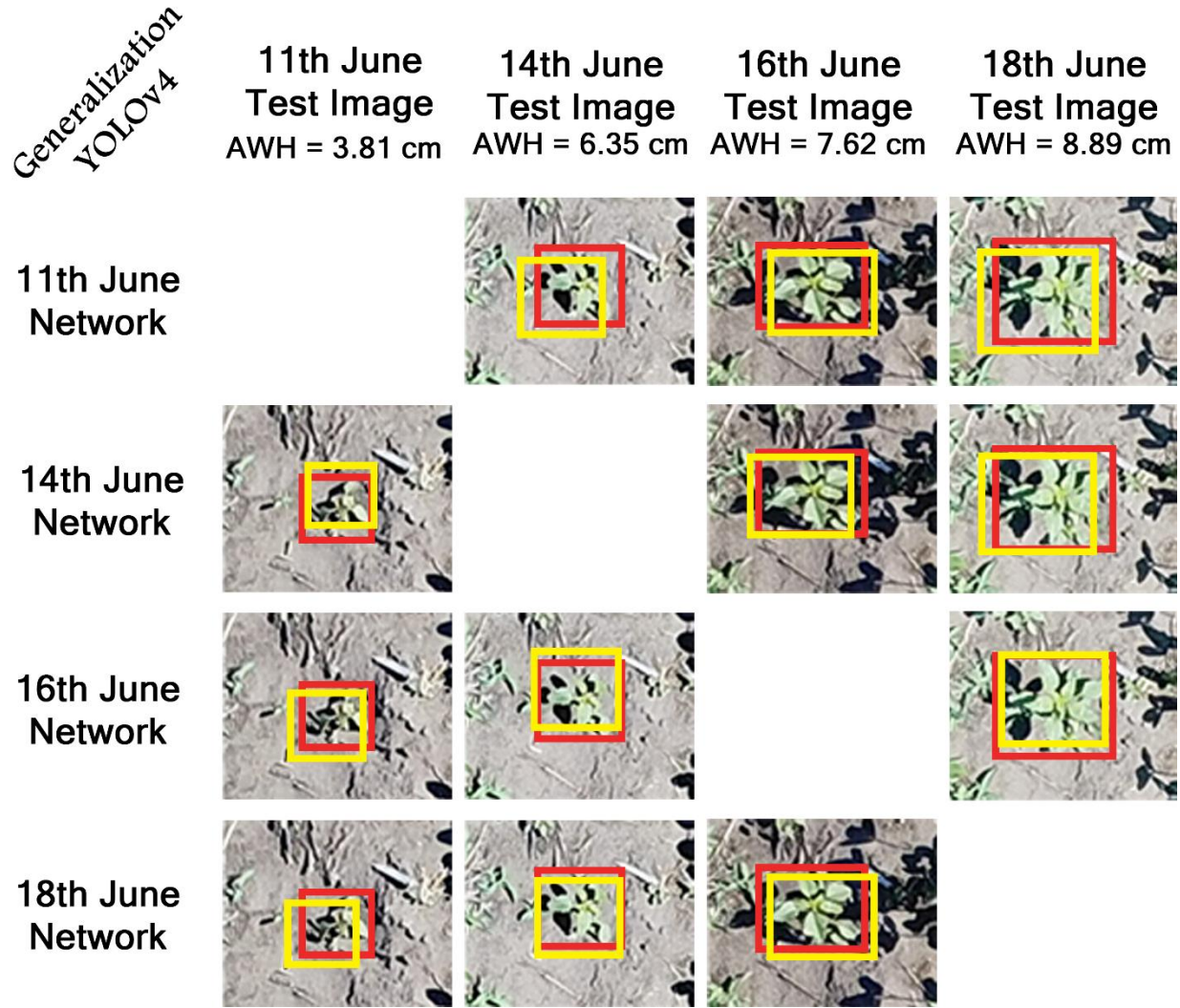


Figure 4.5: YOLOv4 generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer amaranth. The RED bounding box is the ground truth, while the YELLOW bounding box is the prediction.

The combined mean generalization mAP for the four trained YOLOv4 networks is given in Figure 4.6. Out of the four YOLOv4 networks, maximum mean generalization mAP score was obtained for June 16 YOLOv4 network, and the least standard deviation was also obtained for June 16 YOLOv4 network. This shows that June 16 YOLOv4 network was the best performing out of the

four networks. Additionally, there is an overall increase in mAP scores between June 11 and June 18 YOLOv4 networks.

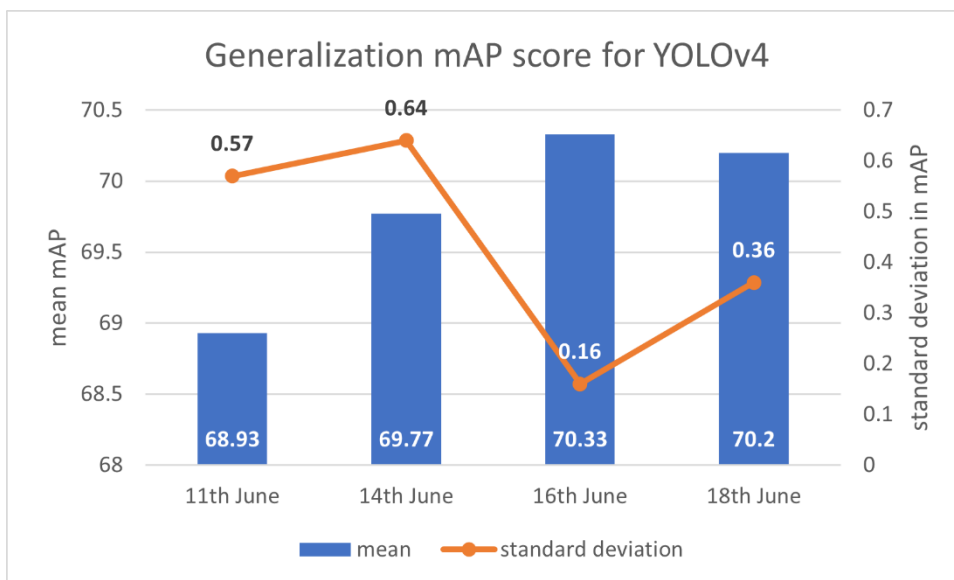


Figure 4.6: Generalization mAP score for YOLOv4

4.1.3 Comparison of Faster RCNN and YOLOv4 for weed identification

In terms of testing the mAP score, Faster RCNN performed better compared to the YOLOv4 model for every crop stage and weed height (Figure 4.7). A higher mAP shows that the two-stage detector is better at weed identification compared to single-stage detector. This could be explained by the presence of the Region Proposal Network in two-stage detectors. Having a separate region proposal and detection network, makes it easier for each of the two stages to optimize in respective tasks of region proposal and detection.

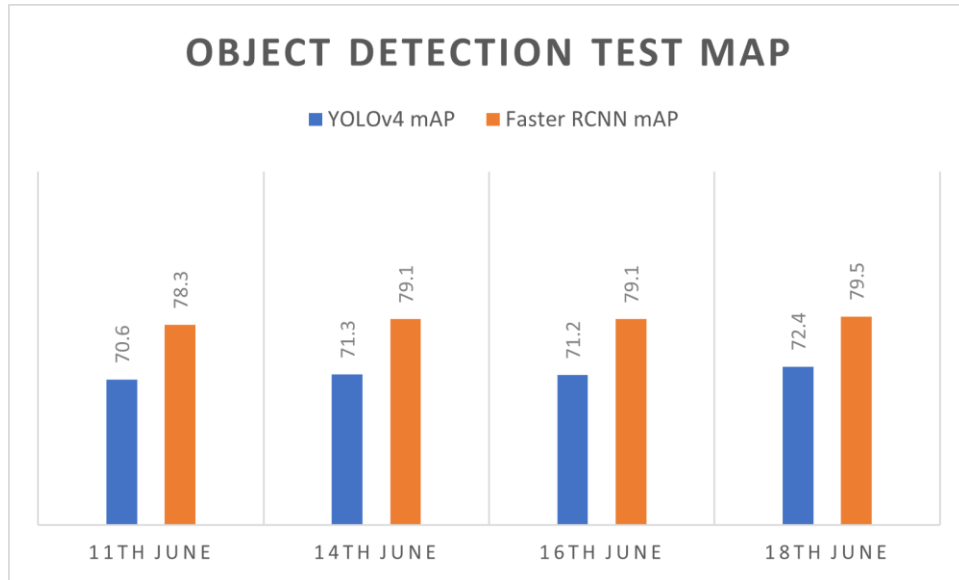


Figure 4.7: Comparison of testing mAP score of Faster RCNN and YOLOv4

Although, the larger network size of Faster RCNN has drawbacks which would prevent it from being the network of choice to be deployed on the edge device. First, it takes 50 % more time to train Faster RCNN compared to YOLOv4. On Gilbreth Cluster, a single YOLOv4 network took around 24 hours to train, while a single Faster RCNN network took 36 hours. Additionally, at the time of inference, Faster RCNN took 1.5 times longer than YOLOv4 to detect the objects in an image. In most cases, because networks could be trained on powerful servers or cloud computers, training time isn't crucial in determining the application of the network on an edge device. But the inference time is a key consideration factor for deploying any deep learning network on an edge device.

The generalization mAP score for Faster RCNN was higher than YOLOv4 for all the dates. This again could be attributed to two separate stages for region proposal and detection in Faster RCNN. Nonetheless, the difference between the generalization mAP score and testing mAP score is negligible for YOLOv4. By speculation, YOLOv4 learns the scale of weeds and thus the height of the weed in the image far better than Faster RCNN. This is advantageous, especially in UAS applications where the scale of weed in the image is directly related to the flight height. And due to the good generalizability of YOLOv4 for weed scale, slight changes in the flight height during field imaging should not adversely affect the performance of weed identification.

The detection results for the 16th June YOLOv4 network in some challenging situations promise its application for weed identification (Figure 4.8). It detected overlapping Soybean and detected both Soybean and Palmer amaranth even in situations when occlusion was occurring due to external objects like the electric fence set up around the field. Moreover, the YOLOv4 network was able to detect in-row weeds and was even able to generalize over the height of Palmer amaranth with minimal standard deviation (Figure 4.6).

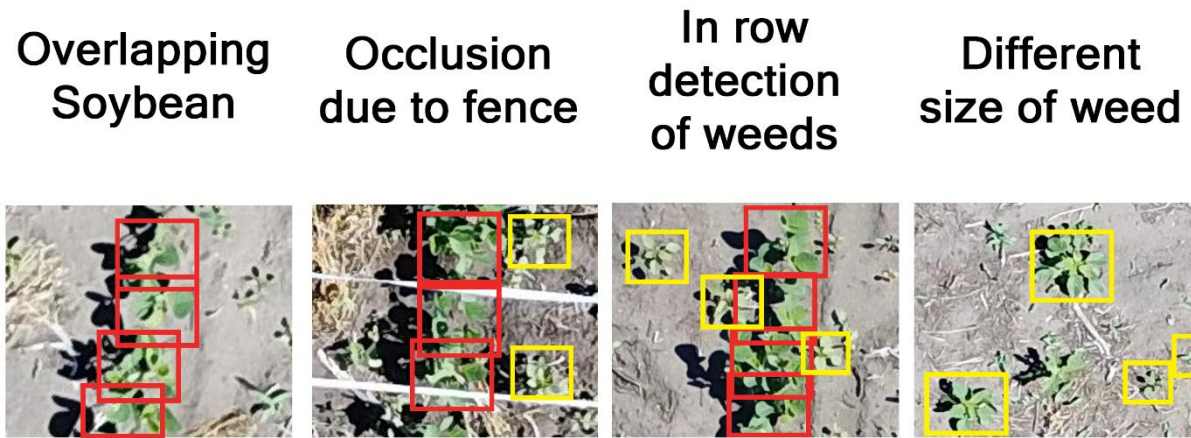


Figure 4.8: YOLOv4 detection results in hard situations. RED bounding box represents Soybean while YELLOW bounding box represent Palmer amaranth.

4.2 Results for Image Segmentation

4.2.1 UNET segmentation results

Summary of results for the two tests a) the testing accuracy of UNET (Figure 4.9) b) the generalization accuracy of UNET (Table 4.5).

UNET, when released in 2015 (Ronneberger et al., 2015b), reported mean IoU scores on two custom cell datasets which were part of the ISBI cell tracking challenge 2014 and 2015 (Arganda-Carreras et al., 2015; M et al., 2014). The first dataset was called “PhC-U373” which contained images of Glioblastoma-astrocytoma U373 cells on a polyacrylimide substrate recorded by phase-contrast microscopy. UNET achieved an IoU score of 92.3 % on this dataset. The second dataset was called “DIC-HeLa” which contained the image of HeLa cells on a flat glass captured by

differential interference contrast microscopy. UNET achieved an IoU score of 77.56 %. Because the dataset used in the current research was significantly different than either of the two cells datasets used in the original study, a direct comparison between the IoU scores cannot be drawn. The capability of the UNET model can still be inferred for its applicability to the current research that resulted in a maximum mean IoU score of 73.5 %.

The mean testing IoU of the eight UNET networks trained on data from June 11 to 18, in order, are as follows: 63.4, 65.3, 67.4, 64.8, 70.6, 71.1, 73.5. The mean IoU increases as the days progress from 11th June to 18th June. This is primarily because the height of Palmer amaranth and size of the Soybean (due to the growth stage of Soybean) increases as the days progressed (

Table 3.3). When the IoU scores for weeds and crops are analyzed separately, it was observed that the mean IoU score for Palmer amaranth ranged between 48-67 % while IoU score for Soybean remained consistent, between 78-79 % (Figure 4.9). This showed that UNET had no trouble identifying Soybean. But when identifying Palmer amaranth, the effect of imaging date and hence the weed height had a major impact on segmentation.

Table 4.4: Growth stage of Soybean and average weed height for Palmer amaranth on Data Collection dates for Object Detection

Date	Crop Growth Stage	Average Weed Height (AWH)
June 11, 2020	Unifoliate Stage	3.81 cm
June 12, 2020	Unifoliate Stage	3.81 cm
June 13, 2020	Unifoliate Stage	5.08 cm
June 14, 2020	V1 stage	6.35 cm
June 15, 2020	V1 stage	6.35 cm
June 16, 2020	V1 stage	7.62 cm
June 1,7 2020	V1 stage	7.62 cm
June 1,8 2020	V2 stage	8.89 cm

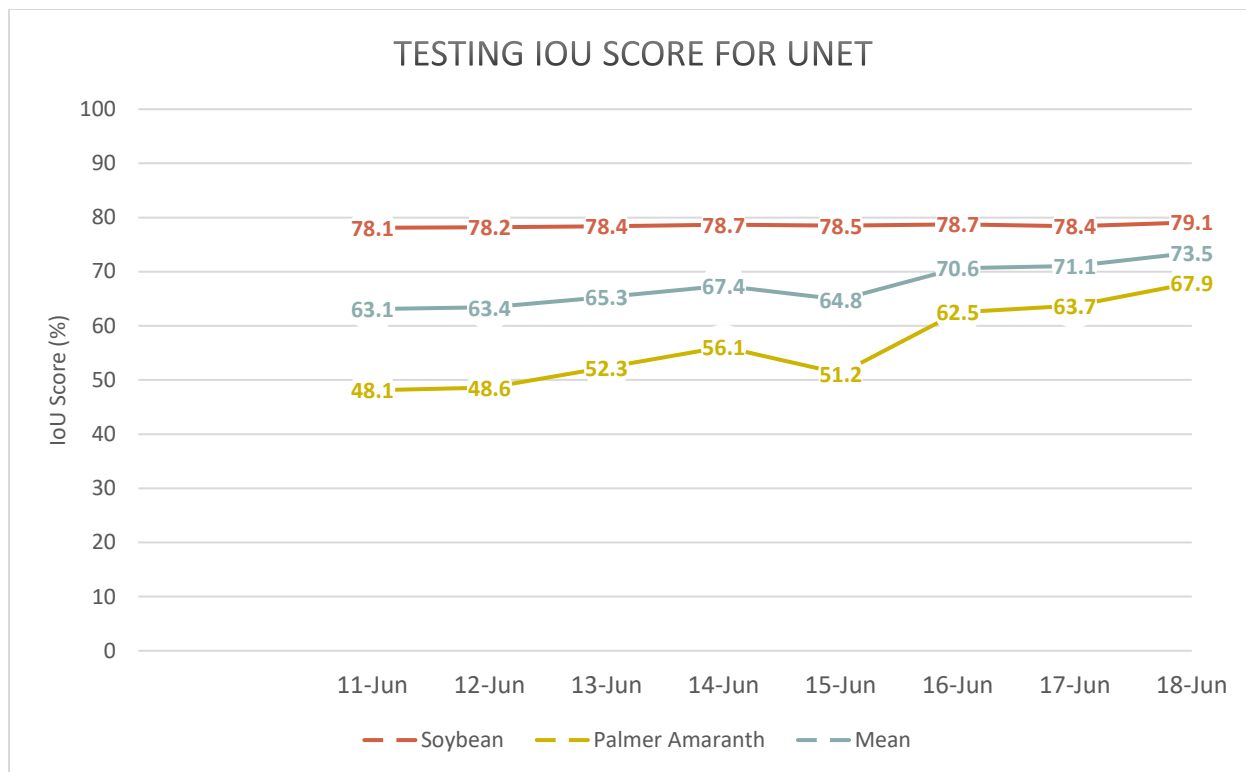


Figure 4.9: IoU score for trained UNET models

For the generalization tests, the UNET network trained on image data from June X, 2020 would be written as of June X UNET network. The four networks were June 11 UNET network, 14 June UNET Network, 16 June UNET Network and 18 June UNET Network.

