MACHINE VISION METHODS FOR EVALUATING PLANT STAND COUNT AND WEED CLASSIFICATION USING OPEN-SOURCE PLATFORMS

A Thesis
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Harsh Pathak

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Program:
Agricultural and Biosystems Engineering

July 2021

Fargo, North Dakota

NORTH DAKOTA STATE UNIVERSITY

Graduate School

Title

MACHINE VISION METHODS FOR EVALUATING PLANT STAND COUNT AND WEED CLASSIFICATION USING OPEN-SOURCE PLATFORMS

By

Harsh Pathak

The supervisory committee certifies that this thesis complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Dr. Igathinathane Cannayen
Chair

Dr. Paulo Flores

Dr. Kirk A. Howatt

Approved:

08/04/2021
Date

Dr. Kenneth Hellevang
Department Chair

ABSTRACT

Evaluating plant stand count or classifying weeds by manual scouting is time-consuming, laborious, and subject to human errors. Proximal remote sensed imagery used in conjunction with machine vision algorithms can be used for these purposes. Despite its great potential, the rate of using these technologies is still slow due to their subscription cost and data privacy issues. Therefore, in this research, open-source image processing software, ImageJ and Python that support in-house processing, was used to develop algorithms to evaluate stand count, develop spatial distribution maps, and classify the four common weeds of North Dakota. A novel sliding and shifting region of interest method was developed for plant stand count. Handcrafted simple image processing and machine learning approaches with shape features were successfully employed for weed species classification. Such tools and methodologies using open-source platforms can be extended to other scenarios and are expected to be impactful and helpful to stakeholders.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my major advisor and committee chair, Dr. Igathinathane Cannayen, Associate Professor, Department of Agricultural and Biosystems Engineering (ABEN), North Dakota State University (NDSU), who hired me as a master's student and provided me an amazing opportunity to work in his research group. I sincerely appreciate all his patience, support, and encouragement throughout the master's program. He taught me how to write articles, conduct independent research, and present research at academic conferences. His ideas, wisdom, and suggestions have helped me sail smoothly through my graduate studies. He also provided me ample opportunities to demonstrate my research work and helped me to explore new fields/ideas. Without his guidance and support, this thesis would not have been possible.

I would like to extend my sincere thanks to my committee members, Dr. Paulo Flores, Assistant Professor ABEN, NDSU, and Dr. Kirk A. Howatt, Associate Professor, Department of Plant Science, NDSU, for their valuable time, guidance, suggestions, and encouragement throughout my research projects.

This project was supported by the North Dakota Corn Council (Fund number: FAR0030691). Research facilities are provided by the Northern Great Plains Research Laboratory (NGPRL), USDA-ARS, Mandan, ND. The funding and facility support from these organizations are highly appreciated.

I must also thank Dr. Ravi Kiran Yellavajjala, Assistant Professor, Department of Civil and Environmental Engineering, NDSU who taught my Machine Learning course, one of the

most demanding courses that developed a useful skillset for my career. I applied the learned skills for my thesis. Many thanks to Dr. Stephanie Day, Associate Professor, Department of Geosciences, NDSU for teaching an amazing course on GIS that helped me to explore the new field of remote sensing. I am thankful to the NGPRL scientists Dr. David Archer and Dr. John Hendrickson for reviewing my manuscript. I would also like to thank Dr. Zhao Zhang for giving his valuable comments and suggestions. Special thanks to Mr. Blaine Schatz, Dr. Kramar David, and Dr. Mark Halvorson from Carrington Research Center, Carrington, ND, for helping with the UAV imagery data collection.

Special thanks to Dr. Rajendra Singh, Dr. Ashish Ratn Mishra, Dr. Arpan Sherring, Dr. Rajesh Singh, Dr. Rajendra Kumar Isaac, and Mrs. Jyotika Kashyap, my bachelor's university professors and school teacher who had encouraged me constantly. I would like to thank Mrs. Tammi Neville and Mr. Andrew Taylor from the Center for Writers for providing continuous feedback on my writing. I would also like to thank Mr. Steve Jeffery and Ms. Nicole Juve for their support in writing the thesis.

Special thanks to Dr. Ruchi Joshi and Ankur Bhardwaj who always trusted me, motivated me, and treated me as their family member. Many thanks to Sunil GC, Subhashree N. Srinivasagan, Dr. Sunoj Shahjahan, and Nitin Rai who supported me throughout my master's program. Also, I would like to thank Sagar Regmi, Subee Singh, Uday Bhanu Prakash Vaddevolu, Billy Graham Ram, Himanshu Chaudhary, Anant Shukla, Lalit Yadav, Pulkit Bakshi, Harshit PareshBhai Shah, Siddhant Khemmani, and Sarthak Bhardwaj for their friendship, support, and encouragement.

I would like to show my appreciation to my family members starting from my grandmother Mrs. Shobha Pathak, my mother Mrs. Kajal Pathak, my father Mr. Anant Pathak,

and my uncle Mr. Naveen Pathak who always supported me and encouraged me to work hard to achieve my goals. I would like to thank my uncle Dr. Dharmendra Saraswat, and my aunt Mrs. Vinita Saraswat whose love, support, and constant motivation always pushed me to do better.

Finally, I would like to thank all the faculty and staff of ABEN, NDSU, and friends who have been very warm, helpful, supportive, and encouraging, and any of this would not have been possible without them.

DEDICATION

I wholeheartedly dedicate this master's thesis to my late grandfather (Lt. Mr. S.N. Pathak) and late maternal grandmother (Lt. Mrs. Anandi Sharma), who always blessed me and motivated me to fulfill my dreams.

TABLE OF CONTENTS

ABSTRACT iii

ACKNOWLEDGEMENTS iv

DEDICATION vii

LIST OF TABLES xi

LIST OF FIGURES xii

LIST OF ABBREVIATIONS xv

LIST OF SYMBOLS xviii

LIST OF APPENDIX FIGURES xix

1. GENERAL INTRODUCTION 1
   - 1.1. Plant Stand Count and Weed Classification 2
   - 1.2. Statement of Hypothesis 4
   - 1.3. Statement of Objectives 4
   - 1.4. Thesis Organization 5

2. A REVIEW OF UNMANNED AERIAL VEHICLE BASED METHODS FOR EVALUATING PLANT STAND COUNT * 7
   - 2.1. Abstract 7
   - 2.2. Introduction 8
   - 2.3. Materials and Methods 12
   - 2.4. Results and Discussion 14
     - 2.4.1. Outcomes of Literature Search using Different Academic Databases . . . 14
     - 2.4.2. Review of Various Stand Count Methods — Computer Vision 16
     - 2.4.3. Research Gaps and Recommendations 29
   - 2.5. Conclusion 31

3. OPEN-SOURCE IMAGEJ BASED EARLY-SEASON CORN STAND COUNT EVALUATION AND SPATIAL DISTRIBUTION USING UAV IMAGERY * 33

3.1. Abstract . . . 33
3.2. Introduction . . . 34
3.3. Materials and Methods . . . 38
3.3.1. Image Data Acquisition . . . 38
3.3.2. Overall Image Processing Approach of Plugin Development . . . 39
3.3.3. Image Analysis Plugin Class Development . . . 40
3.3.4. Crop Row Identification using Profile Plot Method . . . 43
3.3.5. Plant Counting using Sliding and Shifting ROI (SSROI) Method . . . 44
3.3.6. Spatial Distribution Mapping . . . 46
3.4. Results and Discussion . . . 47
3.4.1. Image Segmentation . . . 47
3.4.2. Crop Row Identification . . . 49
3.4.3. SSROI Method of Plant Stand Counting . . . 50
3.4.4. Spatial Plants Distribution Mapping . . . 52
3.4.5. Plant Counting and Spatial Distribution Maps . . . 53
3.4.6. Effect of UAV Flight Height . . . 57
3.4.7. Demonstration of Field-Scale Stand Count and Spatial Distribution Results . . . 60
3.4.8. Limitations and Direction for Future Research . . . 65
3.5. Conclusion . . . 65
4. MACHINE VISION METHODS FOR WEED CLASSIFICATION USING OPEN-SOURCE PLATFORMS * . . . 67
4.1. Abstract . . . 67
4.2. Introduction . . . 68
4.3. Materials and Methods . . . 72
4.3.1. Data Acquisition . . . 72
4.3.2. Image Analysis Plugin Development for Shape Features Analysis . . . 73
4.3.3. Machine Learning Methods . . . 78
4.4. Results and Discussion . . . 82

4.4.1. Weed Identification with Simple Image Processing 82
4.4.2. Performance of Machine Learning Models 85
4.4.3. Research Limitations and Future Research Directions 89
4.5. Conclusion 90
5. GENERAL CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK 92
5.1. General Conclusions 92
5.2. Suggestions for Future Work 94
REFERENCES 96
APPENDIX A. IMAGEJ CODES FOR CROP STAND COUNT EVALUATION 116
APPENDIX B. FUNCTION FOR FEATURE EXTRACTION 122
APPENDIX C. SCRIPTS FOR DATA VISUALIZATION THROUGH R PROGRAMMING 125
APPENDIX D. PYTHON SCRIPTS FOR MACHINE LEARNING MODELS HYPERPARAMETERS TUNING 126
APPENDIX E. PYTHON SCRIPTS FOR MACHINE LEARNING CLASSIFIERS 128
APPENDIX F. PYTHON SCRIPTS FOR TRAINING AND TESTING MACHINE LEARNING . . 130
APPENDIX G. COMMON WEEDS OF NORTH DAKOTA AND THEIR IDENTIFICATION FEATURES 132
G.1. Common Lambsquarters 132
G.2. Common Purslane 133
G.3. Horseweed 134
G.4. Redroot Pigweed 134
G.5. Waterhemp 135
G.6. Common Ragweed 136
G.7. Kochia 136
APPENDIX H. GRAPHICAL REPRESENTATION OF SHAPE FEATURES USED FOR WEED CLASSIFICATION 138

LIST OF TABLES

Table Page

2.1. Details of selected articles through literature search and their citation scores. 17

3.1. Vegetative indices tested for segmentation of the color image of the field plots using the unmanned aerial vehicle. 42

4.1. Extracted shape features value ranges, and overlap of values for the four common weed species of North Dakota (training set; $n = 100$). 85

4.2. Performance of the three machine learning models on four different common weed species of North Dakota. 87

LIST OF FIGURES

Figure Page

2.1. Various remote sensing platforms in agriculture: (A) Powered parachute, (B) Blimp, (C) Unmanned aerial vehicle, (D) Helicopter and (E) Satellite. (*Image Source: Google Images*). 11

2.2. Processing stages of selecting the articles for the literature search on methods of plant stand count using UAV from appropriate keyword combinations and selected databases. 14

2.3. Number of articles published from the year 2000 to the year 2020 based on the three academic databases (Agricola, Scopus, and Web of Science), with the search performed on January 26, 2021. 15

2.4. Process involved in machine learning (ML): (A) Flowchart of developing ML models and evaluation, and (B) Steps for processing agricultural images using ML models and evaluation. 20

2.5. Generalized deep learning model development using convolutional neural network (CNN) architecture showing different CNN layers. 24

3.1. Flowchart of the image processing approach used in the plugin developed to evaluate the corn stand count and spatial distribution. 39

3.2. Integrated development environment of Fiji/ImageJ for plugin development (A), and input panel of the developed plant stand count plugin showing various inputs and output options (B). 40

3.3. Results of different segmentation methods on a sample field segment of 210 ft wide by 690 ft long: (A) ExG, (B) VARI, (C) ExR, (D) ExGR, (E) VI, (F) MExG, (G) GCC, (H) CIVE, and (I) VDVI. 48

3.4. Profile plot for different values of sigma (σ): (A) $\sigma = 0$, (B) $\sigma = 1$, (C) $\sigma = 3$, (D) $\sigma = 5$, (E) $\sigma = 7$, and (F) $\sigma = 9$. 49

3.5. Corn rows identification using profile plot and Gaussian filter. Sample binary image of corn plants (A), raw profile plot of the binary image showing irregularity with several local peaks (B), Gaussian blur filter applied to sample binary image (C), and profile plot based on the filtered image showing smooth and clear peaks for identification of the crop rows (D). 50

3.6. Input color image (Left) and the developed sliding and shifting region of interest (SSROI) method of plant stand counting (Right). 51

3.7. Spatial distribution color-coded map of corn rows showing ideal: green, doubles: blue, single skip: orange, double skip: pink, and multiple skip: red (Left), and simple plot of representative corn plants counted and location (Right). 53

3.8. Off-linear plants rows identification using the sliding and shifting (SSROI) method on a sample field segment of 210 ft wide by 690 ft long. Inserts show magnified view of the sections of the labeled plot. 55

3.9. Plant stand count labeled along the rows in UAV imagery from bottom to top and left to right. 56

3.10. Plant stand count distribution map (Green represents plants under optimum distance, blue represents doubles, orange represents single skip, pink represents double skips, and red represents multiple skips). The sample field segment dimension is 210 ft wide by 690 ft long. 58

3.11. Image segmentation and crop row detection at different heights: (A) 9.144 m, (B) 15.24 m, (C) 22.86 m, and (D) 30.5 m. 59

3.12. Field-scale demonstration of the developed stand count plugin showing original, binary, stand count, spatial distribution, and simple plot of plants. The field dimensions are about 146 ft wide by 760 ft with an area of 2.6 ac. 61

3.13. Results of row identification (A) and results highlight pop-up message box of the plugin (B) on a field-scale operartion. The field dimensions are about 146 ft wide by 760 ft and area of 2.6 ac. 62

3.14. Measured average properties of plants (truncated results) in each row and their grand total and mean values. The area of corn field used for analysis was 2.6 ac. . . 64

4.1. Field soil collection for growing weeds naturally in plastic trays and image preprocessing. (A) Manual clipping of individual plant images from the weeds grown in trays, and (B) Image preprocessing of clipped images using excess green (ExG) preprocessing and segmented denoised binary image. 72

4.2. Overall flowchart of distinguishing four weed species using shape features with three different machine vision methods. 74

4.3. Classification process involved in the machine learning models depicting four different outcomes using a target and other weed species: True positive (TP), false negative (FN), true negative (TN), and false positive (FP). 82

4.4. Variations of five best shape features and their density distribution visualization that can be used to classify the four common weed species using direct image processing. (A) Rectangularity, (B) Ratio of length to perimeter, (C) Ratio of perimeter to equivalent perimeter ratio, (D) Hollowness, and (E) Feret major axis ratio. 84

4.5. Comparison of machine learning models based on F1-score. kNN - k-nearest neighbor, RF - random forest, and SVM - support vector machine. 88

LIST OF ABBREVIATIONS

AC . . . . . Area Compactness

AGL . . . . . Above Ground Level

API . . . . . Application Programming Interface

AR . . . . . Aspect Ratio

CAF . . . . . Convex Area Feret ratio

CDT . . . . . Central Daylight Time

CIVE . . . . . Color Index of Vegetation Extraction

CNN . . . . . Convolutional Neural Network

CV . . . . . Cross Validation

CVI . . . . . Color Vegetation Index

DL . . . . . Deep Learning

DPI . . . . . Dots Per Inch

EqP . . . . . Equivalent Perimeter ratio

ExG . . . . . Excess Green Index

ExGR . . . . . Excess Green Minus Excess Red Index

ExR . . . . . Excess Red Index

FMA . . . . . Feret Major Axis Ratio

FN . . . . . False Negative

FP . . . . . False Positive

GC . . . . . Grum Circularity

GCC . . . . . Green Chromatic Coordinate

HSV . . . . . Hue, Saturation, and Value

JPEG . . . . . Joint Photographic Experts Group

kNN . . . . . K-Nearest Neighborhood

LHW . . . . . Log of ratio of Height to Width

LTP . . . . . Ratio of Length to Perimeter

MC . . . . . Modified Circularity

MExG . . . . . Modified Excess Green Index

ML . . . . . Machine Learning

MLR . . . . . Multiple Linear Regression

MP . . . . . Mega Pixel

ND . . . . . North Dakota

NDSU . . . . . North Dakota State University

NDVI . . . . . Normalized Difference Vegetation Index

NGRDI . . . . . Normalized Green Red Difference Index

NIR . . . . . Near-Infrared

PA . . . . . Precision Agriculture

PRE . . . . . Precision

RAL . . . . . Ratio of Area to Length

RAR . . . . . Reverse Aspect Ratio

REC . . . . . Recall

RF . . . . . Random Forest

RGB . . . . . Red, Green, and Blue

RMSE . . . . . Root Mean Square Error

ROI . . . . . Region of Interest

RPB . . . . . Ratio of Perimeter to Broadness

SSROI . . . . . Sliding and Shifting Region of Interest

SVM . . . . . Support Vector Machine

TN . . . . . True Negative

TP . . . . . True Positive

UAV . . . . . Unmanned Aerial Vehicle

UAS . . . . . Unmanned Aircraft Systems

USDA . . . . . United States Department of Agriculture

VARI . . . . . Visible Atmospherically Resistant Index

VDVI . . . . . Visible Difference Vegetation Index

VEG . . . . . Vegetative Index

LIST OF SYMBOLS

B . . . . . Blue channel intensity value

C . . . . . Regularization parameter

G . . . . . Green channel intensity value

r . . . . . Pearson correlation coefficient

R . . . . . Red channel intensity value

R^* . . . . . Normalized red from RGB color space

R^2 . . . . . Coefficient of determination

T_h . . . . . Threshold for segmentation

T . . . . . Time

X . . . . . Input or independent variable

x . . . . . x-coordinate of image pixel

Y . . . . . Output or dependent variable

y . . . . . y-coordinate of image pixel

σ . . . . . Variance in the 2D Gaussian kernel

γ . . . . . Hyperparameter in SVM

LIST OF APPENDIX FIGURES

Figure Page

G1. Seedlings and matured stage of common lambsquarters weed. 133

G2. Common purslane weed imagery. 133

G3. Horsweed imagery at two different stages. 134

G4. Redroot pigweed imagery at two different stages. 135

G5. Waterhemp imagery at two different stages (*Source*: `https://extension.psu.edu/invasive-pigweeds-palmer-amaranth-and-waterhemp`). . 135

G6. Common ragweed imagery at two different stages (*Source*: `https://blogs.cornell.edu/weedid/field-crops/common-ragweed-ambrosia-artemisiifolia/`). 136

G7. Kochia imagery at two different stages (*Source*: `https://weedid.missouri.edu/weedinfo.cfm?weed_id=148`). 137

H1. Graphical representation of weed shape features (1st set): (A) Ratio of area to length, (B) Log of ratio of length to width, (C) Convex area feret ratio, (D) Feret minor axis ratio, (E) Convex area, (F) Elongation, (G) Ratio of perimeter to broadness, and (H) Modified circularity. 139

H2. Graphical representation of weed shape features (2nd set): (A) Compactness, (B) Circularity, (C) Area compactness, (D) Grum circularity, (E) Reverse aspect ratio, (F) Roundness, (G) Aspect ratio, and (H) Solidity. 140

1. GENERAL INTRODUCTION

Agriculture can be defined as the art and science of cultivating soil, growing crops, and raising livestock that has been practiced by human beings for a long time to obtain food for their survival (Harris and Fuller, 2014). It has evolved from Agriculture 1.0 to Agriculture 4.0 and is in a continuous process to evolve in the future. Agriculture 4.0 uses current emerging technologies such as remote sensing, computer vision, and many more. By employing these technologies in agriculture, a wide range of agricultural parameters can be precisely monitored to improve crop yield; reduce cost and optimize farm inputs based on varying environmental conditions; better understand crop growth status; and optimize weed, pest, and nutrient management (Nukala et al., 2016).

Remote sensing for agriculture can be defined as the process of detecting and monitoring the physical characteristics of an area (field or crop) by measuring the reflected and emitted radiation at a distance (typically from satellites, unmanned aerial vehicle (UAV), or ground-based methods) (USGS, 2021). Remotely sensed images can be used to map soil properties, classify crop species, and scout fields for evaluating plant stand count or for weed identification. Remote sensing in precision agriculture (PA) can be classified based on (i) type of platform (such as satellites, UAV, or ground-based) used for data collection and (ii) types of sensors that are further categorized by the number and width of spectral bands (such as multispectral and hyperspectral camera) and the spatial, temporal, and radiometric resolutions that impact the data collection process (Sami et al., 2017; Sishodia et al., 2020). It is to be noted that the remotely-sensed digital image should have good geometric precision,

spatial and spectral resolution, and quality spectral information for the decision-making process in agriculture (Sami et al., 2017). Simply put, the digital image should have good clarity, resolution, and focus making it suitable for image analysis; hence, most of the research is being conducted in controlled environments (Shajahan, 2019). It was reported that only about 35 % of the total publications employ image processing in field conditions Shajahan (2019).

In comparison with remotely sensed methods, ground-based methods are not as efficient because they are limited to a small space, are time-consuming, and in some cases disturb the samples. Remote sensing using satellites for plant stand count evaluation or weed classification is problematic because of insufficient spatial resolution (Peña et al., 2013). This poor spatial resolution problem can be overcome by using the latest remote sensing platform known as UAV. These platforms have the ability to capture very high spatial resolution images. Neupane et al. (2019), employed deep learning methods for counting banana plants in the field with varying flying heights and found that 40 m provided the best spatial resolution to extract the features.

1.1. Plant Stand Count and Weed Classification

Assessing the plant stand count, which is the number of plants per unit area in the field, based on emergence and weed identification by spatial resolution will help the farmer make important farm management decisions such as replanting if the expected number of plant population is not achieved or herbicide application if weed pressure is high. Plant stand count helps evaluate the planter's efficiency in planting seeds and estimates seed quality by monitoring its germination and emergence. In some crops such as cotton and corn, yield potential in earlier stages is dependent on the plant population, while in the later stage, it is

dependent on weed pressure, climate conditions, and many more. Therefore, detecting the weeds at an early stage will help the farmers carry out immediate actions (e.g., applying herbicides) to reduce yield loss.

Scouting the field to evaluate plant stand count or identify weeds is laborious, time-consuming, and prone to errors, but with the help of remotely sensed images and computer vision algorithms, field scouting can be done easily. Various ground-based and remotely sensed methods are available in the literature to evaluate plant stand count or to identify weeds. But most of these methods are conducted on proprietary software that requires a high subscription cost and hence limits the use among farmers. Apart from this, there are commercial software applications that can evaluate plant stand count and identify weeds, but they are expensive and most of the processing is performed on a cloud platform, which increases the risk of data security, has variable processing time (ranging from hours to days based on the subscription package) and requires a reliable internet connection for efficient use (Pathak et al., 2020). Furthermore, these software applications provide efficient results on straight rows and fail to provide efficient results in off-linear rows, which are quite common in North Dakota (ND).

A cost-effective way to overcome these issues is to use open-source software (where open means freely available and source refers to the main computer program). The use of the open-source software promotes data security and privacy as it provides in-house processing, which means that a user owns the data and doesn't have to upload the data to the cloud platform.

Therefore, with this background, this research work focuses on developing algorithms that can be used to evaluate plant stand count and classify common weed species of ND,

namely, common lambsquarters (*Chenopodium album*), common purslane (*Portulaca oleracea*), horseweed (*Erigeron canadensis*), and redroot pigweed (*Amaranthus retroflexus*) using open-source platforms, namely ImageJ, R, and Python. ImageJ is an open-source, java-based, freely-available image analysis and programming platform that allows users to create custom-coded plugins to solve the image processing problem. While, R is a system for statistical computation and graphics and Python is an interpreted high-level general-purpose programming language that are used respectively for data visualization and machine learning (ML) models development aspects in this research.

1.2. Statement of Hypothesis

1. Intensity profile of the derived binary image of the plants in the field can be used to identify the corn crop rows.
2. Features of individual plants such as centroid, average area, and other shape features can be used to count the plants and generate spatial distribution maps.
3. Weeds can be differentiated based on their shape features.

1.3. Statement of Objectives

The main objectives of this research work were to:

1. Review the methods used for evaluating crop stand count using UAV imagery,
2. Develop an ImageJ image processing plugin that can be used to identify both straight and off-linear crop rows, evaluate plant stand count, and produce a spatial distribution map by taking doubles and skips into consideration, and
3. Develop an ImageJ user-coded direct image processing plugin, and a Python program to employ ML algorithms for classifying the four common weed species of ND.

1.4. Thesis Organization

This thesis is organized into five chapters that consist of (1) a general introduction; (2, 3, and 4) chapters in the form of three peer-reviewed articles that addressed the stated study objectives; and (5) a general conclusion followed by references and seven appendixes. The traditionally presented "Review of Literature" chapter was replaced by a review paper and is entitled "A review of unmanned aerial vehicle based methods for evaluating plant stand count" in Chapter 2. It is to be noted that each chapter (2, 3, and 4) is completely developed and has its own introduction containing a review of literature, materials and methods, results and discussions, and conclusion. While an entire list of references of all the citations from all chapters will be located after the end of the general conclusion (Chapter 5).

Chapter 1, the general introduction, briefly describes an evolution of agricultural practices, the importance of evaluating plant stand count and weed identification, and the methods and drawbacks related to previously studied methods. This chapter also describes what value this research will add and how it will benefit stakeholders such as researchers, students, and farmers. Further, it includes the statement of hypothesis, statement of objectives, and thesis organization.

Chapter 2 provides the review of literature entitled: "A review of unmanned aerial vehicle based methods for evaluating plant stand count." The review covers the various UAV-based methods used to evaluate crop stand count and identifies and summarizes the research gaps. It also discusses the impact of selecting vegetative indices, flying height, data collection time, pixel-resolution, weed pressure, shape, and spectral features on the algorithm's accuracy and provides recommendations to stakeholders.

Chapter 3 presents a method for identifying straight and off-linear crop rows, evaluating the stand count, and producing spatial distribution maps and is entitled: "Open-source ImageJ based early-season corn stand count evaluation and spatial distribution using UAV imagery." The plugin was developed using ImageJ and employed the profile plot, and shifting and sliding region of interest (SSROI) method to count the corn plants and determined the spatial distribution using UAV imagery.

Chapter 4 discusses the different methods by which weeds can be classified starting from simple image processing to the complex machine learning models and is entitled: "Machine vision methods for weed classification using open-source platforms." The algorithms were developed on two open-source platforms (ImageJ and Python) and can be used to classify four weed species.

Chapter 5 entitled "General Conclusion" summarizes the results derived from Chapters 2 through 4 and provides suggestions for future work.

Following Chapter 5, there will be a combined list of all the references under a single "References" unnumbered chapter. While, seven appendixes are enclosed at the end containing the sample codes for data analysis and visualization, image processing for stand count evaluation and weed identification/classification, and information about commonly available weeds in ND.

2. A REVIEW OF UNMANNED AERIAL VEHICLE BASED METHODS FOR EVALUATING PLANT STAND COUNT *

2.1. Abstract

Plant stand count helps in estimating the yield and evaluating the planter's efficiency and seed quality. Traditional methods of counting the plants by manual measurement are time-consuming, laborious, and error-prone. In contrast, the ground-based sensing methods are limited to smaller spaces. High spatial resolution images obtained from unmanned aerial vehicles (UAV) can be used in conjunction with computer vision algorithms to evaluate plant stand count, as it directly influences the yield. In spite of the importance of high-throughput plant stand count in row crop agriculture, no synthesized information in this specific subject matter is available. Therefore, the objective of this paper was to review the current studies that focus on evaluating plant stand count using UAV imagery to provide well-synthesized information, identify research gaps, and provide some recommendations. In this study, a comprehensive literature search was performed on three academic databases (Agricola, Web of Science, Scopus), and a total of 29 articles were found based on search terms and selection criteria for review. From the systematic review, it can be concluded that: appropriate stage after plant emergence without canopy overlap is necessary for image acquisition; optimal flying height should be selected to balance the field coverage and accuracy; L*a*b* color

* This paper is planned to be submitted as a review article in the *Computers and Electronics in Agriculture* journal. Authors: Harsh Pathak, C. Igathinathane, Z. Zhang, D. Archer, and J. Hendrickson. Harsh Pathak performed the literature survey and wrote the manuscript. Dr. Igathinathane Cannayen is the major advisor and the corresponding author who worked with Harsh Pathak throughout the manuscript development. All the co-authors have assisted in the research direction and review of the manuscript.

space can provide better segmentation; hyperspectral camera imagery can provide good discrimination; deep learning with data augmentation and transfer learning models can be used to reduce the computational time and resources; the stand count methodology that is successful with corn and cotton could be extended to other row crops and horticultural crops; and application of direct image processing and use of open-source platforms is required for stakeholder participation. The review will be helpful to the farmers, producers, and researchers in selecting and employing the UAV-based algorithms for evaluating plant stand count.

2.2. Introduction

Agriculture can be defined as the art and science of cultivating soil, growing crops, and raising livestock that human beings have practiced for a long time to obtain food for their survival (Harris and Fuller, 2014). It has evolved from Agriculture 1.0 (traditional) to Agriculture 4.0 (new age) and is in a continuous process to evolve in the future. Agriculture 4.0 uses current emerging technologies such as remote sensing, artificial intelligence, the internet of things, cloud computing, computer vision, and several more (Zhai et al., 2020). By employing these technologies in agriculture, a wide range of agricultural parameters can be precisely monitored to improve crop yield; reduce cost and optimize farm inputs based on varying environmental conditions; better understand crop growth status; and optimize weed, pest, and nutrient management (Nukala et al., 2016). The yield of the crop directly depends on the number of plants present per unit area of the field (Abuzar et al., 2011; Johnson and Harris, 1967; Miller et al., 1991; Robinson et al., 1980). Hence, an evaluation of plant stand count (plant population) early in the season is necessary to make farm management decisions such as possible reseeding if the expected number of plant stand count is not achieved. Even

though plant stand count has a significant impact on the crop yield, there is still no synthesized information available about the different evaluation methods of plant stand count. Therefore, this paper aims to provide well-synthesized information about the methods of plant stand count, identifies research gaps, and provides basic recommendations to stakeholders.