The June 11 UNET network resulted in a mean generalization IoU score for Palmer amaranth on June 14, 16, and 18 images as 32.4 % with a standard deviation of 0.95 %. Compared to the testing IoU score, the generalization IoU score for June 11 UNET network was reduced by 15.7 %.

The generalization IoU score for June 14 UNET network was calculated for June 11, 16 and 18. The mean generalization IoU score for Palmer amaranth was obtained as 49.3 %, with a standard deviation of 7.02 %. Additionally, the difference between mean generalization IoU and testing IoU was observed to be 6.8 %.

For the June 16 UNET network, the mean generalization IoU score for Palmer amaranth was 47.6 %, with a standard deviation of 11.37 %. Compared to June 14 UNET network, the mean generalization IoU score decreased while the standard deviation increased. This indicates that June

16 UNET network was not very good for weed height generalization compared to June 14 UNET model. Moreover, the generalization IoU score was reduced by 14.9 % compared to the testing IoU score.

The last network for which the generalization test was conducted was the June 18 UNET network. The mean generalization IoU score was 49.2 %, with a standard deviation of 15.56 %. Compared to the testing IoU score, the mean generalization IoU score reduced by 18.7 %. This was the highest reduction IoU score among the four generalization networks.

Out of the four generalization networks, the best network for generalizing the segmentation of Palmer amaranth was the 14th June UNET network. This was because it had the highest mean generalization IoU score (49.3 %) and the least standard deviation (7.02 %). Moreover, the difference between the mean generalization IoU score and testing IoU score was minimum for the 14th June model (6.8 %). Regardless, the generalization performance for UNET wasn't very promising (Table 4.5, Figure 4.10).

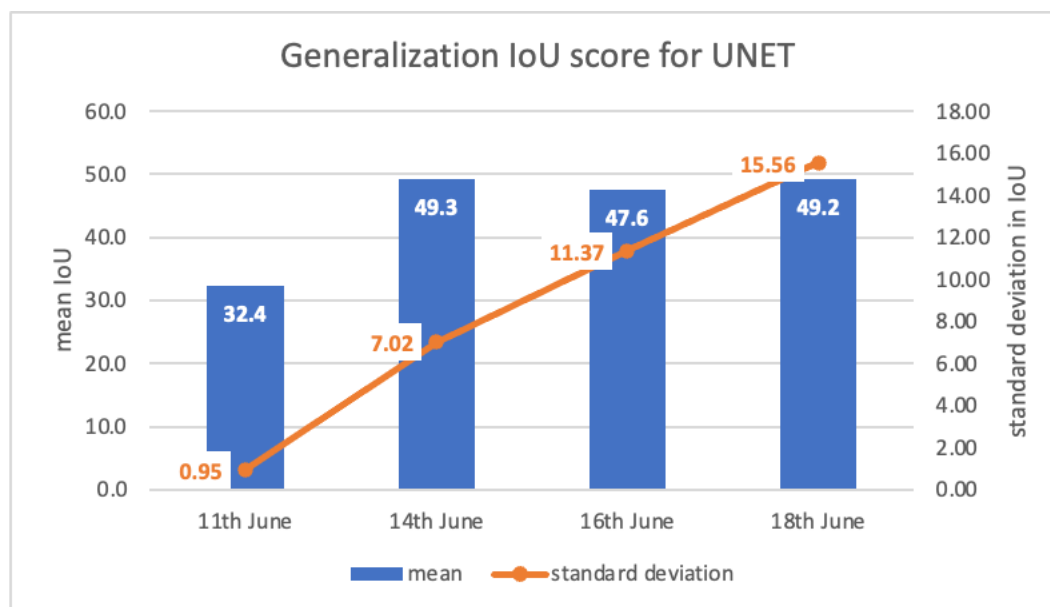


Figure 4.10: UNET's generalization IoU score for Palmer amaranth

Table 4.5: UNET results for generalization tests

UNET (IoU % for Palmer Amaranth)		Testing Data Date			
		11 th June	14 th June	16 th June	18 th June
Training Data Date	11 th June	-	32.4	31.5	33.4
	14 th June	43.5	-	57.1	47.3
	16 th June	35.6	49.1	-	58.2
	18 th June	33.4	49.7	64.5	-

4.2.2 DeeplabV3+ segmentation results

Similar to UNET, two tests were performed to test a) testing accuracy of DeeplabV3+ (Figure 4.11) b) generalization accuracy of DeeplabV3+ (Figure 4.12).

DeeplabV3+ network achieved a baseline IoU score of 89.0 % on PASCAL VOC 2012 dataset and 82.1 % on the Cityscapes dataset (L.-C. Chen et al., 2018). Although, the reported IoU score was with the Xception-JFT backbone (Chollet, 2017). With the Resnet-101 backbone, the reported IoU score was 80.57 % on PASCAL VOC 2012 dataset. In this research, the DeeplabV3+ network uses Resnet-50 as the backbone. While a direct comparison was not possible with the results reported in the original paper, some other implementations of DeeplabV3+ are available on Github (Fang, 2019). With Resnet-50 as the backbone, DeeplabV3+ showed a mean IoU score of 77.2 %. In this research, the mean IoU score for testing for the eight trained UNET networks ranged between 67 and 79 %.

The mean testing IoU of the eight DeeplabV3+ networks trained on data from June 11 to 18 in order is as follows: 67.1, 65.9, 69.8, 70.8, 68.9, 74.7, 77.0, 78.1. An overall trend of increasing IoU scores with each passing day was observed. This again would result from the increasing height of Palmer amaranth and increasing size of Soybean owing to progressing Soybean growth stage (

Table 3.3). As with UNET, when analyzing individual IoU scores for Soybean and Palmer amaranth, it was observed that the IoU score for Palmer amaranth was much lower than Soybean. In fact, it ranged between 52 – 69 %. On the other hand, the IoU for Soybean remained above 81 % for all the dates (Figure 4.11). Hence, it could be concluded that the DeeplabV3+ could easily segment Soybean regardless of its growth stage. At the same time, its segmentation capability for Palmer amaranth was directly proportional to the height of Palmer amaranth.

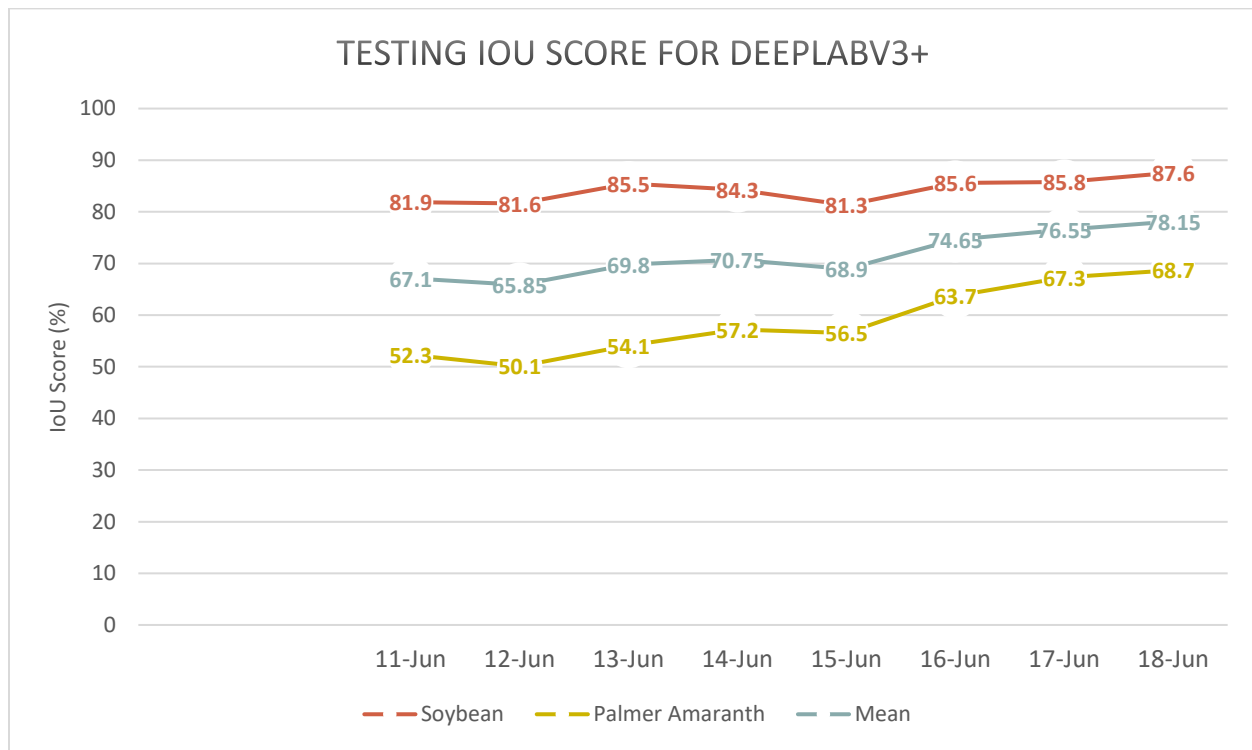


Figure 4.11: IoU score results for trained DeeplabV3+

For the generalization tests, the DeeplabV3+ network trained on image data from June X would be written as June X DeeplabV3+ network. The four networks were June 11 DeeplabV3+ network, 14 June DeeplabV3+ Network, 16 June DeeplabV3+ Network and 18 June DeeplabV3+ Network. The June 11 DeeplabV3+ network was evaluated on images from June 14, 16, 18. The mean generalization IoU score of 35.5 % was observed for Palmer amaranth with a standard deviation of 2.42 %. Compared to testing the IoU score for the June 11 network, the mean generalization IoU score for the network was reduced by 12.6 %.

The mean generalization IoU score for June 14 DeeplabV3+ network was observed to be 52.3 %. The network was tested on images from June 11, 16 and 18. Moreover, the standard deviation for generalization was 4.40 %. Additionally, the mean generalization IoU score was reduced by 3.8 % compared to the testing IoU score for June 14 network.

For June 16 DeeplabV3+ network, the mean generalization IoU score for Palmer amaranth was observed to be 51.5 %, with a standard deviation of 5.44 %. The difference between the generalization IoU score and testing IoU score for June 16 DeeplabV3+ network was 11.0 %.

The final network for which the generalization test was conducted was the June 18 DeeplabV3+ network. The mean generalization IoU score for this network was 50.8 %, with a standard deviation of 13.05 %. Compared to the testing IoU score, the mean generalization IoU score reduced by 17.1 %. This was the highest reduction IoU score among the four generalization networks. It could be indicative of the fact that training DeeplabV3+ on images of higher Palmer amaranth would be detrimental to the network capability of detecting shorter Palmer amaranth.

Out of the four DeeplabV3+ generalization networks, the network from 14th June performed the best. This is indicated by a) highest mean generalization IoU score for Palmer amaranth (52.3), b) second least standard deviation in generalization IoU score (4.40 %), and c) least reduction in generalization IoU score from the testing IoU score (3.8 %) (Figure 4.12, Table 4.6, Figure 4.9).

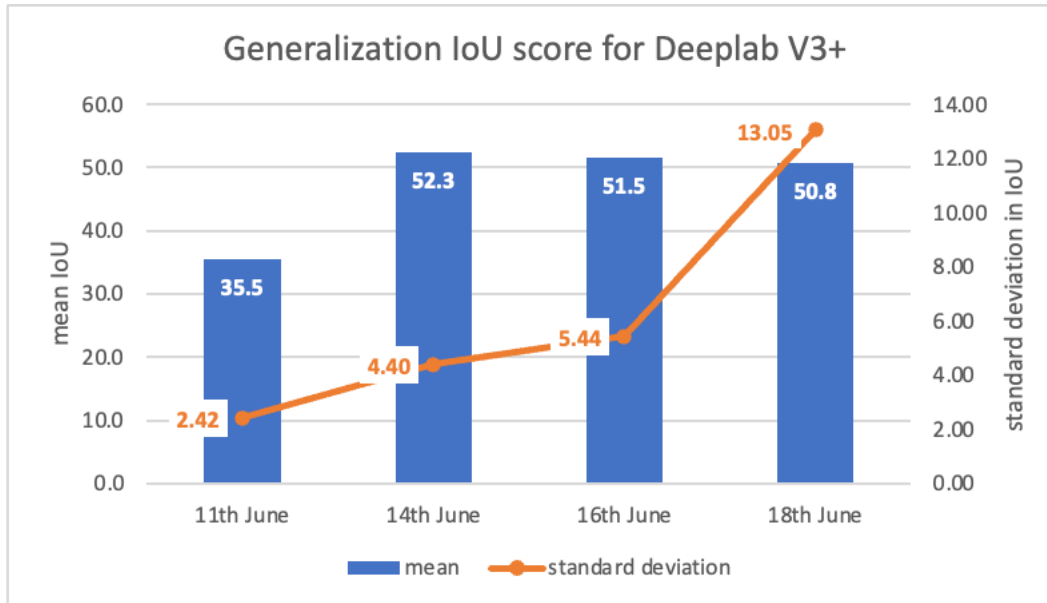


Figure 4.12: DeeplabV3+'s generalization IoU score for Palmer amaranth

Table 4.6: DeepLabV3+ results for generalization tests

DeepLabV3+ (IoU % for Palmer Amaranth)		Testing Data Date			
		11 th June	14 th June	16 th June	18 th June
Training Data Date	11 th June	-	37.7	32.9	35.8
	14 th June	48.9	-	57.3	50.8
	16 th June	46.5	54.9	-	59.4
	18 th June	38.8	49.0	64.7	-

4.2.3 Comparison of UNET and DeeplabV3+ segmentation networks

The generalization IoU scores for UNET and DeeplabV3+ are tabulate in Table 4.5 and Table 4.6 respectively. Based on generalization IoU scores alone, DeeplabV3+ was a better segmentation network than UNET. This can be attributed to the fact that DeeplabV3+'s decoder refines the segmentation along the object's edges. This phenomenon can also be observed by comparing the generalization images for UNET (Figure 4.13) and DeeplabV3+ (Figure 4.14). DeeplabV3+ has smoother edges that match closely to the ground truth compared to UNET.

Moreover, because we are dealing with shorter weeds in the early season, due to a lack in sensor resolution, the edges are not captured well. Hence, the segmentation of Palmer amaranth was poor regardless of the segmentation network. The IoU score for the 11th June test image was reduced drastically (Figure 4.13 and Figure 4.14-Row 1(Image)).

Although, because of the large size of DeeplabV3+, it took longer to train than the UNET network. DeeplabV3+ took around 42 hours to train, while the UNET model took close to 31 hours to train. Hence, if training time is a consideration, UNET would be a more suitable network. Additionally, DeeplabV3+ took three times more time to process input images during inference due to its larger size.

DeeplabV3+ was found more suitable for early season weed segmentation, as seen in the generalization results and detection images despite the disadvantage of longer computation time. Moreover, on June 14, DeeplabV3+ (i.e., trained on AWH of 6.35 cm) performed the best among the eight generalized networks (four UNET networks plus four DeeplabV3+ networks). In summary, the results indicated that the best generalizing network for weed segmentation from UAS images would be the DeeplabV3+ network trained on images with an AWH of 6.35 cm.

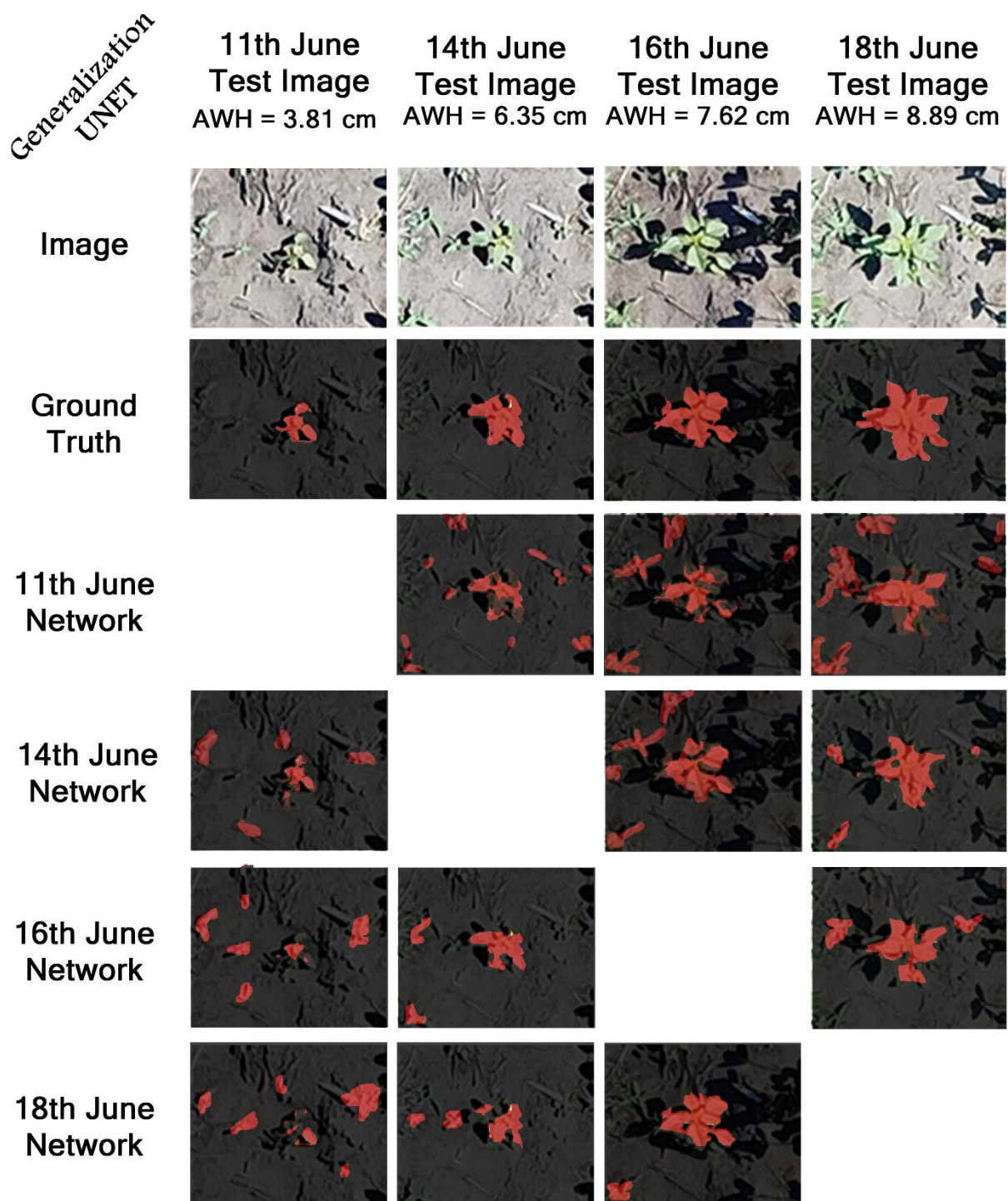


Figure 4.13: UNET generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer amaranth. The RED color in represents Palmer amaranth, and the black color represents the background

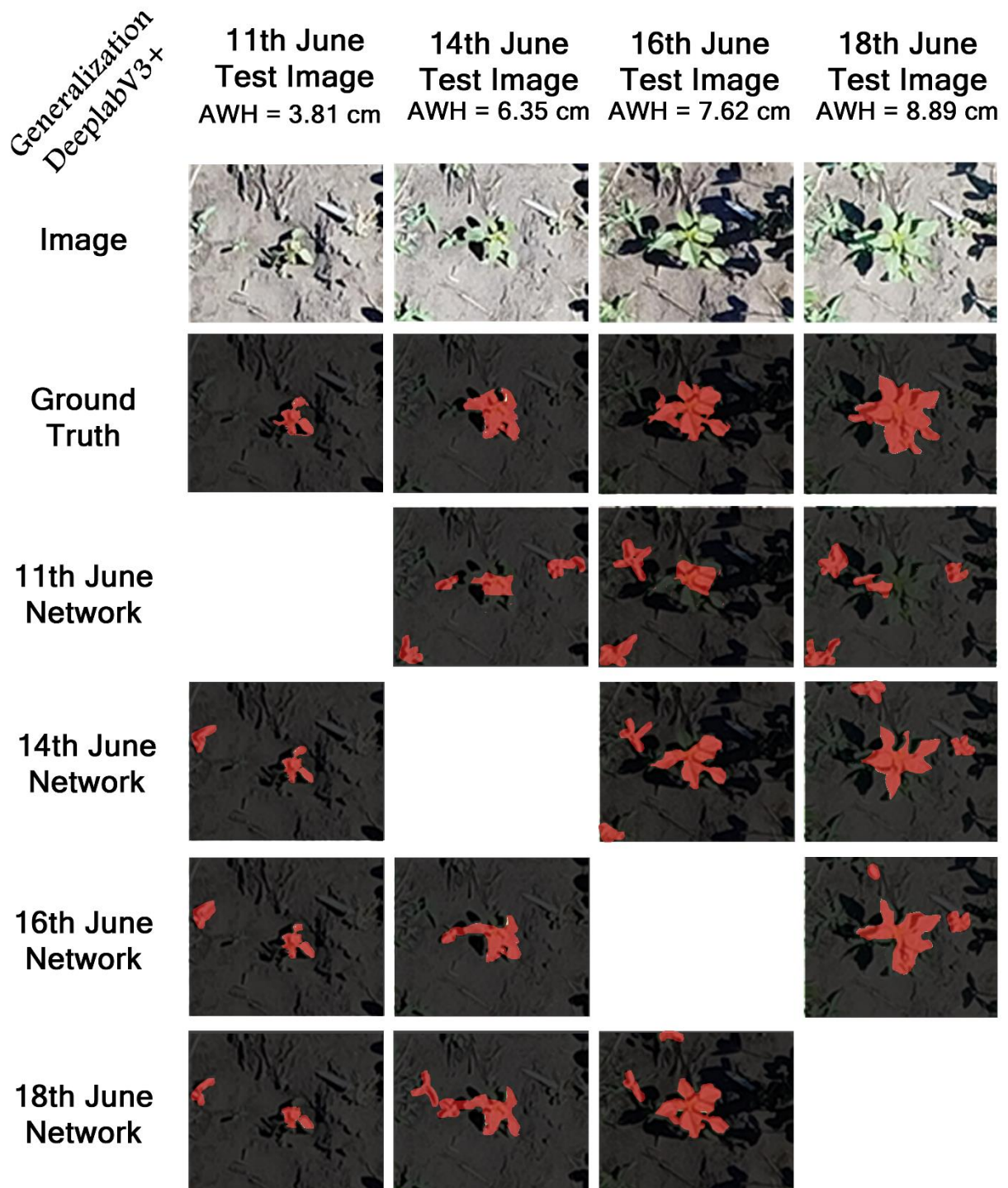


Figure 4.14: DeeplabV3+ generalization detection results for the four networks on the same patch of field. The patch contains a single instance of Palmer Amaranth. The RED color in represents Palmer amaranth, and the black color represents the background

4.3 Results for Network Deployment on Edge devices

4.3.1 Object detection on Edge devices

Among the two object detection networks, YOLOv4 was selected for deploying on the edge device. Although the performance of Faster R-CNN was higher, the added computational requirements of the two stages of the network don't make faster R-CNN an ideal network for edge devices. Additionally, YOLOv4 showed promising results with mAP score higher than 70% with detection in challenging situations and better generalization on image scale than Faster R-CNN.

Optimization results for NVIDIA Nano are shown in Figure 4.15. YOLOv4's native network runs on a compiled C code. The base fps for the native network is 1.6. When the network was converted into the ONNX model, the fps remained the same at 1.6. This was expected because the ONNX standard is not an optimized model but acts as a standard exchange format for various deep learning networks.

The highest increase in the fps was observed with tensorrt optimizations. With FP32 precision, 2.9 fps was obtained, while with FP16 precision, 4.6 fps was obtained.

While being a specialized library for YOLOv4, tkDNN did not perform better, if not worse, than tensorRT. With FP32 precision, tkDNN optimization resulted in 2.8 fps, about 0.1 fps less than tensorRT, while with FP16 precision, tkDNN optimization resulted in 3.9 fps, about 0.7 fps less than tensorRT.

From the results, another conclusion can be drawn. By reducing the floating-point precision, the inference time can be decreased, thus leading to higher fps. This could be explained as follows: MAC operations are the majority of the calculations performed at the time of inference in deep learning models. The inference time is considerably reduced because less computation has to be performed for Multiply and Accumulate Operation (MAC) with reduced floating-point (H. Zhang et al., 2018). For tensorRT and tkDNN optimizations, the highest fps was obtained with the Floating-point precision of FP16. A speedup of almost three times was seen in tensorRT optimizations with FP16 precision compared to the native network.

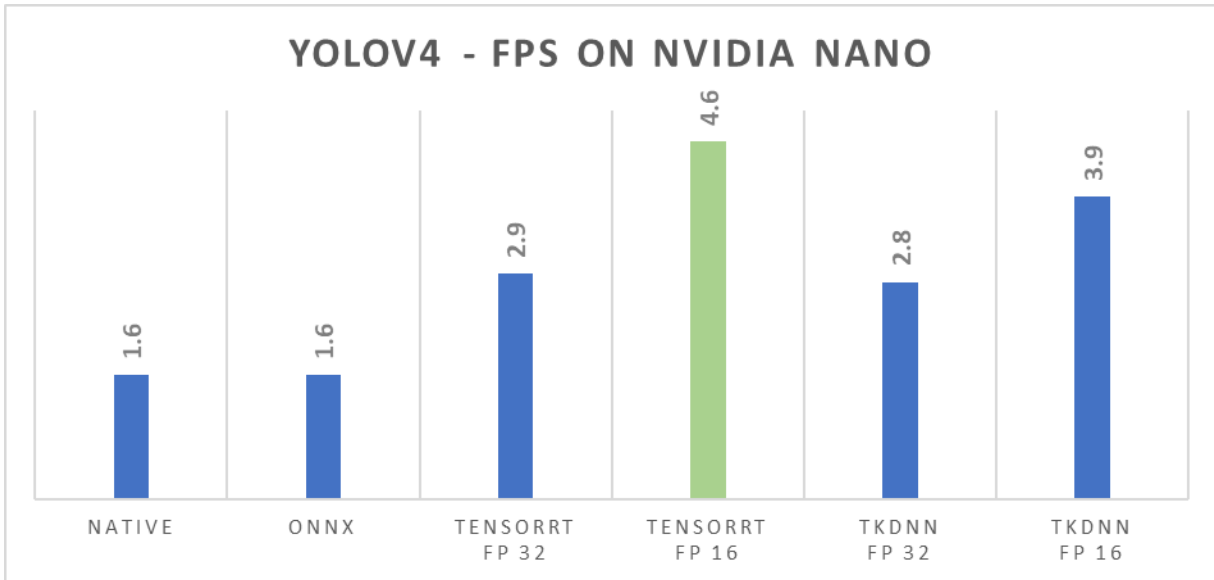


Figure 4.15: Optimization results for YOLOv4 on NVIDIA Nano. The green bar represents the highest FPS

YOLOv4 was also optimized for NVIDIA Xavier NX (Figure 4.16). Native YOLOv4 network ran at 5.2 fps, about three times higher than on NVIDIA Nano. This is because Xavier NX has more number of GPU cores as well as more GPU memory, both of which provide more computational power to Xavier NX.

ONNX network resulted in similar fps to the native network.

tensorRT network resulted in the highest fps among all the optimizations. With FP16 precision, 27.8 fps was observed, while with FP 32 precision, 12.1 fps was observed. Comparing the performance of tkDNN with tensorRT, fps delivered by tkDNN lacked. With FP 16 precision, 24 fps were obtained, while with FP 32 precision, 12.1 fps were obtained.

In conclusion, for YOLOv4, the closest to real-time performance was obtained with tensorRT optimization with FP16 precision on NVIDIA Xavier NX.

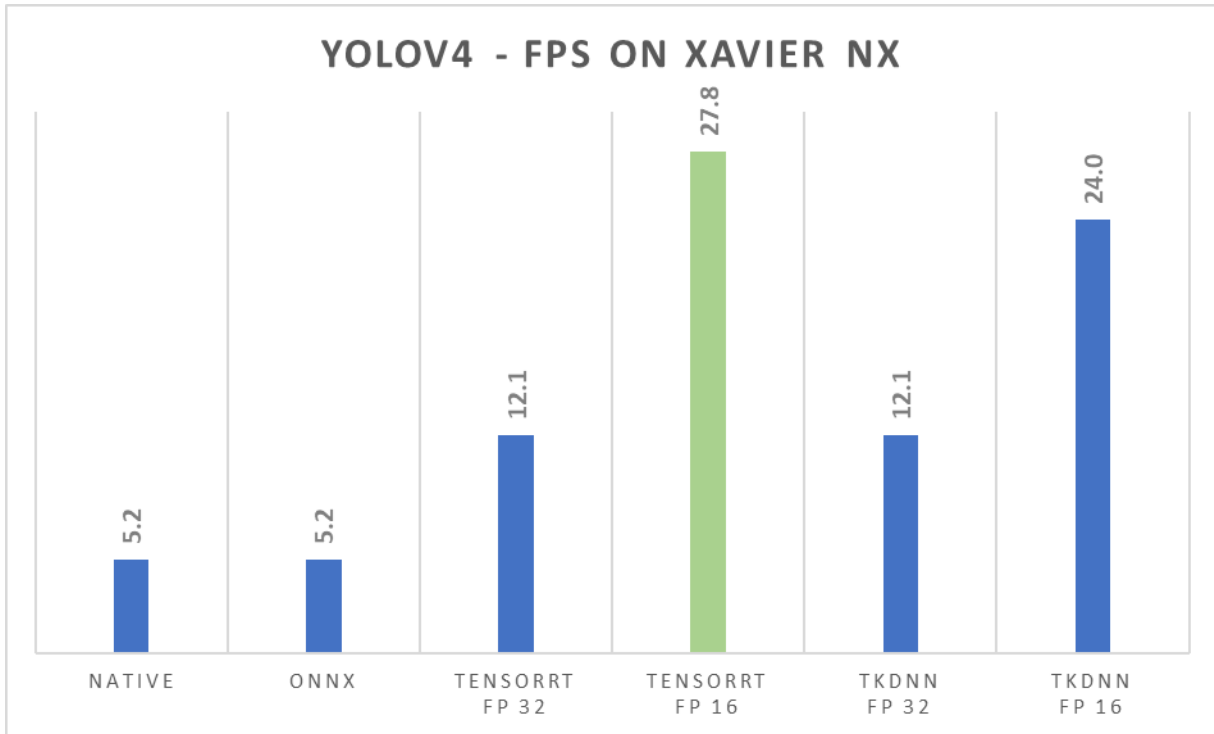


Figure 4.16: Optimization results for YOLOv4 on NVIDIA Xavier NX. The green bar represents the highest FPS

4.3.2 Image Segmentation on Edge devices

DeeplabV3+ was the best performing network for Image segmentation. Hence, it was optimized for evaluating its performance on the edge device.