Optimizing plant population or plant stand count to achieve a good yield is important to farmers and researchers alike. Plant stand count also helps to check the planter's efficiency in planting the seeds and seed quality, such as emergence and germination rates. By counting the plants during emergence and harvest, the survival rate during the seedling phase can be evaluated (Ampatzidis and Partel, 2019). Plant population and uniformity are influenced by many factors, such as nutrient and water availability (including soil texture, moisture content, and temperature) seed depth, herbicide injury, weather, pest infestations, and sunlight (Al-Kaisi and Hanna, 2006; Carter and Nafziger, 1990; Nielsen, 1993). These factors ultimately affect the overall yield and quality of production. Corn yield also declines because of many other reasons, such as aborted kernels, barren stalks, or nonuniform population (Hashemi et al., 2005; Wright, 2018).

Counting plants is not only limited to the field but also performed at nurseries (Leiva et al., 2016). Manual counting performed at both the field and nursery level is labor-intensive, time-consuming, and has human errors and subjectivity (Bullock et al., 1998). Traditional counting methods, other than just visual inspection, include the use of hoops and graduated sticks for counting plants in the field (Conley and Robinson, 2007). In these methods, data were collected at different plot sites, and the average of the different sites was multiplied by a pre-defined multiplier for that specific crop to estimate the number

of plants in the field (Meaghan, 2020). With these methods, farmers were not able to obtain accurate plant stand count as well as good measures of stand variability across the field. Therefore, there is a need for automated sensing technology to identify and count the plants in a field to address these challenges.

Ground-based sensing methods use computers for data analysis while a digital camera is mounted on a vehicle for data collection, or a person walks around the field for data collection. Several ground-based sensing methods were developed that used machine vision to identify and count plants in the field (Jia and Krutz, 1992; Shrestha and Steward, 2005; Tang and Tian, 2008a; Zhang et al., 2020d). Ground-based methods have certain limitations such as: (i) they are time-consuming and tedious and are not efficient for large fields (Rudd et al., 2017; Yang et al., 2003), (ii) they have mosaicking errors due to the motion of the video sampling platform when it travels on large soil clods caused by the tractor tire treads (Tang and Tian, 2008a), or (iii) they have estimation or identification errors when the plants are damaged due to trampling by the tractor tires (Tang and Tian, 2008a). To avoid trampling damage to the plants, a 3D printed robot was used, but it underestimated the plant population due to tracking failure (Zhang et al., 2020d). Therefore, there is a need for a method that can take less time, less labor, and cover a large area of land and remote sensing is one such approach.

Remote sensing can be defined as the process of detecting and monitoring the physical characteristics of an area in agriculture, the reflected and emitted radiation of a field or crop is measured at a distance without touching it by using various remote sensing platforms (NASA, 2021). In comparison to the ground-based sensing platforms, some of the remote sensing platforms are lightweight, small, and can cover large areas without causing any

damage to the crops. Some of the remote sensing platforms used in agriculture (fig. 2.1) are parachutes, blimps, helicopters, satellites, and unmanned aerial vehicles (UAVs)(Ampatzidis and Partel, 2019; Bryant et al., 2003; Gnädinger and Schmidhalter, 2017; Peña et al., 2013; Sankaran et al., 2017; Sugiura et al., 2003). Parachutes and blimps have payload limitations and cannot be flown on windy days (Sankaran et al., 2015), while using helicopters for field scouting will be costly.

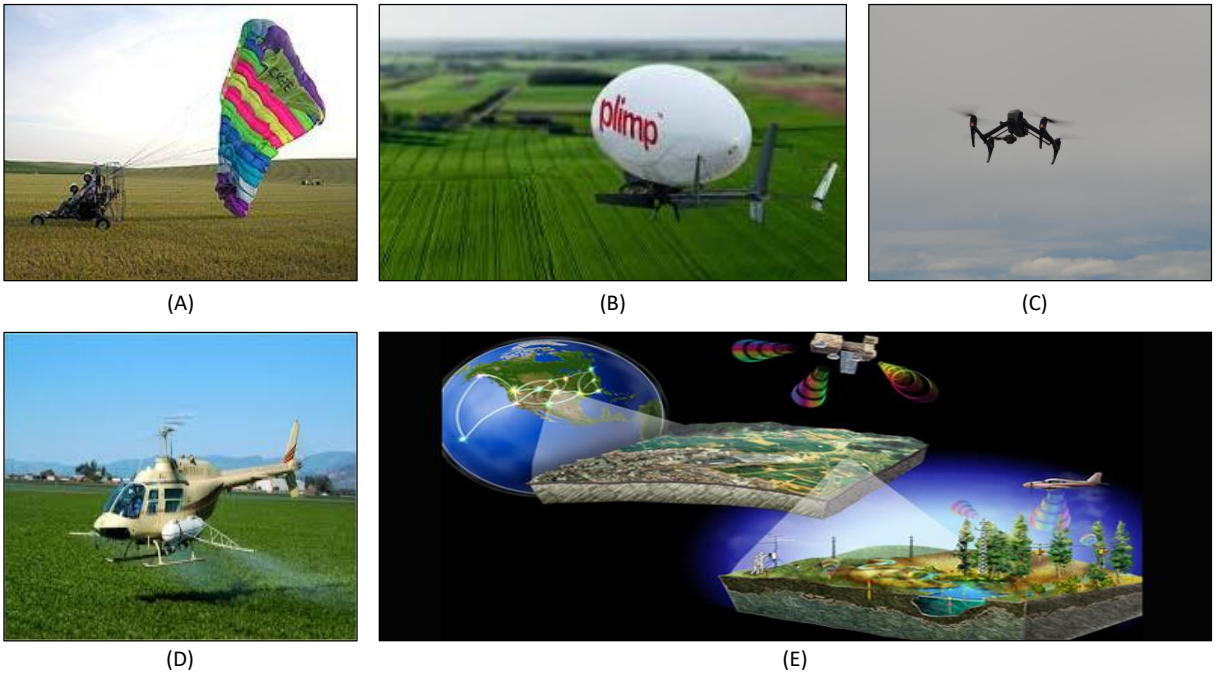


Figure 2.1. Various remote sensing platforms in agriculture: (A) Powered parachute, (B) Blimp, (C) Unmanned aerial vehicle, (D) Helicopter and (E) Satellite. (*Image Source: Google Images*).

Therefore, for functions like field scouting for stand count evaluation or weed identification, UAVs are better suited than satellites because of their (i) on-demand visit, (ii) ability to provide high spatial and temporal resolution (Peña et al., 2013; Sankaran et al., 2015), (ii) manageable data acquisition process, (iv) ability to collect data during cloudy conditions (Myers et al., 2015), and (v) low cost as compared to satellite or manned aircraft

(Alsalam et al., 2017). Therefore, UAVs have been used in research for stand count evaluation (Gnädinger and Schmidhalter, 2017; Sankaran et al., 2017; Shajahan, 2019; Suresh Babu, 2018; Varela et al., 2018).

Several image processing algorithms were developed to calculate the plant stand count using UAV imagery, but to our best knowledge, no review focusing on plant stand count evaluation using UAV was found. Therefore, a comprehensive review focused on UAV imagery-based plant stand count evaluation is necessary to capture the state-of-the-art and current research. This review study aims to synthesize how the methods used for evaluating the plant stand count have evolved with time, thereby allowing to identify the research gaps and provide recommendations. Thus, the objective of this review is to perform a literature search on three academic databases, namely, Agricola, Web of Science, and Scopus, to find the number of articles published in the past 20 years that utilize remotely sensed images to evaluate the plant stand count. The review will be helpful to the farmers, producers, and researchers to select and employ the UAV algorithms for stand count evaluation.

2.3. Materials and Methods

A literature search was performed on three of the most common academic databases, Agricola (`https://www.proquest.com/agricola/`), Web of Science (`https://www.webofknowledge.com`), and Scopus (`https://www.scopus.com`) to find the number of published peer-reviewed articles in the subject area of plant stand count in combination with the use of remote sensing techniques, while the primary goal of finding articles of UAV application in plant stand count. Access to the Agricola, Scopus, and Web of Science was provided through an institutional subscription, while restricted access will be available for free. The two conditions applied while searching articles were: (i) the entered

keywords should be present in either the article title or abstract or in the keywords, and (ii) the articles published between the years 2000-2020. The twenty-year range was selected because this represents the major period of research outputs in this subject area. Whereas, a literature search before this period (< 2000) using the keywords of "UAV" and "agriculture" resulted in only one peer-reviewed article from Web of Science and Agricola, while no articles were found from Scopus.

Articles search and selection for review were evaluated in three stages: identification, screening, and eligibility. In the identification stage, the keywords depicted in the figure (fig. 2.2) were used to obtain the papers of interest. In the screening stage, the results from different databases were gathered, and then duplicate articles were removed. Finally, in the eligibility stage, the abstract of the articles was read to determine whether they met the eligibility criteria of the subject relevance. For example, articles that contain information about the plant stand count or plant population related to remote sensing were successful in the eligibility stage as articles related to other areas (e.g., coal mine-related studies or evapotranspirational models studies that passed through the two stages) were rejected. The number of articles passed through the final eligibility stage was selected for the review (fig. 2.2).

In the literature search, different boolean and proximity or positional operators were used to find the number of articles published to evaluate plant stand count (fig. 2.2). For example, in Scopus the search phrase was: ["machine learning" OR "deep learning" OR "image process*"] AND ["crop" OR "plant" with "W/5" (population OR count OR counting OR density OR emergence)] AND ["unmanned aerial" OR "UAV" OR "UAS" OR "drone" OR "remote sensing"], where W/5 means that the set of five words, namely, population, count,

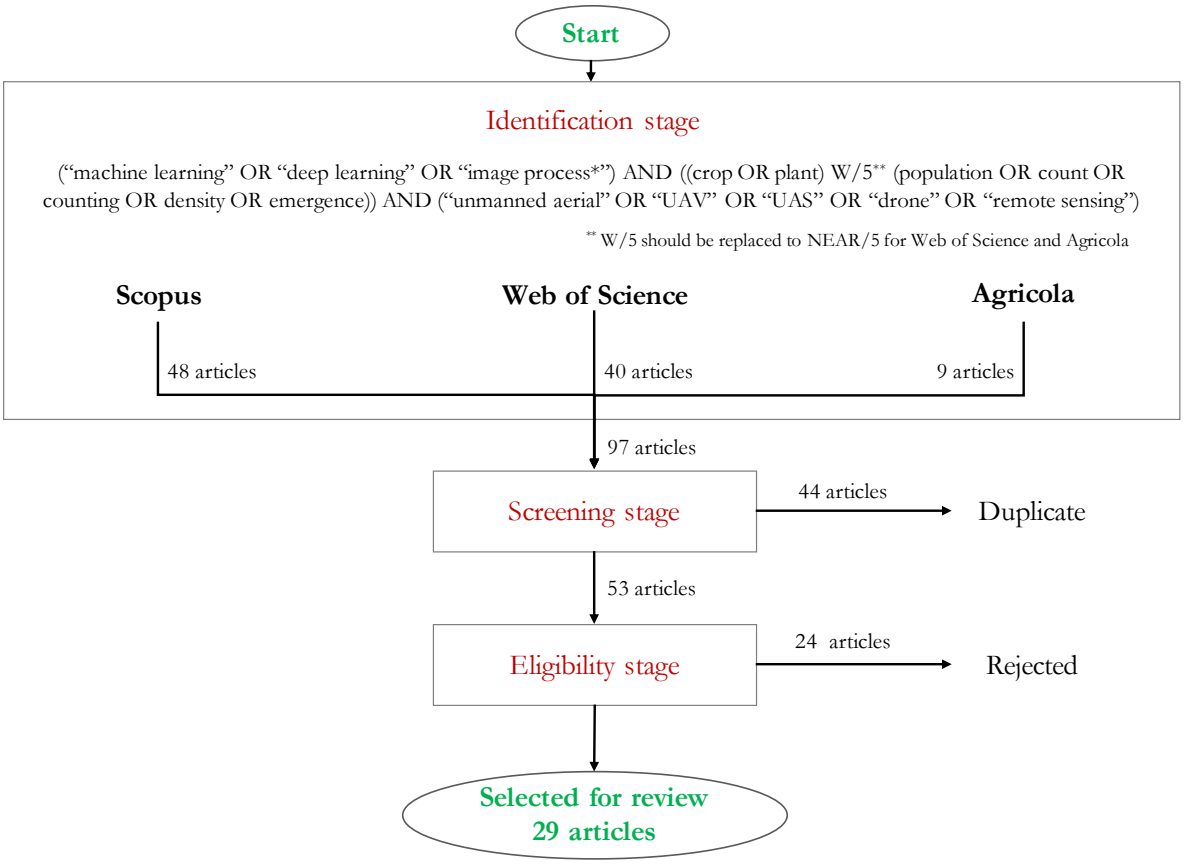


Figure 2.2. Processing stages of selecting the articles for the literature search on methods of plant stand count using UAV from appropriate keyword combinations and selected databases.

counting, density or emergence should be within five words from the preceding words (crop or plant) used in the search. It is to be noted that "within" (W) is proximity or positional operator and 'OR', 'AND' are the boolean operators.

2.4. Results and Discussion

2.4.1. Outcomes of Literature Search using Different Academic Databases

In the identification stage, a total of 97 peer-reviewed journal articles were obtained out of which 48 were from Scopus, 40 were from Web of Science, and 9 were from Agricola. While in the screening stage, 44 duplicate articles were removed, leaving 53 unique articles (fig. 2.2). In the eligibility stage, after reading the abstracts and verifying the contents of the

53 articles, it was found that only 29 articles were of our interest as they focused on the methods to evaluate plant stand count, population or density. It should be noted that if the search was about the general “precision agriculture,” where the present subject matter is a part, would have returned numerous articles.

The trend of the number of articles published during the 20-year time period (2000–2020) shows much of the research and publication activities in the present subject took off from 2015, while the number of articles was between 0 to 3 in the 15-year range of 2000–2015 (fig. 2.3).

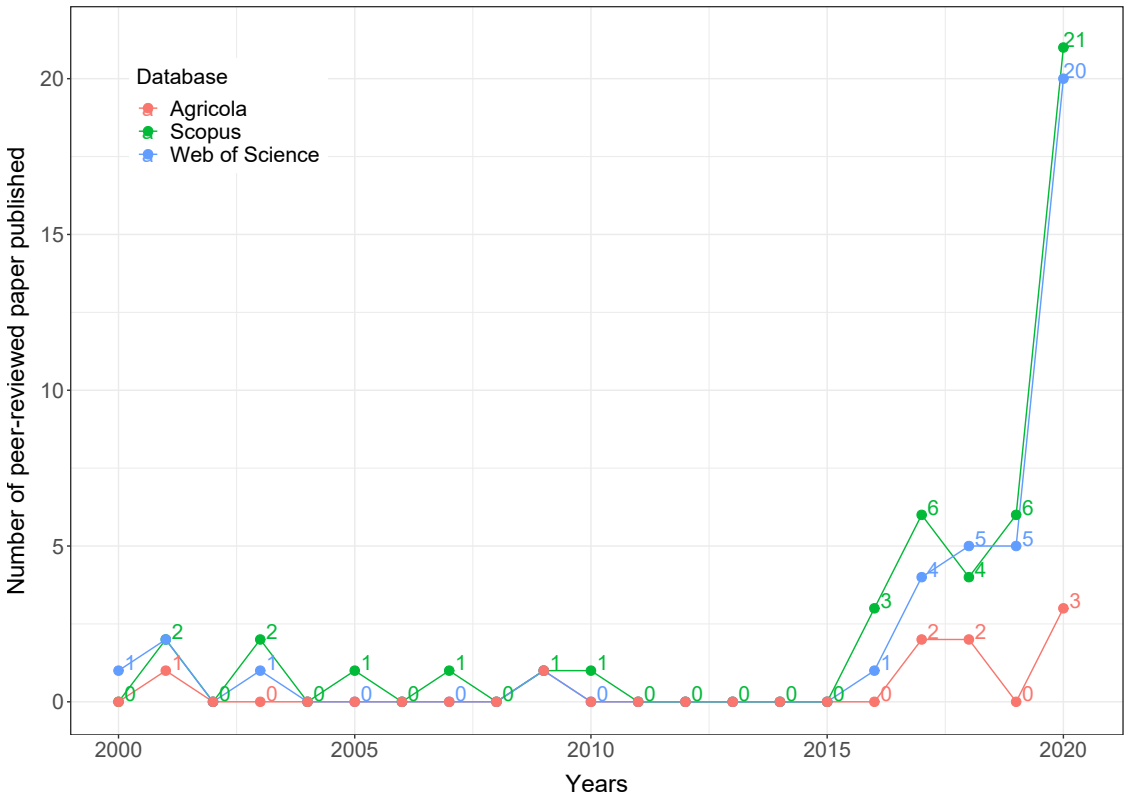


Figure 2.3. Number of articles published from the year 2000 to the year 2020 based on the three academic databases (Agricola, Scopus, and Web of Science), with the search performed on January 26, 2021.

Overall, it can be clearly seen from the trend that there are very few research articles available in the literature that focus on methods evaluating plant stand count. These 29 articles were further processed to develop well-synthesized information, identify research gaps, and formulate recommendations. The list of the 29 articles and relevant brief details are presented in Table 2.1. It can be observed that the articles published earlier (2016–2018) received more citations (average: 61.5/article) than the latest (2019–2020) publications (average: 11.7/article). However, with the demonstrated interest in this subject (fig. 2.3) it is expected that more citations will be received in the future.

2.4.2. Review of Various Stand Count Methods — Computer Vision

Computer vision is a field of artificial intelligence (AI) that enables computers and systems the ability to derive information from digital images and videos. This interdisciplinary field automates the elements of human vision using sensors, computers, and different image analysis algorithms. Computer vision has two stages, namely, image acquisition (data collection) and image analysis (Patrício and Rieder, 2018). Image analysis can be performed in three ways: (i) traditional image processing, (ii) machine learning (ML), and (iii) deep learning (DL). Most part of this review focuses on image analysis, while some information on image acquisition is also provided.

2.4.2.1. Traditional image processing

Traditional image processing can be used to process UAV imagery to evaluate the number of plants in the field. Traditional image processing consists of various tasks that can be performed to manipulate images using digital computers (da Silva and Mendonça, 2005; Patrício and Rieder, 2018). These tasks include image enhancement, image restoration, and image analysis (da Silva and Mendonça, 2005). Traditional image processing can help in

Table 2.1. Details of selected articles through literature search and their citation scores.

S.No.	Paper title	Journal	Number of citations	Reference
1	TasselNetV2+: A fast implementation for high-throughput plant counting from high-resolution RGB imagery	Frontiers in Plant Science	3	Lu and Cao (2020)
2	Automated crop plant counting from very high-resolution aerial imagery	Precision Agriculture	8	Valente et al. (2020)
3	Improved crop row detection with deep neural network for early-season maize stand count in UAV imagery	Computers and Electronics in Agriculture	1	Pang et al. (2020)
4	Identifying and mapping individual plants in a highly diverse high-elevation ecosystem using UAV imagery and deep learning	ISPRS Journal of Photogrammetry and Remote Sensing	7	Zhang et al. (2020a)
5	Evaluation of cotton emergence using UAV-based imagery and deep learning	Computers and Electronics in Agriculture	6	Feng et al. (2020a)
6	Counting of *Pennisetum alopecuroides* at heading stage in a grazed pasture using images from an unmanned aerial vehicle	Grassland Science	2	Yuba et al. (2020)
7	Plant counting of cotton from UAS imagery using deep learning-based object detection framework	Remote Sensing	7	Oh et al. (2020)
8	A low-cost UAV framework towards ornamental plant detection and counting in the wild	ISPRS Journal of Photogrammetry and Remote Sensing	1	Bayraktar et al. (2020)
9	Prediction of soybean plant density using a machine learning model and vegetation indices extracted from RGB images taken with a UAV	Agronomy	9	Ranđelović et al. (2020)
10	A convolutional neural network approach for counting and geolocating citrus-trees in UAV multispectral imagery	ISPRS Journal of Photogrammetry and Remote Sensing	39	Osco et al. (2020)
11	Mask R-CNN refitting strategy for plant counting and sizing in UAV imagery	Remote Sensing	9	Machefer et al., (2020)
12	Mapping crop stand count and planting uniformity using high resolution imagery in maize crop	Biosystems Engineering	0	Shirzadifar et al., (2020)
13	Ananas comosus crown image thresholding and crop counting using a colour space transformation scheme	Telkomnika	1	Rahimi et al. (2020)
14	Using open-source software and digital imagery to efficiently and objectively quantify cover density of an invasive alien plant species	Journal of Environmental Management	1	Carlier et al. (2020)
15	Agroview: Cloud-based application to process, analyze and visualize UAV-collected data for precision agriculture applications utilizing artificial intelligence	Computers and Electronics in Agriculture	19	Ampatzidis et al. (2020)
16	Evaluation of cotton emergence using UAV-based narrow-band spectral imagery with customized image alignment and stitching algorithms	Remote Sensing	2	Feng et al. (2020b)
17	Automatic plant counting and location based on a few-shot learning technique	IEEE Journal	6	Karami et al. (2020)
18	Capturing maize stand heterogeneity across yield-stability zones using unmanned aerial vehicles (UAV)	Sensors	6	Shuai et al. (2019)
19	Deep learning based banana plant detection and counting using high-resolution red-green-blue (RGB) images collected from unmanned aerial vehicle (UAV)	PloS one	24	Neupane et al. (2019)
20	Ear density estimation from high resolution RGB imagery using deep learning technique	Agricultural and Forest Meteorology	82	Madec et al. (2019)
21	Rapeseed seedling stand counting and seeding performance evaluation at two early growth stages based on unmanned aerial vehicle imagery	Frontiers in Plant Science	27	Zhao et al. (2018)
22	Aerial imagery analysis - quantifying appearance and number of sorghum heads for applications in Breeding and Agronomy	Frontiers in Plant Science	53	Guo et al. (2018)
23	Monitoring cotton (*Gossypium hirsutum* L.) germination using ultrahigh-resolution UAS images	Precision Agriculture	51	Chen et al., (2018)
24	Reaching new heights: Can drones replace current methods to study plant population dynamics?	Plant Ecology	23	Tay et al. (2018)
25	Digital counts of maize plants by unmanned aerial vehicles (UAVs)	Remote Sensing	116	Gnädinger and Schmidhalter (2017)
26	High-resolution aerial imaging based estimation of crop emergence in potatoes	American Journal of Potato Research	14	Sankaran et al. (2017)
27	Estimates of plant density of wheat crops at emergence from very low altitude UAV imagery	Remote Sensing of Environment	220	Jin et al., (2017)
28	Potential of low altitude multispectral imaging for in-field apple tree nursery inventory mapping	IFAC-PapersOnLine	7	Quirós and Khot (2016)
29	Mapping crop planting quality in sugarcane from UAV imagery: A pilot study in Nicaragua	Remote Sensing	43	Luna and Lobo (2016)

Note: Following articles were searched on the freely available Google Scholar (`https://scholar.google.com/`) on Date- July 17, 2021, and their corresponding number of citations are reported.

extracting useful information from the image automatically. Image processing is widely used to solve various agriculture problems such as classification (Sunoj et al., 2018a), object detection (Shajahan et al., 2017), object counting (Gnädinger and Schmidhalter, 2017) and many more. They usually involve a direct approach (e.g., shape parameters, pixel-, dimensional-information) and require a relatively smaller dataset. In traditional image processing, the predictive handcrafted models were created to minimize the errors.

To count the number of cotton plants in a field, a method was developed on ArcMap (`https://desktop.arcgis.com/en/arcmap/`) that uses a supervised maximum likelihood classifier to classify the leaf and non-leaf objects and average plant size to count plants (Chen et al., 2018). The average estimation accuracy of 88.6% was reported to estimate the number of cotton plants in the field. They suggested that to further improve the accuracy of discriminating between plant and non-plant near-infrared or multi-spectral cameras should be used; since multi-spectral cameras provide more spectral information (Chen et al., 2018). A workflow was generated on ERDAS (for raster analysis; `https://www.hexagongeospatial.com/products/power-portfolio/erdas-imagine`) and ArcGIS (for vector analysis) to count the number of apple trees present in the nursery using multispectral imagery and overall accuracy of 95% was reported when the flying height was 10 m (Quirós and Khot, 2016). Furthermore, by using ArcGIS the number of potato plants of two different varieties was estimated with the Pearson's correlation coefficient ($r = 0.82$) between image-based and manual count (Sankaran et al., 2017).

Template matching, a digital image processing technique, is used to locate a region of interest (ROI, certain object) in an image. An object-based image analysis algorithm based on template matching technique was developed to count safflower plants in the field using

remotely sensed RGB imagery (Koh et al., 2019). A group of 10 templates was collected in the green channel and found that the estimated seedling counts have an R^2 of 0.87 for the year 2017 and R^2 of 0.86 for the year 2018 (Koh et al., 2019). García-Martínez et al. (2020) also used template matching and normalized cross-correlation to count the number of corn plants in the field. The precision ranging from 95 % to 98 % was found in counting when the corn plants were at the V5 growth stage with the selection of four, eight, and twelve samples. They found that the V5 stage provided the best results in comparison with the V9 and V2 stages (García-Martínez et al., 2020). The higher accuracy at the V5 stage compared to V2 and V9 growth could be due to fewer plants in the field at the V2 stage and a higher overlap between the plants at the V9 stage. Another method developed to count corn plants by adapting a decorrelation stretch contrast enhancement process to produce higher contrast gave a correlation R^2 up to 0.89 between in-situ and image-based counted plants (Gnädinger and Schmidhalter, 2017).

2.4.2.2. Machine learning methods

The ML is an effective way to process a large amount of data collected from remote sensing platforms in agriculture (Li et al., 2019; Varela et al., 2018). The ML models provide machines (computers) the ability to learn and adapt without being explicitly programmed to make precise predictions on unseen datasets (Liakos et al., 2018; Patrício and Rieder, 2018). The ML models differ from the traditional image processing models as the predictive handcrafted models are replaced by the training phase that helps to learn and predict precise results on the unseen dataset. Data cleaning is an essential step where noise and outliers are removed to avoid overfitting. These models require a large amount of data for training and testing to increase the robustness and prediction accuracy, since with a lesser amount of data

the model can be underfitted. To increase the size of the dataset, in place of additional original data, a special data augmentation technique is often used.

For processing agriculture imagery, ML process consists of following steps (fig. 2.4): (i) data acquisition, (ii) image segmentation, (iii) feature engineering, (iv) model selection, (v) hyperparameter tuning, (vi) model training, and (vii) performance evaluation or rating (Patrício and Rieder, 2018).

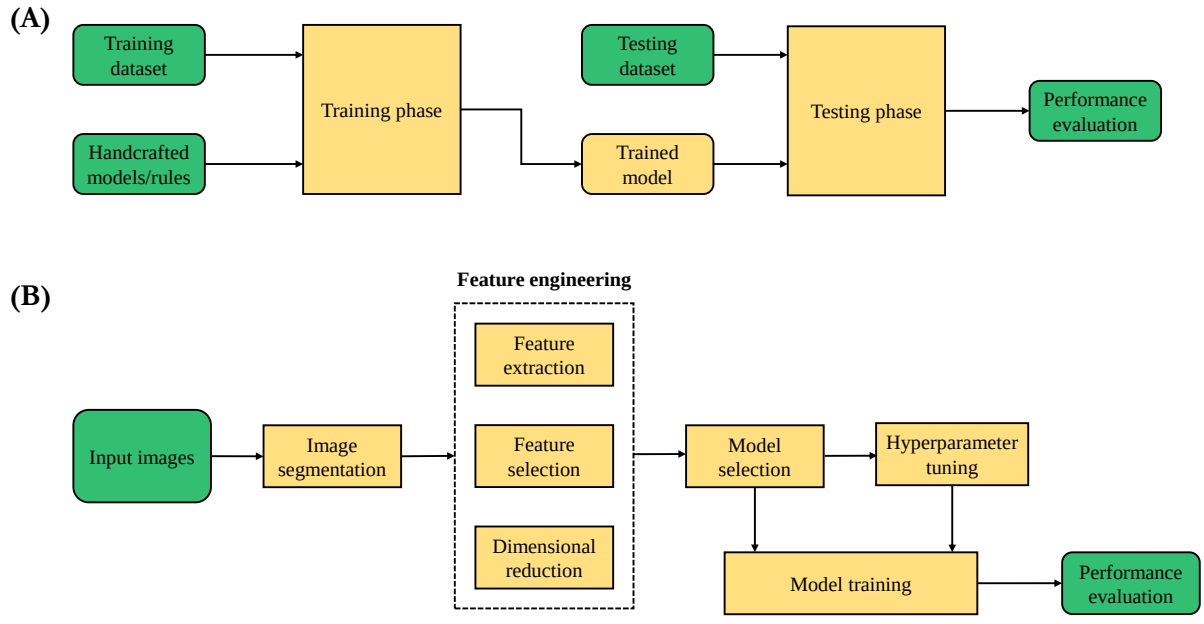


Figure 2.4. Process involved in machine learning (ML): (A) Flowchart of developing ML models and evaluation, and (B) Steps for processing agricultural images using ML models and evaluation.

Feature engineering can be defined as a process of selecting and transforming variables for creating ML models. It includes tasks such as feature extraction, feature selection, and dimensional reduction. The feature extraction process involves extracting features (e.g., geometric features, color moments) from the image that can be further used to perform different tasks like classification and prediction. Feature selection, among several

other insignificant features, helps to achieve reliable and accurate predictions as well as reduces the computational cost. Dimensional reduction helps visualize data easily and decreases computational cost due to projecting higher dimensional data to a lower-dimensional space. The model selection consists of selecting the suitable model that can be used to solve a particular problem (Patrício and Rieder, 2018). Hyperparameters influence the accuracy and computational cost of the model, hence important. For example, selecting the number of trees in a random forest (RF) classifier, or selecting the value of 'k' in k-nearest neighbors. Once the model is trained, it can be used to obtain a prediction for the new data (Patrício and Rieder, 2018). For rating the ML model performance, different accuracy metrics, such as precision, recall, and f1-score can be used.