In the case of DeeplabV3+, only tensorRT optimizations were applied. The optimization results for NVIDIA Nano are shown in Figure 4.17. The native DeeplabV3+ network resulted in 1.2 fps, while similar fps was observed with the ONNX network. The highest fps was seen with tensorRT. With FP16 precision, a frame rate of 2.9 was observed, while with FP32 precision, a frame rate of 2.9 was observed. An increase in frame rate by almost 240% is obtained with tensorRT-FP16 optimizations. Although this value is less than 30 fps (which is required for real-time performance), it is still the highest obtained fps in the research literature for weed identification using Image Segmentation with DeeplabV3+.

It is evident that DeeplabV3+ was inherently slower than YOLOv4. This was because the Image segmentation task provide more granular information for an image compared to object detection. This granular information comes at the cost of increased inference time.

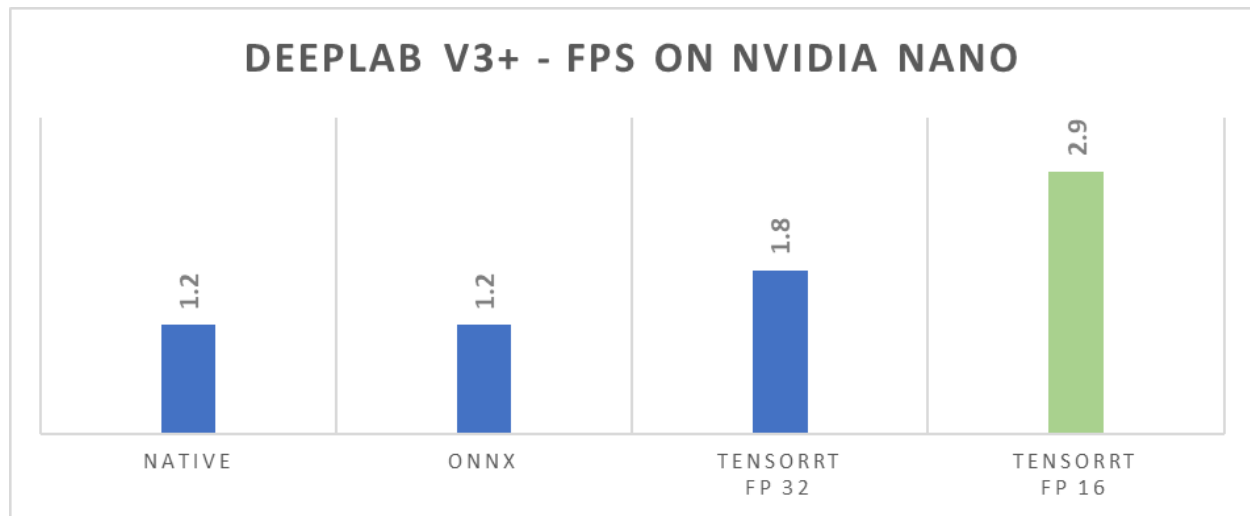


Figure 4.17: Optimization results for DeeplabV3+ on NVIDIA Nano. The green bar represents the highest FPS

DeeplabV3+ was additionally optimized for NVIDIA Xavier NX (Figure 4.18). The fps for the native DeeplabV3+ network was observed to be 4.3 fps. Compared to NVIDIA Nano, the fps increased by almost 3.5 times. This was primarily because of the increased computational power available in Xavier NX. Additionally, no increase in fps was observed with the ONNX network.

A four-fold increase in fps was observed with tensorRT optimizations for DeeplabV3+. With FP 16 precision, fps was 13.9 was obtained, while with FP 32 precision, fps of 8.3 was obtained. Hence, even for DeeplabV3+, the best optimizations were tensorRT with FP16 precision, and the best device was NVIDIA Xavier NX. While the fps was not even half the required real-time fps, it is a step in the right direction where more research is needed.

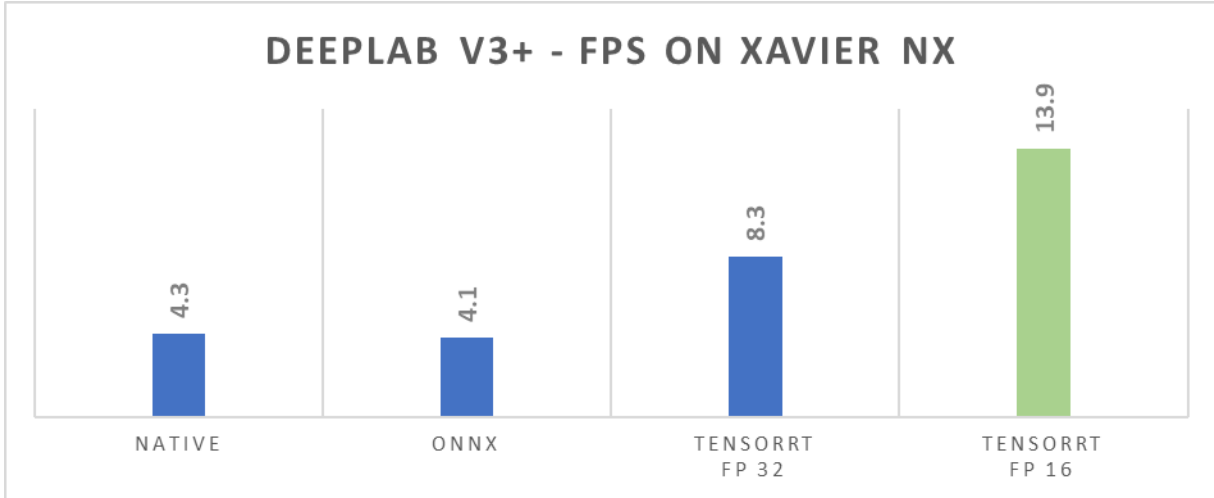


Figure 4.18: Optimization results for DeeplabV3+ on NVIDIA Xavier NX. The green bar represents the highest FPS

4.4 Comparison of results with other studies

The research demonstrated the potential of utilizing object detection and image segmentation for early season weed identification in Soybean using a) deep learning, b) in field conditions, c) from UAS images, and d) by deploying a trained model on edge devices. While other studies have not targeted all aspects included in this research, they have addressed some aspects. Hence, it becomes important to compare their performance with this research.

A study conducted for detecting weeds in mid- and late-season UAV imagery resulted in precision of 0.65 and recall of 0.68 for Faster R-CNN (Sivakumar et al., 2020b). The study did not target a specific weed, but rather multiple species of weeds were present at the test site. These included waterhemp (*Amaranthus tuberculatus*), Palmer amaranthus (*Amaranthus palmeri*), common lambsquarters (*Chenopodium album*), velvetleaf (*Abutilon theophrasti*), and foxtail. A direct comparison cannot be drawn with this research because the study did not report mAP score (primary evaluation metric for object detection (Padilla et al., 2021)). Although, mAP is a combination of precision and recall, a general idea could be drawn about the performance. The highest mAP obtained with Faster R-CNN in this research was 79.1 %, much higher than the individual precision or recall of the study. Moreover, this research targeted early-season weeds while the study was conducted for mid- and late-season weeds.

Another study targeted the detection of bermudagrass turfgrass. Detectnet, an object detection network, was used for weed identification (Yu et al., 2019). The study reported an F1 score greater than 0.99. While this research doesn't come close to the performance of the study, it is important to note two critical differences between this research and the study to track down the reason for the difference in performance. First, in the study, the images were acquired using a handheld camera. Thus, the images contained more detailed, distinguishing features than the UAS images used in this research. Second, the images were acquired of small patches of the field rather than an aerial view of the field. Hence each image contained a targeted instance of weeds rather than multiple instances of the weed in a single image. Both of these differences could be the reason for improved performance in the study.

A study conducted in canola fields used Segnet and UNET to segment crop, weed, and canola (Asad & Bais, 2020). With Segnet, an IoU score of 66.48 % was obtained for weed segmentation, while with UNET an IoU score of 66.22 % was obtained. The results in this research are in line with the performance of this study wherein DeeplabV3+ provided a maximum IoU score for Palmer amaranth of 68.7 %. Although, the study reported results on handheld images while this research reports it on UAS images.

Similar mean IoU score of 80% was obtained in the study conducted for segmenting sunflower, weed, and background (Fawakherji, Youssef, et al., 2019). In comparison, in this research, a mean IoU of 78.15 % was obtained. Another study used a custom segmentation network (CED-Net) for weed and crop segmentation (Khan et al., 2020). The custom network's performance was compared against UNET and DeeplabV3. The weed IoU for UNET, DeeplabV3, and CED-Net was 66.6 %, 61.4 %, and 70.2 %. These results are comparable to this research wherein DeeplabV3+ obtained a maximum IoU score for Palmer amaranth of 68.7 %. Although, merit of this research over the two studies (Fawakherji, Youssef, et al., 2019; Khan et al., 2020) is that none of them were conducted for UAS images.

None of the studies mentioned so far reported network optimization for real-time deployment on edge devices which is an important factor for agricultural robots (Hu et al., 2021). Some studies which have targeted real-time deployment have utilized devices like JetsonTX2 (N. Li et al., 2019),

Terasic DE1-SoC (Lammie et al., 2019), Jetson AGX Xavier (Lan et al., 2021). When using VGG16 with the Resnet10 backbone for weed segmentation, the frame rate of 5.6 fps was obtained on Jetson TX2. For segmentation of weeds and crops, fps of 45.05 were reported using MobilenetV2-UNet and tensorRT optimization with FP16 precision. This is a higher frame rate for segmentation compared to this research. Although, the deep learning network used in the study contained a lesser number of layers than DeeplabV3+. More number of layers usually increases the computation time for inference due to added MAC operation. Thus, explaining the higher fps for the study compared to this research.

Field programmable gated arrays (FPGAs) are specialized hardware programmed for one particular task. For example, an FPGA could be explicitly programmed for DeeplabV3+. In contrast to a general-purpose computer or edge device, FPGA would be faster because of the specialized nature of the hardware, which is optimized for one task alone. The benefit could be seen for image classification on the Deepweeds dataset (Olsen et al., 2019). Using an FPGA frame rate of 650 fps was obtained (Lammie et al., 2019). While the study was conducted for classification, which is computationally less expensive than object detection or image segmentation (thus the higher fps), yet the potential of FPGA for object detection and image segmentation must be explored as it could be possible to achieve even higher fps than obtained in this research.

5. CONCLUSION

The challenging task of early-season weed identification was addressed using deep learning networks. The generalization ability of four networks, namely, Faster R-CNN, YOLOv4, UNET, and DeepLabV3+, were analyzed for Palmer amaranth identification from RGB images captured by UAS in two computer vision tasks, i.e., object detection and image segmentation. The soybean field images were captured using a 4000x3000 pixels camera mounted on a UAS at a flight height of 5 m above ground level. The images included situations where plants overlapped or were occluded by external objects like a fence. Another challenging problem that was analyzed using deep learning networks was the presence of in-row weeds where overlapping weeds and crops make separating them using computer vision a hard problem.

The results indicate that a ground sampling distance of 0.17 cm/pixel was sufficient to accurately locate the weeds and crops in UAS images. Additionally, both Faster R-CNN and YOLOv4 were able to identify in-row instances of Palmer amaranth and identify overlapping soybean, even at the V2 stage when overlapping between soybean plants was much more than overlapping at the unifoliate stage. The highest testing mAP score of 75.1 % was achieved with the Faster R-CNN network on June 14, when the soybean was at the V1 growth stage, and Palmer amaranth's average height (AH) was 6.35 cm. Moreover, YOLOv4 was a better network at generalizing for different weed heights and soybean growth stages included in the current research. A recommendation emerges that the YOLOv4 network trained on images of the V1 soybean growth stage and Palmer amaranth's AH of 7.62 cm generalizes the best. It was able to identify soybean from the Unifoliate stage up to the V2 stage and was even able to identify Palmer amaranth with an AH of 3.81 cm.

Image segmentation did not achieve good performance for early-season Palmer amaranth identification. The highest generalization IoU score of 52.3 % was achieved with DeeplabV3+ trained on images of the V1 growth stage of soybean and 6.35 cm AH of Palmer amaranth. Despite poor overall performance in image segmentation, it was observed that compared to UNET, DeeplabV3+ was better at segmenting the edges of the weeds. Hence, for future image segmentation applications, the DeeplabV3+ algorithm needs further exploration.

The results demonstrate the advantage of deploying deep learning networks on edge devices with the TensorRT optimization library. On NVIDIA Xavier NX, YOLOv4 ran at a frame rate of almost 28 fps, just two fps short of real-time. Because Xavier NX weighed a little less than 60 gm and required just 15 W of power, it would be a suitable device for mounting on UAS and ground robots. In comparison to other studies, this research is unique in a number of ways. Through this research, the generalizability accuracy of deep learning networks is determined for the first time for soybean crops and Palmer amaranth weed. In addition, the highest fps is reported compared to other studies for the task of both object detection and image segmentation with large deep learning networks (over 100 layers) on an edge device. For example, when using ERFNet to segment sugarbeet and weed, a maximum frame rate of 10 fps was reported on Jetson TX2 (Milioto and Stachniss, 2019). In comparison, in this research, a frame rate of 14 fps was reported with DeeplabV3+ despite DeeplabV3+ being more computationally expensive than ERFNet (Romera et al., 2018). Similarly, another recent study utilized a Coral USB accelerator and Raspberry Pi 4 to detect weeds (Czymmek et al., 2021). The study reported an average of 53 fps by using a MobileNetv2-SSD object detection network. MobileNetv2-SSD is a shallow network compared to the YOLOv3 network and results in an accuracy drop of about 7 %, as reported in the study. Hence the faster fps came at the expense of accuracy, while in current research, despite using a full YOLOv4 network, close to 28 fps were obtained. A study targeted towards weed detection for organic farming utilized a modified YOLOv3 network (Czymmek et al., 2019). It reported 40 fps on a powerful server-grade GPU (NVIDIA GeForce GTX 1080 Ti), which consumed close to 250 W of power. In contrast, the current research achieved 28 fps by implementing YOLOv4 on an edge device that required only a fraction of the power, an equivalent of 15 W. Not to mention that YOLOv4 is a more accurate network compared to YOLOv3 (Bochkovskiy et al., 2020).

While this research exhibits the potential of using deep learning networks on resource-limited edge devices for early-season weed identification, more work could be done to further this research and address the following shortcomings.

- 1) The research was not designed to validate the generalizability of the deep learning networks for identifying the presence of Palmer amaranth weed while the soybean growth stage varied from the cotyledon to the V3-V4. Finding a single deep learning network that could

accurately detect weeds from the V1 to V4 stage of soybean would be very useful as multiple studies have reported that controlling weeds in soybean before these stages reduce the overall impact of weeds on crop yield (Coulter and Nafziger, 2007; Dalley et al., 2004; Eyherabide and Cendoya, 2002; Soltani et al., 2019).

- 2) The poor generalizability results for image segmentation is a problem that could be addressed in the future by exploring other deep learning networks or by developing custom segmentation algorithms like CED-NET (Khan et al., 2020).
- 3) Vegetative indices derived from RGB images could increase the image segmentation performance, as demonstrated by another study (Milioto et al., 2018).
- 4) While the networks were optimized for deploying on the edge device, it was not integrated with the UAS or a ground robot. Integration with the robot could potentially reduce the fps a little due to the transmission overhead of image from the camera to the edge device. Further reduction in fps is possible due to the computational overhead of transmitting the output of the edge device to a remote operator or storing the output on a disk. Although, the delay due to computational overhead could be reduced by utilizing DeepStream SDK (NVIDIA, 2021b).
- 5) INT 8 optimization has the potential to increase the fps for deep learning networks further. However, it requires re-training the network with new quantization nodes within the network. The process is called Quantization Aware Training (QAT). NVIDIA has reported achieving fps of almost 60 on YOLOv4, although QAT is yet to be explored for weed identification purposes (NVIDIA, 2021a).

REFERENCES

- Abbas, T., Zahir, Z. A., Naveed, M., & Kremer, R. J. (2018). Chapter Five - Limitations of Existing Weed Control Practices Necessitate Development of Alternative Techniques Based on Biological Approaches. In D. L. Sparks (Ed.), *Advances in Agronomy* (Vol. 147, pp. 239–280). Academic Press.
- Adão, T., Hruška, J., Pádua, L., Bessa, J., Peres, E., Morais, R., & Sousa, J. J. (2017). Hyperspectral Imaging: A Review on UAV-Based Sensors, Data Processing and Applications for Agriculture and Forestry. *Remote Sensing*, 9(11), 1110. <https://doi.org/10.3390/rs9111110>
- Aghi, D., Mazzia, V., & Chiaberge, M. (2020). Local Motion Planner for Autonomous Navigation in Vineyards with a RGB-D Camera-Based Algorithm and Deep Learning Synergy. *Machines* 2020, Vol. 8, Page 27, 8(2), 27. <https://doi.org/10.3390/MACHINES8020027>
- Ahmad, J., Muhammad, K., Ahmad, I., Ahmad, W., Smith, M. L., Smith, L. N., Jain, D. K., Wang, H., & Mehmood, I. (2018). Visual features based boosted classification of weeds for real-time selective herbicide sprayer systems. *Computers in Industry*, 98, 23–33.
- Al Mansoori, S. H., Kunhu, A., & Al Ahmad, H. (2018). Automatic palm trees detection from multispectral UAV data using normalized difference vegetation index and circular Hough transform. In B. Huang, S. López, & Z. Wu (Eds.), *High-Performance Computing in Geoscience and Remote Sensing VIII* (Vol. 10792, Issue 9, p. 3). SPIE. <https://doi.org/10.1117/12.2325732>
- Ale, L., Sheta, A., Li, L., Wang, Y., & Zhang, N. (2019). Deep learning based plant disease detection for smart agriculture. *2019 IEEE Globecom Workshops, GC Wkshps 2019 - Proceedings*. <https://doi.org/10.1109/GCWKSHPS45667.2019.9024439>
- Amziane, A., Losson, O., Mathon, B., Dumenil, A., & Macaire, L. (2020, November 9). Frame-based reflectance estimation from multispectral images for weed identification in varying illumination conditions. *2020 10th International Conference on Image Processing Theory, Tools and Applications, IPTA 2020*. <https://doi.org/10.1109/IPTA50016.2020.9286692>
- Andrea, C. C., Mauricio Daniel, B., & Jose Misael, J. B. (2018). Precise weed and maize classification through convolutional neuronal networks. *2017 IEEE 2nd Ecuador Technical Chapters Meeting, ETCM 2017, 2017-Janua*, 1–6. <https://doi.org/10.1109/ETCM.2017.8247469>

- Andújar, D., Weis, M., & Gerhards, R. (2012). An ultrasonic system for weed detection in cereal crops. *Sensors (Switzerland)*, 12(12), 17343–17357. <https://doi.org/10.3390/s121217343>
- Arganda-Carreras, I., Turaga, S. C., Berger, D. R., Cireşan, D., Giusti, A., Gambardella, L. M., Schmidhuber, J., Laptev, D., Dwivedi, S., Buhmann, J. M., Liu, T., Seyedhosseini, M., Tasdizen, T., Kamensky, L., Burget, R., Uher, V., Tan, X., Sun, C., Pham, T. D., ... Seung, H. S. (2015). Crowdsourcing the creation of image segmentation algorithms for connectomics. *Frontiers in Neuroanatomy*, 0(November), 142. <https://doi.org/10.3389/FNANA.2015.00142>
- Asad, M. H., & Bais, A. (2020). Weed detection in canola fields using maximum likelihood classification and deep convolutional neural network. *Information Processing in Agriculture*, 7(4), 535–545. <https://doi.org/10.1016/j.inpa.2019.12.002>
- Bah, M. D., Hafiane, A., & Canals, R. (2017). Weeds detection in UAV imagery using SLIC and the hough transform. *Proceedings of the 7th International Conference on Image Processing Theory, Tools and Applications, IPTA 2017, 2018-Janua*, 1–6. <https://doi.org/10.1109/IPTA.2017.8310102>
- Bah, M., Hafiane, A., & Canals, R. (2018). Deep Learning with Unsupervised Data Labeling for Weed Detection in Line Crops in UAV Images. *Remote Sensing*, 10(11), 1690. <https://doi.org/10.3390/rs10111690>
- Bakhshipour, A., & Jafari, A. (2018a). Evaluation of support vector machine and artificial neural networks in weed detection using shape features. *Computers and Electronics in Agriculture*. <https://www.sciencedirect.com/science/article/pii/S0168169917301333>
- Bakhshipour, A., & Jafari, A. (2018b). Evaluation of support vector machine and artificial neural networks in weed detection using shape features. *Computers and Electronics in Agriculture*, 145, 153–160. <https://doi.org/10.1016/j.compag.2017.12.032>
- Barrero, O., & Perdomo, S. A. (2018a). RGB and multispectral UAV image fusion for Gramineae weed detection in rice fields. *Precision Agriculture*, 19(5), 809–822. <https://doi.org/10.1007/s11119-017-9558-x>
- Barrero, O., & Perdomo, S. A. (2018b). RGB and multispectral UAV image fusion for Gramineae weed detection in rice fields. *Precision Agriculture*, 19(5), 809–822. <https://doi.org/10.1007/S11119-017-9558-X/FIGURES/10>

- Bay, H., Tuytelaars, T., & Van Gool, L. (2006). SURF: Speeded up robust features. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/11744023_32
- Behmann, J., Mahlein, A. K., Rumpf, T., Römer, C., & Plümer, L. (2015). A review of advanced machine learning methods for the detection of biotic stress in precision crop protection. *Precision Agriculture*, 16(3), 239–260. <https://doi.org/10.1007/S11119-014-9372-7/FIGURES/5>
- Bhong, V. S., Waghmare, D. L., Jadhav, A. A., Bahadure, N. B., Bhaldar, H. K., & Vibhute, A. S. (2020). Design farming robot for weed detection and herbicides applications using image processing. *Techno-Societal 2018 - Proceedings of the 2nd International Conference on Advanced Technologies for Societal Applications*. https://doi.org/10.1007/978-3-030-16848-3_38
- Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020a). YOLOv4. *CVPR Workshop on The Future of Datasets in Vision*.
- Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020b). YOLOv4: Optimal Speed and Accuracy of Object Detection. *ArXiv:2004.10934 [Cs, Eess]*. <http://arxiv.org/abs/2004.10934>
- Brilhador, A., Gutoski, M., Hattori, L. T., De Souza Inacio, A., Lazzaretti, A. E., & Lopes, H. S. (2019). Classification of Weeds and Crops at the Pixel-Level Using Convolutional Neural Networks and Data Augmentation. *2019 IEEE Latin American Conference on Computational Intelligence, LA-CCI 2019*, 1–6. <https://doi.org/10.1109/LA-CCI47412.2019.9037044>
- Buddha, K., Nelson, H. J., Zermas, D., & Papanikolopoulos, N. (2019). Weed detection and classification in high altitude aerial images for robot-based precision agriculture. *27th Mediterranean Conference on Control and Automation, MED 2019 - Proceedings*. <https://doi.org/10.1109/MED.2019.8798582>
- Caiati, C., Pollice, P., Favale, S., & Lepera, M. E. (2019). The Herbicide Glyphosate and Its Apparently Controversial Effect on Human Health: An Updated Clinical Perspective. *Endocrine, Metabolic & Immune Disorders - Drug Targets*, 20(4), 489–505. <https://doi.org/10.2174/1871530319666191015191614>

- Calderón, R., Navas-Cortés, J. A., Lucena, C., & Zarco-Tejada, P. J. (2013). High-resolution airborne hyperspectral and thermal imagery for early detection of Verticillium wilt of olive using fluorescence, temperature and narrow-band spectral indices. *Remote Sensing of Environment*, 139, 231–245. <https://doi.org/10.1016/J.RSE.2013.07.031>
- Castro, A. I. de, Torres-Sánchez, J., Peña, J. M., Jiménez-Brenes, F. M., Csillik, O., & López-Granados, F. (2018). An Automatic Random Forest-OBIA Algorithm for Early Weed Mapping between and within Crop Rows Using UAV Imagery. *Remote Sensing 2018, Vol. 10, Page 285, 10(2)*, 285. <https://doi.org/10.3390/RS10020285>
- Champ, J., Mora-Fallas, A., Goëau, H., Mata-Montero, E., Bonnet, P., & Joly, A. (2020). Instance segmentation for the fine detection of crop and weed plants by precision agricultural robots. *Applications in Plant Sciences*. <https://doi.org/10.1002/aps3.11373>
- Chen, C. J., Huang, Y. Y., Li, Y. S., Chen, Y. C., Chang, C. Y., & Huang, Y. M. (2021). Identification of Fruit Tree Pests with Deep Learning on Embedded Drone to Achieve Accurate Pesticide Spraying. *IEEE Access*, 9, 21986–21997. <https://doi.org/10.1109/ACCESS.2021.3056082>
- Chen, J., & Ran, X. (2019). Deep Learning With Edge Computing: A Review. *Proc. IEEE*, 107(8), 1655–1674.
- Chen, L. C., Papandreou, G., Schroff, F., & Adam, H. (2017). Rethinking atrous convolution for semantic image segmentation. In *arXiv*.
- Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-030-01234-2_49
- Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. *Proceedings of the European Conference on Computer Vision (ECCV)*, 801–818.
- Chen, Y., Lin, Z., Zhao, X., Wang, G., & Gu, Y. (2014). Deep Learning-Based Classification of Hyperspectral Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6), 2094–2107. <https://doi.org/10.1109/JSTARS.2014.2329330>

- Cheng, B., & Matson, E. T. (2015). A Feature-Based Machine Learning Agent for Automatic Rice and Weed Discrimination. *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, 9119, 517–527. https://doi.org/10.1007/978-3-319-19324-3_46
- Cheng, H. D., Jiang, X. H., Sun, Y., & Wang, J. (2001). Color image segmentation: advances and prospects. *Pattern Recognition*, 34(12), 2259–2281. [https://doi.org/10.1016/S0031-3203\(00\)00149-7](https://doi.org/10.1016/S0031-3203(00)00149-7)
- Che'ya, N. N., Dunwoody, E., & Gupta, M. (2021). Assessment of Weed Classification Using Hyperspectral Reflectance and Optimal Multispectral UAV Imagery. *Agronomy* 2021, Vol. 11, Page 1435, 11(7), 1435. <https://doi.org/10.3390/AGRONOMY11071435>
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1251–1258.
- Christensen, S., Heisel, T., Walter, A. M., & Graglia, E. (2003). A decision algorithm for patch spraying. *Weed Research*, 43(4), 276–284. <https://doi.org/10.1046/J.1365-3180.2003.00344.X>
- Chun, B. G., Ihm, S., Maniatis, P., Naik, M., & Patti, A. (2011). CloneCloud: Elastic execution between mobile device and cloud. *EuroSys'11 - Proceedings of the EuroSys 2011 Conference*, 11, 301–314. <https://doi.org/10.1145/1966445.1966473>
- Cisco, U. (2020). *Cisco annual internet report (2018–2023) white paper*. <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/whitepaper-c11-741490.html>
- Cook, A., Robinson, M., Ferrag, M. A., Maglaras, L. A., He, Y., Jones, K., & Janicke, H. (2018). *Internet of Cloud: Security and Privacy Issues*. 271–301. https://doi.org/10.1007/978-3-319-73676-1_11
- Coulter, J. A., & Nafziger, E. D. (2007). Planting Date and Glyphosate Timing on Soybean. *Weed Technology*, 21(2), 359–366. <https://doi.org/10.1614/WT-06-025.1>
- Cox, D. R. (1959). The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 21(1), 238–238. <https://doi.org/10.1111/j.2517-6161.1959.tb00334.x>
- Craven, B. D., & Islam, S. M. N. (2011). Ordinary least-squares regression. *The SAGE Dictionary of Quantitative Management Research*, 224–228.

- Cruz, A. C., El-Kereamy, A., & Ampatzidis, Y. (2017). Vision-based Grapevine Pierce's disease detection system using artificial intelligence. *Elibrary.Asabe.Org*. <https://doi.org/10.13031/aim.201800148>
- Czymmek, V., Harders, L. O., Knoll, F. J., & Hussmann, S. (2019). Vision-based deep learning approach for real-time detection of weeds in organic farming. *I2MTC 2019 - 2019 IEEE International Instrumentation and Measurement Technology Conference, Proceedings, 2019-May*. <https://doi.org/10.1109/I2MTC.2019.8826921>
- Czymmek, V., Moller, C., Harders, L. O., & Hussmann, S. (2021). Deep Learning Approach for high Energy efficient Real-Time Detection of Weeds in Organic Farming. *Conference Record - IEEE Instrumentation and Measurement Technology Conference, 2021-May*. <https://doi.org/10.1109/I2MTC50364.2021.9459943>
- Dalley, C. D., Kells, J. J., & Renner, K. A. (2004). Effect of Glyphosate Application Timing and Row Spacing on Weed Growth in Corn (*Zea mays*) and Soybean (*Glycine max*). *Weed Technology*, 18(1), 177–182. <https://doi.org/10.1614/02-150B>
- Dautov, R., & Song, H. (2019). Towards IoT Diversity via Automated Fleet Management. *MDE4IoT/ModComp@ MoDELS*, 47–54.
- de Castro, A., Torres-Sánchez, J., Peña, J., Jiménez-Brenes, F., Csillik, O., & López-Granados, F. (2018). An Automatic Random Forest-OBIA Algorithm for Early Weed Mapping between and within Crop Rows Using UAV Imagery. *Remote Sensing*, 10(3), 285. <https://doi.org/10.3390/rs10020285>
- Debauche, O., Mahmoudi, S., Mahmoudi, S. A., Manneback, P., Bindelle, J., & Lebeau, F. (2020). Edge Computing and Artificial Intelligence for Real-time Poultry Monitoring. *Procedia Computer Science*, 175, 534–541. <https://doi.org/10.1016/J.PROCS.2020.07.076>
- Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data Via the EM Algorithm . *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1), 1–22. <https://doi.org/10.1111/j.2517-6161.1977.tb01600.x>
- Deng, J., Zhong, Z., Huang, H., Lan, Y., Han, Y., & Zhang, Y. (2020a). Lightweight Semantic Segmentation Network for Real-Time Weed Mapping Using Unmanned Aerial Vehicles. *Applied Sciences*, 10(20), 7132. <https://doi.org/10.3390/app10207132>