A supervised ML algorithm, a support vector machine with radial function and tuned with particle swarm optimization algorithm, was used to count the number of wheat plants present in RGB field imagery (Jin et al., 2017). For counting rapeseed plants at two different stages, two different regression models were developed, and R^2 values of 0.845 and 0.867 were reported. However, a single regression model over these two stages resulted in an R^2 of 0.846 (Zhao et al., 2018).

Counting plants like rapeseed and wheat is difficult due to complex overlap and variable spacing between them (Zhao et al., 2018). Plants such as corn and cotton are nearly evenly spaced and have minimal overlap in the early stage; therefore, they can be counted easily. For counting corn plants in the field, a workflow was developed that uses geometric features extracted from the segmented image to train a decision tree classifier for retrieving information about stand count and their locations (Varela et al., 2018). The overall accuracy of the combined model was 0.93 for object classification, but no plant counting was

performed at the field level (Varela et al., 2018). The decision tree model is not robust with a small number of tree branches, while it overfits with many branches. Therefore, AdaBoost that combines many decision trees were used to count the number of cotton seedlings in the field (Feng et al., 2020b). The algorithm was able to count seedling clusters with an accuracy of 84.1 % (Feng et al., 2020b). Furthermore, a random forest (RF) classifier was used to predict soybean density using eight vegetation indices (Ranđelović et al., 2020). During the validation stage, the correlation coefficient (R^2) between the real and predicted number of plants was 0.87 (Ranđelović et al., 2020).

The k-means clustering segmentation algorithm provided better results than the excess green segmentation method in identifying corn plants in the field. The algorithm was able to identify 91 % of corn plants in the field using RGB imagery (Shirzadifar et al., 2020). Before applying the k-means, the image was transformed into L*a*b* color space (CIELAB) to reduce the color dimensions and to remove luminance variation occurring in the RGB image due to natural lighting condition (Shirzadifar et al., 2020; Zheng et al., 2018). Although the RGB camera is of low cost and is less sensitive to lighting conditions it can be potentially affected by sunlight (Zhao et al., 2018). Moreover, the crop color is also affected by the soil reflectance in RGB imagery (Feng et al., 2020b). In contrast, the multispectral camera provides good spectral information for crop identification but lags in providing good resolution; for example, the Micasense RedEdge-M (Micasense Inc, Seattle, WA, USA) has 1.2 M pixel resolution, whereas the RGB camera can provide high-resolution images, for example, the ZENMUSE X7 a product of (DJI, Shenzhen, China) has 24 M pixel resolution. While the high resolution narrow-band hyperspectral camera can solve the problems of color distortion of the RGB camera and low resolution of the multispectral camera. A pushbroom

hyperspectral camera was used to collect the images of cotton plants present in a field. Since the default image stitching software had many restrictions such as high processing georeferencing, low and consistent flight velocity, less wind, and many more, a novel image alignment and stitching algorithm was developed (Feng et al., 2020b).

2.4.2.3. Deep learning

The DL extends ML by adding more depth or complexity into its model (Kamilaris and Prenafeta-Boldú, 2018). Because of these complex models, DL can solve advanced problems of computer vision for the agriculture domain, such as plant detection (Kitano et al., 2019; Liu et al., 2020), plant identification (Ghazi et al., 2017; Grinblat et al., 2016), plant disease detection (Mohanty et al., 2016; Ramcharan et al., 2017), (TÜRKOĞLU and Hanbay, 2019), plant and fruit counting (Chen et al., 2017; Oh et al., 2020), and object detection (Neupane et al., 2019). The DL models require a large amount of data to learn the models to provide accurate results for problems involving classification, prediction, and object detection. Out of the four different architectures of DL, the convolutional neural network (CNN) serves as the backbone or tool for image classification and object detection. The general architecture of the CNN model is shown in Figure 2.5. Because of CNN's ability to learn data characteristics using convolution operation, compared to the other three architectures, namely unsupervised pre-trained networks, recurrent neural network, recursive neural network, CNN is suitable for image analysis operations (Patrício and Rieder, 2018). As per the recent review of literature performed by Kamilaris and Prenafeta-Boldú (2018), around 42 % of the researchers employed CNN for solving agricultural problems.

The CNN is designed by using combinations of different numbers of convolution, pooling, activation, and full-connection layers. Convolution and pooling layers include

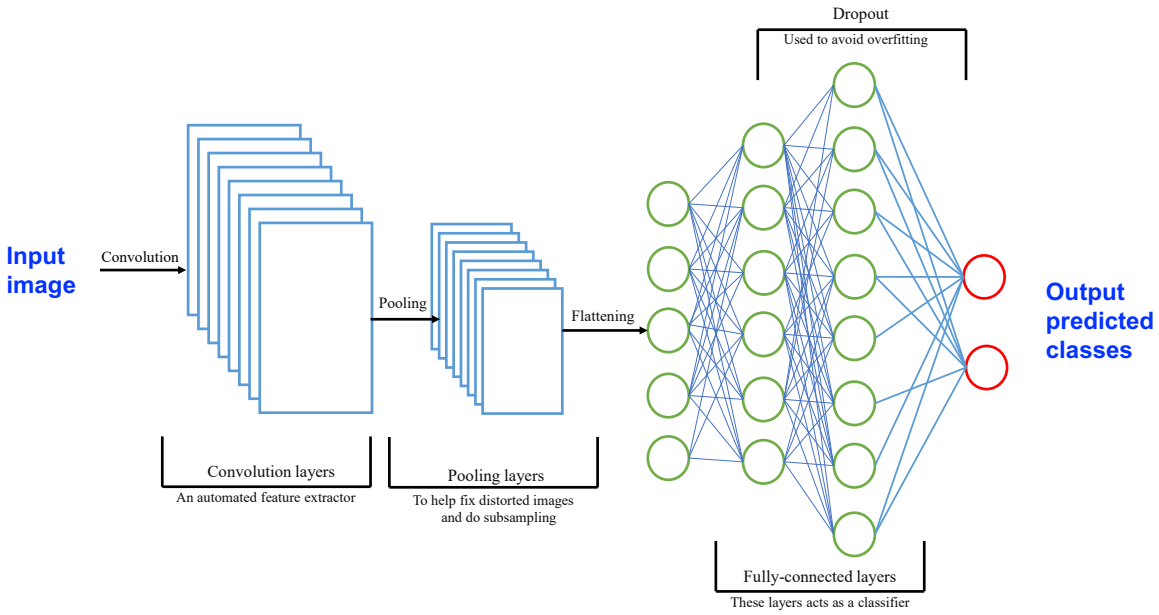


Figure 2.5. Generalized deep learning model development using convolutional neural network (CNN) architecture showing different CNN layers.

feature detector/filter boxes that scan the image while sliding from top to bottom and left to right and generate the feature maps. The convolution process helps to extract the image features while keeping the spatial correlation of image features. The pooling layer helps in down-sampling to reduce the complexity for further layers (Albawi et al., 2017). While, the activation function includes non-linear functions (Kuo, 2016), such as rectified linear unit (ReLU), sigmoid, and parameterized ReLU (PReLU). Fully connection layers are simply the feed-forward neural network (similar to fig. 2.5).

The DL models can be developed in two ways either by using the earlier defined network architecture (commonly known as transfer learning) or designing your own deep learning models (Kamilaris and Prenafeta-Boldú, 2018). It is to be noted that designing your own deep learning models from scratch requires a considerable amount of data for training and validation, high training time, and high computational cost, and is quite challenging for

agricultural applications. These challenges can be overcome by employing transfer learning. Several commonly used previously defined DL models are AlexNet (Krizhevsky et al., 2012), VGG (Simonyan and Zisserman, 2014), GoogleNet (Szegedy et al., 2015), ResNet (He et al., 2016), and many more. Some of these models often serve as the backbone to detect or count objects, and the application of DL models for plant stand count evaluation will be subsequently presented.

The overall emergence of plants compared to the number of seeds planted is indicated by stand counts. Two major problems that exist in counting objects (in our case plants) in agricultural sectors are: (i) arbitrary size of the objects and (ii) objects are often covered by branches or leaves of other objects (Zhang et al., 2020b). Other factors that affect the early-stage detection of plants are the characteristics of the field environment (e.g., illumination, soil properties, or weed pressure) and image acquisition parameters (e.g., altitude, velocity, camera resolution, or time of data collection) (Karami et al., 2020). Methods of object counting can be classified into four categories: segmentation-, detection-, regression-, and density function-based models (Zhang et al., 2020b).

2.4.2.3.1. Regression-based method

Regression-based methods are the simplest and most straightforward methods for counting the plants in the field (Jiang and Li, 2020). Generally, the softmax layer of the CNN model is replaced by a single neuron for regressing the numeric values. Ribera et al. (2017) tested the performance of different CNN models for sorghum plant counting and found that Inception-v3 has the smallest mean absolute percentage error (MAPE) of 6.7 %. A MAPE value of 4.3 % and R^2 of 0.95 was observed for counting cotton plants in a field using the ResNet18 model through transfer learning (Feng et al., 2020a). This low MAPE was due to

good segmented images provided by the decorrelation stretch segmentation method during the image pre-processing stage.

Lu et al. (2017) developed a two-step method for counting tassels, wherein the first step, a local low-capacity CNN model was used to predict the number of tassels in each sub-image, while in the second step, the estimated number of counts in each sub-image was averaged based on the number of pixels available in that sub-image. TasselNet achieved higher accuracy in comparison to the other conventional methods of counting objects. To further increase the accuracy, the receptive field of TasselNet was increased, and a global regressor was used for counting (Xiong et al., 2019). Since TasselNet and TasselNetV2 were implemented on proprietary software (MATLAB), they were infeasible for practical deployment. Therefore, TasselNetV2+ was brought into practice with minor modifications on PyTorch, an open-source platform (Lu and Cao, 2020). In TasselNetV2+, the efficient normalizer, trimmed counter, and improved encoder improved the processing speed. Lu and Cao (2020) suggested that these models could be employed for counting the plants in the field because these models were robust to the size variation of plants. Regression-based models have certain advantages such as (i) simpler manual annotation (generally requires dots rather than bounding boxes), (ii) simplified network, (iii) computationally less expensive, and (iv) can deal with the problems of occlusion and background clutter (Lu and Cao, 2020). TasselNetV2+, however, has some limitations, such as (i) poor generalization due to the limited number of annotated images and (ii) cannot provide any spatial information in the input image (Jiang and Li, 2020; Khaki et al., 2020).

2.4.2.3.2. Detection-based models

Detection-based methods can be classified into two types, namely two-stage and single-stage. Detection-based methods use object detection models to accurately detect the objects and then count them. In the two-stage detection method, the algorithm in the first step extracts ROIs using selective search methods, and then classification and bounding box regression are conducted in the second separate step (Jiang and Li, 2020). Jiang and Li (2020) stated that accurate detection of objects helps in accurate counting. Accurately detecting the objects (or geolocating) will help explore the problems that occurred during the seeding process, which can further be overcome or researched. Faster R-CNN (region-based CNN) with Inception ResNet v2 was used to count the number of cotton seedlings with an R^2 of 0.98 by using video frames collected using ground-based methods (Jiang et al., 2019). Further, the Faster R-CNN with Inception as a base model was used to detect and count banana plants using UAV imagery (Neupane et al., 2019). The algorithm provided the highest accuracy of 97.9 % for the dataset collected at the height of 40 m, followed by 91.5 % for 50 m and 87.2 % for 60 m. Flying at a higher altitude reduces the flight time but also decreases the accuracy.

In addition to the two-stage detectors such as Faster RCNN, one-stage detectors such You Only Look Once (Yolo) are being widely used for object detection and counting. The one-stage detectors are faster than the two-stage detectors as they are trained to perform classification and bounding box regression at the same time. To count the cotton seedlings in the field, Yolo version 3 (Yolov3) was used and achieved an R^2 of 0.96 to 0.97 when the plant density was $0-14$ plants per linear meter of row (Oh et al., 2020). Furthermore, Yolov3 served as the first step in a two-step process for evaluating phenotypic characteristics (such as

detecting and counting citrus trees) using multispectral aerial imagery (Ampatzidis and Partel, 2019). In the first step, the algorithm finds the initial number of trees, out of which some are incorrectly identified due to weed patches that resemble tree canopies, and a few were missing due to small canopies. More precise detection is performed in the second step, which reduces the false positives by 92 % and false negatives by 82 % (Ampatzidis and Partel, 2019). In another study, to further improve object detection accuracy, Faster R-CNN was used instead of Yolov3 (Ampatzidis et al., 2020). To reduce the computational time, an anchor-free method such as CenterNet based with an Hourglass-104 architecture was used to detect the location of the maize center and was used for counting (Karami et al., 2020). The results were further compared with the RetinaNet and Yolov3 and found that CenterNet outperformed the other two architecture results (Karami et al., 2020). Both detection-based methods provide good results when canopies have some distance between them, but their performance decreases when the density among objects is high. Furthermore, these models are computationally expensive and require a lot of time for data annotation and pre-splitting the images as needed by the model for training, and inference renders inefficient image analysis.

2.4.2.3.3. Density-based models

With density-based models, one can also train CNN to count the number of plants in the field, especially in the places where plant density is high. A density-based approach was used to count and geolocate citrus trees in a highly dense orchard using multispectral imagery (Osco et al., 2020). The initial part of the model uses VGG16 to obtain a feature map that was further used to generate a confidence map. The model resulted in high precision

(0.95), recall (0.96), and f1-score (0.95) when compared with two other object-detection methods (Faster R-CNN and RetinaNet; Osco et al. (2020)).

2.4.2.3.4. Segmentation-based models

In this approach, the CNN architecture first obtains the mask of plants and post-processes them using conventional computer vision methods to isolate them for further counting (Jiang and Li, 2020). A set of Residual U-Nets with a sequence of input scales was integrated and termed as Scale Sequence Residual U-Nets (SS Res U-Net) (Zhang et al., 2020a). The SS Res U-Net was used to identify and map two acaulirosula species (*Espeletia boyacensis Cuatrec* and *Espeletia congestiflora Cuatrec*). Due to multi-scale characteristics, SS Res U-Net outperforms the other DL methods, such as U-Net, Residual U-Net, Fully convolutional network, and Multiple residual U-Net, and achieved an accuracy of 91.67 % (Zhang et al., 2020a). Although segmentation-based methods provide accurate semantic masks, they still lag in counting the objects (Jiang and Li, 2020). An algorithm was developed to estimate corn stand count at the V6 growth stage by using a modified version of Mask RCNN, MaxArea Mask Scoring RCNN, for finding crop rows and geometric descriptors for estimating the stand count with an overall accuracy of 95.8 % (Pang et al., 2020).

2.4.3. Research Gaps and Recommendations

Based on this available literature following is a list of potential areas that need further investigation along with some recommendations. It is to be noted that these areas were not entirely covered by the existing research articles and are expected to be addressed in the future.

- Computer vision algorithms have been widely used for evaluating plant stand count, especially for corn and cotton crops, and gave promising results (Chen et al., 2018;

Feng et al., 2020b; Pang et al., 2020; Varela et al., 2018). Therefore, the research should be further extended to other row crops such as sunflower, soybean, and many more, as variation in plant population in any crop affects their yield (Johnson and Harris, 1967; Robinson et al., 1980), and this is important to the stakeholders.

- Getting the accurate number of plant stand count is challenging because of the occlusion and uneven emergence of plants (García-Martínez et al., 2020). Therefore, the best growth stage for collecting the data needs to be investigated.

- It was suggested that the use of the hyperspectral camera would provide better results in comparison to the RGB and multispectral camera, as it provides more spectral information and good image resolution and has been promising results in counting the cotton plants in the field (Feng et al., 2020b). However, more investigation is required to know the difference in their accuracy.

- The accuracy of the algorithm in detecting the banana trees decreased with the increase in the flying height (Neupane et al., 2019). So, the optimal flying height of the UAV that balances time to cover the field and accuracy of the algorithm in detecting the plants needs to be investigated for different crops such as corn, cotton, and other row crops and horticultural crops.

- In a survey, it was found that 40 % of the farmers discontinued using the software because of the subscription cost (DeLay, 2020). Therefore, open-source software should be used for developing algorithms to evaluate plant stand count.

- Counting objects in agriculture is problematic because of the varying size of the objects (Zhang et al., 2020b). The DL models such as TasselNetV2+ could be tested to count

the plants present in the field due to their robustness to size variation (Lu and Cao, 2020).

- Along with advanced ML and DL algorithms, simple and image processing-based methods that work directly with image objects after segmentation (background extraction and preprocessing) and shapes should be developed using open-source platforms such as ImageJ (`https://imagej.nih.gov/ij/`), R (`https://www.r-project.org`), and Python (`https://www.python.org`).

- Use of other image acquisition platforms like low altitude aircraft, fixed-wing UAVs, phenocam, and other near remote sensing devices should be employed and stand count analysis methods specific to these platforms should be developed.

2.5. Conclusion

This paper reviewed plant stand count evaluation methods using the unmanned aerial vehicle (UAV) or remote sensing imagery. Using three academic databases, Agricola, Web of Science, and Scopus, search keywords with boolean combination, screening out duplicates, and eligibility evaluation to select the subject matter relevant articles, in the year range between 2000 and 2020, only 29 peer-reviewed journal articles were found. A growing trend of increased research in this subject after 2015 was observed, showing a great interest in this precision agriculture area.

Specific conclusions that can be derived from the review are: (i) the period when the plants have emerged with enough foliage to be detected and have no canopy overlap is the best for image acquisition; (ii) optimal flying height should be selected to balance field coverage and accuracy; (iii) compared to RGB, the L*a*b* color space can provide better

segmentation results; (iv) hyperspectral camera imagery can provide good discrimination and reduces color distortion; (v) data augmentation should be performed when using the deep learning models if sufficient original data is not available or it is expensive to collect; and (vi) transfer learning models, based on previously developed pertained models, can be used to reduce the computational time and resources.

In the reviewed articles, it was found that all three types of computer vision methods, namely, traditional image processing, machine learning, and deep learning, have been used to count plants in the field. The use of computer vision algorithms provided promising results in stand count especially with corn and cotton with good accuracy, hence the stand count of row crops is feasible with appropriate techniques. This methodology and can be extended to count other row crops and horticultural crops. Various factors such as the flying height of the UAV, plant growth stages during the time of data collection, lighting condition, and camera (type) selection affect the quality of the image during the image acquisition process that intern affects the accuracy of the model. More research and investigation on finding the optimal parameters and camera selection for the data collection process are required. Furthermore, the use of open-source platforms for stand count analysis through direct image processing algorithm development should be promoted to increase the stakeholder participation.

3. OPEN-SOURCE IMAGEJ BASED EARLY-SEASON CORN STAND COUNT EVALUATION AND SPATIAL DISTRIBUTION USING UAV IMAGERY *

3.1. Abstract

Assessing the plant stand count and spatial distribution at the early season is crucial as it helps the farmer decide to either go for replanting or continue with the crop. Traditional stand count methods in the field are laborious, costly, and time-consuming. Nowadays, several image processing software programs are available to evaluate the plant stand count and spatial distribution, but they all require an annual subscription and process all data on the cloud platform, which may affect data security. Apart from these, algorithms also enforce certain restrictions like high-resolution images and straight crop rows and require a high processing time. Therefore, an image processing algorithm using ImageJ open-source software platform that can count the number of plants and evaluate the spatial distribution of the plants in the field was developed. The input image for the plugin is the RGB color image obtained using an unmanned aerial vehicle (UAV) flown at different heights. The excess green method produced the segmented binary image and the profile plot method with an appropriate cutoff value found the crop row coordinates. A novel "sliding and shifting ROI"

* This paper is planned to be submitted as a research article in the *Biosystems Engineering* journal. Authors: Harsh Pathak, C. Igathinathane, Paulo Flores, S. Sunoj, and D. Archer. Title: Open-source ImageJ based early-season corn stand count evaluation and spatial distribution using UAV imagery. Harsh Pathak did the investigation, developed codes, and wrote the original draft. Dr. Igathinathane Cannayen is the major advisor and the corresponding author who worked with Harsh Pathak throughout the manuscript and code development. All the co-authors have assisted in the research direction and review of the manuscript.

(SSROI) method developed efficiently counted the plants along the rows and covered the whole field, even when the plants in the row were off-linear. Using the centroid distances of the plants, a color-coded spatial distribution map that visualizes the skips (single or double), doubles, and ideal cases were also output from the plugin, which helps in making the site-specific farm management decisions. The developed methodology of the plugin can be readily extended to evaluate the stand count of other row crops, such as sunflower, safflower, other agricultural, and horticultural crops.

3.2. Introduction

Corn (*Zea mays* L.) is grown in almost all parts of the United States, and about 37 231 080 ha (92 million acres) of land were planted with the corn in 2020 (USDA, 2020). The United States produced around 366×10^6 Mg (14.42 billion bushels) of corn and exported around 14.3 % of the entire production in 2018/19 (US-Grains-Council, 2019). Corn yield is closely related to plant stand count and spatial distribution (Leng and Huang, 2017). An early estimation of stand count is necessary as it can help in making replanting decisions, if necessary when the expected plant population was not achieved. However, simpler and user-friendly methods of stand count that are accessible and affordable to the producers are yet to be developed. Therefore, this study aims to develop a user-friendly image processing methodology for early corn stand count and distribution using open-source software such as ImageJ — "Image Processing and Analysis in Java" (Ferreira and Rasband, 2012; Rasband et al., 1997).

Stand count along with stand distribution influences the crop yield. At a specific plant stand count, the corn yield is maximized for a given amount of nutrients (Duncan, 1958). Stand count is influenced by stand uniformity, soil temperature, seed depth, herbicide injury,

weather, and pest attacks (Al-Kaisi and Hanna, 2006; Carter and Nafziger, 1990; Nielsen, 1993). Even though stand count is directly proportional to yield, spatial distribution also significantly influences crop yield. When there is a missing plant in a row, then the plants on either side of it receive a yield loss of 47 % and 19 % in a low-density and high-density agricultural field, respectively (Nafziger, 1996). Manually collecting such data is time-consuming and may have human errors (Bullock et al., 1998). Therefore, there is a need for a user-friendly, reliable, and automated method for plant stand count and the spatial distribution of row crops.

Several researchers have used remote sensing platforms (Tang and Tian, 2008a; Varela et al., 2018) and proximal sensors or cameras with a ground vehicle to evaluate the plant population (Shrestha and Steward, 2003; Zhang et al., 2020d). Ground-based methods can count the number of plants more quickly and accurately when compared to manual scouting. The machine vision technique was used to count the number of corn plants at V3 to V4 stages in a field (Shrestha and Steward, 2003). For counting the plants, plant row features such as a total number of plant pixels and the plants median position to identify the corn plant were used; however, the algorithm failed to differentiate between corn and weed plants as the identification was based only on reflectance (Shrestha and Steward, 2003). Inclusion of shape features, namely roundness and plant canopy areas was attempted to increase the accuracy in classifying the weeds and the corn plants (Shrestha and Steward, 2005).

Mosaicking errors in computer vision algorithms is another cause of concern. These errors, especially with ground-based sensing, are caused when driving over bumpy soil surfaces caused by the tractor tire treads or large soil clods. Estimation or identification error can occur when corn plants are damaged by tractor tires (Tang and Tian, 2008b). Recently, a

3D printed field robot was developed for phenotyping, which does not cause damage to the plants while moving in a field but has tracking failures (Zhang et al., 2020d). Ground-based sensing methods take from minutes to hours to collect measurements depending on the field size (Sankaran et al., 2015); therefore, it is limited to small areas (Varela et al., 2018).

Remote sensing methods include sensing using satellites, manned aerial vehicles, and unmanned aerial vehicles (UAV), and are developed to address the limitations of ground-based sensing. These methods have the potential to reduce the cost of field scouting by automating the process. However, remote sensing using satellite images was found to have insufficient spatial resolution (Peña et al., 2013), and data may be altered due to cloud cover (Rudd et al., 2017). On the other hand, UAVs can provide high spatial and temporal resolution imagery with greater operational flexibility such as on-demand missions at varying flight height (Sankaran et al., 2015).

A UAV was employed for data collection to digitally count the number of corn plants in a field using the decorrelation stretch contrast enhancement process on MATLAB (Gnädinger and Schmidhalter, 2017). This method utilized the color differences between the old and young plants, and only the light green color was used to count plants to avoid overlapping. Since the entire algorithm works on spectral information, it may lead to an error in areas of the field with greater weed pressure (Gnädinger and Schmidhalter, 2017). A decision tree classifier was implemented to count the corn plants in real field conditions using the information of the geometric descriptors (Varela et al., 2018). Potato stand count was performed on ArcGIS using multispectral imagery that achieved good correlation ($r = 0.82$) between the manual and image-based plant count, which clearly shows how high-resolution aerial imagery can be used for estimating the emergence of row crops (Sankaran et al., 2017).

Most of the research reported is conducted on expensive, annually subscribed proprietary software like MATLAB or ArcGIS because of their built-in libraries and packages, which makes image analysis coding easier. There also exist some commercial, ready-to-use software like Agremo, Precisionhawk, and Plot Phenix that can count the plants in a field; but they all require an annual subscription, data should be uploaded to the cloud platform, and require straight crop rows for effective operation. Due to the subscription cost, around 40 % of producers stopped using these commercial software packages, while around 12 % of the producers do not use these software packages because of privacy concerns related to cloud computing data uploads (DeLay, 2020).

To overcome issues such as subscription cost, privacy concerns, a user-coded program using open-source software that supports in-house processing is required. Therefore, in this study, Fiji (ImageJ), a freely available open-source, image processing and analysis program was used (Schneider et al., 2012). Several researchers have used ImageJ for developing specific agricultural applications: identifying shape and particle size distribution (Igathinathane et al., 2008), measuring the orthogonal dimensions of food grains (Igathinathane et al., 2009), chlorpyrifos (pest control chemical) partitioning from soil and plants (Rogers and Stringfellow, 2009), measuring sunflower floral dimensions (Sunoj et al., 2018c), calibrating color of digital images (Sunoj et al., 2018b), and several others. Therefore, a user-coded plugin can address the needs of plant stand count evaluation, reduce the subscription cost, and can enforce data security because there are no interactions with the cloud platform. The specific objectives of this research were: (a) automatically identify corn plant rows that may not fall in a straight line from UAV imagery; (b) evaluate corn stand

count; and (c) determine plant spacing and produce a plant spacing distribution map to visualize doubles and skips in crop stand.

3.3. Materials and Methods

3.3.1. Image Data Acquisition

The image data was collected from the Carrington Research Center, Carrington, ND, USA (47°30′29.5″N, 99°7′14.5″W). The corn seeds were planted on May 21, 2020, with an expected target population of 32 000/ac. The spacing between the target rows was 0.76 m (30 in) and between plants was 0.15 m (6 in). Image data was collected on June 2020, four weeks after planting, when the growth stage of corn was V3 to V4. While another set of data was collected on June 2021, seven weeks after planting, when the growth stage of corn was V4 to V5. The expected target population for year 2021 was 34 500/ac.

A quadcopter DJI Phantom 4 Pro (`https://www.dji.com/phantom-4-pro`) that has a 20 MP RGB camera and has a vertical take-off and landing ability was used to take the aerial images over the corn field. The flight route was programmed and controlled by the software provided by PIX4DCapture IOS application. The flying height of the UAV was set to 15.24 m (50 ft) above ground level, and the overlap between the images, both front and side, was set to 75 % for the year 2020. While for the year 2021, data was collected at four different heights (9.144 m, 15.24 m, 22.86 m, and 30.5 m), and the overlap between the images was set to 80 %. The images were collected using the 20 MP RGB camera. The collected images were transmitted from the camera to the computer and were orthomosaicked (image stitching) using the PIX4Dmapper software by PIX4D licensed through the university subscription (`https://www.pix4d.com/product/pix4dmapper-photogrammetry-software`).

3.3.2. Overall Image Processing Approach of Plugin Development

The Figure 3.1 illustrates the various stages of image processing involved in the plugin development as a process flowchart. Overall, the input color image was segmented using the method selected by the user and the binary image was created. The binary image was easy to work with as it contains only two types of pixels (values: 0 and 255). The binary image was used to identify the crop rows, count the plants based on the developed sliding region of interest (ROI) method, and draw the spatial distribution map with color-coding based on plant distances and ideal plant spacing. Detailed information on all the processes is presented in appropriate subsections.

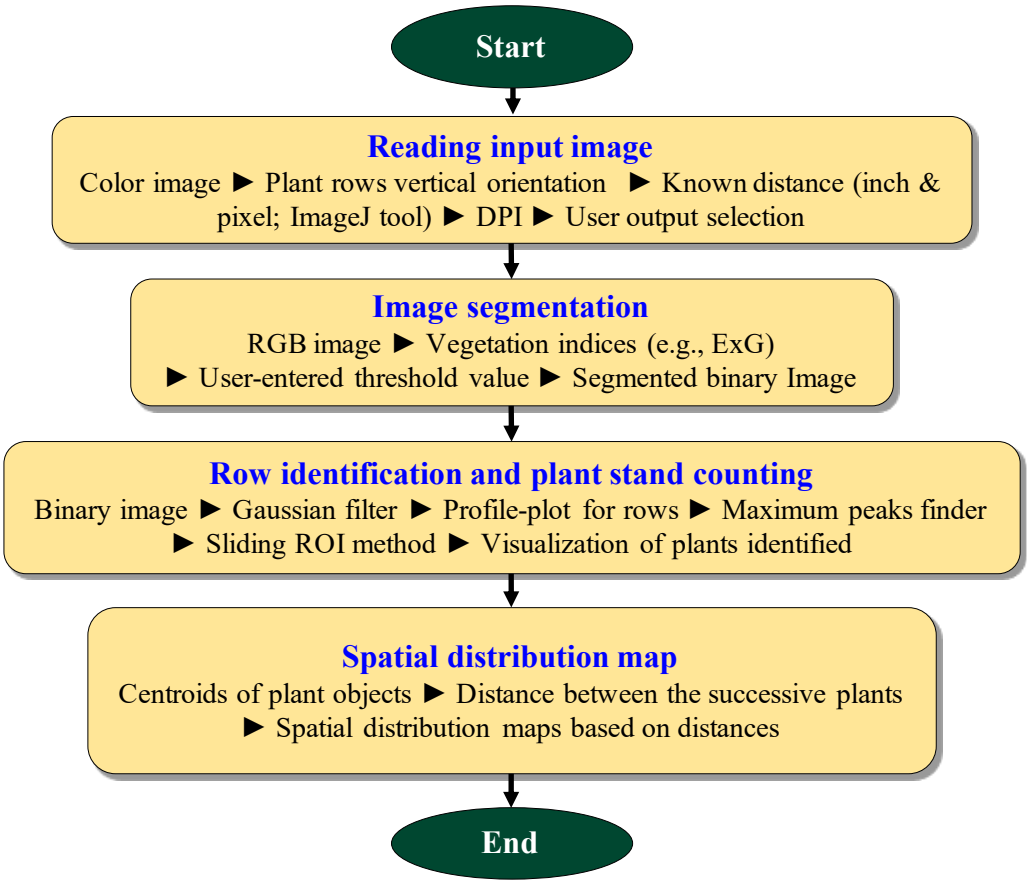
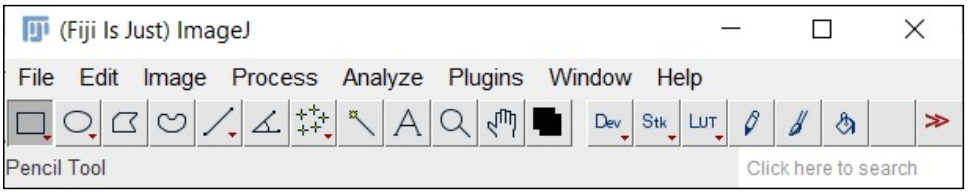


Figure 3.1. Flowchart of the image processing approach used in the plugin developed to evaluate the corn stand count and spatial distribution.