- Deng, J., Zhong, Z., Huang, H., Lan, Y., Han, Y., & Zhang, Y. (2020b). Lightweight Semantic Segmentation Network for Real-Time Weed Mapping Using Unmanned Aerial Vehicles. *Applied Sciences* 2020, Vol. 10, Page 7132, 10(20), 7132. <https://doi.org/10.3390/APP10207132>
- dos Santos Ferreira, A., Matte Freitas, D., Gonçalves da Silva, G., Pistori, H., & Theophilo Folhes, M. (2017). Weed detection in soybean crops using ConvNets. *Computers and Electronics in Agriculture*. <https://doi.org/10.1016/j.compag.2017.10.027>
- Elstone, L., How, K. Y., Brodie, S., Ghazali, M. Z., Heath, W. P., & Grieve, B. (2020). High Speed Crop and Weed Identification in Lettuce Fields for Precision Weeding. *Sensors* 2020, Vol. 20, Page 455, 20(2), 455. <https://doi.org/10.3390/S20020455>
- Espejo-Garcia, B., Mylonas, N., Athanasakos, L., Fountas, S., & Vasilakoglou, I. (2020). Towards weeds identification assistance through transfer learning. *Computers and Electronics in Agriculture*, 171. <https://doi.org/10.1016/j.compag.2020.105306>
- Espinoza, M. A. M., Le, C. Z., Raheja, A., & Bhandari, S. (2020). Weed identification and removal using machine learning techniques and unmanned ground vehicles. *Autonomous {Air} and {Ground} {Sensing} {Systems} for {Agricultural} {Optimization} and {Phenotyping} {V}*, 11414, 114140J. <https://doi.org/10.1117/12.2557625>
- Esposito, M., Crimaldi, M., Cirillo, V., Sarghini, F., & Maggio, A. (2021). Drone and sensor technology for sustainable weed management: a review. *Chemical and Biological Technologies in Agriculture*, 8(1), 1–11. <https://doi.org/10.1186/S40538-021-00217-8/TABLES/4>
- Eyherabide, J. J., & Cendoya, M. G. (2002). Critical periods of weed control in soybean for full field and in-furrow interference. *Weed Science*, 50(2), 162–166. [https://doi.org/10.1614/0043-1745\(2002\)050\[0162:CPOWCI\]2.0.CO;2](https://doi.org/10.1614/0043-1745(2002)050[0162:CPOWCI]2.0.CO;2)
- Fang, G. (2019). *DeepLabv3, DeepLabv3+ and pretrained weights on VOC & Cityscapes*. <https://github.com/VainF/DeepLabV3Plus-Pytorch>
- Farooq, A., Hu, J., & Jia, X. (2018). Weed classification in hyperspectral remote sensing images via deep convolutional neural network. *International Geoscience and Remote Sensing Symposium (IGARSS)*. <https://doi.org/10.1109/IGARSS.2018.8518541>

- Fawakherji, M., Potena, C., Bloisi, D. D., Imperoli, M., Pretto, A., & Nardi, D. (2019). *UAV Image Based Crop and Weed Distribution Estimation on Embedded GPU Boards* (M. Vento, G. Percannella, S. Colantonio, D. Giorgi, B. J. Matuszewski, H. Kerdegari, & M. Razaak, Eds.; pp. 100–108). Springer International Publishing. https://doi.org/10.1007/978-3-030-29930-9_10
- Fawakherji, M., Youssef, A., Bloisi, D., Pretto, A., & Nardi, D. (2019). Crop and weeds classification for precision agriculture using context-independent pixel-wise segmentation. *2019 Third IEEE International Conference on Robotic Computing (IRC)*, 146–152.
- Federal Aviation Administration. (2020). *FAA Aerospace Forecast: Fiscal Years 2020- 2040*.
- Fennimore, S. A., & Cutulle, M. (2019a). Robotic weeders can improve weed control options for specialty crops. *Pest Management Science*, 75(7), 1767–1774. <https://doi.org/10.1002/ps.5337>
- Fennimore, S. A., & Cutulle, M. (2019b). Robotic weeders can improve weed control options for specialty crops. *Pest Management Science*, 75(7), 1767–1774. <https://doi.org/10.1002/ps.5337>
- Fernández-Quintanilla, C., Peña, J., ... D. A.-W., & 2018, undefined. (2018). Is the current state of the art of weed monitoring suitable for site-specific weed management in arable crops? *Wiley Online Library*, 58(4), 259–272. <https://doi.org/10.1111/wre.12307>
- Fernández-Quintanilla, C., Peña, J. M., Andújar, D., Dorado, J., Ribeiro, A., & López-Granados, F. (2018). Is the current state of the art of weed monitoring suitable for site-specific weed management in arable crops? *Weed Research*, 58(4), 259–272. <https://doi.org/10.1111/wre.12307>
- Fickett, N. D., Boerboom, C. M., & Stoltenberg, D. E. (2013). Soybean Yield Loss Potential Associated with Early-Season Weed Competition across 64 Site-Years. *Weed Science*, 61(3), 500–507. <https://doi.org/10.1614/WS-D-12-00164.1>
- Friedman, J. H. (2007). Rejoinder: Multivariate Adaptive Regression Splines. *The Annals of Statistics*, 19(1), 1–67. <https://doi.org/10.1214/aos/1176347973>
- Gaba, S., Gabriel, E., Chadœuf, J., Bonneau, F., & Bretagnolle, V. (2016). Herbicides do not ensure for higher wheat yield, but eliminate rare plant species. *Scientific Reports 2016 6:1*, 6(1), 1–10. <https://doi.org/10.1038/srep30112>

- Gago, J., Douthe, C., Coopman, R. E., Gallego, P. P., Ribas-Carbo, M., Flexas, J., Escalona, J., & Medrano, H. (2015). UAVs challenge to assess water stress for sustainable agriculture. *Agricultural Water Management*, 153, 9–19. <https://doi.org/10.1016/J.AGWAT.2015.01.020>
- Gaines, T. A., Duke, S. O., Morran, S., Rigon, C. A. G., Tranel, P. J., Küpper, A., & Dayan, F. E. (2020). Mechanisms of evolved herbicide resistance. *Journal of Biological Chemistry*, 295(30), 10307–10330. <https://doi.org/10.1074/jbc.REV120.013572>
- Gao, J., French, A. P., Pound, M. P., He, Y., Pridmore, T. P., & Pieters, J. G. (2020). Deep convolutional neural networks for image-based {Convolvulus} sepium detection in sugar beet fields. *Plant Methods*, 16(1), 29. <https://doi.org/10.1186/s13007-020-00570-z>
- Gibson, K. D., Dirks, R., Medlin, C. R., & Johnston, L. (2004). Detection of Weed Species in Soybean Using Multispectral Digital Images. *Weed Technology*, 18(3), 742–749. <https://doi.org/10.1614/WT-03-170R1>
- Gower, S. A., Loux, M. M., Cardina, J., Harrison, S. K., Sprankle, P. L., Probst, N. J., Bauman, T. T., Bugg, W., Curran, W. S., Currie, R. S., Harvey, R. G., Johnson, W. G., Kells, J. J., Owen, M. D. K., Regehr, D. L., Slack, C. H., Spaur, M., Sprague, C. L., Vangessel, M., & Young, B. G. (2003). Effect of Postemergence Glyphosate Application Timing on Weed Control and Grain Yield in Glyphosate-Resistant Corn: Results of a 2-Yr Multistate Study. *Weed Technology*, 17(4), 821–828. <https://doi.org/10.1614/P02-200>
- Grassini, P., Eskridge, K. M., & Cassman, K. G. (2013). Distinguishing between yield advances and yield plateaus in historical crop production trends. *Nature Communications*, 4. <https://doi.org/10.1038/ncomms3918>
- Guijarro, M., Pajares, G., Riomoros, I., Herrera, P. J., Burgos-Artizzu, X. P., & Ribeiro, A. (2011). Automatic segmentation of relevant textures in agricultural images. *Computers and Electronics in Agriculture*, 75(1), 75–83. <https://doi.org/10.1016/J.COMPAG.2010.09.013>
- Gupta, M., Abdelsalam, M., Khorsandroo, S., & Mittal, S. (2020). Security and Privacy in Smart Farming: Challenges and Opportunities. *IEEE Access*, 8, 34564–34584. <https://doi.org/10.1109/ACCESS.2020.2975142>
- Hamuda, E., Glavin, M., & Jones, E. (2016). A survey of image processing techniques for plant extraction and segmentation in the field. *Computers and Electronics in Agriculture*, 125, 184–199. <https://doi.org/10.1016/J.COMPAG.2016.04.024>

- Hassanein, M., & El-Sheimy, N. (2018). An efficient weed detection procedure using low-cost UAV imagery system for precision agriculture applications. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*. <https://doi.org/10.5194/isprs-archives-XLII-1-181-2018>
- He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, 2961–2969.
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2015.2389824>
- He, Y., & Weng, Q. (2018). *High spatial resolution remote sensing: data, analysis, and applications*.
- Hestir, E. L., Khanna, S., Andrew, M. E., Santos, M. J., Viers, J. H., Greenberg, J. A., Rajapakse, S. S., & Ustin, S. L. (2008). Identification of invasive vegetation using hyperspectral remote sensing in the California Delta ecosystem. *Remote Sensing of Environment*, 112(11), 4034–4047. <https://doi.org/10.1016/j.rse.2008.01.022>
- Hu, K., Wang, Z., Coleman, G., Bender, A., Yao, T., Zeng, S., Song, D., Schumann, A., & Walsh, M. (2021). Deep Learning Techniques for In-Crop Weed Identification: A Review. *ArXiv.Org*. <https://arxiv.org/abs/2103.14872v1>
- Huang, H., Deng, J., Lan, Y., Yang, A., Deng, X., Wen, S., Zhang, H., & Zhang, Y. (2018). Accurate weed mapping and prescription map generation based on fully convolutional networks using UAV imagery. *Sensors (Switzerland)*, 18(10), 3299. <https://doi.org/10.3390/s18103299>
- Huang, Y., Lee, M. A., Thomson, S. J., & Reddy, K. N. (2016). Ground-based hyperspectral remote sensing for weed management in crop production. *International Journal of Agricultural and Biological Engineering*, 9(2), 98–109. <https://doi.org/10.3965/j.ijabe.20160902.2137>
- Hung, C., Xu, Z., & Sukkarieh, S. (2014). Feature Learning Based Approach for Weed Classification Using High Resolution Aerial Images from a Digital Camera Mounted on a UAV. *Remote Sensing*, 6(12), 12037–12054. <https://doi.org/10.3390/rs61212037>

- Hunter, M. C., Smith, R. G., Schipanski, M. E., Atwood, L. W., & Mortensen, D. A. (2017). Agriculture in 2050: Recalibrating Targets for Sustainable Intensification. *BioScience*, 67(4), 386–391. <https://doi.org/10.1093/biosci/bix010>
- Iqbal, F., Lucieer, A., & Barry, K. (2018). Simplified radiometric calibration for UAS-mounted multispectral sensor. *European Journal of Remote Sensing*. <https://doi.org/10.1080/22797254.2018.1432293>
- Jaggard, K. W., Qi, A., & Ober, E. S. (2010). Possible changes to arable crop yields by 2050. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1554), 2835–2851. <https://doi.org/10.1098/rstb.2010.0153>
- Jasiński, M., Mączak, J., Szulim, P., & Radkowski, S. (2018). Autonomous Agricultural Robot – Testing of the Vision System for Plants/Weed Classification. *Advances in Intelligent Systems and Computing*, 743, 473–482. https://doi.org/10.1007/978-3-319-77179-3_44
- Jha, K., Doshi, A., Patel, P., & Shah, M. (2019). A comprehensive review on automation in agriculture using artificial intelligence. *Artificial Intelligence in Agriculture*, 2, 1–12. <https://doi.org/10.1016/j.aiia.2019.05.004>
- Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, 32(3), 241–254. <https://doi.org/10.1007/BF02289588>
- Jouppi, N. P., Young, C., Patil, N., Patterson, D., Agrawal, G., Bajwa, R., Bates, S., Bhatia, S., Boden, N., Borchers, A., Boyle, R., Cantin, P. L., Chao, C., Clark, C., Coriell, J., Daley, M., Dau, M., Dean, J., Gelb, B., ... Yoon, D. H. (2017). In-datacenter performance analysis of a tensor processing unit. *Proceedings - International Symposium on Computer Architecture*. <https://doi.org/10.1145/3079856.3080246>
- Jumarie, C., Aras, P., & Boily, M. (2017). Mixtures of herbicides and metals affect the redox system of honey bees. *Chemosphere*, 168, 163–170. <https://doi.org/10.1016/J.CHEMOSPHERE.2016.10.056>
- Kawaguchi, K., Kaelbling, L. P., & Bengio, Y. (2017). Generalization in Deep Learning. *ArXiv.Org*. <https://arxiv.org/abs/1710.05468v6>
- Khan, A., Ilyas, T., Umraiz, M., Mannan, Z. I., & Kim, H. (2020). Ced-net: Crops and weeds segmentation for smart farming using a small cascaded encoder-decoder architecture. *Electronics (Switzerland)*. <https://doi.org/10.3390/electronics9101602>

- Kim, S. K., & Huh, J. H. (2020). Consistency of medical data using intelligent neuron faster r-cnn algorithm for smart health care application. *Healthcare (Switzerland)*. <https://doi.org/10.3390/healthcare8020185>
- Knezevic, S. Z., Evans, S. P., & Mainz, M. (2003). Row Spacing Influences the Critical Timing for Weed Removal in Soybean (*Glycine max*). *Weed Technology*, 17(4), 666–673. <https://doi.org/10.1614/WT02-49>
- Kounalakis, T., Triantafyllidis, G. A., & Nalpantidis, L. (2016). Weed recognition framework for robotic precision farming. *IST 2016 - 2016 IEEE International Conference on Imaging Systems and Techniques, Proceedings*. <https://doi.org/10.1109/IST.2016.7738271>
- Kounalakis, T., Triantafyllidis, G. A., & Nalpantidis, L. (2018). Image-based recognition framework for robotic weed control systems. *Multimedia Tools and Applications*. <https://doi.org/10.1007/s11042-017-5337-y>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25. <http://code.google.com/p/cuda-convnet/>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84–90. <https://doi.org/10.1145/3065386>
- Krutz, L. J., Senseman, S. A., Zablotowicz, R. M., & Matocha, M. A. (2005). Reducing herbicide runoff from agricultural fields with vegetative filter strips: a review. *Weed Science*, 53(3), 353–367. <https://doi.org/10.1614/WS-03-079R2>
- Kwon, Y. (2018). *Yolo_Label: GUI for marking bounded boxes of objects in images for training neural network*. https://github.com/developer0hye/Yolo_Label
- Lammie, C., Olsen, A., Carrick, T., & Rahimi Azghadi, M. (2019). Low-Power and High-Speed Deep FPGA Inference Engines for Weed Classification at the Edge. *IEEE Access*, 7, 51171–51184. <https://doi.org/10.1109/ACCESS.2019.2911709>
- Lan, Y., Huang, K., Yang, C., Lei, L., Ye, J., Zhang, J., Zeng, W., Zhang, Y., & Deng, J. (2021). Real-time Identification of Rice Weeds by UAV Low-Altitude Remote Sensing Based on Improved Semantic Segmentation Model. *Remote Sensing 2021, Vol. 13, Page 4370*, 13(21), 4370. <https://doi.org/10.3390/RS13214370>