3.3.3. Image Analysis Plugin Class Development

The plugin, a computer program and java class that runs in the ImageJ environment, was developed using Fiji (ImageJ) platform. The plugin works on RGB images that should be loaded/opened in the ImageJ environment. Once the image is read, the plugin opens a "Generic Dialog box" in which the user provides certain inputs and chooses the desired results by selecting the corresponding options coded in the plugin (fig. 3.2). The algorithms

(A)



(B)

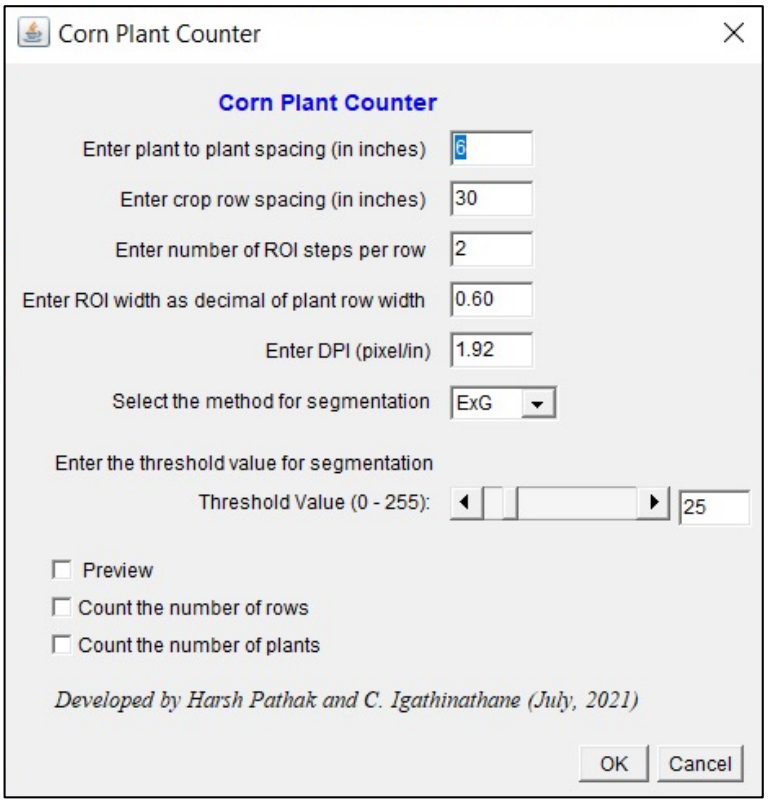


Figure 3.2. Integrated development environment of Fiji/ImageJ for plugin development (A), and input panel of the developed plant stand count plugin showing various inputs and output options (B).

were coded to identify rows, count the number of corn plants and produce a spatial distribution map.

3.3.3.1. Image preprocessing

The initial step in almost all image processing plant detection approaches is to segment the image by dividing it into two different classes: plant (crops and weeds) and background (soil and residues). Dividing the image into two classes, one representing the vegetation and the other the background, is termed segmentation. Segmentation is an important step in image processing because inappropriate segmentation may lead to misclassification and hence will affect the accuracy of the algorithm (Hamuda et al., 2016). The common approach of segmentation of color image through grayscale will not provide good results because the plants and background pixels have similar grayscale values (Tian and Slaughter, 1998).

Several methods have been developed for segmenting the crop from the background, but the most common are the color index–based methods, learning–based methods, and threshold–based methods (Hamuda et al., 2016). In this research, the color index–based method was used for image segmentation as learning–based methods increase the computational time, and the threshold–based methods require several adjustments with the varying light intensity (Hamuda et al., 2016).

Several types of color–index based indices are defined in literature such as normalized difference index — NDI (Woebbecke et al., 1993), excess green index — ExG (Woebbecke et al., 1995), color index of vegetation extraction — CIVE (Kataoka et al., 2003), excess green minus excess red index (Neto, 2004), and modified excess green index — MExG

(Burgos-Artizzu et al., 2011). In this research, several vegetative indices defined in the table 3.1 were tested for color image segmentation to extract the plants from the background.

Table 3.1. Vegetative indices tested for segmentation of the color image of the field plots using the unmanned aerial vehicle.

Segmentation method	Formula
Excess green segmentation (ExG)	$2G-R-B$
Visible atmospherically resistant index (VARI)	$\frac{G-R}{(G+R-B)}$
Red green ratio index (RGRI)	$\frac{R}{G}$
Excess red index (ExR)	$1.3R-G$
Excess green minus excess red (ExGR)	ExG − ExR
Normalized green - red difference index (NGRDI)	$\frac{(G-R)}{(G+R)}$
Vegetative index (VI)	$\frac{G}{R^a B^{(1-a)}}$
Modified excess green index (MExG)	$1.262G-0.884R-0.311B$
Green chromatic coordinate (GCC)	$\frac{G}{(R+G+B)}$
Color index vegetation extraction (CIVE)	$0.441R-0.811G+0.385B$
Visible difference vegetation index (VDVI)	$\frac{(2G-R-B)}{(2G+R+B)}$

Note: R, G, B stands for red, green, blue pixel values from the RGB color image.

The R, G, B values of an image were used to calculate the various vegetative index values (table 3.1). Then by applying the appropriate threshold (T_h), the RGB image was transformed into a binary image. The condition used for thresholding was:

$$\text{Object pixel}_{(x,y)} = \begin{cases} \text{Plant, value} = 255 & \text{Index value} \geq T_h \\ \text{Background, value} = 0 & \text{Otherwise} \end{cases} \tag{3.1}$$

The user has been provided with the choice to change the segmentation method by selecting their preferred method from a drop-down list. By testing the threshold value on several images, the default threshold value was set to 20, but the plugin also provided the choice to the user to change the threshold value either by manually entering in the text–box or by using a slider. After the segmentation process, the "Particle Analyzer API" was used to create a mask (image copy) using the area threshold to remove the noise and other artifacts (filtering out smaller area particles) that exist in the segmented image.

3.3.4. Crop Row Identification using Profile Plot Method

Crop row patterns are decided in a manner so that the agricultural implement gets the longest and straightest passes (Poncet et al., 2012). Crop row identification is the basic step for counting the plants in the field. Row identification is necessary for the automatic navigation of agricultural robots and tractors as it can assist in the automation of different on-farm activities (Jiang et al., 2016; Zhang et al., 2018). Several methods have been proposed in the past for crop row identification such as Hough transform, linear regression, stereo vision, and horizontal strips (Åstrand and Baerveldt, 2005; Varela et al., 2018), but these methods have their limitations (especially with high weed pressure). Therefore, there is a need for a simple, reliable, and automated process for crop row identification.

For identifying the crop rows, the approach used was to plot the profile of segmented plant pixels along the crop rows across the field and finding the peaks. For the rectangular section of the field with crop rows oriented vertically, the profile plot displays the column's average plot, where the x-axis shows the distance and the y-axis shows the averaged pixel intensity. The intensity profile plots for the crop rows may not be smooth and produce multiple local maxima (breakpoints or peeks). Simply using these local maxima as an

indication of crop rows resulted in an excessive number of crop rows. To overcome the issue of breakpoints in the curve, a 2D “Gaussian Filter” was employed to smooth and remove the noise from the original segmented image. The following equation represents a two-dimensional Gaussian kernel:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.2}$$

where σ is the variance, x is the distance from the origin in the horizontal axis, and y is the distance from the origin in the vertical axis.

With the smoothed profile plot using the ImageJ “Profile Plot” method, after the Gaussian filter, the “MaximumFinder()” class was used to extract the peak locations along the image’s width, at which the intensity profile had the maximum values. One of the significant challenges is selecting the profile cutoff value for crop row identification, which sometimes is hard-coded or taken as the average (Shajahan, 2019). Therefore, an automatic cutoff determination method was developed and applied to the profile plot of the segmented image. If the intensity value at a particular location was greater than 25 % of the intensity’s maximum value, it was termed plant row. This method of row identification was limited to straight rows and cannot be used directly to identify off-linear rows. The extracted peak values were stored in the array for further processing during stand count.

3.3.5. Plant Counting using Sliding and Shifting ROI (SSROI) Method

Even though it is possible to count the number of plants in a field as a bulk measure using the ImageJ ParticleAnalyzer() method on the segmented binary image, it is not possible to count plants in each row, find the with-in row variability, and overall spatial distribution of

plants. Therefore, for counting the plants in rows, and determining their spatial distribution in the field, a novel “Sliding and Shifting ROI” (SSROI) method was developed. The SSROI method performs the stand count along the identified crop rows.

In the SSROI method, the rectangular ROI was overlaid on the segmented binary image and allowed to slide from the bottom to the top of the image. The center of lower and upper edges of SSROI coincides with the profile plot array values. The length of the SSROI was based on the number of desired shifts along the vertical direction, and the width was based on the row-to-row and the width coverage for including the plants along the row or excluding the objects (possible weeds) between the rows.

From the known physical dimensions of the field (sometimes from row spacing values) and the corresponding number of pixels in the image (ImageJ “line” or “rectangle” selection tool used), the dots per inch (DPI) value of the image was established. This DPI value, evaluated with the image open and ImageJ measurement tools, is one of the inputs of the plugin (fig. 2.2). The DPI was used to convert pixel dimensions in the image to represent the physical dimensions in the field, which is required to judge the identified image’s plant spacings against the desired plant-to-plant spacing in the field.

The number of plants inside the SSROI was counted using the ParticleAnalyzer() method (operates inside the ROI) while it slides from the bottom to the top of the image, and their centroid coordinates were obtained. The centroid points of all the plants were used further to identify the crop rows, both off-linear and straight. The ROI march method was programmed to ignore the plants that touch ROI edges, resulting in fewer plant counts while counting. To avoid the counting error, the code was written such that if the top of the ROI encounters the object (plants), then the ROI slides its position along the vertical direction to

accommodate the touching plant and counts it. Markers were drawn on the identified centroid using the ImageJ “fillOval()” method.

The bounding boxes around the plant were also drawn to provide spatial information using the drawRect() method of ImageJ. The labels representing the number of plants were generated using the “TextRoi()” class and placed at a slight offset along the x-shift from the centroid of the plant. The labeled image was used for validating the plugin performance in counting the plants with respect to the manual counting the plants in the image by visual inspection. A line connecting the centroid will help in visualizing how plants that are not in the straight-line, but within the SSROI were included in the counting. Apart from these results, a pop–up window showing important results such as the number of plants, doubles, and skips is also developed.

3.3.6. Spatial Distribution Mapping

In order to develop the spatial distribution map, the distance between the centroid of the successive plants present in the crop row was calculated using the distance formula. Further, a color-coded crop spatial distribution map based on five classes, namely, single (ideal), double, single skip, double skip, and multiple skips, was also generated to visualize the entire field better and help make the site-specific decisions. The distance between the plants was color-coded as rectangles using the ImageJ “fillRect()” method. The filling color for the rectangle was selected from a list of five colors (green: for ideal, blue: for doubles, orange: for single skip, pink: for double skip, and red: for multiple skips) based on the spacing between them.

3.4. Results and Discussion

The developed plugin was validated using the field images using the RGB camera mounted on the UAV. The various results of image segmentation, crop row identification, plant stand count, and spatial distribution map generation are presented and discussed.

3.4.1. Image Segmentation

Different vegetation indices were tested on the portion of the field imagery and found that ExG, ExR, and VI performed better compared to other segmentation methods (fig. 3.3). It is to be noted that VARI, ExGR, MExG, and GCC also performed well, but the segmented image has some outliers or noise in it. At the same time, CIVE loses certain features in the image, and NGDRI and RGRI did not perform well with our dataset. In this research, the ExG segmentation method was used to create the binary image as it provides a clear contrast between the plants and soil. In addition, ExG of segmentation is more accurate than ExR and is less sensitive to background errors and lighting conditions, and can be used in both cloudy and sunny weather (Basso and de Freitas, 2020; Hamuda et al., 2016).

Figure 3.3. Results of different segmentation methods on a sample field segment of 210 ft wide by 690 ft long: (A) ExG, (B) VARI, (C) ExR, (D) ExGR, (E) VI, (F) MExG, (G) GCC, (H) CIVE, and (I) VDVI.

3.4.2. Crop Row Identification

The intensity profile plot generated by using the ProfilePlot API of ImageJ directly on the segmented image has a lot of breakpoints and was not continuous, due to which counting the number of rows and identifying their starting point was hard. By applying the Gaussian filter on the segmented image, the profile plot curve became smooth and continuous. Out of the different values of $\sigma(x)$ and $\sigma(y)$ and it was found that $\sigma(x) = \sigma(y) = 9$ worked best for the data (fig. 3.4). The smoothed image was used to extract the profile plot to count the number of rows. The MaxFinder() API, along with the array thresholds resulted in accurate detection of the initial start point along the x-direction.

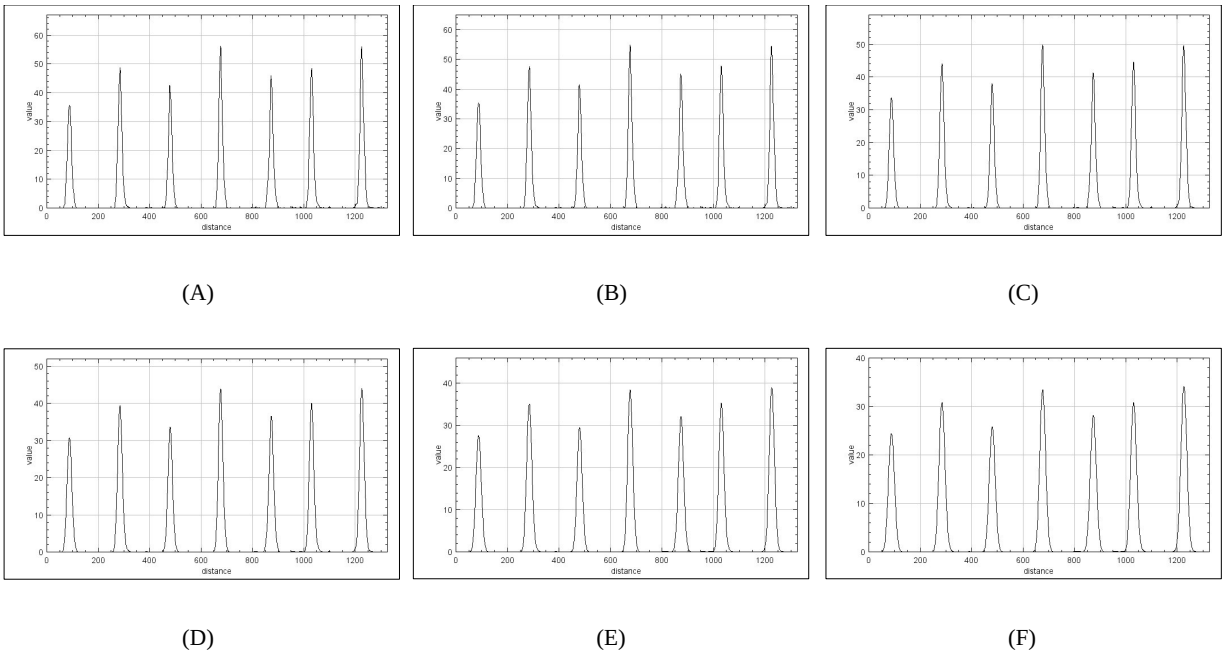


Figure 3.4. Profile plot for different values of sigma (σ): (A) $\sigma = 0$, (B) $\sigma = 1$, (C) $\sigma = 3$, (D) $\sigma = 5$, (E) $\sigma = 7$, and (F) $\sigma = 9$.

Overall, the Figure 3.5 illustrates the application of crop row identification using profile plot and Gaussian filter. This method resulted in a corn crop row identification

accuracy of 100 %, when the crop rows are straight. Due to its good accuracy, the method can be further used to identify the other row crop rows such as sunflower as long as they are straight. This method fails to identify off-linear crop rows, which can be further identified with the SSROI method.

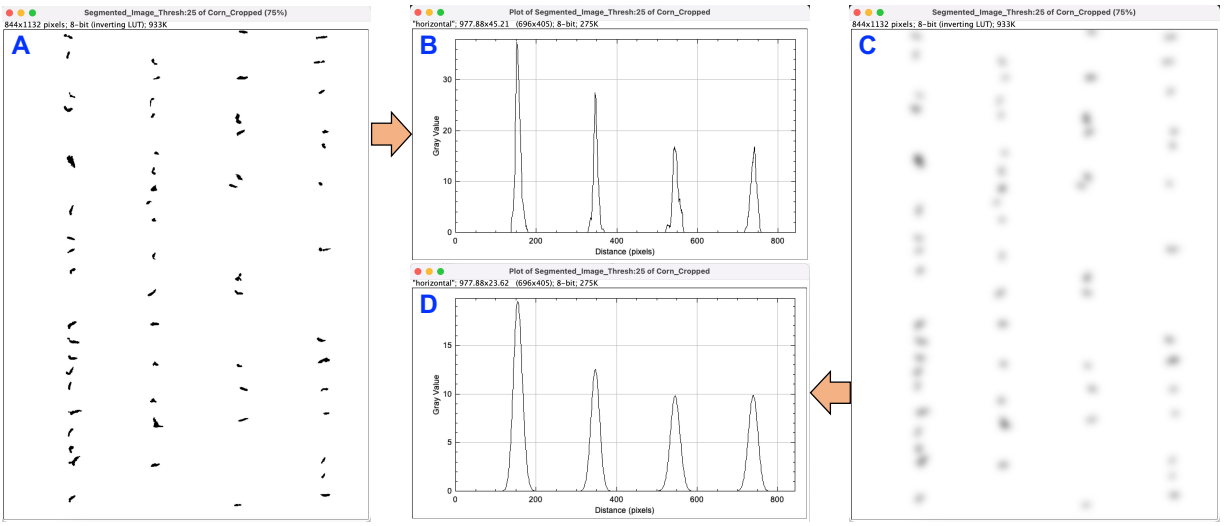


Figure 3.5. Corn rows identification using profile plot and Gaussian filter. Sample binary image of corn plants (A), raw profile plot of the binary image showing irregularity with several local peaks (B), Gaussian blur filter applied to sample binary image (C), and profile plot based on the filtered image showing smooth and clear peaks for identification of the crop rows (D).

3.4.3. SSROI Method of Plant Stand Counting

The ImageJ “ParticleAnalyzer()” method on the segmented binary image (fig. 3.3) derived from the color input image (fig. 3.6 Left) can count the number of plants in a field as bulk measurement, but it is not possible to count plants in each row, find the with-in row variability and overall spatial distribution of plants. Therefore, for counting the plants in rows, and determine their spatial distribution in the field, a novel SSROI method was developed. The outline of SSROI was included in the labeled stand count output for better visualization (fig. 3.6 Right).

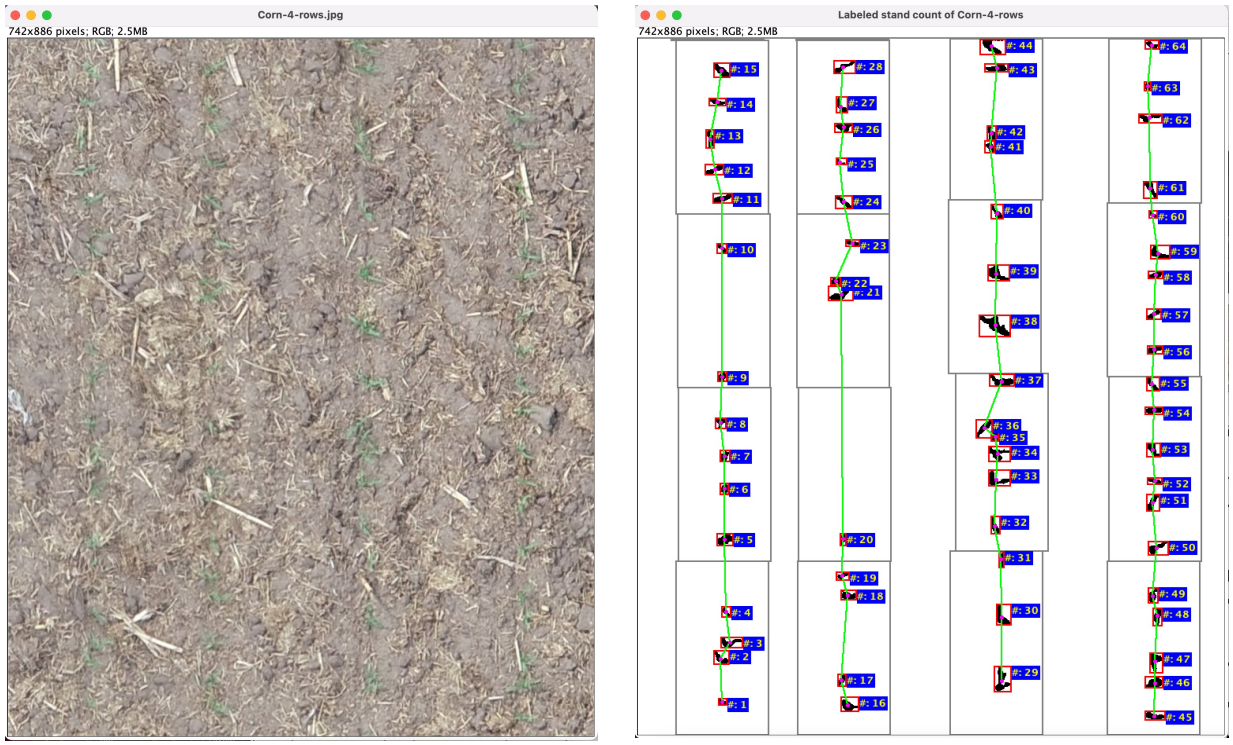


Figure 3.6. Input color image (Left) and the developed sliding and shifting region of interest (SSROI) method of plant stand counting (Right).

The SSROI method performs the stand count along the identified crop rows. In the SSROI method, the rectangular ROI was overlaid on the segmented binary image and allowed to "slide" (y-direction) from the bottom to the top of the image. The position of the subsequent SSROI is based on the average centroid of the plants in the previous SSROI and this positioning allowed to "shift" the SSROI. Such shifted SSROI tends to trace the plants in the crop rows based on the previous plants' locations, thereby addressing the off-linear plants in the row. The center of lower and upper edges of SSROI coincides with the profile plot peak values. The length of the SSROI was based on the number of desired shifts along the vertical direction, and the width was based on the row-to-row and the width coverage for including the plants along the row or excluding the objects (possible weeds) between the rows.

The developed algorithm uses simple image processing techniques such as using vegetative indices for creating a binary image, intensity plots, and SSROI sliding for row identification and stand count evaluation rather than the black box methods such as machine learning and deep learning models, where users generally are not aware of the processes that go into the models. In comparison with the deep learning detection models such as Faster R-CNN, You only look once (Yolo v3) and several more, this method is simple and does not require much domain expertise and rigorous training. Training deep learning models even through transfer learning is computationally expensive and time-consuming (Rai and Flores, 2021). Neupane et al. (2019) tried to train a deep learning model to count the number of banana trees in the field, and the minimum time it took to train the model on RGB imagery was 90 minutes.

3.4.4. Spatial Plants Distribution Mapping

To develop the spatial distribution map, the distance between the centroid of the successive plants present in the crop row was calculated using the distance formula. A color-coded crop spatial distribution map based on five classes, namely, “ideal,” “doubles,” and “single skip,” “double skip,” and “multiple skips,” was also generated to visualize the entire field better and help make the site-specific decisions. The distance between the plants was color-coded as rectangles (fig. 3.7 Left). The filling color for the rectangle was selected from a list of five colors (green: for ideal, blue: for doubles, orange: for a single skip, pink: for a double skip, and red: for multiple skips). A simple plot with green circles representing the centroids of counted plants for easy visualization was also produced as an output (fig. 3.7 Right).

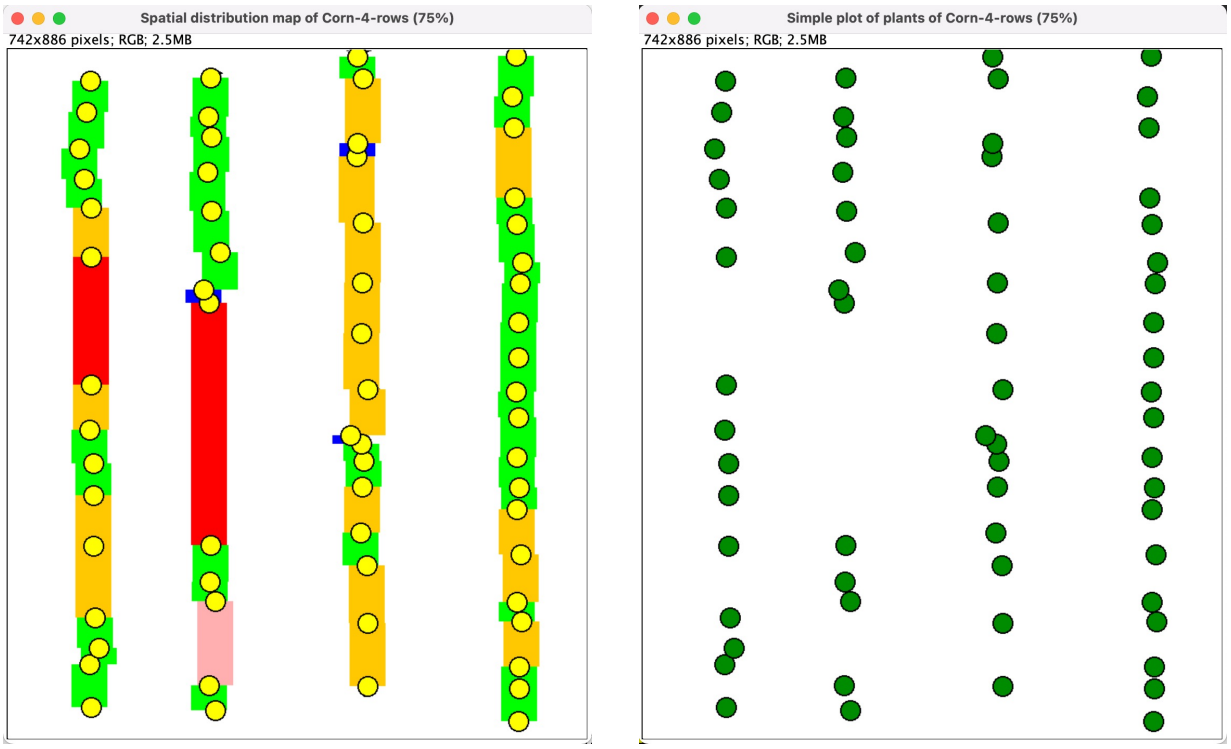


Figure 3.7. Spatial distribution color-coded map of corn rows showing ideal: green, doubles: blue, single skip: orange, double skip: pink, and multiple skip: red (Left), and simple plot of representative corn plants counted and location (Right).

3.4.5. Plant Counting and Spatial Distribution Maps

The SSROI method was efficient in counting the number of plants in the field. It was found that with the increase in ROI height, the overall time to process the image and count the number of plants decreases. In contrast, with an increase in ROI width, the overall time to process the image for counting the number of plants increases. To assess the performance of the plugin, the number of plants counted by the plugin was compared with the manual count performed by visually zooming in the image. During the validation, it was found that the number of plants counted by the plugin was 555, while the number of plants manually counted was 568. Some of the plants were very close to each other and were overlapping, due to which the algorithm counted them as one plant. The algorithm counted the number of

corn plants in the field with an accuracy of 97.71 %. Chen et al. (2018) used the supervised maximum likelihood method to count the number of cotton plants and reported an accuracy of 88.6 %.

The SSROI method also addresses the off-linear rows when the plants are not exactly a straight line (fig. 3.8). This method is similar but tends to be more efficient than the search-hand method (Suresh Babu, 2018), in terms of computational efficiency and time, in which the pixel marches at seven different angles to find the plants. A smaller height of the SSROI will be effective in capturing the off-linear rows, but the algorithm can be improved by ROI rotation as well. In the SSROI method, the plants inside the ROI are identified, their centers are located, and the line joining the centers of different plants are the crop rows. Figure 3.9 shows a closer view of the plant stand count using the SSROI method in identifying the off-linear crop rows.