- Law, H., & Deng, J. (2019). CornerNet: Detecting Objects as Paired Keypoints. *ArXiv:1808.01244 [Cs]*.
- Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., & Yan, S. (2017). Perceptual generative adversarial networks for small object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1222–1230.
- Li, N., Zhang, X., Zhang, C., Guo, H., Sun, Z., & Wu, X. (2019). Real-Time Crop Recognition in Transplanted Fields With Prominent Weed Growth: A Visual-Attention-Based Approach. *IEEE Access*, 7, 185310–185321. <https://doi.org/10.1109/ACCESS.2019.2942158>
- Li, Y., Al-Sarayreh, M., Irie, K., Hackell, D., Bourdot, G., Reis, M. M., & Ghamkhar, K. (2021). Identification of Weeds Based on Hyperspectral Imaging and Machine Learning. *Frontiers in Plant Science*, 11. <https://doi.org/10.3389/fpls.2020.611622>
- Liu, H., Lee, S. H., & Saunders, C. (2014). Development of a machine vision system for weed detection during both of off-season and in-season in broadacre no-tillage cropping lands. *American Journal of Agricultural and Biological Sciences*, 9(2), 174–193. <https://doi.org/10.3844/ajabssp.2014.174.193>
- Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path Aggregation Network for Instance Segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. <https://doi.org/10.1109/CVPR.2018.00913>
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. *ArXiv:1512.02325 [Cs]*, 9905, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2
- Lloyd, S. P. (1982). Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2), 129–137. <https://doi.org/10.1109/TIT.1982.1056489>
- LÓPEZ-GRANADOS, F. (2011). Weed detection for site-specific weed management: mapping and real-time approaches. *Weed Research*, 51(1), 1–11. <https://doi.org/10.1111/J.1365-3180.2010.00829.X>
- López-Granados, F., Torres-Sánchez, J., De Castro, A. I., Serrano-Pérez, A., Mesas-Carrascosa, F. J., & Peña, J. M. (2016a). Object-based early monitoring of a grass weed in a grass crop using high resolution UAV imagery. *Agronomy for Sustainable Development*, 36(4). <https://doi.org/10.1007/s13593-016-0405-7>

- López-Granados, F., Torres-Sánchez, J., De Castro, A. I., Serrano-Pérez, A., Mesas-Carrascosa, F. J., & Peña, J. M. (2016b). Object-based early monitoring of a grass weed in a grass crop using high resolution UAV imagery. *Agronomy for Sustainable Development*, 36(4), 1–12. <https://doi.org/10.1007/s13593-016-0405-7>
- Lottes, P., Behley, J., Milioto, A., & Stachniss, C. (2018). Fully convolutional networks with sequential information for robust crop and weed detection in precision farming. *IEEE Robotics and Automation Letters*, 3(4), 2870–2877. <https://doi.org/10.1109/LRA.2018.2846289>
- Lottes, P., Khanna, R., Pfeifer, J., Siegwart, R., & Stachniss, C. (2017). UAV-based crop and weed classification for smart farming. *Proceedings - IEEE International Conference on Robotics and Automation*, 3024–3031. <https://doi.org/10.1109/ICRA.2017.7989347>
- Louargant, M., Jones, G., Faroux, R., Paoli, J.-N., Maillot, T., Gée, C., & Villette, S. (2018). Unsupervised Classification Algorithm for Early Weed Detection in Row-Crops by Combining Spatial and Spectral Information. *Remote Sensing*, 10(5), 761. <https://doi.org/10.3390/rs10050761>
- Louargant, M., Villette, S., Jones, G., Vigneau, N., Paoli, J. N., & Gée, C. (2017). Weed detection by UAV: simulation of the impact of spectral mixing in multispectral images. *Precision Agriculture*, 18(6), 932–951. <https://doi.org/10.1007/S11119-017-9528-3/FIGURES/11>
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- M, M., V, U., D, S., P, M., P, M., C, E., A, U., T, E., S, V., DM, B., P, K., T, B., M, S., C, C., S, C., N, H., K, R., KE, M., J, J., ... C, O.-S. (2014). A benchmark for comparison of cell tracking algorithms. *Bioinformatics (Oxford, England)*, 30(11), 1609–1617. <https://doi.org/10.1093/BIOINFORMATICS/BTU080>
- Machleb, J., Peteinatos, G. G., Kollenda, B. L., Andújar, D., & Gerhards, R. (2020). Sensor-based mechanical weed control: Present state and prospects. In *Computers and Electronics in Agriculture*. <https://doi.org/10.1016/j.compag.2020.105638>
- Mahmoodi, S., & Rahimi, A. (2009). Estimation of critical period for weed control in corn in Iran. *World Academy of Science, Engineering and Technology*, 49, 67–72.

- Mateos Matilla, D., Lozano Murciego, Á., Jiménez-Bravo, D. M., Sales Mendes, A., & Leithardt, V. R. Q. (2021). Low-cost Edge Computing devices and novel user interfaces for monitoring pivot irrigation systems based on Internet of Things and LoRaWAN technologies. *Biosystems Engineering*. <https://doi.org/10.1016/J.BIOSYSTEMSENG.2021.07.010>
- Matese, A., Toscano, P., Di Gennaro, S. F., Genesio, L., Vaccari, F. P., Primicerio, J., Belli, C., Zaldei, A., Bianconi, R., & Gioli, B. (2015). Intercomparison of UAV, Aircraft and Satellite Remote Sensing Platforms for Precision Viticulture. *Remote Sensing*, 7(3), 2971–2990. <https://doi.org/10.3390/rs70302971>
- Mazzia, V., Khaliq, A., Salvetti, F., & Chiaberge, M. (2020). Real-time apple detection system using embedded systems with hardware accelerators: An edge AI application. *IEEE Access*, 8, 9102–9114. <https://doi.org/10.1109/ACCESS.2020.2964608>
- Messina, G., & Modica, G. (2020). Applications of UAV Thermal Imagery in Precision Agriculture: State of the Art and Future Research Outlook. *Remote Sensing 2020, Vol. 12, Page 1491*, 12(9), 1491. <https://doi.org/10.3390/RS12091491>
- Meyer, G. E., Neto, J. C., Jones, D. D., & Hindman, T. W. (2004). Intensified fuzzy clusters for classifying plant, soil, and residue regions of interest from color images. *Computers and Electronics in Agriculture*, 42(3), 161–180. <https://doi.org/10.1016/j.compag.2003.08.002>
- Micasense. (2019). *Automatic Calibration Panel Detection (QR Mode) – MicaSense Knowledge Base*. <https://support.micasense.com/hc/en-us/articles/360018618774>
- Milioto, A., Lottes, P., & Stachniss, C. (2017). Real-time Blob-wise Sugar Beets vs Weeds Classification for Monitoring Fields Using Convolutional Neural Networks. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 4.
- Milioto, A., Lottes, P., & Stachniss, C. (2018). Real-Time Semantic Segmentation of Crop and Weed for Precision Agriculture Robots Leveraging Background Knowledge in CNNs. *Proceedings - IEEE International Conference on Robotics and Automation*, 2229–2235. <https://doi.org/10.1109/ICRA.2018.8460962>
- Milioto, A., & Stachniss, C. (2019). Bonnet: An open-source training and deployment framework for semantic segmentation in robotics using CNNs. *Proceedings - IEEE International Conference on Robotics and Automation*, 2019-May, 7094–7100. <https://doi.org/10.1109/ICRA.2019.8793510>

- Mitchell, T. M. (1997). Machine learning. In *Machine Learning* (Vol. 45, Issue 13). McGraw-Hill.
<https://books.google.ca/books?id=EoYBngEACAAJ&dq=mitchell+machine+learning+1997&hl=en&sa=X&ved=0ahUKEwiomdqfj8TkAhWGslkKHRCbAtoQ6AEIKjAA>
- Neyshabur, B., Bhojanapalli, S., McAllester, D., & Srebro, N. (2017). Exploring Generalization in Deep Learning. *Advances in Neural Information Processing Systems, 2017-December*, 5948–5957. <https://arxiv.org/abs/1706.08947v2>
- NVIDIA. (2021a). *Achieving FP32 Accuracy for INT8 Inference Using Quantization Aware Training with NVIDIA TensorRT* . <https://developer.nvidia.com/blog/achieving-fp32-accuracy-for-int8-inference-using-quantization-aware-training-with-tensorrt/>
- NVIDIA. (2021b). *Embedded Systems Developer Kits & Modules from NVIDIA Jetson*. <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/>
- NVIDIA. (2021c). *NVIDIA CUDA*. <https://developer.nvidia.com/cuda-zone>
- NVIDIA. (2021d). *NVIDIA cuDNN*. <https://developer.nvidia.com/cudnn>
- NVIDIA. (2021e). *NVIDIA DeepStream SDK* . <https://developer.nvidia.com/deepstream-sdk>
- NVIDIA. (2021f). *NVIDIA TensorRT*. <https://developer.nvidia.com/tensorrt>
- NVIDIA Newsroom. (2019). *NVIDIA Announces Jetson Xavier NX, World's Smallest Supercomputer for AI at the Edge*. <https://nvidianews.nvidia.com/news/nvidia-announces-jetson-xavier-nx-worlds-smallest-supercomputer-for-ai-at-the-edge>
- O'Grady, M. J., Langton, D., & O'Hare, G. M. P. (2019). Edge computing: A tractable model for smart agriculture? *Artificial Intelligence in Agriculture*, 3, 42–51. <https://doi.org/10.1016/j.aiia.2019.12.001>
- Olsen, A., Konovalov, D. A., Philippa, B., Ridd, P., Wood, J. C., Johns, J., Banks, W., Girgenti, B., Kenny, O., Whinney, J., Calvert, B., Azghadi, M. R., & White, R. D. (2019). DeepWeeds: A Multiclass Weed Species Image Dataset for Deep Learning. *Scientific Reports 2019 9:1*, 9(1), 1–12. <https://doi.org/10.1038/s41598-018-38343-3>
- Osorio, K., Puerto, A., Pedraza, C., Jamaica, D., & Rodríguez, L. (2020). A Deep Learning Approach for Weed Detection in Lettuce Crops Using Multispectral Images. *AgriEngineering*, 2(3), 471–488. <https://doi.org/10.3390/agriengineering2030032>
- Padilla, R., Passos, W. L., Dias, T. L. B., Netto, S. L., & da Silva, E. A. B. (2021). A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit. *Electronics 2021, Vol. 10, Page 279, 10(3)*, 279. <https://doi.org/10.3390/ELECTRONICS10030279>

- Pantazi, X. E., Tamouridou, A. A., Alexandridis, T. K., Lagopodi, A. L., Kashefi, J., & Moshou, D. (2017). Evaluation of hierarchical self-organising maps for weed mapping using UAS multispectral imagery. *Computers and Electronics in Agriculture*, 139, 224–230. <https://doi.org/10.1016/j.compag.2017.05.026>
- Pérez-Ortiz, M., Peña, J. M., Gutiérrez, P. A., Torres-Sánchez, J., Hervás-Martínez, C., & López-Granados, F. (2016). Selecting patterns and features for between- and within- crop-row weed mapping using UAV-imagery. *Expert Systems with Applications*, 47, 85–94. <https://doi.org/10.1016/J.ESWA.2015.10.043>
- Peterson, G. E. (1967). The Discovery and Development of 2,4-D. *Agricultural History*, 41(3), 243–254. <http://www.jstor.org/stable/3740338>
- Petrich, L., Lohrmann, G., Neumann, M., Martin, F., Frey, A., Stoll, A., & Schmidt, V. (2020). Detection of *Colchicum autumnale* in drone images, using a machine-learning approach. *Precision Agriculture*, 21(6), 1291–1303. <https://doi.org/10.1007/s11119-020-09721-7>
- Potena, C., Nardi, D., & Pretto, A. (2017). Fast and accurate crop and weed identification with summarized train sets for precision agriculture. *Advances in Intelligent Systems and Computing*, 531, 105–121. https://doi.org/10.1007/978-3-319-48036-7_9
- Quinlan, J. (1992). Learning with continuous classes. Proceedings of AI'92. World Scientific. *5th Australian Joint Conference on Artificial Intelligence*, 92, 343–348.
- Rasti, S., Bleakley, C. J., Silvestre, G. C. M., Holden, N. M., Langton, D., & O'Hare, G. M. P. (2021). Crop growth stage estimation prior to canopy closure using deep learning algorithms. *Neural Computing and Applications*, 33(5), 1733–1743. <https://doi.org/10.1007/S00521-020-05064-6/FIGURES/3>
- Reddy, K. N., Huang, Y., Lee, M. A., Nandula, V. K., Fletcher, R. S., Thomson, S. J., & Zhao, F. (2014). Glyphosate-resistant and glyphosate-susceptible Palmer amaranth (*Amaranthus palmeri* S. Wats.): hyperspectral reflectance properties of plants and potential for classification. *Pest Management Science*, 70(12), 1910–1917. <https://doi.org/10.1002/ps.3755>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *ArXiv:1506.02640 [Cs]*.

- Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. *ArXiv:1506.01497 [Cs]*. <http://arxiv.org/abs/1506.01497>
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2016.2577031>
- Romera, E., Alvarez, J. M., Bergasa, L. M., & Arroyo, R. (2018). ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1), 263–272. <https://doi.org/10.1109/TITS.2017.2750080>
- Ronneberger, O., Fischer, P., & Brox, T. (2015a). U-net: Convolutional networks for biomedical image segmentation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-319-24574-4_28
- Ronneberger, O., Fischer, P., & Brox, T. (2015b). U-Net: Convolutional Networks for Biomedical Image Segmentation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9351, 234–241. https://doi.org/10.1007/978-3-319-24574-4_28
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), 211–252. <https://doi.org/10.1007/s11263-015-0816-y>
- Sa, I., Chen, Z., Popovic, M., Khanna, R., Liebisch, F., Nieto, J., & Siegwart, R. (2018). WeedNet: Dense Semantic Weed Classification Using Multispectral Images and MAV for Smart Farming. *IEEE Robotics and Automation Letters*, 3(1), 588–595. <https://doi.org/10.1109/LRA.2017.2774979>
- Sa, I., Popović, M., Khanna, R., Chen, Z., Lottes, P., Liebisch, F., Nieto, J., Stachniss, C., Walter, A., & Siegwart, R. (2018). WeedMap: A Large-Scale Semantic Weed Mapping Framework Using Aerial Multispectral Imaging and Deep Neural Network for Precision Farming. *Remote Sensing 2018, Vol. 10, Page 1423*, 10(9), 1423. <https://doi.org/10.3390/RS10091423>

- Saha, A. K., Saha, J., Ray, R., Sircar, S., Dutta, S., Chattopadhyay, S. P., & Saha, H. N. (2018). IOT-based drone for improvement of crop quality in agricultural field. *2018 IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC 2018, 2018-January*, 612–615. <https://doi.org/10.1109/CCWC.2018.8301662>
- Sanders, J., Everman, W. J., Austin, R., Roberson, G. T., & Richardson, R. J. (2019). *Weed species differentiation using spectral reflectance land image classification*. 24. <https://doi.org/10.1117/12.2519306>
- Sapkota, B., Singh, V., Cope, D., Valasek, J., & Bagavathiannan, M. (2020). Mapping and Estimating Weeds in Cotton Using Unmanned Aerial Systems-Borne Imagery. *AgriEngineering*, 2(2), 350–366. <https://doi.org/10.3390/agriengineering2020024>
- Satyanarayanan, M., Bahl, P., Cáceres, R., & Davies, N. (2009). The case for VM-based cloudlets in mobile computing. *IEEE Pervasive Computing*, 8(4), 14–23. <https://doi.org/10.1109/MPRV.2009.82>
- Scherrer, B. J. (2019). *Weed and crop discrimination with hyperspectral imaging and machine learning*. Montana State University - Bozeman, College of Engineering.
- Shafie, F. A., & Rennie, D. (2012). Consumer Perceptions Towards Organic Food. *Procedia - Social and Behavioral Sciences*, 49, 360–367. <https://doi.org/10.1016/J.SBSPRO.2012.07.034>
- Shepherd, M., Turner, J. A., Small, B., & Wheeler, D. (2020). Priorities for science to overcome hurdles thwarting the full promise of the ‘digital agriculture’ revolution. *Journal of the Science of Food and Agriculture*, 100(14), 5083–5092. <https://doi.org/10.1002/JSFA.9346>
- Shi, P., Xu, X., Ni, J., Xin, Y., Huang, W., & Han, S. (2021). Underwater Biological Detection Algorithm Based on Improved Faster-RCNN. *Water 2021, Vol. 13, Page 2420, 13(17)*, 2420. <https://doi.org/10.3390/W13172420>
- Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge {Computing}: {Vision} and {Challenges}. *IEEE Internet of Things Journal*, 3(5), 637–646. <https://doi.org/10.1109/IIOT.2016.2579198>
- Shi, W., & Dustdar, S. (2016). The Promise of Edge Computing. *Computer*, 49(5), 78–81. <https://doi.org/10.1109/MC.2016.145>

- Shirzadifar, A., Bajwa, S., Nowatzki, J., & Shojaeiarani, J. (2020). Development of spectral indices for identifying glyphosate-resistant weeds. *Computers and Electronics in Agriculture*, 170, 105276. <https://doi.org/10.1016/j.compag.2020.105276>
- Silva, G. (2018). Feeding the World in 2050 and Beyond--Part 1: Productivity Challenges. *Michigan State University Extension--3 December*.
- Sivakumar, A. N. V., Li, J., Scott, S., Psota, E., Jhala, A. J., Luck, J. D., & Shi, Y. (2020a). Comparison of object detection and patch-based classification deep learning models on mid-to late-season weed detection in UAV imagery. *Remote Sensing*. <https://doi.org/10.3390/rs12132136>
- Sivakumar, A. N. V., Li, J., Scott, S., Psota, E., Jhala, A. J., Luck, J. D., & Shi, Y. (2020b). Comparison of object detection and patch-based classification deep learning models on mid-to late-season weed detection in UAV imagery. *Remote Sensing*. <https://doi.org/10.3390/rs12132136>
- Slaughter, D. C. (2014). The Biological Engineer: Sensing the Difference Between Crops and Weeds. *Automation: The Future of Weed Control in Cropping Systems*, 71–95. https://doi.org/10.1007/978-94-007-7512-1_5
- Slaughter, D. C., Giles, D. K., Fennimore, S. A., & Smith, R. F. (2008). Multispectral Machine Vision Identification of Lettuce and Weed Seedlings for Automated Weed Control. *Weed Technology*, 22(2), 378–384. <https://doi.org/10.1614/WT-07-104.1>
- Soltani, N., Dille, J. A., Burke, I. C., Everman, W. J., VanGessel, M. J., Davis, V. M., & Sikkema, P. H. (2016). Potential Corn Yield Losses from Weeds in North America. *Weed Technology*, 30(4), 979–984. <https://doi.org/10.1614/WT-D-16-00046.1>
- Soltani, N., Dille, J. A., Burke, I. C., Everman, W. J., Vangessel, M. J., Davis, V. M., & Sikkema, P. H. (2017). Perspectives on Potential Soybean Yield Losses from Weeds in North America. *Weed Technology*, 31(1), 148–154. <https://doi.org/10.1017/wet.2016.2>
- Soltani, N., Nurse, R. E., Jhala, A. J., & Sikkema, P. H. (2019). The critical weed-free period in glyphosate-resistant soybean in ontario is similar to previous estimates using glyphosate-susceptible soybean. *Canadian Journal of Plant Science*, 99(4), 437–443. <https://doi.org/10.1139/CJPS-2018-0049>
- Soviany, P., & Ionescu, R. T. (2018). Optimizing the Trade-off between Single-Stage and Two-Stage Object Detectors using Image Difficulty Prediction. *ArXiv:1803.08707 [Cs]*.

- Suh, H. K., Hofstee, J. W., IJsselmuiden, J., & van Henten, E. J. (2018). Sugar beet and volunteer potato classification using Bag-of-Visual-Words model, Scale-Invariant Feature Transform, or Speeded Up Robust Feature descriptors and crop row information. *Biosystems Engineering*. <https://doi.org/10.1016/j.biosystemseng.2017.11.015>
- Sujaritha, M., Annadurai, S., Satheeshkumar, J., Kowshik Sharan, S., & Mahesh, L. (2017). Weed detecting robot in sugarcane fields using fuzzy real time classifier. *Computers and Electronics in Agriculture*, 134, 160–171. <https://doi.org/10.1016/J.COMPAG.2017.01.008>
- Swain, K. C., Nørremark, M., Jørgensen, R. N., Midtby, H. S., & Green, O. (2011). Weed identification using an automated active shape matching (AASM) technique. *Biosystems Engineering*, 110(4), 450–457. <https://doi.org/10.1016/J.BIOSYSTEMSENG.2011.09.011>
- Sylvester, G. (2018a). *E-agriculture in action: Drones for agriculture*. Food and Agriculture Organization of the United Nations and International~....
- Sylvester, G. (2018b). *E-agriculture in action: Drones for agriculture*. Food and Agriculture Organization of the United Nations and International~....
- T., S., T., S., G.S., S. G., S., S., & Kumaraswamy, R. (2019). Performance Comparison of Weed Detection Algorithms. *2019 International Conference on Communication and Signal Processing (ICCSP)*, 843–847. <https://doi.org/10.1109/ICCSP.2019.8698094>
- Taghadomi-Saberi, S., ... A. H.-E. I. C., & 2015, undefined. (2015). Improving field management by machine vision-a review. *Cigrjournal.Org*, 17(3). <http://cigrjournal.org/index.php/Ejournal/article/view/3250>
- Taghizadeh, M., Gowen, A. A., & O'Donnell, C. P. (2011). Comparison of hyperspectral imaging with conventional RGB imaging for quality evaluation of *Agaricus bisporus* mushrooms. *Biosystems Engineering*, 108(2), 191–194. <https://doi.org/10.1016/J.BIOSYSTEMSENG.2010.10.005>
- Tahir Ata-Ul-Karim, S., Liu, X., Lu, Z., Yuan, Z., Zhu, Y., & Cao, W. (2016). In-season estimation of rice grain yield using critical nitrogen dilution curve. *Field Crops Research*, 195, 1–8. <https://doi.org/10.1016/J.FCR.2016.04.027>
- Tang, J. L., Zhang, Z. G., Wang, D., Xin, J., & He, L. J. (2018). Research on weeds identification based on K-means feature learning. *Soft Computing*, 22(22), 7649–7658. <https://doi.org/10.1007/S00500-018-3125-X>

- Teimouri, N., Dyrmann, M., Nielsen, P. R., Mathiassen, S. K., Somerville, G. J., & Jørgensen, R. N. (2018). Weed Growth Stage Estimator Using Deep Convolutional Neural Networks. *Sensors* 2018, Vol. 18, Page 1580, 18(5), 1580. <https://doi.org/10.3390/S18051580>
- Thompson, N. C., Greenewald, K., Lee, K., & Manso, G. F. (2020). The Computational Limits of Deep Learning. *ArXiv.Org*. <https://arxiv.org/abs/2007.05558v1>
- Tkachenko, M., Malyuk, M., Shevchenko, N., Holmanyuk, A., & Liubimov, N. (2020). *Label Studio: Data labeling software*.
- Tona, E., Calcante, A., & Oberti, R. (2018). The profitability of precision spraying on specialty crops: a technical–economic analysis of protection equipment at increasing technological levels. *Precision Agriculture*, 19(4), 606–629.
- Tong, K., Wu, Y., & Zhou, F. (2020). Recent advances in small object detection based on deep learning: A review. *Image and Vision Computing*, 97, 103910. <https://doi.org/10.1016/J.IMAVIS.2020.103910>
- Torres-Sánchez, J., López-Granados, F., De Castro, A. I., & Peña-Barragán, J. M. (2013). Configuration and Specifications of an Unmanned Aerial Vehicle (UAV) for Early Site Specific Weed Management. *PLoS ONE*, 8(3), e58210. <https://doi.org/10.1371/journal.pone.0058210>
- Travlos, I. S., Economou, G., & Kanatas, P. J. (2011). Corn and Barnyardgrass Competition as Influenced by Relative Time of Weed Emergence and Corn Hybrid. *Agronomy Journal*, 103(1), 1–6. <https://doi.org/10.2134/AGRONJ2010.0245>
- Trimble. (n.d.). {GreenSeeker} {Crop} {Sensing} {System}. In *Trimble Agriculture*.
- Tryon, R. C. (1957). Communality of a variable: Formulation by cluster analysis. *Psychometrika*, 22(3), 241–260. <https://doi.org/10.1007/BF02289125>
- Tursun, N., Datta, A., Budak, S., Kantarci, Z., & Knezevic, S. (2015). Row spacing impacts the critical period for weed control. *Springer*. <https://doi.org/10.1007/s12600-015-0494-x>
- Tyohemba, R. L., Pillay, L., & Humphries, M. S. (2021). Bioaccumulation of current-use herbicides in fish from a global biodiversity hotspot: Lake St Lucia, South Africa. *Chemosphere*, 284, 131407. <https://doi.org/10.1016/J.CHEMOSPHERE.2021.131407>
- United Nations. (2019). *World population prospects Highlights, 2019 revision Highlights, 2019 revision*.
- USDA ERS. (2021). *Organic Market Summary and Trends*.

- Verucchi, M., Brilli, G., Sapienza, D., Verasani, M., Arena, M., Gatti, F., Capotondi, A., Cavicchioli, R., Bertogna, M., & Solieri, M. (2020). A Systematic Assessment of Embedded Neural Networks for Object Detection. *IEEE Symposium on Emerging Technologies and Factory Automation, ETFA*, 2020-September, 937–944. <https://doi.org/10.1109/ETFA46521.2020.9212130>
- Wan, S., & Goudos, S. (2020). Faster R-CNN for multi-class fruit detection using a robotic vision system. *Computer Networks*. <https://doi.org/10.1016/j.comnet.2019.107036>
- Wang, A., Xu, Y., Wei, X., & Cui, B. (2020). Semantic Segmentation of Crop and Weed using an Encoder-Decoder Network and Image Enhancement Method under Uncontrolled Outdoor Illumination. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2020.2991354>
- Wang, A., Zhang, W., & Wei, X. (2019a). A review on weed detection using ground-based machine vision and image processing techniques. *Computers and Electronics in Agriculture*, 158, 226–240. <https://doi.org/10.1016/j.compag.2019.02.005>
- Wang, A., Zhang, W., & Wei, X. (2019b). A review on weed detection using ground-based machine vision and image processing techniques. *Computers and Electronics in Agriculture*, 158, 226–240. <https://doi.org/10.1016/J.COMPAG.2019.02.005>
- Wang, L., Lan, Y., Yue, X., Ling, K., Cen, Z., Cheng, Z., Liu, Y., & Wang, J. (2019). Vision-based adaptive variable rate spraying approach for unmanned aerial vehicles. *International Journal of Agricultural and Biological Engineering*, 12(3), 18–26. <https://doi.org/10.25165/IJABE.V12I3.4358>
- Wang, Q., Cheng, M., Xiao, X., Yuan, H., Zhu, J., Fan, C., & Zhang, J. (2021). An image segmentation method based on deep learning for damage assessment of the invasive weed *Solanum rostratum* Dunal. *Computers and Electronics in Agriculture*, 188, 106320. <https://doi.org/10.1016/J.COMPAG.2021.106320>
- Weed Science. (2021). *Chronological Increase in Resistant Weeds Globally*. <http://weedsience.org/Pages/Graphs/SOAGraph.aspx>
- Weis, M., & Gerhards, R. (2008). Detection of weeds using image processing and clustering. *Bornimer Agrartechnische Berichte*, 69(2008), 138–144.
- Westwood, J. H., Charudattan, R., Duke, S. O., Fennimore, S. A., Marrone, P., Slaughter, D. C., Swanton, C., & Zollinger, R. (2018). Weed Management in 2050: Perspectives on the Future of Weed Science. *Weed Science*, 66(3), 275–285. <https://doi.org/10.1017/wsc.2017.78>

- Wilf, P., Zhang, S., Chikkerur, S., Little, S. A., Wing, S. L., & Serre, T. (2016). Computer vision cracks the leaf code. *Proceedings of the National Academy of Sciences of the United States of America*. <https://doi.org/10.1073/pnas.1524473113>
- Wu, L., Zhu, Z., & E, W. (2017). Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes. *Ariv.Org*. <https://arxiv.org/abs/1706.10239v2>
- Wu, S., Zhang, H. R., Valiant, G., & Ré, C. (2020). On the generalization effects of linear transformations in data augmentation. *37th International Conference on Machine Learning, ICML 2020*.
- Xiang, H., & Tian, L. (2011). Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV). *Biosystems Engineering*, 108(2), 174–190. <https://doi.org/10.1016/j.biosystemseng.2010.11.010>
- Yang, W., Li, Z., Wang, C., & Li, J. (2020). A multi-task Faster R-CNN method for 3D vehicle detection based on a single image. *Applied Soft Computing Journal*. <https://doi.org/10.1016/j.asoc.2020.106533>
- Young, S. L. (2018). Beyond Precision Weed Control: A Model for True Integration. *Weed Technology*, 32(1), 7–10. <https://doi.org/10.1017/wet.2017.70>
- Young, S. L., Pierce, F. J., & Nowak, P. (2014). Introduction: Scope of the Problem—Rising Costs and Demand for Environmental Safety for Weed Control. In S. L. Young & F. J. Pierce (Eds.), *Automation: The Future of Weed Control in Cropping Systems* (pp. 1–8). Springer Netherlands.
- Yu, J., Schumann, A. W., Sharpe, S. M., Li, X., & Boyd, N. S. (2020). Detection of grassy weeds in bermudagrass with deep convolutional neural networks. *Weed Science*, 68(5), 545–552. <https://doi.org/10.1017/wsc.2020.46>
- Yu, J., Sharpe, S. M., Schumann, A. W., & Boyd, N. S. (2019). Deep learning for image-based weed detection in turfgrass. *European Journal of Agronomy*, 104, 78–84. <https://doi.org/10.1016/j.eja.2019.01.004>
- Yun, S., Han, D., Chun, S., Oh, S. J., Choe, J., & Yoo, Y. (2019). CutMix: Regularization strategy to train strong classifiers with localizable features. *Proceedings of the IEEE International Conference on Computer Vision*. <https://doi.org/10.1109/ICCV.2019.00612>

- Zhang, H., Lee, H. J., & Ko, S. B. (2018). Efficient Fixed/Floating-Point Merged Mixed-Precision Multiply-Accumulate Unit for Deep Learning Processors. *Proceedings - IEEE International Symposium on Circuits and Systems*, 2018-May. <https://doi.org/10.1109/ISCAS.2018.8351354>
- Zhang, J., Basso, B., Price, R. F., Putman, G., & Shuai, G. (2018). Estimating plant distance in maize using Unmanned Aerial Vehicle (UAV). *PLOS ONE*, 13(4), e0195223. <https://doi.org/10.1371/JOURNAL.PONE.0195223>
- Zhang, S., Guo, J., & Wang, Z. (2019). Combing K-means Clustering and Local Weighted Maximum Discriminant Projections for Weed Species Recognition. *Frontiers in Computer Science*, 1, 4. <https://doi.org/10.3389/fcomp.2019.00004>
- Zhang, S., Huang, W., & Wang, Z. (2021). Combing modified Grabcut, K-means clustering and sparse representation classification for weed recognition in wheat field. *Neurocomputing*, 452, 665–674. <https://doi.org/10.1016/j.neucom.2020.06.140>
- Zhang, W., Hansen, M. F., Volonakis, T. N., Smith, M., Smith, L., Wilson, J., Ralston, G., Broadbent, L., & Wright, G. (2018). Broad-Leaf Weed Detection in Pasture. *2018 3rd IEEE International Conference on Image, Vision and Computing, ICIVC 2018*, 101–105. <https://doi.org/10.1109/ICIVC.2018.8492831>
- Zhang, Z., Kayacan, E., Thompson, B., & Chowdhary, G. (2020). High precision control and deep learning-based corn stand counting algorithms for agricultural robot. *Autonomous Robots* 2020 44:7, 44(7), 1289–1302. <https://doi.org/10.1007/S10514-020-09915-Y>
- Zhao, Y., Gong, L., Huang, Y., & Liu, C. (2016). A review of key techniques of vision-based control for harvesting robot. *Computers and Electronics in Agriculture*, 127, 311–323. <https://doi.org/10.1016/J.COMPAG.2016.06.022>
- Zou, K., Chen, X., Zhang, F., Zhou, H., & Zhang, C. (2021). A Field Weed Density Evaluation Method Based on UAV Imaging and Modified U-Net. *Remote Sensing*, 13(2), 310. <https://doi.org/10.3390/rs13020310>