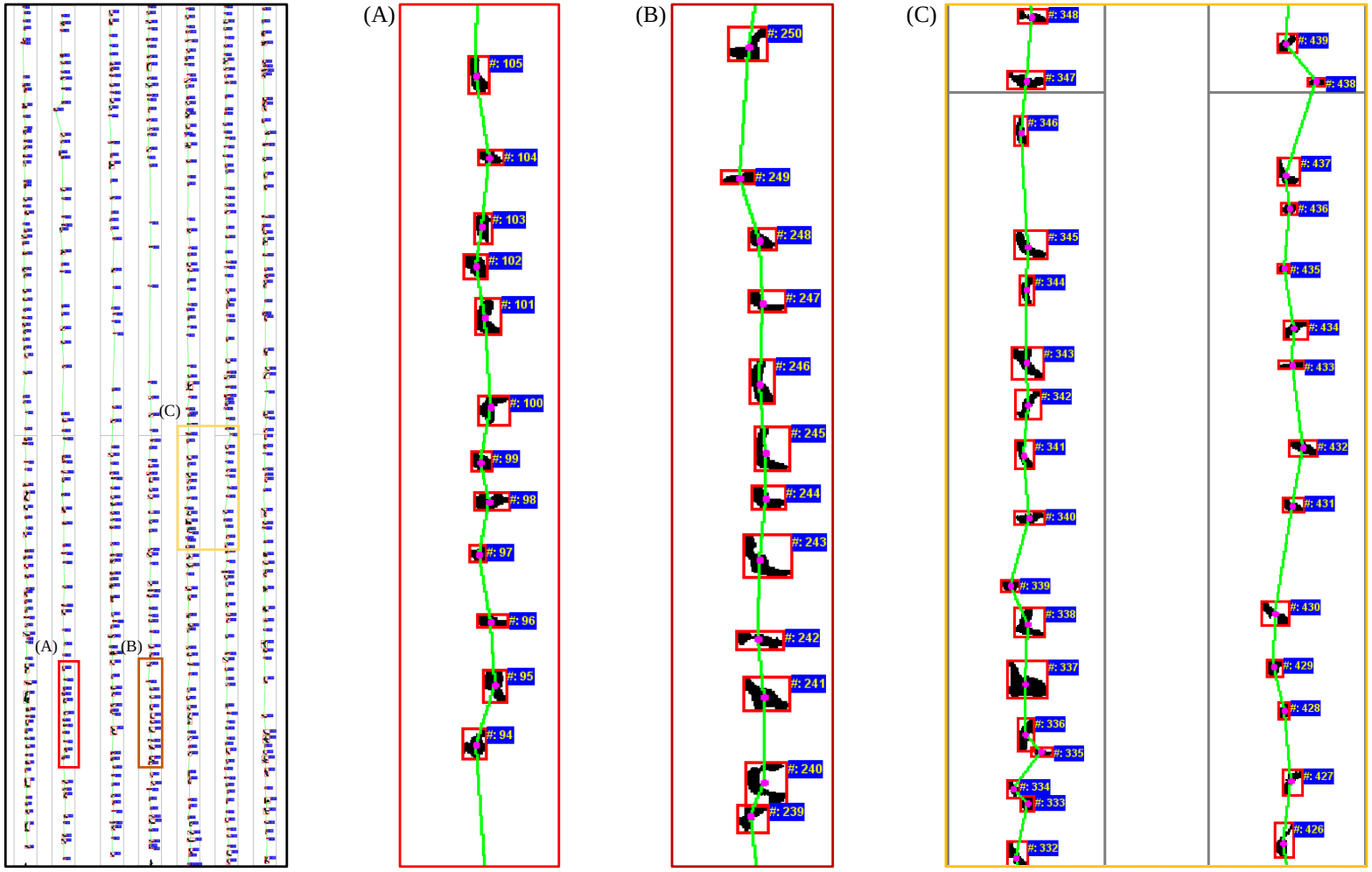


Figure 3.8. Off-linear plants rows identification using the sliding and shifting (SSROI) method on a sample field segment of 210 ft wide by 690 ft long. Inserts show magnified view of the sections of the labeled plot.

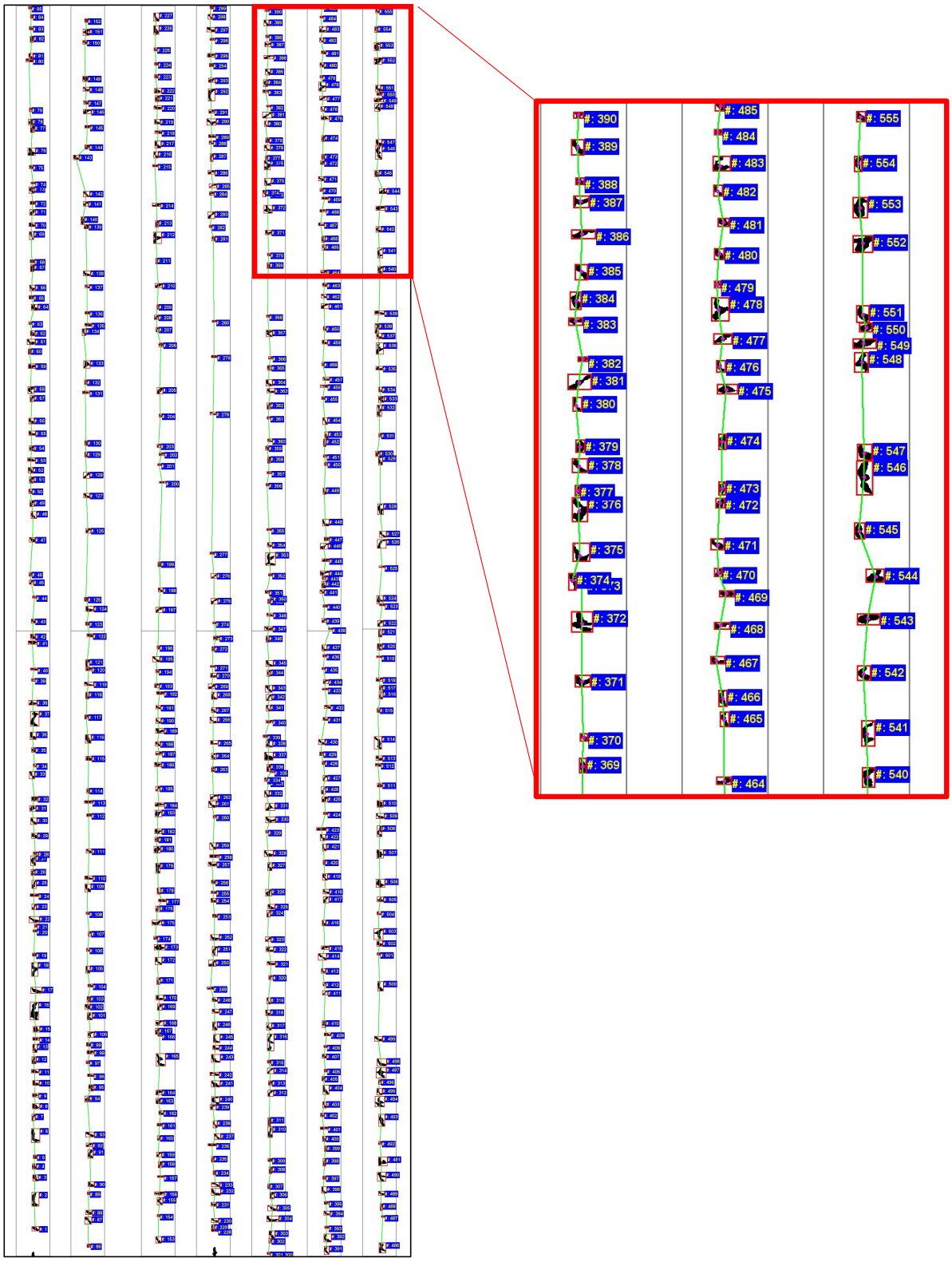


Figure 3.9. Plant stand count labeled along the rows in UAV imagery from bottom to top and left to right.

The plugin produced a color-coded spatial distribution map with five classes (fig. 3.10). The green color shows the ideal situation where the plant spacing was uniform; blue color indicates the multiple plants were close to each other; orange color depicts that there is a single plant missing; pink color represents that two plants are missing, and red color reveals that more than two plants are missing. This map will help farmers evaluate planter's performance in placing the seeds, seed quality and can help to investigate the reasons for variable spacing or non-uniform emergence rate. In general, the multiples are due to the planter's metering mechanism and the size of the seed. While the skips can be due to either partially filled cells, seed viability, environmental problems (e.g., loss of soil fertility) or pests attacks (Shajahan, 2019).

3.4.6. Effect of UAV Flight Height

The developed algorithm was further tested to identify crop rows from the data collected in June 2021 at four different heights (9.144 m, 15.24 m, 22.86 m, and 30.5 m) and at a different growth stage (fig. 3.11). Finding the optimal flying height that balances time to cover the field and accuracy needs to be addressed. It was found that the ExG segmentation method performed well in creating a binary image at all four different heights, and the ROI march method also performed well in identifying the crop rows. However, due to the overlapping between the plants, it is challenging to accurately count the corn plants, as counting accuracy decreases due to occlusion (García-Martínez et al., 2020). It is expected that collecting the high-resolution imagery data early in the growing season at (V3–V4) growth stage with a flying height of 22.86 m (75 ft) will provide good accuracy.

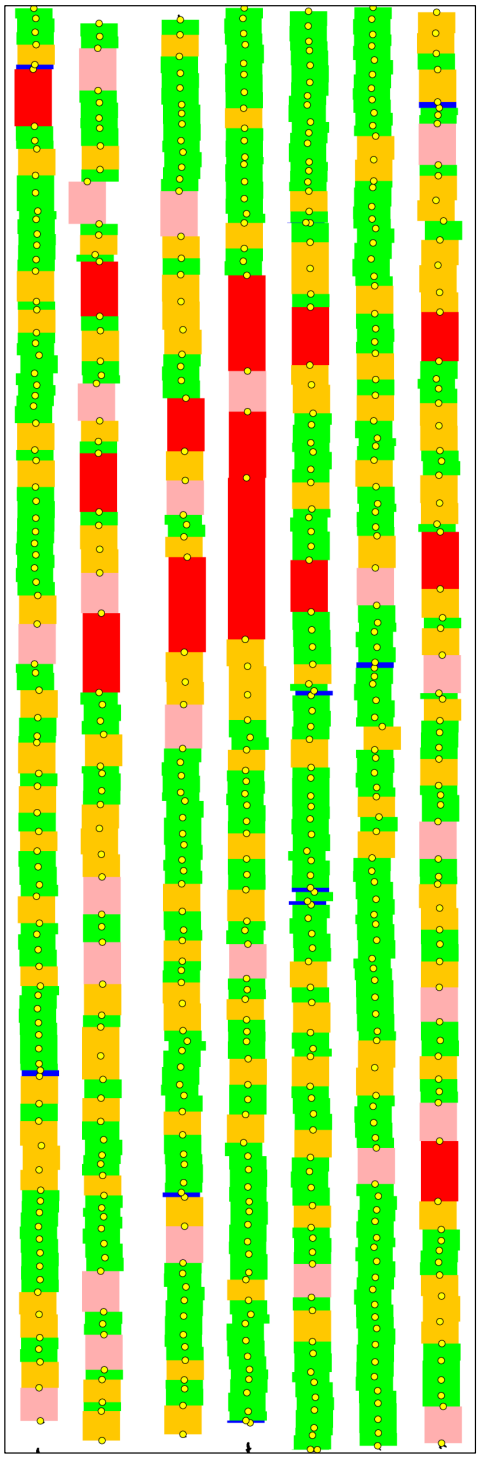


Figure 3.10. Plant stand count distribution map (Green represents plants under optimum distance, blue represents doubles, orange represents single skip, pink represents double skips, and red represents multiple skips). The sample field segment dimension is 210 ft wide by 690 ft long.

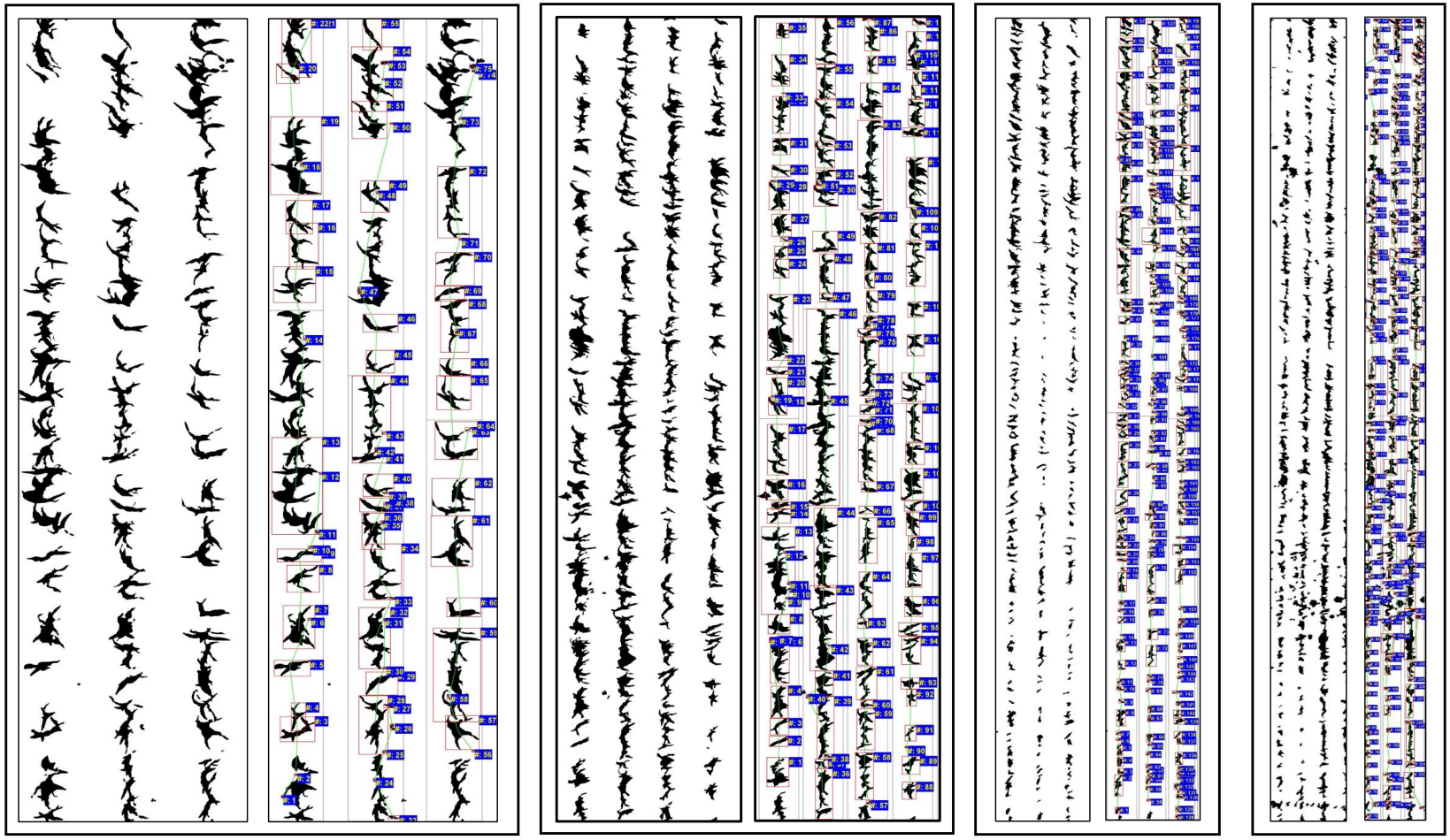


Figure 3.11. Image segmentation and crop row detection at different heights: (A) 9.144 m, (B) 15.24 m, (C) 22.86 m, and (D) 30.5 m.

3.4.7. Demonstration of Field-Scale Stand Count and Spatial Distribution Results

As a demonstration of a field-scale operation, an orthomosaicked image of the corn field (CREC, NDSU, June 17, 2020) of about 2.6 ac having a dimension of about 146 ft wide by 760 ft long was used as input to the developed plugin. The Figure 3.12 shows the original color image, segmented binary image, labeled plant stand count, spatial distribution, and simple plot of plants in the field. The spatial stand distribution and the simple plot representation of the field illustrate that a good plant stand and good distribution was obtained, which was indicated by predominant green bars of "ideal" spacing in the distribution plot and green dots in simple plot, respectively.

Also, it can be seen that the plant rows were straight and the four sliding steps adequately captured all the plants in the SSROIs and covered the whole field well. Overall, the segmentation with appropriate ExG threshold captured the plants effectively, which is the critical stage, on which other successive operations (labeling, distribution, and simple plots) depend.

The profile and Gaussian filtering operation on the segmented image produced 59 prominent peaks (fig. 3.13A) that represented the crop rows. Heights of the peaks were indicative of the number of plants as well as their size. A quick view of the results highlights of the plugin was provided by the pop-up message box (fig. 3.13B), and these results with additional information were also presented in the ImageJ "log" window (not shown). The highlights indicated that the field had 59 rows, plant stand count of 72 145, while 5.24 % of plants were left out (possibly weeds that germinated between rows). Increasing the SSROI width, from the considered 0.6, would have counted some of the plants as they would be accommodated.

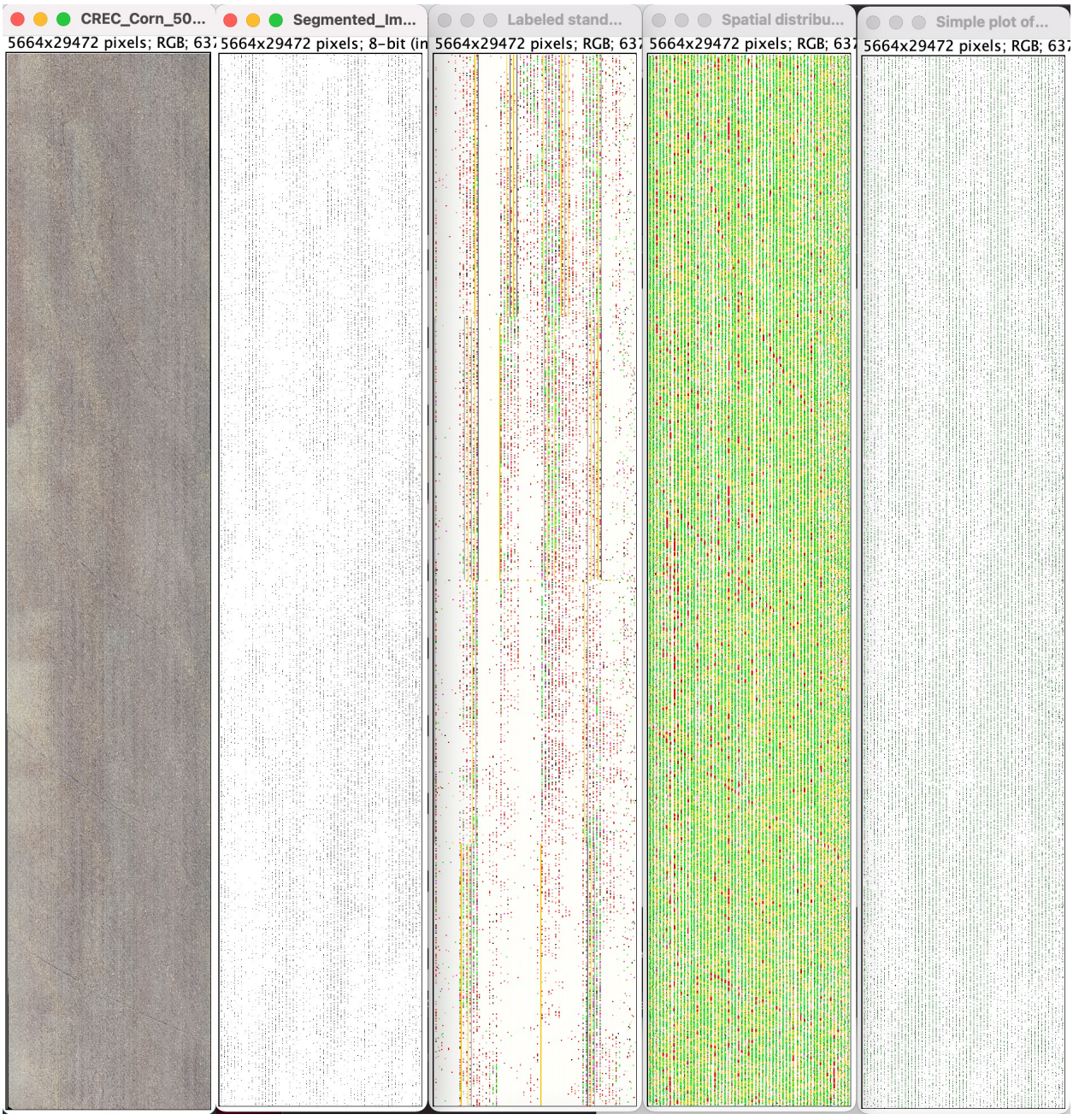
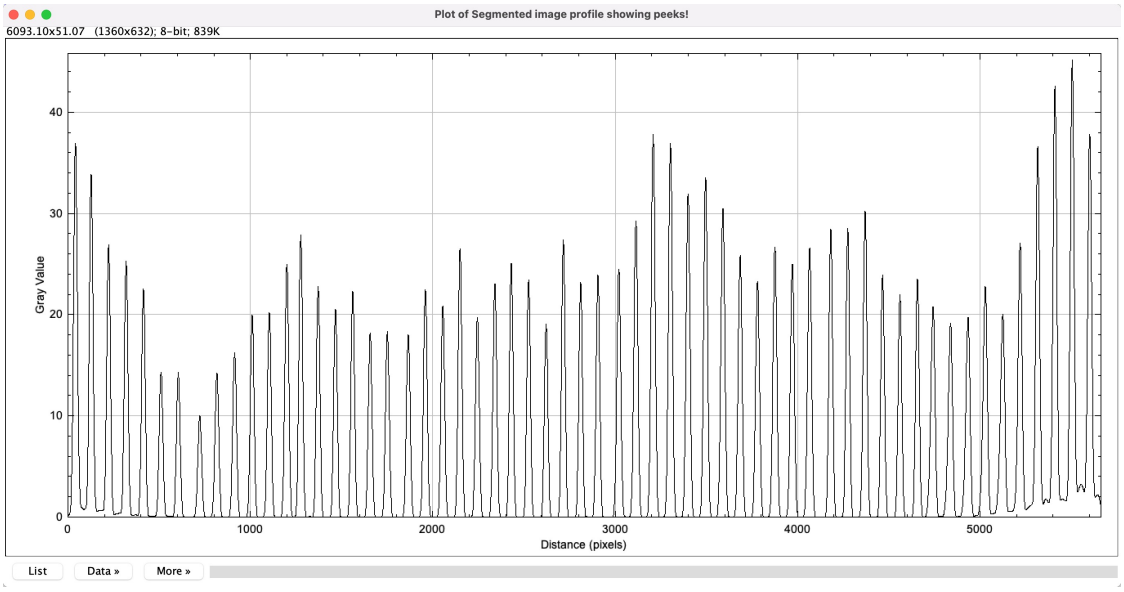


Figure 3.12. Field-scale demonstration of the developed stand count plugin showing original, binary, stand count, spatial distribution, and simple plot of plants. The field dimensions are about 146 ft wide by 760 ft with an area of 2.6 ac.

(A)



(B)

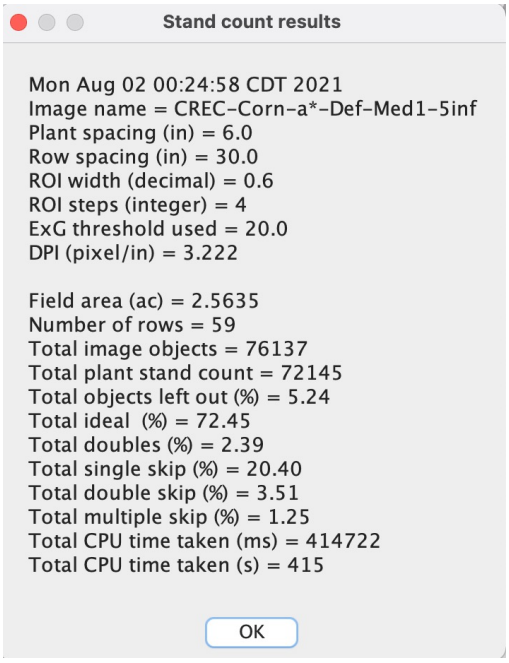


Figure 3.13. Results of row identification (A) and results highlight pop-up message box of the plugin (B) on a field-scale operartion. The field dimensions are about 146 ft wide by 760 ft and area of 2.6 ac.

The message-box also shows the results of the plants' spatial distribution (fig. 3.13B), which indicated that most of the plants were ideally distributed (72.5 %) and only a few belonged to doubles (2.4 %). However, single skips were dominant (20.4 %), while double skips (3.5 %) and multiple skips (1.6 %) were comparable to doubles. Overall, the stand distribution of this field is good and may not warrant any management measure regarding germination and plant health.

The CPU time taken for the evaluation of this field, using the binary image as input, was 415 s, which was about 2.7 min/ac of the field. This result can be considered efficient when compared to cloud-based services. Computational speed can be improved if the number of SSROI along the row was reduced (from 4 applied; fig. 3.13B), especially when the rows are straight. Scenario analysis of the plugin with the threshold limit for creating the binary image, and SSROI steps and width and validation with the ground truth (sometimes performed by manually counting the plants from the original image) will produce the optimized value of these parameters that affect the stand count and distribution.

Along with the graphical outputs and other outputs (pop-up message and ImageJ log window), several textual outputs as ImageJ "ResultsTable" such as (i) properties of individual plants (e.g., area, perimeter, centroid, bounding box, and so on; not shown) along each row, and (ii) average values of selected properties of each row (e.g., number of plants, area, perimeter, bounding box width and height, Feret diameter, equivalent diameter, and CPU time) were coded in the plugin. Figure 3.14 presents the row-averaged results applied to the field-scale image (fig. 3.12), which is a "csv" version of the ResutlsTable output opened in Microsoft Excel.

	A	B	C	D	E	F	G	H	I
1	PlantRow	Number_Plants	Avg_Area (in^2)	Avg_Peri (in)	Avg_BoxWidth (in)	Avg_BoxHeight (in)	Avg_Feret (in)	Avg_EqDia (in)	Time (ms)
2	1	1552	5.17	9.71	3	2.93	3.96	2.38	12058
3	2	1516	4.8	9.36	2.87	2.89	3.84	2.31	8048
4	3	1054	5.5	9.76	2.86	3.03	3.86	2.41	6648
5	4	1392	4.26	8.83	2.81	2.66	3.69	2.22	5787
6	5	1234	4.86	9.49	3.01	2.84	3.89	2.37	5821
7	6	1048	3.78	8.27	2.64	2.49	3.47	2.08	5268
8	7	1040	4.54	9.34	2.9	2.84	3.86	2.28	5239
9	8	1024	4.23	9	2.77	2.75	3.72	2.18	5313
10	9	1094	5.14	10.22	3.11	3.05	4.13	2.42	5283
11	10	1180	6.41	11.73	3.48	3.48	4.68	2.66	5334
12	11	1329	4.94	10.2	3.04	3.14	4.16	2.4	5616
13	12	966	6.41	11.59	3.39	3.48	4.54	2.7	5228
14	...	...	...	...	...	...	...	...	...
15	...	...	...	...	...	...	...	...	...
16	...	...	...	...	...	...	...	...	...
17	44	1110	6.49	11.38	3.66	3.31	4.68	2.79	6146
18	45	1163	6.39	11.16	3.44	3.36	4.52	2.76	5781
19	46	1193	5.98	10.98	3.49	3.26	4.52	2.66	5782
20	47	1396	4.46	9.26	2.89	2.84	3.87	2.3	6411
21	48	1050	5.52	10.25	3.03	3.16	4.08	2.49	5710
22	49	1434	4.13	8.87	2.75	2.75	3.74	2.19	5903
23	50	1439	4.29	9.04	2.8	2.77	3.77	2.22	5718
24	51	1025	6.15	10.91	3.07	3.47	4.31	2.61	6731
25	52	1379	3.9	8.47	2.56	2.71	3.58	2.12	6199
26	53	1302	5.74	10.63	3.04	3.38	4.33	2.54	5920
27	54	1291	6.52	11.25	3.23	3.53	4.53	2.69	5854
28	55	1310	7.02	11.74	3.28	3.7	4.65	2.75	5914
29	56	1396	7.7	12.28	3.41	3.84	4.83	2.86	6393
30	57	1404	7.98	12.57	3.39	4.01	4.9	2.92	5749
31	58	1480	7.64	12.08	3.32	3.8	4.76	2.82	5735
32	59	1777	5.6	10.08	2.87	3.2	4.06	2.45	6431
33	---	---	---	---	---	---	---	---	---
34	Total/Avg	72145	5.58	10.5	3.18	3.19	4.28	2.52	414.72 (s)
35	---	---	---	---	---	---	---	---	---
36									

Figure 3.14. Measured average properties of plants (truncated results) in each row and their grand total and mean values. The area of corn field used for analysis was 2.6 ac.

The result presents the number of plants in each row, along with the grand total. This number of plants in each row is a piece of good information that helps in management decisions. This result showing the various row-averaged properties can be used used to gauge the health of the plants in a row. All the properties in the results with higher values indicate healthier plants and lower values otherwise. For example, the plants in rows 57 and 58 are healthier compared to plants in rows 49 and 50, and the individual values can be quickly checked with the calculated averages shown at the bottom (fig. 3.14).

3.4.8. Limitations and Direction for Future Research

This research was conducted when the corn was at the V3-V4 growth stage when the plants were not overlapping, and the algorithm performed well in counting the corn plants. While no test was performed at the later stage when the plants overlap each other. Counting the objects (plants) in the agricultural sector is challenging because of two significant reasons: (i) arbitrary size of the objects and (ii) overlap between the branches or leaves (Zhang et al., 2020b). Advanced algorithms can be developed that use average area and various segmentation techniques (such as watershed segmentation) to divide the overlap objects into two and helps in counting. In this study, the area was used as a filtering feature, and it was expected that no weeds would exist after applying the filter. However, in the future, other shape factors such as circularity, elongation, roundness, and many more can be used to separate weeds before counting to get more accurate results. The plugin provided accurate results in counting corn plants in the field, therefore it can be further extended to count other row crops such as sunflower. The SSROI rotation so as to address curvilinear planted rows, specifically found on field edges, should also be tried. Yield prediction models can be also be developed by using the stand count information.

3.5. Conclusion

The study successfully illustrated the potential of the user-coded plugin developed using an open-source platform, ImageJ, that can help process the UAV imagery to evaluate plant stand count and produce spatial distribution maps. Of the different image segmentation methods tested, the excess green (ExG) segmentation method was found suitable, while several others are also good enough for segmenting corn plants at an early stage.
The intensity profile of crop rows with appropriate cut-off value was efficient in identifying

the crop rows. The novel sliding and shifting ROI (SSROI) method developed to count the number of plants along the rows and cover the entire field is found efficient. This method by shifting the ROI successfully accommodated for the off-linear plants on the row.
The developed plugin gave an overall counting accuracy of 97.71 %, and also produced color-coded spatial distribution maps to highlight doubles, skips, and ideal conditions to plan site-specific farm management decisions further. It was found that the plugin can efficiently identify crop rows even when the flying height of the UAV was 30.5 m or more as long as good contrast and enough pixels define a plant, but failed to count plants due to the amount of overlap between the plants.

Future research should include developing a plugin that can count plants even at the later stages of their growth when there is overlap between them, rotational ROI to accommodate edge curvilinear rows. Other shape factors should be used to increase the robustness and accuracy of the model. Similar plugins should be developed to evaluate the stand count of other row crops, such as sunflower, safflower, other agricultural, and horticultural crops.

4. MACHINE VISION METHODS FOR WEED CLASSIFICATION USING OPEN-SOURCE PLATFORMS *

4.1. Abstract

Weed management practices strive to reduce weeds, which compete with crops for nutrients, sunlight, and water and are thus important for maintaining production. In most weed management practices, the first step is to identify or classify weeds. However, efficient identification and classification of weeds are challenging using conventional manual methods such as field visits. Therefore, in this study, we propose to classify four common weeds in the corn field of North Dakota (common lambsquarters, common purslane, horseweed, and redroot pigweed), also applicable to other regions, using computer vision methods. Weeds were grown in plastic trays under natural conditions from the field soil, and the images were collected using an RGB camera. For each weed species, 21 shape features were extracted through a developed ImageJ plugin. The handcrafted simple image processing approach was highly successful with a few weed species (common lambsquarters and horseweed, and horsweed and redroot pigweed) in distinguishing each other, while did not perform well with other species combinations and classifying them collectively owing to the similarity of the shapes. However, the three advanced non-parametric machine learning (ML) models, namely, k-nearest neighbor (kNN), random forest (RF), and support vector machine (SVM), resulted

* This paper is planned to be submitted as a research article in the *BioSystems* journal. Authors: Harsh Pathak, C. Igathinathane, Kirk Howatt, and Z. Zhang. Title: Machine vision methods for weed classification using open-source platforms. Harsh Pathak did the investigation, developed codes, and wrote the original draft. Dr. Igathinathane Cannayen is the major advisor and the corresponding author who worked with Harsh Pathak throughout the manuscript and code development. All the co-authors have assisted in the research direction and review of the manuscript.

in high accuracies with RF outperforming the others (F1-score of 0.91). It is recommended that the handcrafted simple image processing algorithms should be tried first for the identification and classification of weeds before resorting to the advanced and complex versatile ML modeling approaches. Such tools and methodologies using open-source platforms will be helpful to farmers, producers, and other users related to crop production.

4.2. Introduction

Weed pressure annually causes corn and soybean production to face a yield and financial loss of 50 % and 52 % amounting to 26.7 billion US dollars and 16.2 billion U.S. dollars, respectively. (Soltani et al., 2016,1). Weeds pose a large threat to production as they reduce yield by blocking irrigation canals and by competing against crops for nutrients, water, and sunlight (Lin, 2010; Pulido et al., 2017). Furthermore, weeds harbor pests and release chemicals that are inhibitive to the growth of crops. Corn seeds have the ability to sense whether weeds are growing above the ground by the light that weeds reflect that penetrates the soil. This sense lurks and triggers cellular changes that delay germination and hence cause yield losses (Swanton, 2016). Weed interference causes delayed soybean maturity, reduced plant height and seed yield, and also affects seed quality (Vollmann et al., 2010). Therefore, efficient quantification and assessment of weeds are necessary for weed management to ensure high yield and quality of these crops.

Cultural practices such as crop rotation, growing cover crops, and optimizing planting date, planting density, and row spacing were followed to reduce weeds in the field (Wicks et al., 1999). To handle weeds during the growing season, chemical methods of weed control can be followed. Chemical methods that involve the use of herbicides serve as the most common and highly effective tool for weed management practices to maintain quality and

quantity of crop production (Dadashzadeh et al., 2020; Pérez-Ortiz et al., 2016; Zheng et al., 2017).

Apart from the problems of excessive use of herbicides, such as blanket application, water contamination, biodiversity reduction, loss in nutrient content, and health problems to the operator and consumers (Dadashzadeh et al., 2020; Pérez-Ortiz et al., 2016; Zheng et al., 2017), it gives rise to different herbicide-resistance weeds (e.g., Palmer amaranth (*Amaranthus palmeri*) and waterhemp (*Amaranthus tuberculatus*)) (Louargant et al., 2018). As per the Weed Science Society of America (WSSA), around 43 varieties of glyphosate-resistant (Roundup) weeds exist all across the globe (Heap, 2019).
The site-specific weed management (SSWM) method is designed to reduce the environmental and economic problems caused by the excessive use of herbicides, wherein the herbicide is applied to only those places where weeds are present by using new technologies to collect and process spatial information of weeds present in the field (Peña et al., 2013).

An alternative to chemical methods is to employ mechanical weeding systems that can be self-propelled or tractor-driven, which is commonly followed in organic farming. Organic farming contributes 4 % to the total US food sales and is the fastest-growing sector in the US (Zimdahl, 2018). Mechanical weeding systems can handle incoming weeds during the season. The automated version of mechanical weeding (robots) decreases labor inputs and uses electronic sensors and global positioning system-based guidance along with complex software and hardware technologies to work in non-row crop fields (Cordill and Grift, 2011; Dadashzadeh et al., 2020). Automated systems have a high initial price and management costs, poor effectiveness on intra-row weeds, and depend on pedoclimatic conditions (especially soil texture and moisture), weed species, and their growth stage (Scavo and

Mauromicale, 2020). In general, mechanical weeding systems cause damage to crops, soil erosion, and nutrient loss.

For successful implementation of either chemical or mechanical weed control methods, accurate identification of weeds is necessary. Therefore, there is a need for an efficient, robust, reliable, and low-cost method that requires less processing and computational time for weed identification.

Nowadays, most weed identification research usually employs digital image processing and computer vision methods (Liu and Bruch, 2020). The most common ways to discriminate weeds are using the visual features that can be further divided into spectral features, biological morphology, visual textures, spatial contexts, and patterns present in the digital image (Wang et al., 2019). The major drawback of using the spectral features for weed classification is that the algorithm is not robust and efficient because of three main reasons (1) loading limit of the device, (2) spectral information changes in an uncontrolled field environment, and (3) physiological stress affects leaf reflectance properties (Louargant et al., 2018). Furthermore, algorithms that use spectral information for classification can underperform if two plants have similar colors (Wang et al., 2019) and are expensive and complex. The spatial context features take into account the location or position of the plant for discrimination. A machine vision method using spatial context was developed to discriminate weeds from a crop, resulting in the identification of inter-row weeds and failure to detect intra-row weeds (Liu et al., 2014).

The biological morphology includes the shape and structure of plants or their parts and has been widely used for weed identification, especially the shape features (Slaughter et al., 2008; Wang et al., 2019). For classifying weeds from crop plants, two machine

learning (ML) classifiers, namely Bayesian classifier (BC) and support vector classifier (SVM), were tested and found that SVM outperformed BC (Ahmed et al., 2011). The SVM was further used to classify crops and weeds using several size-independent shape features, color features, and moments; and accuracy of 97.3 % was reported (Ahmed et al., 2012).
A shape-based weed detection algorithm was developed to identify weeds from sugar beets, and an accuracy of 92.30 ± 2.53 % was reported using SVM on testing dataset (Bakhshipour and Jafari, 2018). The k-Nearest Neighbor (kNN) was used to classify sugar beets and weeds using textural features and achieved an accuracy of around 91 % (Khurana and Bawa, 2021). Most of the reported research was conducted using proprietary software packages like MATLAB because of easier coding experience from the in-built libraries, and hence, limits its use for farmers (DeLay, 2020).

Therefore, the major objective of the study is to use open-source software and develop a simple image processing approach and advanced ML models to classify four common weed species, namely, common lambsquarters (*Chenopodium album*), common purslane (*Portulaca oleracea*), horseweed (*Erigeron canadensis*), and redroot pigweed (*Amaranthus retroflexus*). The simple handcrafted image processing approach that involves shape features extraction and algorithm development, Fiji (ImageJ) image processing system will be employed; and advanced ML model development, Python with scikit-learn will be used. The performance of the methodologies will be assessed through standard performance measures of model development. The methodologies developed are expected to be extended to other weed species as well.

4.3. Materials and Methods

4.3.1. Data Acquisition

In this research, weeds were grown in plastic trays filled with field soil, collected from the field of Northern Great Plains Research Laboratory, USDA-ARS, Mandan, ND, USA (46°48′38.8″N, 100°54′48.1″W), in the natural environment rather than in the greenhouse (fig. 4.1A). The trays were watered once a day. Two weeks after plant emergence, a dose of 'Miracle Grow' solution that had 15-30-15 N-P-K was applied during watering.

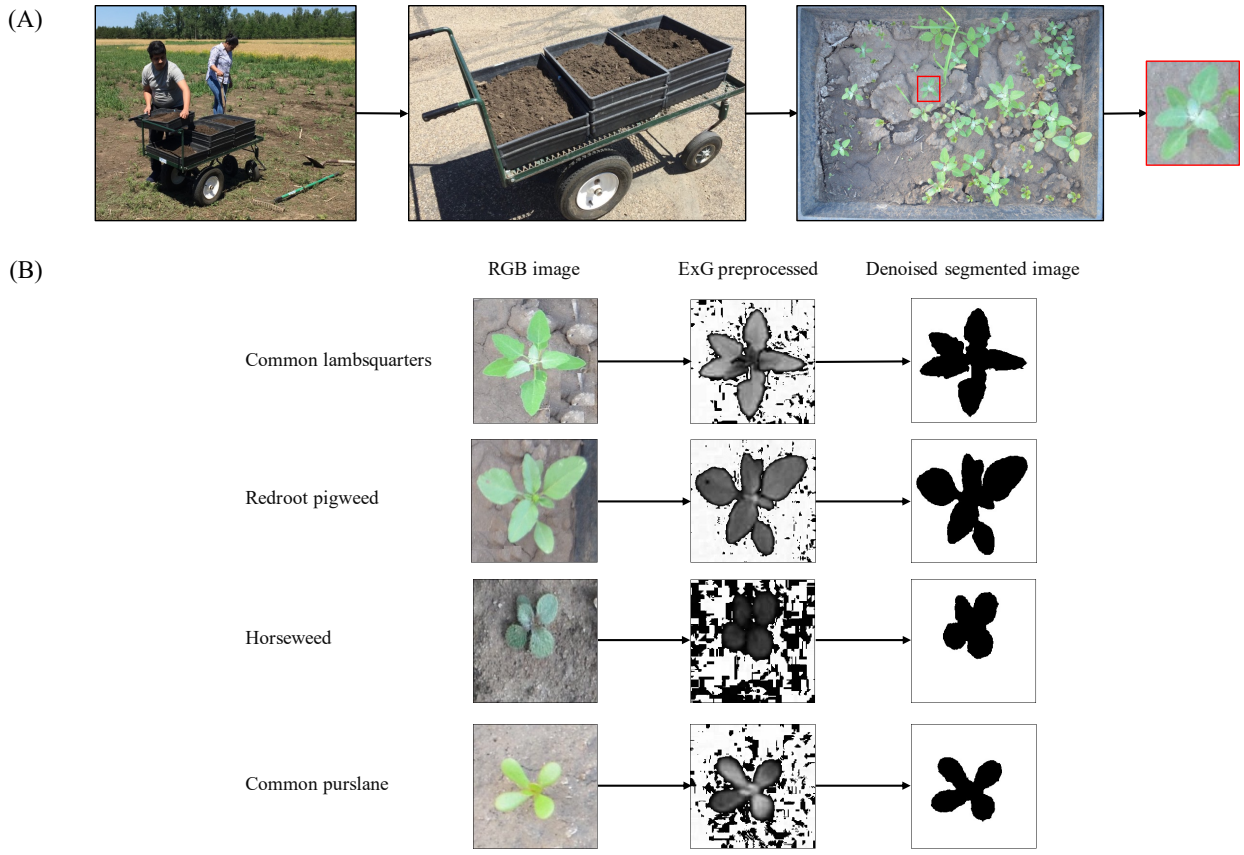


Figure 4.1. Field soil collection for growing weeds naturally in plastic trays and image preprocessing. (A) Manual clipping of individual plant images from the weeds grown in trays, and (B) Image preprocessing of clipped images using excess green (ExG) preprocessing and segmented denoised binary image.

A digital single-lens reflex (DSLR) camera (Model Nikon D5100, Nikon Corporation) was used to capture RGB images of the weeds growing in the tray using the auto mode for data collection. To capture the weed images, the camera was held perpendicular to the soil surface at a distance ranging from 0.5 m to 1.0 m. The collected images were pre-processed by clipping the image so that each frame contained only one seedling (fig. 4.1B). For this clipping of multiple seedlings from the same image, a user-coded Fiji (ImageJ) (Schneider et al., 2012) plugin was developed. Using the plugin, the region of interest (ROI) can be drawn around each seedling and the cropped image will be stored sequentially in a specified folder path.

A dataset of 200 images of each weed species were obtained (total: $4 \times 200 = 800$). The cropped images were resized to 256×256 to have a uniform set of images for further processing. The data augmentation methods such as shear, rotation, horizontal and vertical flips, and zoom were also employed to increase the size of the dataset for feeding into the ML classifiers. The overall process flowchart of distinguishing four weed species using shape features with handcrafted simple direct image processing approach and three different machine learning algorithms showing various stages of development is presented in Figure 4.2.

4.3.2. Image Analysis Plugin Development for Shape Features Analysis

After the data acquisition process, the next step was to create a binary image that could be further processed to extract shape features of the plant to identify different weed species. The process of creating a binary image by dividing it into two classes: plant (weed) and background (soil or residue) is known as segmentation. It is an essential step in image

processing because inappropriate segmentation can lead to misclassification and decreases the accuracy of the algorithm (Hamuda et al., 2016).

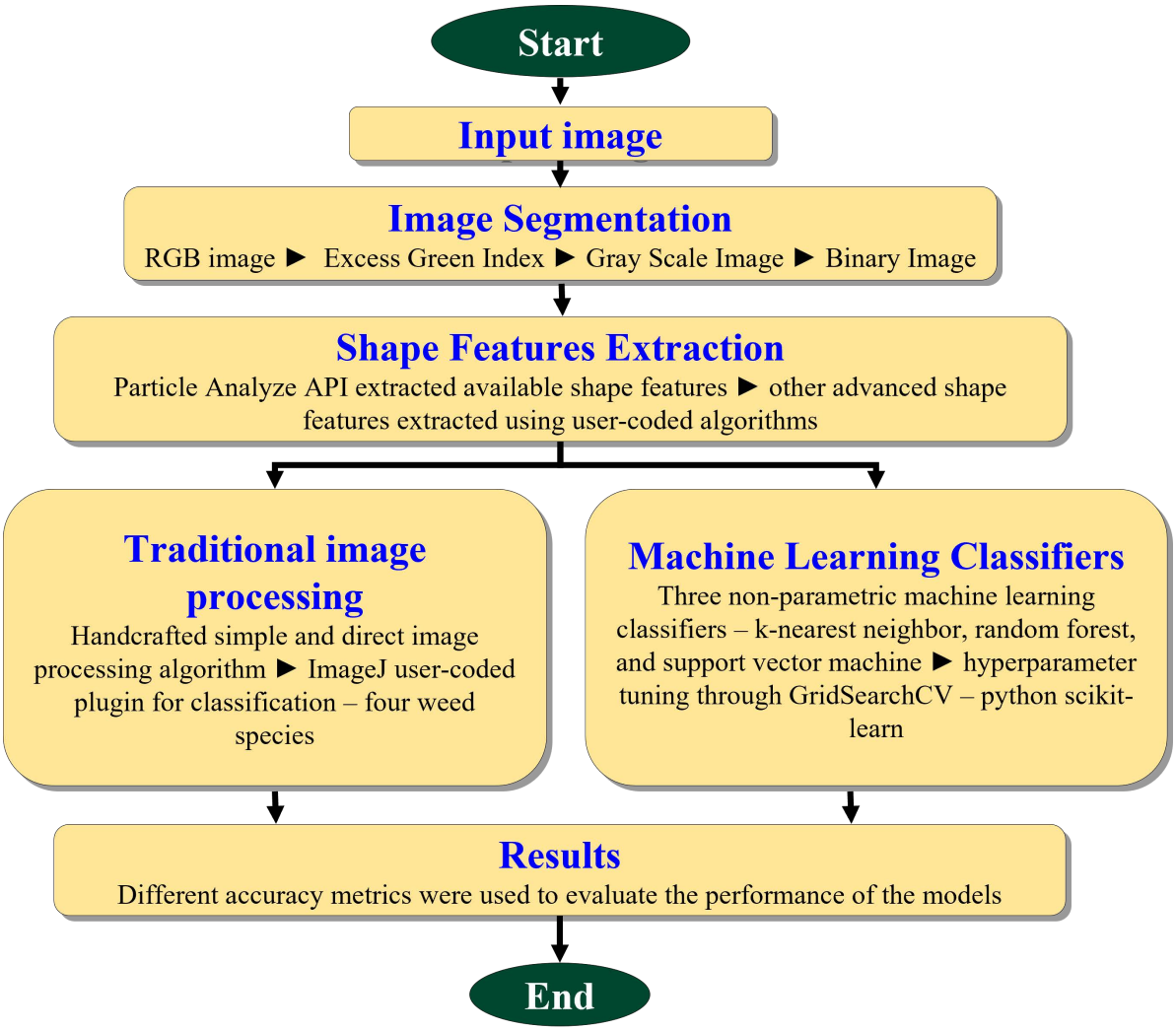


Figure 4.2. Overall flowchart of distinguishing four weed species using shape features with three different machine vision methods.

The excess green segmentation (ExG) method (Woebbecke et al., 1995), a type of color index-based segmenting method was used because of its low sensitivity to background errors and lighting conditions (Basso and de Freitas, 2020; Hamuda et al., 2016).

$$\text{Object pixel}_{(x,y)} = \begin{cases} \text{Plant, value} = 255 & 2G - B - R \geq T_h \\ \text{Background, value} = 0 & \text{Otherwise} \end{cases} \tag{4.1}$$

where, x and y are the coordinates of the image pixels; R, G, and B are the red, green, and blue intensity values of pixels; and T_h is the threshold value.

After segmentation, following the logic (eq. 4.1), the "ParticleAnalyzer" ImageJ command (which is coded in the developed plugin) was used to create a binary mask for the area threshold with a lower limit of 1000 pixel^2 to remove noise and other artifacts.

Several shape features describing the geometric properties of the weed species were extracted from the binary image obtained from segmentation using the ParticleAnalyzer application programming interface (API) (Cho et al., 2002; Lee et al., 1999; Suresh Babu, 2018; Tian et al., 2000). This API provides basic features, such as area, perimeter, Feret major and minor axes, bounding box width and height, and ellipse major and minor axes, along with standard shape features, such as aspect ratio (AR), roundness, circularity, and solidity. Since the basic shape features, such as area, perimeter has dimensions involved with them, their value changes with the resolution of the image and the size of the seedlings due to growth stages (Lin, 2010). Because of these limitations, these parameters cannot be used directly in classification, but can be used to extract other advanced features.

A plugin using Fiji (ImageJ) was developed to derive 21 advanced shape features from the binary images. Out of the 21 features, four were derived directly by the ParticleAnalyzer, while the other advanced shape features, which are non-dimensional, were user-coded. The four standard ImageJ non-dimensional shape features are as follows:

$$\text{Circularity} = \frac{4\pi \times \text{Area}}{\text{Perimeter}^2} \tag{4.2}$$

$$\text{Aspect Ratio (AR)} = \frac{\text{Major axis}}{\text{Minor axis}} \tag{4.3}$$

$$\text{Roundness} = \frac{4 \times \text{Area}}{\pi \times \text{Major axis}^2} \tag{4.4}$$

$$\text{Solidity} = \frac{\text{Area}}{\text{Convex area}} \tag{4.5}$$

and the dimensions of area, convex area, and perimeter of object were in pixel^2, pixel^2, and pixel, respectively. Convex area can be defined as the area of the convex hull (polygon) that wraps the object. The dimension of major and minor axes were in pixel and corresponded to the orthogonal axes of the fitted ellipse, whose area was equal to the area of the object. The other 17 advanced shape features that were user-coded are:

$$\text{Convex area} = \frac{\text{Area}}{\text{Solidity}} \tag{4.6}$$

$$\text{Hollowness} = \frac{\text{Convex area - Area}}{\text{Convex area}} \tag{4.7}$$

$$\text{Reverse aspect ratio (RAR)} = \frac{1}{\text{Aspect ratio}} \tag{4.8}$$

$$\text{Rectangularity} = \frac{\text{Area}}{\text{Bounding rectangle area}} \tag{4.9}$$

$$\text{Feret major axis ratio (FMA)} = \frac{\text{Feret diameter}}{\text{Major axis}} \tag{4.10}$$

$$\text{Convex area Feret ratio (CAF)} = \frac{\text{Convex area}}{\text{Feret diameter}^2} \tag{4.11}$$

$$\text{Compactness} = \frac{\text{Area}}{\text{Feret diameter}} \tag{4.12}$$

$$\text{Ratio of area to length (RAL)} = \frac{\text{Area}}{\text{Major axis}^2} \tag{4.13}$$

$$\text{Log of ratio of height to width (LHW)} = \log\left(\frac{\text{Height}}{\text{Width}}\right) \tag{4.14}$$

$$\text{Elongation} = \frac{\text{Major axis - Minor axis}}{\text{Major axis + Minor axis}} \tag{4.15}$$

$$\text{Ratio of perimeter to broadness (RPB)} = \frac{\text{Perimeter}}{2 \times \text{Width} + \text{Height}} \tag{4.16}$$

$$\text{Ratio of length to perimeter (LTP)} = \frac{\text{Major axis}}{\text{Perimeter}} \tag{4.17}$$

$$\text{Modified circularity (MC)} = \frac{4 \times \text{Area}}{\text{Perimeter} \times \text{Feret diameter}} \tag{4.18}$$

$$\text{Grum circularity (GC)} = \frac{16 \times \text{Area}^2}{4\pi \times \text{Perimeter} \times \text{Feret Diameter}^3} \tag{4.19}$$

$$\text{Area compactness (AC)} = \frac{4 \times \text{Area}}{\pi \times \text{Feret diameter}^2} \tag{4.20}$$

The bounding rectangle area can be defined as the area of the smallest enclosing rectangle of the object and its dimension is pixel^2, while the dimension of Feret diameter is pixel and can be defined as the maximum diameter of the particle. It is to be noted that all other shape features are dimensionless, and the equation of “Convex area” is an intermediate parameter with pixel^2 as unit.

$$\text{Equivalent perimeter ratio (EqP)} = \frac{\text{Perimeter}}{\text{Equivalent perimeter}} \tag{4.21}$$

where equivalent perimeter can be further calculated as

$$\text{Equivalent perimeter} = 2.0 \times \sqrt{\pi \times \text{Area}} \tag{4.22}$$

For employing handcrafted simple and direct image processing approach, values of all the shape features were first plotted for visualization of the variation of the shape parameters

of all weed species using ggplot (Wickham, 2016) library of R software. The graphs were further analyzed to find which shape features could help to discriminate the weeds. In the analysis, the shape features which have very limited overlap, visualized through density distribution plotting, between the values of different weed species will be used for discriminating the weed species. After the analysis, the threshold values for that particular shape features were used to create a plugin (handcrafted model) to classify/identify the weed species.

4.3.3. Machine Learning Methods

The 21 extracted shape features were further fed into the three non-parametric ML classifiers namely k-nearest neighbor (kNN), random forest (RF), and support vector machine (SVM) as these methods were found to produce high classification accuracies in agricultural applications such as weed classification (Ahmad et al., 2011; Rojas et al., 2017), disease detection and classification (Samajpati and Degadwala, 2016; Suresha et al., 2017), and several more applications. These models are described subsequently.

4.3.3.1. k-nearest neighbor classifier

The kNN is the simplest, most popular supervised ML method used for performing classification (Akbulut et al., 2017). This algorithm is also known as a lazy or instance-based learning method because it does not have specialized learning involved other than keeping track of the labeled data. The theory behind kNN is that the algorithm finds K samples based on the distance values that are nearest to the new test samples. The label of each test sample is determined by the majority of votes of its K nearest neighbors (Thanh Noi and Kappas, 2018). Therefore, K is the main hyperparameter in this classifier and needs to be tuned as it directly affects the model's performance.

Therefore, the GridSearchCV (GSCV) API of sci-kit-learn library was used (Pedregosa et al., 2011) for selecting the optimal value of K for our dataset. The GSCV provides the best combination of parameters from the list supplied in the parameter grid, based on a scoring metric. A list of K values was provided that ranged from 2 to 30. The choice for p-value was given as 1 or 2, where 1 means Manhattan distance and 2 means Euclidean distance. The resulting combination of parameters was used to train the model for classifying the four weed species.

4.3.3.2. Support vector machine (SVM) classifier

The SVM is a supervised ML algorithm and has been widely used to address classification problems in agriculture (Bakhshipour and Jafari, 2018; Lin, 2010). SVM finds a hyperplane in N-dimensional space that can classify distinct classes rather than finding the nearest neighbors. The hyperplane maximizes the margins and minimizes the generalization errors (Lin, 2010). A key parameter in the SVM is the kernel function that transforms the training set of data into a higher dimension space so that a non-linear decision surface could be transformed into a linear surface to facilitate distinction of data (Zhang et al., 2020c). Apart from the kernel functions, values of gamma (γ) and regularization (C) directly influence the model's accuracy. An exponentially growing sequence of γ and C values is a common approach to find good parameter values (Hsu et al., 2003). Therefore, finding the optimal values for parameters such as kernel type, γ, and C is necessary. Since there were four weed species, the one versus rest feature of the sci-kit-learn SVM class was used (Pedregosa et al., 2011).

To find the optimal values of the kernel, γ, C, and GSCV were used. Three different types of kernels, namely "linear," "polynomial," and "Gaussian" kernel, were provided along

with the list of exponential growing sequence (2^k), where k ranges from $[-4, 4]$. The resulting parameters were used for training the model to further classify the four weed species.

4.3.3.3. Random forest (RF) classifier

The RF (Breiman, 2001), also known as an ensemble learning algorithm, is a supervised learning model that comprises a large number of decision trees that operate together because a single decision tree might have more chance of overfitting (Flores et al., 2021). This algorithm has been widely used in agriculture for different purposes, such as disease detection and classification (Samajpati and Degadwala, 2016), land cover classification (Rodriguez-Galiano et al., 2012), weed classification (De Castro et al., 2018), and many more because it can handle high dimensionality data, has a fast operation speed, is insensitive to overfitting, and provides good accuracy (Belgiu and Drăguţ, 2016). The basic theory behind random forest is that each tree in the random forest provides some outcome for the test data, and the majority vote of all the trees is the label for the test data.
The accuracy of the model depends on the number of trees (Flores et al., 2021). Therefore, finding the optimal value of the number of trees is necessary. Furthermore, the number of features (max_features) in each split needs to be investigated (Thanh Noi and Kappas, 2018).

To obtain the optimal/best set of combinations, GSCV was used. The parameters for the number of trees was a list of integers starting from 10 to 200, while for the number of features, two different types, namely ‘sqrt’ and ‘log2,’ were provided. The parameters provided by the GSCV were used to train the model for performing classification tasks.

4.3.3.4. Machine learning model evaluation

For the ML model evaluation, the dataset was split into two groups, the first was the training group that comprised 80 % of the data and the second the testing group that contained the remaining 20 % of the data. Before feeding the data into each ML model, feature scaling was performed using scikit-learn StandardScaler API (Pedregosa et al., 2011). The performance of the model was evaluated by comparing the predicted labels with the actual labels. Each classification model was evaluated using testing accuracy, precision, recall, and F1-score. The formulas for calculating these metrics are as follows:

$$\text{Testing accuracy} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN} + \text{FP}} \tag{4.23}$$

$$\text{Precision (PRE)} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.24}$$

$$\text{Recall (REC)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.25}$$

$$\text{F1-score} = 2 \times \frac{\text{PRE} \times \text{REC}}{\text{PRE} + \text{REC}} \tag{4.26}$$

where TP stands for true positive, FP stands for false positive, TN stands for true negative, and FN stands for false negative.

As an illustration, referring to Figure 4.3, if positive class meant common lambsquarters and negative class meant some other weed species, TP can be defined as an outcome where the model predicted the correct label of positive class (for example, common lambsquarters identified as common lambsquarters). TN is an outcome where the model correctly predicted the incorrect label of negative class (for example, model identified the other class weed as another class, not common lambsquarters). FN can be defined as an

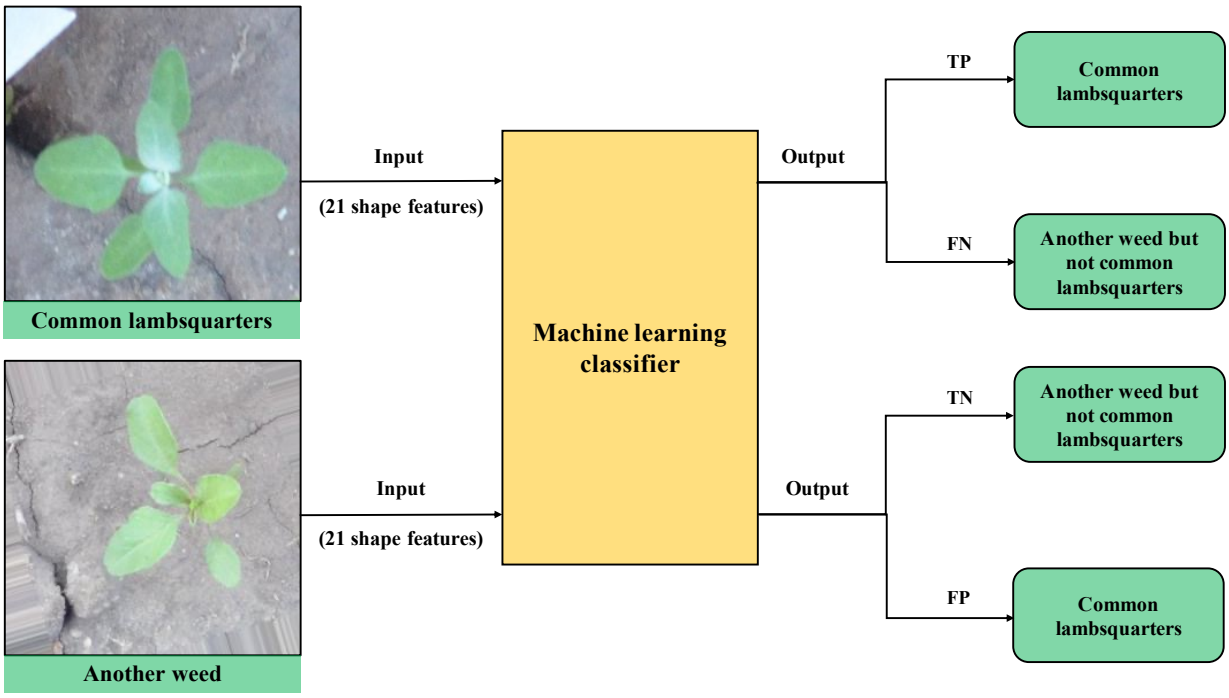


Figure 4.3. Classification process involved in the machine learning models depicting four different outcomes using a target and other weed species: True positive (TP), false negative (FN), true negative (TN), and false positive (FP).

outcome where the model predicted the wrong label for the positive class (for example, common lambsquarters identified as some other weed). FP can be defined as the outcome where the model predicted the wrong label for the negative class (for example, some other weed species identified as common lambsquarters). Testing accuracy can be defined as the number of datasets correctly identified or classified by the model.

4.4. Results and Discussion

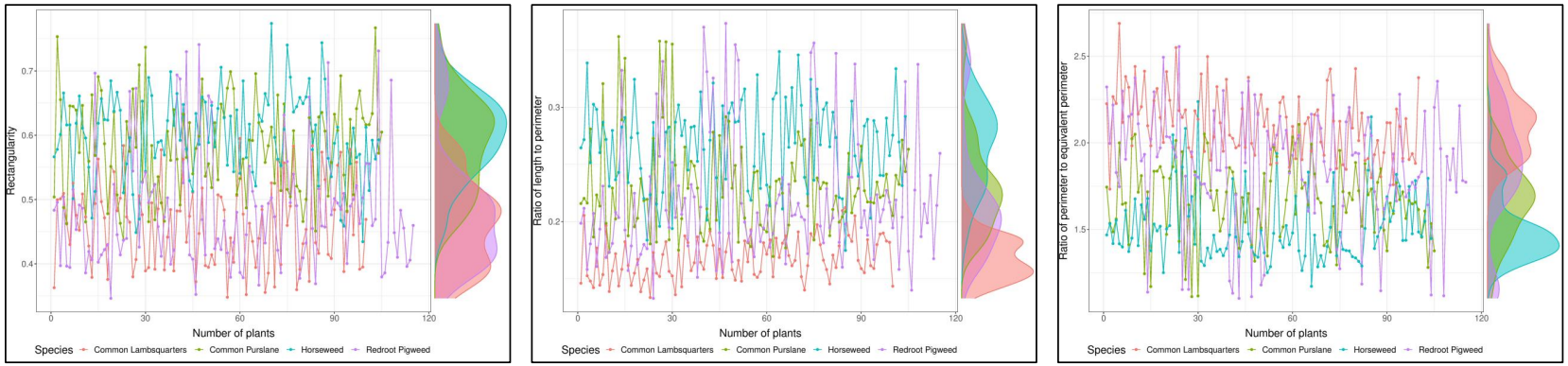
4.4.1. Weed Identification with Simple Image Processing

Geometrical shape features have been widely used for classifying objects present in agricultural imagery. Roundness and area features were used to classify corn and weeds in field imagery (Shrestha and Steward, 2005). Shape features were used to identify and count soybean aphids using digital images (Shajahan et al., 2017). For the simple image processing,

100 images of each species were collected, and features were extracted and plotted on graphs (fig. 4.4) using R-scripts and all the tested geometrical feature data and overlap bar in Table 4.1.

The FMA and EqP were the top features that could be used to classify common lambsquarters from horseweed, common purslane, and redroot pigweed. Hollowness can be used to classify horseweed from all the other three weed species. To classify common purslane from all other weed species, hollowness and rectangularity can be used. Furthermore, rectangularity and LTP could be used to classify redroot pigweed from horseweed, common purslane, and common lambsquarters.

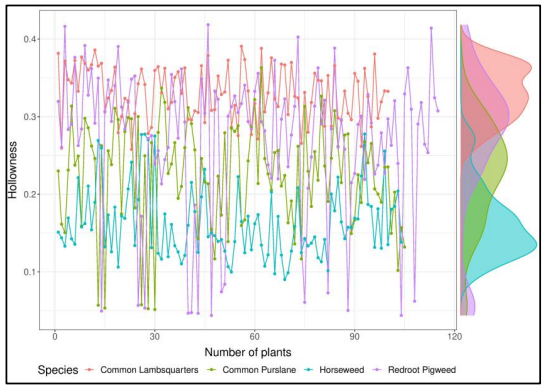
Twenty images of each weed species were selected to evaluate the performance of the developed model. The algorithm correctly identified 100 % of common lambsquarters and 95 % of horseweed. While the algorithm insufficiently identified common purslane and redroot pigweed correctly. In the analysis, 8 out of 20 common purslane were correctly identified, while only 5 out of 20 redroot pigweed were correctly identified, resulting in the accuracy of 40 % and 25 %, respectively. The overall accuracy of the algorithm was 65 %. These classification accuracies are quite similar to that found by Søgaard (2005) in classifying three weed species using active shape models. It is to be noted when classifying weeds using shape-features, the accuracy of the model incorrectly classifying the weeds decreases if the resemblance between them is higher (Søgaard, 2005). In this present research, 13 out of 20 redroot pigweed were identified as common lambsquarters, while 9 out of 20 common purslane were identified as horseweed because of the shape resemblance as a whole plant although they all have different leaf structures.



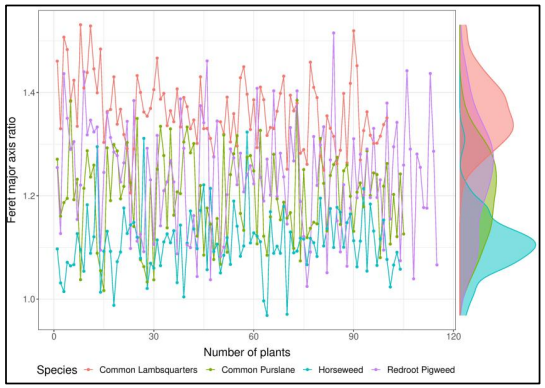
(A)

(B)

(C)



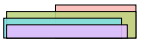
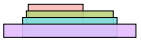
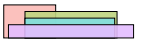
(D)



(E)

Figure 4.4. Variations of five best shape features and their density distribution visualization that can be used to classify the four common weed species using direct image processing. (A) Rectangularity, (B) Ratio of length to perimeter, (C) Ratio of perimeter to equivalent perimeter ratio, (D) Hollowness, and (E) Feret major axis ratio.

Table 4.1. Extracted shape features value ranges, and overlap of values for the four common weed species of North Dakota (training set; $n = 100$).

Features	Range of features				Overlap bars
	CL	CP	HW	RP	
Circularity	0.14–0.33	0.23–0.81	0.20–0.73	0.15–0.82	
Aspect ratio	1.03–1.62	1.02–2.43	1.08–2.51	1.06–2.44	
Roundness	0.62–0.97	0.41–0.98	0.40–0.92	0.41–0.95	
Solidity	0.61–0.74	0.64–0.95	0.72–0.91	0.58–0.96	
Reverse aspect ratio	0.62–0.97	0.41–0.98	0.40–0.92	0.41–0.95	
Compactness	0.18–0.42	0.29–1.03	0.25–0.93	0.20–1.05	
Ratio of area to length	0.49–0.77	0.32–0.77	0.31–0.72	0.32–0.74	
Elongation	0.01–0.24	0.01–0.42	0.04–0.43	0.03–0.42	
Ratio of perimeter to broadness	1.02–1.72	0.82–1.45	0.82–1.44	0.82–1.50	
Rectangularity	0.35–0.60	0.44–0.77	0.43–0.77	0.35–0.74	
Log of ratio of height to width	−0.15–0.14	−0.16–0.30	−0.18–0.32	−0.28–0.42	
Ratio of length to perimeter	0.13–0.21	0.17–0.36	0.17–0.35	0.13–0.37	
Equivalent perimeter ratio	1.73–2.69	1.11–2.11	1.17–2.24	1.10–2.56	
Hollowness	0.26–0.39	0.05–0.36	0.09–0.28	0.04–0.42	
Feret major axis ratio	1.21–1.53	1.02–1.38	0.97–1.32	1.02–1.52	
Feret minor axis ratio	1.31–2.11	1.21–2.56	1.27–2.64	1.33–2.98	
Convex area feret ratio	0.42–0.64	0.35–0.67	0.35–0.65	0.28–0.63	
Modified circularity	0.22–0.44	0.31–0.70	0.31–0.69	0.24–0.77	
Grum circularity	0.08–0.26	0.11-0.43	0.12–0.50	0.07–0.56	
Area compactness	0.34–0.60	0.34-0.70	0.36–0.73	0.28–0.72	

Note: CL - common lambsquarters: ■, CP - common purslane: ■, HW - horsweed: ■, and RP - redroot pigweed: ■

4.4.2. Performance of Machine Learning Models

The extracted shape features, that were used in the direct image processing, can also be used to train the ML classifiers, such as k-NN, RF, and SVM. All the classifiers were trained on the same data. The details of each model are described below:

4.4.2.1. k-nearest neighbor classifier

The smaller value of k, in kNN, will affect the result due to noise, while a larger value of k increases the computational load and makes the classification boundaries less distinct

(Flores et al., 2021). In the list of 30 values ranging from 2 to 30, the GSCV provided a k value of 3, while the p-value was 1 (Manhattan distance) using the three-fold cross-validation. The cross-validation approach reduces the chance of overfitting (Ahmed et al., 2012; Hsu et al., 2003). By using the k-value of 3 and using Manhattan distance, the testing accuracy of the model was 87.70 %. The means for precision, recall, and F1-score were 0.88. While the ten-fold cross-validation accuracy of the model was 88.86 %.

4.4.2.2. Support vector machine (SVM) classifier

In SVM, the γ (gamma) parameter defines how far the effect of a single training point influences the model. With low values of γ, the model cannot capture the actual shape of the data, but it is computationally efficient. While, with higher values of γ, it might be restricted to itself (Ashraf and Khan, 2020). Another parameter of influence in SVM is C; with a larger value of C, the model is overfitted and fails to work efficiently with the new dataset, while there exists a trade-off with a lower value of C. The value of kernel type, γ, and regularization or error bound parameter C were determined through three-fold cross-validation by GSCV. The optimal parameter values were kernel type as Gaussian, γ as $2^3(8)$, and C as $2^2(4)$. By using the kernel type as Gaussian and optimal γ and C values, the ten-fold cross-validation accuracy was 88.52 %. While the testing accuracy of the model was 89.44 %. Furthermore, the means for precision, recall, and F1-score were 0.90.

4.4.2.3. Random forest classifier

In RF, larger numbers of trees can provide more stable estimates of variable importance and results, and the bootstrap provides better generalization (Hassanijalilian et al., 2020; Liaw et al., 2002). The GSCV with the three-fold cross-validation provided values for the number of trees and max_features as 115 and square root, respectively.

The bootstrap setting was set to 'true' to increase the model performance on an unseen dataset. By using the above parameter settings the mean accuracy of the model using ten-fold cross-validation was 90.11 %. The testing accuracy of the RF model was 90.72 % and the means for precision, recall, and F1-score were 0.91.

4.4.2.4. Performance evaluation of ML models

Classification performance results of the three non-parametric ML models (kNN, RF, and SVM) in classifying four weed species are presented in Table 4.2. All three ML models had comparable performance, with both average precision and recall in the range of 0.88 to 0.91. However, the RF classifier was slightly better than SVM and kNN (table 4.2).

Table 4.2. Performance of the three machine learning models on four different common weed species of North Dakota.

Model	Class	Precision	Recall	Accuracy
k-nearest neighbor	1	0.83	0.89	0.89
	2	0.87	0.88	0.87
	3	0.92	0.93	0.92
	4	0.88	0.82	0.82
	Average	0.88	0.88	0.88
Random Forest	1	0.90	0.93	0.92
	2	0.91	0.90	0.89
	3	0.91	0.93	0.93
	4	0.91	0.88	0.88
	Average	0.91	0.91	0.90
Support vector machine (Gaussian kernal)	1	0.92	0.90	0.90
	2	0.86	0.92	0.92
	3	0.94	0.90	0.90
	4	0.87	0.86	0.86
	Average	0.90	0.90	0.90

Note: Classes 1, 2, 3, and 4 refers to common lambsquarters, common purslane, horseweed, and redroot pigweed, respectively. Refer equations for precision (eq. 4.23), and recall (eq. 4.24).

The performance of the ML classifiers can also be easily compared using the F1-score, which also indicates the robustness and accuracy of the classifier. From the plotted bar graph (fig. 4.5), the F1-score comparison indicates clearly that RF was the best classifier followed by SVM and kNN. However, the RF (f1: 0.90–0.92) and SVM (f1: 0.89–0.92) had the most similar performance, but the kNN is not far behind (f1: 0.85–0.92).

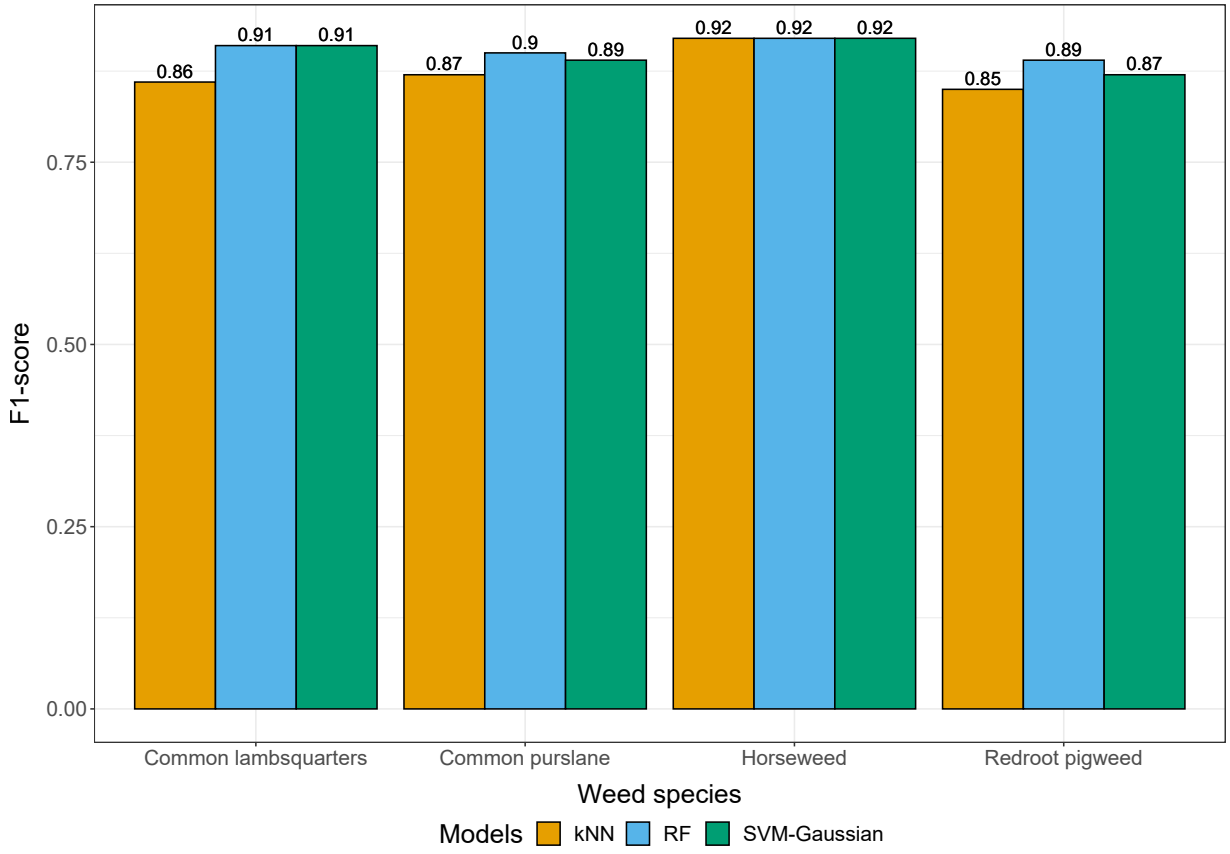


Figure 4.5. Comparison of machine learning models based on F1-score. kNN - k-nearest neighbor, RF - random forest, and SVM - support vector machine.

In the present study, out of the different ML models, RF performed best, followed by SVM and kNN. As observed in the present study, Ashraf and Khan (2020) found that RF outperformed SVM in classifying images based on grassy weed densities into three categories. While detecting the weeds in the chilli field using the unmanned aerial vehicle (UAV) imagery,

it was found that RF performed the best and kNN performed the worst (Islam et al., 2021). The RF has high-performance accuracy because the RF classifiers are trained with different sub-samples and subsets of datasets resulting in high generalization and performance on an unseen dataset (Hassanijalilian et al., 2020). While classifying volunteer barley in a field of oilseed rape, SVM performed better than kNN because SVM has complex function classes such as radial functions that can help construct non-linear separation functions (Weis et al., 2009).

Overall, the ML models have performed better than the handcrafted direct simple image processing algorithms. The handcrafted image processing algorithms were accurate in identifying two out of four species tested (horseweed and common lambsquarters: 95 %–100 %). While this direct approach was not efficient, with the methodology of shape factors tested, they were simple and direct and could be improved with more samples and other algorithms of shape and texture identification. It should also be noted that all the shape factors derived are also served as intermediate inputs to the advanced ML models.
In addition, ML models are more complex than the handcrafted models — that are simpler and do not require rigorous training and testing cycle with cross-validation as followed with ML. It is recommended that if the classification problem supports the use of handcrafted simple image processing algorithms, it is better to resort to them, otherwise, the versatile ML modeling approach should be followed.

4.4.3. Research Limitations and Future Research Directions

In this research, images of four commonly found weed species of North Dakota were used to train the ML models and develop a handcrafted simple image processing model for weed classification, while the deep learning model was avoided due to the limited amount of dataset. In the future, more weed species should be considered, and a huge weed imagery

dataset should be developed and tested with ML and deep learning models. As the weeds were grown in plastic trays in mineral soil under the field condition, the methodology is expected to be applicable in the field, especially during the initial stages; however, field testing should be performed for it is expected to present several challenges (e.g., image background variation, other weeds present, lack of dimensions reference). It will be interesting to test the current methodology with field data to check the performance of ML and handcrafted simple image processing models. Furthermore, these shape features can be used to distinguish crops from the weeds and can help in generating a weed distribution map, which can reduce the use of herbicides and this aspect should be explored.

4.5. Conclusion

The developed methodologies that used shape features to classify four different weed species (common lambsquarters, common purslane, horseweed, and redroot pigweed) commonly found in the corn fields of North Dakota using handcrafted simple image processing algorithms and three non-parametric machine learning (ML) models (kNN, SVM, and RF) were successful. The handcrafted simple image processing approach was highly successful with a few weed species in distinguishing each other (common lambsquarters *vs* horseweed; accuracy $\geq$ 95 %), while did not perform well with other species combinations (common lambsquarters *vs* redroot pigweed, and horseweed *vs* common purslane; accuracy 25 %–40 %) and classified them collectively owing to the similarity of the shapes. All the ML models, however, resulted in high precision, recall, F1-score, and testing accuracies in classifying all the weed species considered. Out of the different ML models, RF performed best (f1: 0.90–0.92), followed by SVM (f1: 0.89–0.92), and kNN (f1: 0.85–0.92).

It is recommended that the handcrafted simple image processing algorithms should be tried first for identification and classification studies before resorting to the advanced and complex versatile ML modeling approaches. Future research should include testing these models in field conditions, increasing the size of the weed imagery dataset, and employing and comparing the deep learning models along with other ML models. Using the shape features in conjunction with the image processing and ML models that can help prepare weed distribution maps should also be explored.

5. GENERAL CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

5.1. General Conclusions

Corn is grown in all parts of the United States, and its yield is closely related to its stand count and spatial distribution of plants. In addition, weeds also reduce the crop yield by competing with the crop for nutrients, sunlight, and water. The traditional method of scouting the field to evaluate plant stand count, know the spatial distribution of corn plants, or identify weeds in the field is laborious, time-consuming, and prone to human errors. However, digital image processing can be used to evaluate plant stand count, generate spatial distribution maps, and identify weeds. Several commercially available software exist that can be used for evaluating plant stand count or for identifying weeds in the field, but they all require high subscription fees and take a long processing time to process the field data. Therefore, in this study, open-source software, namely ImageJ, R, and Python, were used to develop the algorithms to evaluate plant stand count, produce spatial distribution maps, and classify four common weed species of North Dakota.

- Objective 1 (*Review of UAV-based method for evaluating plant stand count*): A detailed literature analysis was performed to provide well-synthesized information about the methods for evaluating plant stand count using UAV imagery. The total number of journal articles published related to the subject was identified using different combinations of keywords on three academic databases, namely Agricola, Scopus, and

Web of Science, and were reviewed to identify research gaps and provide some recommendations. This review will be helpful to farmers, producers, and researchers to select and employ UAV-based algorithms for evaluating plant stand count.

- Objective 2 (*Plant stand count evaluation and spatial distribution using UAV imagery*) To count the number of corn plants in the field and produce spatial distribution maps, a user-coded plugin was developed using ImageJ, an open-source image processing platform. Of the different image segmentation methods tested, the excess green (ExG) segmentation method was found suitable, while several others are also good enough for segmenting corn plants at an early stage. The intensity profile of crop rows with appropriate cut-off value was efficient in identifying the crop rows. The novel sliding-and-shifting region-of-interest (SSROI) method developed to count the number of plants along the rows, even off-linear plants, and cover the entire field is found efficient. The overall counting accuracy of the algorithm was 97.71 %, and the accuracy depends on the segmentation and stage of plant growth. The spatial distribution map generated by the plugin was able to visualize five classes, namely doubles, single skip, double skip, ideal, and multiple skips, with different colors. It was found that the plugin can efficiently identify crop rows even when the flying height of the UAV was 30.5 m or more as long as good contrast, and enough pixels define a plant but failed to count plants due to overlap between the plants. Future research should include developing a plugin that can count plants even at the later stages of their growth when there is overlap between them, rotational ROI to accommodate edge curvilinear rows.

- Objective 3 (*Weed classification using image processing and ML models*) To classify the four common weeds of North Dakota, namely, common lambs-quarters, common purslane, horseweed, and redroot pigweed, hand-crafted simple image processing model and non-parametric ML models (kNN, SVM, and RF) were used employing extracted 21 shape features. The handcrafted simple image processing approach was highly successful only with a few weed species in distinguishing each other (common lambsquarters vs horse-weed; accuracy $\geq 95\,\%$). All the ML models, however, resulted in high precision, recall, f1-score, and testing accuracies in classifying all the weed species considered. Out of the different ML models, RF performed best (f1: 0.90–0.92), followed by SVM (f1: 0.89–0.92), and kNN (f1: 0.85–0.92). It is recommended that the handcrafted simple image processing algorithms should be tried first for identification and classification studies before resorting to the advanced and complex versatile ML modeling approaches. Future research should include testing these models in field conditions, increasing the size of the weed imagery dataset, and employing and comparing the deep learning models along with other ML models.

5.2. Suggestions for Future Work

The future research works suggested here pertain to possible work that can be extended from each of the objectives considered in this thesis.

1. Use of various shape parameters in identification and performing the plant stand count rather than default individual objects should be explored.

2. Develop a simple image processing plant counting algorithm to differentiate plants from weeds.

3. Develop a plugin to evaluate plant population and generate spatial distribution maps for other row crops.

4. Create yield prediction model taking into consideration of plant stand counts and other factors such as plant height using simple and advanced machine learning approaches.

5. Explore other geometric parameters and other algorithms for handcrafted image processing methods for identifying weeds.

6. More weed species data and field-level tests should be performed for evaluating the performance of the weed classification algorithms.

7. Application of convolutional neural network and deep learning method for weed identification and detection should be explored.

REFERENCES

Abuzar, M., Sadozai, G., Baloch, M., Baloch, A., Shah, I., Javaid, T., Hussain, N., et al. (2011). Effect of plant population densities on yield of maize. *J. Anim. & Plant Sci.*, 21(4), 692–695.

Ahmad, I., Siddiqi, M. H., Fatima, I., Lee, S., & Lee, Y.-K. (2011). Weed classification based on haar wavelet transform via k-nearest neighbor (k-NN) for real-time automatic sprayer control system. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication* (pp. 1–6).

Ahmed, F., Al-Mamun, H. A., Bari, A. H., Hossain, E., & Kwan, P. (2012). Classification of crops and weeds from digital images: A support vector machine approach. *Crop Prot.*, 40, 98–104.

Ahmed, F., Bari, A., Hossain, E., Al-Mamun, H. A., & Kwan, P. (2011). Performance analysis of support vector machine and bayesian classifier for crop and weed classification from digital images. *World Appl. Sci. J.*, 12(4), 432–440.

Akbulut, Y., Sengur, A., Guo, Y., & Smarandache, F. (2017). NS-k-NN: Neutrosophic set-based k-nearest neighbors classifier. *Symmetry*, 9(9), 179.

Al-Kaisi, M. & Hanna, H. M. (2006). Field soil variability and its impact on crop stand uniformity. Accessed: July 1, 2021.

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)* (pp. 1–6).: IEEE.

Alsalam, B. H. Y., Morton, K., Campbell, D., & Gonzalez, F. (2017). Autonomous UAV with vision based on-board decision making for remote sensing and precision agriculture. In *2017 IEEE Aerospace Conference* (pp. 1–12).: IEEE.

Ampatzidis, Y. & Partel, V. (2019). UAV-based high throughput phenotyping in citrus utilizing multispectral imaging and artificial intelligence. *Remote Sens.*, 11(4), 410.

Ampatzidis, Y., Partel, V., & Costa, L. (2020). Agroview: Cloud-based application to process, analyze and visualize UAV-collected data for precision agriculture applications utilizing artificial intelligence. *Comput. Electron. Agric.*, 174, 105457.

Ashraf, T. & Khan, Y. N. (2020). Weed density classification in rice crop using computer vision. *Comput. Electron. Agric.*, 175, 105590.

Åstrand, B. & Baerveldt, A.-J. (2005). A vision based row-following system for agricultural field machinery. *Mechatronics*, 15(2), 251–269.

Bakhshipour, A. & Jafari, A. (2018). Evaluation of support vector machine and artificial neural networks in weed detection using shape features. *Comput. Electron. Agric.*, 145, 153–160.

Basso, M. & de Freitas, E. P. (2020). A UAV guidance system using crop row detection and line follower algorithms. *J. Intell. & Rob. Syst.*, 97(3), 605–621.

Bayraktar, E., Basarkan, M. E., & Celebi, N. (2020). A low-cost UAV framework towards ornamental plant detection and counting in the wild. *ISPRS J. Photogramm. Remote Sens.*, 167, 1–11.

Belgiu, M. & Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions. *ISPRS J. Photogramm. Remote Sens.*, 114, 24–31.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5–32.

Bryant, R., Moran, M. S., McElroy, S. A., Holifield, C., Thome, K. J., Miura, T., & Biggar, S. F. (2003). Data continuity of Earth Observing 1 (EO-1) Advanced Land I satellite imager (ALI) and Landsat TM and ETM+. *IEEE Trans. Geosci. Remote Sens.*, 41(6), 1204–1214.

Bullock, D. G., Bullock, D. S., Nafziger, E. D., Doerge, T. A., Paszkiewicz, S. R., Carter, P. R., & Peterson, T. A. (1998). Does variable rate seeding of corn pay? *Agron. J.*, 90(6), 830–836.

Burgos-Artizzu, X. P., Ribeiro, A., Guijarro, M., & Pajares, G. (2011). Real-time image processing for crop/weed discrimination in maize fields. *Comput. Electron. Agric.*, 75(2), 337–346.

Carlier, J., Davis, E., Ruas, S., Byrne, D., Caffrey, J. M., Coughlan, N. E., Dick, J. T., & Lucy, F. E. (2020). Using open-source software and digital imagery to efficiently and objectively quantify cover density of an invasive alien plant species. *J. Environ. Manage.*, 266, 110519.

Carter, P. R. & Nafziger, E. D. (1990). Uneven emergence in corn. Accessed: July 1, 2021.

Chen, R., Chu, T., Landivar, J. A., Yang, C., & Maeda, M. M. (2018). Monitoring cotton (*Gossypium hirsutum* L.) germination using ultrahigh-resolution UAS images. *Precis. Agric.*, 19(1), 161–177.

Chen, S. W., Shivakumar, S. S., Dcunha, S., Das, J., Okon, E., Qu, C., Taylor, C. J., & Kumar, V. (2017). Counting apples and oranges with deep learning: A data-driven approach. *IEEE Rob. Autom. Lett.*, 2(2), 781–788.

Cho, S., Lee, D., & Jeong, J. (2002). Ae—automation and emerging technologies: Weed–plant discrimination by machine vision and artificial neural network. *Biosyst. Eng.*, 83(3), 275–280.

Conley, S. & Robinson, A. (2007). *Thin soybean stands: Should I replant, fill in, or leave them alone. Coop. Ext. Serv*. Technical report, SPS-104-W. Purdue Univ., West Lafayette, IN.

Cordill, C. & Grift, T. (2011). Design and testing of an intra-row mechanical weeding machine for corn. *Biosyst. Eng.*, 110(3), 247–252.

da Silva, E. A. & Mendonça, G. V. (2005). Digital image processing. In W.-K. CHEN (Ed.), *The Electrical Engineering Handbook* (pp. 891–910). Burlington: Academic Press.

Dadashzadeh, M., Abbaspour-Gilandeh, Y., Mesri-Gundoshmian, T., Sabzi, S., Hernández-Hernández, J. L., Hernández-Hernández, M., & Arribas, J. I. (2020). Weed classification for site-specific weed management using an automated stereo computer-vision machine-learning system in rice fields. *Plants*, 9(5), 559.

De Castro, A. I., 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 Sens.*, 10(2), 285.

DeLay, N. (January 23, 2020). Farm data usage in commercial agriculture. `https://ag.purdue.edu/commercialag/home/resource/2020/01/farm-data-usage-in-commercial-agriculture/`. Accessed: July 1, 2021.

Duncan, W. (1958). The relationship between corn population and yield. *Agron. J.*, 50(2), 82–84.

Feng, A., Zhou, J., Vories, E., & Sudduth, K. A. (2020a). Evaluation of cotton emergence using UAV-based imagery and deep learning. *Comput. Electron. Agric.*, 177, 105711.

Feng, A., Zhou, J., Vories, E., & Sudduth, K. A. (2020b). Evaluation of cotton emergence using UAV-based narrow-band spectral imagery with customized image alignment and stitching algorithms. *Remote Sens.*, 12(11), 1764.

Ferreira, T. & Rasband, W. (2012). ImageJ User Guide — imageJ/Fiji 1.16r. https://imagej.nih.gov/ij/docs/guide/.

Flores, P., Zhang, Z., Igathinathane, C., Jithin, M., Naik, D., Stenger, J., Ransom, J., & Kiran, R. (2021). Distinguishing seedling volunteer corn from soybean through greenhouse color, color-infrared, and fused images using machine and deep learning. *Ind. Crops Prod.*, 161, 113223.

García-Martínez, H., Flores-Magdaleno, H., Khalil-Gardezi, A., Ascencio-Hernandez, R., Tijerina-Chávez, L., Vázquez-Peña, M. A., & Mancilla-Villa, O. R. (2020). Digital count of corn plants using images taken by unmanned aerial vehicles and cross correlation of templates. *Agronomy*, 10(4), 469.

Ghazi, M. M., Yanikoglu, B., & Aptoula, E. (2017). Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing*, 235, 228–235.

Gnädinger, F. & Schmidhalter, U. (2017). Digital counts of maize plants by unmanned aerial vehicles (UAVs). *Remote Sens.*, 9(6), 544.

Grinblat, G. L., Uzal, L. C., Larese, M. G., & Granitto, P. M. (2016). Deep learning for plant identification using vein morphological patterns. *Comput. Electron. Agric.*, 127, 418–424.

Guo, W., Zheng, B., Potgieter, A. B., Diot, J., Watanabe, K., Noshita, K., Jordan, D. R., Wang, X., Watson, J., Ninomiya, S., et al. (2018). Aerial imagery analysis–quantifying appearance and number of sorghum heads for applications in breeding and agronomy. *Front. Plant Sci.*, 9, 1544.

Hamuda, E., Glavin, M., & Jones, E. (2016). A survey of image processing techniques for plant extraction and segmentation in the field. *Comput. Electron. Agric.*, 125, 184–199.

Harris, D. R. & Fuller, D. Q. (2014). *Agriculture: Definition and Overview*, (pp. 104–113). Springer New York: New York, NY.

Hashemi, A. M., Herbert, S. J., & Putnam, D. H. (2005). Yield response of corn to crowding stress. *Agron. J.*, 97(3), 839–846.

Hassanijalilian, O., Igathinathane, C., Bajwa, S., & Nowatzki, J. (2020). Rating iron deficiency in soybean using image processing and decision-tree based models. *Remote Sens.*, 12(24), 4143.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778).

Heap, I. (February , 2019). International herbicide-resistant weed database. `http://weedscience.org/Pages/Country.aspx`. Accessed: July 1, 2021.

Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. (2003). A practical guide to support vector classification.

Igathinathane, C., Pordesimo, L., & Batchelor, W. (2009). Major orthogonal dimensions measurement of food grains by machine vision using ImageJ. *Food Res. Int.*, 42(1), 76–84.

Igathinathane, C., Pordesimo, L., Columbus, E., Batchelor, W., & Methuku, S. (2008). Shape identification and particles size distribution from basic shape parameters using ImageJ. *Comput. Electron. Agric.*, 63(2), 168–182.

Islam, N., Rashid, M. M., Wibowo, S., Xu, C.-Y., Morshed, A., Wasimi, S. A., Moore, S., & Rahman, S. M. (2021). Early weed detection using image processing and machine learning techniques in an Australian chilli farm. *Agriculture*, 11(5), 387.

Jia, J. & Krutz, G. (1992). Location of the maize plant with machine vision. *J. Agric. Eng. Res.*, 52, 169–181.

Jiang, G., Wang, X., Wang, Z., & Liu, H. (2016). Wheat rows detection at the early growth stage based on hough transform and vanishing point. *Comput. Electron. Agric.*, 123, 211–223.

Jiang, Y. & Li, C. (2020). Convolutional neural networks for image-based high-throughput plant phenotyping: A review. *Plant Phenomics*, 2020, 4152816.

Jiang, Y., Li, C., Paterson, A. H., & Robertson, J. S. (2019). Deepseedling: Deep convolutional network and kalman filter for plant seedling detection and counting in the field. *Plant Methods*, 15(1), 1–19.

Jin, X., Liu, S., Baret, F., Hemerlé, M., & Comar, A. (2017). Estimates of plant density of wheat crops at emergence from very low altitude UAV imagery. *Remote Sens. Environ.*, 198, 105–114.

Johnson, B. J. & Harris, H. B. (1967). Influence of plant population on yield and other characteristics of soybeans. *Agron. J.*, 59(5), 447–449.

Kamilaris, A. & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Comput. Electron. Agric.*, 147, 70–90.

Karami, A., Crawford, M., & Delp, E. J. (2020). Automatic plant counting and location based on a few-shot learning technique. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 13, 5872–5886.

Kataoka, T., Kaneko, T., Okamoto, H., & Hata, S. (2003). Crop growth estimation system using machine vision. In *Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003)*, volume 2 (pp. b1079–b1083).: IEEE.

Khaki, S., Pham, H., Han, Y., Kent, W., & Wang, L. (2020). High-throughput image-based plant stand count estimation using convolutional neural networks. *arXiv preprint arXiv:2010.12552*.

Khurana, G. & Bawa, N. K. (2021). Weed detection approach using feature extraction and kNN classification. In *Advances in Electromechanical Technologies* (pp. 671–679). Springer.

Kitano, B. T., Mendes, C. C., Geus, A. R., Oliveira, H. C., & Souza, J. R. (2019). Corn plant counting using deep learning and UAV images. *IEEE Geosci. Remote Sens. Lett.*, (pp. 1–5).

Koh, J. C., Hayden, M., Daetwyler, H., & Kant, S. (2019). Estimation of crop plant density at early mixed growth stages using UAV imagery. *Plant Methods*, 15(1), 1–9.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 1097–1105.

Kuo, C.-C. J. (2016). Understanding convolutional neural networks with a mathematical model. *J. Visual Commun. Image Represent.*, 41, 406–413.

Lee, W. S., Slaughter, D., & Giles, D. (1999). Robotic weed control system for tomatoes. *Precis. Agric.*, 1(1), 95–113.

Leiva, J. N., Robbins, J., Saraswat, D., She, Y., & Ehsani, R. (2016). Effect of plant canopy shape and flowers on plant count accuracy using remote sensing imagery. *Agric. Eng. Int.: CIGR Journal*, 18(2), 73–82.

Leng, G. & Huang, M. (2017). Crop yield response to climate change varies with crop spatial distribution pattern. *Scientific Reports*, 7(1), 1–10.

Li, B., Xu, X., Han, J., Zhang, L., Bian, C., Jin, L., & Liu, J. (2019). The estimation of crop emergence in potatoes by UAV RGB imagery. *Plant Methods*, 15(1), 1–13.

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine learning in agriculture: A review. *Sensors*, 18(8), 2674.

Liaw, A., Wiener, M., et al. (2002). Classification and regression by Random Forest. *R news*, 2(3), 18–22.

Lin, C. (2010). A support vector machine embedded weed identification system.

Liu, B. & Bruch, R. (2020). Weed detection for selective spraying: A review. *Current Robotics Reports*, 1(1), 19–26.

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. *Am. J. Agri. Biol. Sci.*, 9.

Liu, H., Sun, H., Li, M., & Iida, M. (2020). Application of color featuring and deep learning in maize plant detection. *Remote Sens.*, 12(14), 2229.

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 Sens.*, 10(5), 761.

Lu, H. & Cao, Z. (2020). TasselNetV2+: A fast implementation for high-throughput plant counting from high-resolution RGB imagery. *Front. Plant Sci.*, 11, 1929.

Lu, H., Cao, Z., Xiao, Y., Zhuang, B., & Shen, C. (2017). Tasselnet: Counting maize tassels in the wild via local counts regression network. *Plant Methods*, 13(1), 1–17.

Luna, I. & Lobo, A. (2016). Mapping crop planting quality in sugarcane from UAV imagery: A pilot study in Nicaragua. *Remote Sens.*, 8(6), 500.

Madec, S., Jin, X., Lu, H., De Solan, B., Liu, S., Duyme, F., Heritier, E., & Baret, F. (2019). Ear density estimation from high resolution RGB imagery using deep learning technique. *Agric. For. Meteorol.*, 264, 225–234.

Meaghan, A. (January 23, 2020). Stand assessments - soybean. `https://crops.extension.iastate.edu/encyclopedia/stand-assessments-soybean`. Accessed: July 1, 2021.

Miller, B. C., Hill, J. E., & Roberts, S. R. (1991). Plant population effects on growth and yield in water-seeded rice. *Agron. J.*, 83(2), 291–297.

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci.*, 7, 1419.

Myers, D., Ross, C. M., & Liu, B. (2015). A review of unmanned aircraft system (UAS) applications for agriculture. In *2015 ASABE Annual International Meeting* (pp.Ĩ).: American Society of Agricultural and Biological Engineers.

Nafziger, E. D. (1996). Effects of missing and two-plant hills on corn grain yield. *J. Prod. Agric.*, 9(2), 238–240.

NASA (2021). Remote sensing. `https://earthdata.nasa.gov/learn/remote-sensing`. Accessed: July 1, 2021.

Neto, J. C. (2004). *A combined statistical-soft computing approach for classification and mapping weed species in minimum-tillage systems*. The University of Nebraska-Lincoln.

Neupane, B., Horanont, T., & Hung, N. D. (2019). Deep learning based banana plant detection and counting using high-resolution red-green-blue (RGB) images collected from unmanned aerial vehicle (UAV). *PloS one*, 14(10), e0223906.

Nielsen, R. (1993). Stand establishment variability in corn. Agronomy Department, Purdue University. Accessed: July 1, 2021.

Nukala, R., Panduru, K., Shields, A., Riordan, D., Doody, P., & Walsh, J. (2016). Internet of things: A review from 'farm to fork'. In *2016 27th Irish Signals and Systems Conference (ISSC)* (pp. 1–6).

Oh, S., Chang, A., Ashapure, A., Jung, J., Dube, N., Maeda, M., Gonzalez, D., & Landivar, J. (2020). Plant counting of cotton from UAS imagery using deep learning-based object detection framework. *Remote Sens.*, 12(18), 2981.

Osco, L. P., de Arruda, M. d. S., Junior, J. M., da Silva, N. B., Ramos, A. P. M., Moryia, É. A. S., Imai, N. N., Pereira, D. R., Creste, J. E., Matsubara, E. T., et al. (2020). A convolutional neural network approach for counting and geolocating citrus-trees in UAV multispectral imagery. *ISPRS J. Photogramm. Remote Sens.*, 160, 97–106.

Pang, Y., Shi, Y., Gao, S., Jiang, F., Veeranampalayam-Sivakumar, A.-N., Thompson, L., Luck, J., & Liu, C. (2020). Improved crop row detection with deep neural network for early-season maize stand count in UAV imagery. *Comput. Electron. Agric.*, 178, 105766.

Pathak, H., Subhashree, S., Sunoj, S., & Igathinathane, C. (August,2020). Data security and privacy in precision agriculture - is open-source software a possible solution? USDA Integrator.

Patrício, D. I. & Rieder, R. (2018). Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Comput. Electron. Agric.*, 153, 69–81.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12, 2825–2830.

Peña, J. M., Torres-Sánchez, J., de Castro, A. I., Kelly, M., & López-Granados, F. (2013). Weed mapping in early-season maize fields using object-based analysis of unmanned aerial vehicle (UAV) images. *PloS one*, 8(10), e77151.

Pérez-Ortiz, M., Gutiérrez, P. A., Peña, J. M., Torres-Sánchez, J., López-Granados, F., & Hervás-Martínez, C. (2016). Machine learning paradigms for weed mapping via unmanned aerial vehicles. In *2016 IEEE symposium series on computational intelligence (SSCI)* (pp. 1–8).: IEEE.

Poncet, A., Fulton, J., McDonald, T., Pate, G., & Tisseyre, B. (2012). Maximizing agriculture equipment capacity using precision agriculture technologies.

Pulido, C., Solaque, L., & Velasco, N. (2017). Weed recognition by SVM texture feature classification in outdoor vegetable crop images. *Ingeniería e Investigación*, 37(1), 68–74.

Quirós, J. J. & Khot, L. R. (2016). Potential of low altitude multispectral imaging for in-field apple tree nursery inventory mapping. *IFAC-PapersOnLine*, 49(16), 421–425.

Rahimi, W. N. S., Ali, M. S. A. M., et al. (2020). Ananas comosus crown image thresholding and crop counting using a colour space transformation scheme. *Telkomnika*, 18(5), 2472–2479.

Rai, N. & Flores, P. (2021). Leveraging transfer learning in ArcGIS Pro to detect “doubles” in a sunflower field. In *ASABE 2021 Annual International Meeting*: American Society of Agricultural and Biological Engineers.

Ramcharan, A., Baranowski, K., McCloskey, P., Ahmed, B., Legg, J., & Hughes, D. P. (2017). Deep learning for image-based cassava disease detection. *Front. Plant Sci.*, 8, 1852.

Ranđelović, P., Đorđević, V., Milić, S., Balešević-Tubić, S., Petrović, K., Miladinović, J., & Đukić, V. (2020). Prediction of soybean plant density using a machine learning model and vegetation indices extracted from RGB images taken with a UAV. *Agronomy*, 10(8), 1108.

Rasband, W. S. et al. (1997). ImageJ. `https://imagej.nih.gov/ij/`.

Ribera, J., Chen, Y., Boomsma, C., & Delp, E. J. (2017). Counting plants using deep learning. In *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)* (pp. 1344–1348).: IEEE.

Robinson, R., Ford, J., Luenschen, W., Rabas, D., Smith, L., Warnes, D., & Wiersma, J. (1980). Response of sunflower to plant population. *Agron. J.*, 72(6), 869–871.

Rodriguez-Galiano, V. F., Ghimire, B., Rogan, J., Chica-Olmo, M., & Rigol-Sanchez, J. P. (2012). An assessment of the effectiveness of a random forest classifier for land-cover classification. *ISPRS J. Photogramm. Remote Sens.*, 67, 93–104.

Rogers, M. R. & Stringfellow, W. T. (2009). Partitioning of chlorpyrifos to soil and plants in vegetated agricultural drainage ditches. *Chemosphere*, 75(1), 109–114.

Rojas, C. A. P., Guzman, L. E. S., & Toledo, N. F. V. (2017). A comparative analysis of weed images classification approaches in vegetables crops. *Eng. J.*, 21(2), 81–98.

Rudd, J. D., Roberson, G. T., & Classen, J. J. (2017). Application of satellite, unmanned aircraft system, and ground-based sensor data for precision agriculture: A review. In *2017 ASABE Annual International Meeting* (pp.1̃).: American Society of Agricultural and Biological Engineers.

Samajpati, B. J. & Degadwala, S. D. (2016). Hybrid approach for apple fruit diseases detection and classification using random forest classifier. In *2016 International conference on communication and signal processing (ICCSP)* (pp. 1015–1019).: IEEE.

Sami, K., John, F., Ellzabeth, H., Kaylee, P., & Andrew, K. (March , 2017). https://ohioline.osu.edu/factsheet/fabe-5541.

Sankaran, S., Khot, L. R., Espinoza, C. Z., Jarolmasjed, S., Sathuvalli, V. R., Vandemark, G. J., Miklas, P. N., Carter, A. H., Pumphrey, M. O., Knowles, N. R., et al. (2015). Low-altitude, high-resolution aerial imaging systems for row and field crop phenotyping: A review. *Eur. J. Agron.*, 70, 112–123.

Sankaran, S., Quirós, J. J., Knowles, N. R., & Knowles, L. O. (2017). High-resolution aerial imaging based estimation of crop emergence in potatoes. *Am. J. Potato Res.*, 94(6), 658–663.

Scavo, A. & Mauromicale, G. (2020). Integrated weed management in herbaceous field crops. *Agronomy*, 10(4), 466.

Schneider, C. A., Rasband, W. S., & Eliceiri, K. W. (2012). NIH image to ImageJ: 25 years of image analysis. *Nat. Methods*, 9(7), 671–675.

Shajahan, S. (2019). *Agricultural Field Applications of Digital Image Processing Using an Open Source ImageJ Platform*. PhD thesis, North Dakota State University.

Shajahan, S., Sivarajan, S., Maharlooei, M., Bajwa, S. G., Harmon, J. P., Nowatzki, J. F., & Igathinathane, C. (2017). Identification and counting of soybean aphids from digital images using shape classification. *Trans. ASABE*, 60(5), 1467–1477.

Shirzadifar, A., Maharlooei, M., Bajwa, S. G., Oduor, P. G., & Nowatzki, J. F. (2020). Mapping crop stand count and planting uniformity using high resolution imagery in a maize crop. *Biosyst. Eng.*, 200, 377–390.

Shrestha, D. S. & Steward, B. L. (2003). Automatic corn plant population measurement using machine vision. *Trans. ASAE*, 46(2), 559.

Shrestha, D. S. & Steward, B. L. (2005). Shape and size analysis of corn plant canopies for plant population and spacing sensing. *Appl. Eng. Agric.*, 21(2), 295–303.

Shuai, G., Martinez-Feria, R. A., Zhang, J., Li, S., Price, R., & Basso, B. (2019). Capturing maize stand heterogeneity across yield-stability zones using unmanned aerial vehicles (UAV). *Sensors*, 19(20), 4446.

Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sishodia, R. P., Ray, R. L., & Singh, S. K. (2020). Applications of remote sensing in precision agriculture: A review. *Remote Sens.*, 12(19), 3136.

Slaughter, D., Giles, D., & Downey, D. (2008). Autonomous robotic weed control systems: A review. *Comput. Electron. Agric.*, 61(1), 63–78.

Søgaard, H. T. (2005). Weed classification by active shape models. *Biosyst. Eng.*, 91(3), 271–281.

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 Technol.*, 30(4), 979–984.

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 Technol.*, 31(1), 148–154.

Sugiura, R., Fukagawa, T., Noguchi, N., Ishii, K., Shibata, Y., & Toriyama, K. (2003). Field information system using an agricultural helicopter towards precision farming. In *Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003)*, volume 2 (pp. 1073–1078).: IEEE.

Sunoj, S., Igathinathane, C., & Jenicka, S. (2018a). Cashews whole and splits classification using a novel machine vision approach. *Postharvest Biol. Technol.*, 138, 19–30.

Sunoj, S., Igathinathane, C., Saliendra, N., Hendrickson, J., & Archer, D. (2018b). Color calibration of digital images for agriculture and other applications. *ISPRS J. Photogramm. Remote Sens.*, 146, 221–234.

Sunoj, S., Subhashree, S., Dharani, S., Igathinathane, C., Franco, J., Mallinger, R., Prasifka, J., & Archer, D. (2018c). Sunflower floral dimension measurements using digital image processing. *Comput. Electron. Agric.*, 151, 403–415.

Suresh Babu, D. (2018). Plant-stand count and weed identification mapping using unmanned aerial vehicle images. Master's thesis, North Dakota State University.

Suresha, M., Shreekanth, K. N., & Thirumalesh, B. V. (2017). Recognition of diseases in paddy leaves using kNN classifier. In *2017 2nd International Conference for Convergence in Technology (I2CT)* (pp. 663–666).

Swanton, C. (August 10, 2016). What plants sense and “say” may impact the future of weed control. http://wssa.net/2016/08/what-plants-sense-and-say-may-impact-the-future-of-weed-control.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1–9).

Tang, L. & Tian, L. F. (2008a). Plant identification in mosaicked crop row images for automatic emerged corn plant spacing measurement. *Trans. ASABE,* 51(6), 2181–2191.

Tang, L. & Tian, L. F. (2008b). Real-time crop row image reconstruction for automatic emerged corn plant spacing measurement. *Trans. ASABE,* 51(3), 1079–1087.

Tay, J. Y., Erfmeier, A., & Kalwij, J. M. (2018). Reaching new heights: can drones replace current methods to study plant population dynamics? *Plant Ecolog.*, 219(10), 1139–1150.

Thanh Noi, P. & Kappas, M. (2018). Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using Sentinel-2 imagery. *Sensors*, 18(1), 18.

Tian, L., Slaughter, D., & Norris, R. (2000). Machine vision identification of tomato seedlings for automated weed control. *Trans. ASAE.*, 40(6), 1761–1768.

Tian, L. F. & Slaughter, D. C. (1998). Environmentally adaptive segmentation algorithm for outdoor image segmentation. *Comput. Electron. Agric.*, 21(3), 153–168.

TÜRKOĞLU, M. & Hanbay, D. (2019). Plant disease and pest detection using deep learning-based features. *Turk. J. Electr. Eng. Comput. Sci.*, 27(3), 1636–1651.

US-Grains-Council (2019). Corn, production and exports. https://grains.org/buying-selling/corn. Accessed: July 01, 2021.

USDA (July 30, 2020). Acreage. National Agricultural Statistics Service, United States Department Of Agriculture. Accessed: July 1, 2021.

USGS (2021). What is remote sensing and what is it used for? https://www.usgs.gov/faqs/what-remote-sensing-and-what-it-used/. Accessed: July 1, 2021.

Varela, S., Dhodda, P. R., Hsu, W. H., Prasad, P., Assefa, Y., Peralta, N. R., Griffin, T., Sharda, A., Ferguson, A., & Ciampitti, I. A. (2018). Early-season stand count determination in corn via integration of imagery from unmanned aerial systems (UAS) and supervised learning techniques. *Remote Sens.*, 10(2), 343.

Vollmann, J., Wagentristl, H., & Hartl, W. (2010). The effects of simulated weed pressure on early maturity soybeans. *Eur. J. Agron.*, 32(4), 243–248.

Wang, A., Zhang, W., & Wei, X. (2019). A review on weed detection using ground-based machine vision and image processing techniques. *Comput. Electron. Agric.*, 158, 226–240.

Weis, M., Rumpf, T., Gerhards, R., & Plümer, L. (2009). Comparison of different classification algorithms for weed detection from images based on shape parameters. *Bornimer Agrartechn. Ber*, 69, 53–64.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.

Wicks, G. A., Martin, A. R., & Lyon, D. J. (1999). *Cultural practices to improve weed control in winter wheat*. Cooperative Extension, Institute of Agriculture and Natural Resources.

Woebbecke, D. M., Meyer, G. E., Von Bargen, K., & Mortensen, D. A. (1993). Plant species identification, size, and enumeration using machine vision techniques on near-binary images. In *Optics in Agriculture and Forestry*, volume 1836 (pp. 208–219).: International Society for Optics and Photonics.

Woebbecke, D. M., Meyer, G. E., Von Bargen, K., & Mortensen, D. A. (1995). Color indices for weed identification under various soil, residue, and lighting conditions. *Trans. ASAE.*, 38(1), 259–269.

Wright, M. (2018). Trends in US field corn plant population.

Xiong, H., Cao, Z., Lu, H., Madec, S., Liu, L., & Shen, C. (2019). TasselNetv2: in-field counting of wheat spikes with context-augmented local regression networks. *Plant Methods*, 15(1), 1–14.

Yang, C., Greenberg, S., Everitt, J., Sappington, T., Norman, J., et al. (2003). Evaluation of cotton defoliation strategies using airborne multispectral imagery. *Trans. ASAE.*, 46(3), 869.

Yuba, N., Kawamura, K., Yasuda, T., Lim, J., Yoshitoshi, R., Kurokawa, Y., & Maeda, T. (2020). Counting of pennisetum alopecuroides at heading stage in a grazed pasture using images from an unmanned aerial vehicle. *Grassland Sci.*, 66(4), 285–292.

Zhai, Z., Martínez, J. F., Beltran, V., & Martínez, N. L. (2020). Decision support systems for Agriculture 4.0: Survey and challenges. *Comput. Electron. Agric.*, 170, 105256.

Zhang, C., Atkinson, P. M., George, C., Wen, Z., Diazgranados, M., & Gerard, F. (2020a). Identifying and mapping individual plants in a highly diverse high-elevation ecosystem using UAV imagery and deep learning. *ISPRS J. Photogramm. Remote Sens.*, 169, 280–291.

Zhang, Q., Liu, Y., Gong, C., Chen, Y., & Yu, H. (2020b). Applications of deep learning for dense scenes analysis in agriculture: A review. *Sensors*, 20(5), 1520.

Zhang, X., Li, X., Zhang, B., Zhou, J., Tian, G., Xiong, Y., & Gu, B. (2018). Automated robust crop-row detection in maize fields based on position clustering algorithm and shortest path method. *Comput. Electron. Agric.*, 154, 165–175.

Zhang, Z., Flores, P., Igathinathane, C., L Naik, D., Kiran, R., & Ransom, J. K. (2020c). Wheat lodging detection from UAS imagery using machine learning algorithms. *Remote Sens.*, 12(11), 1838.

Zhang, Z., Kayacan, E., Thompson, B., & Chowdhary, G. (2020d). High precision control and deep learning-based corn stand counting algorithms for agricultural robot. *Auton. Rob.*, (pp. 1–14).

Zhao, B., Zhang, J., Yang, C., Zhou, G., Ding, Y., Shi, Y., Zhang, D., Xie, J., & Liao, Q. (2018). Rapeseed seedling stand counting and seeding performance evaluation at two early growth stages based on unmanned aerial vehicle imagery. *Front. Plant Sci.*, 9, 1362.

Zheng, X., Lei, Q., Yao, R., Gong, Y., & Yin, Q. (2018). Image segmentation based on adaptive k-means algorithm. *EURASIP Journal on Image and Video Processing*, 2018(1), 1–10.

Zheng, Y., Zhu, Q., Huang, M., Guo, Y., & Qin, J. (2017). Maize and weed classification using color indices with support vector data description in outdoor fields. *Comput. Electron. Agric.*, 141, 215–222.

Zimdahl, R. L. (2018). Chapter 11 - Weed management in organic farming systems. In R. L. Zimdahl (Ed.), *Fundamentals of Weed Science (Fifth Edition)* (pp. 337–357). Academic Press, Fifth Edition edition.

APPENDIX A. IMAGEJ CODES FOR CROP STAND COUNT EVALUATION

This appendix provides the pseudocode of plant stand count and spatial distribution plugin with important package import and task specific Java methods. Comments are also provided in order to ease the understanding of the codes.

```
/*****************************************************************************
                    Plant stand counting and spatial distribution using UAV imagery
            Developed by Harsh Pathak & C. Igathinathane; ABEN, NDSU
*****************************************************************************/
// Importing the necessary libraries of Fiji (ImageJ)
import ij.IJ;
import ij.ImagePlus;
import ij.process.ImageProcessor;
import ij.plugin.filter.ExtendedPlugInFilter;
import ij.plugin.filter.PlugInFilterRunner;
import ij.gui.DialogListener;
import ij.gui.GenericDialog;
import ij.gui.NewImage;
import ij.WindowManager;
import ij.gui.Plot;
import java.util.Arrays;
import java.awt.Font;
import ij.plugin.filter.MaximumFinder;
import ij.plugin.filter.ParticleAnalyzer;
import ij.plugin.filter.Analyzer;
import ij.measure.ResultsTable;
import ij.plugin.filter.GaussianBlur;
import ij.measure.ResultsTable;
import ij.plugin.filter.ExtendedPlugInFilter;
import ij.measure.Measurements;
import java.awt.Color;
import java.awt.AWTEvent;
import ij.gui.TextRoi;
import ij.gui.ProfilePlot;
import ij.gui.Overlay;
```

```
import ij.text.TextWindow;
import ij.gui.Wand;
/*************************************************************************/

public class cornplantcounter implements PlugIn, Measurements{
   // Data variables
   ResultsTable rt,rt1;
   ImagePlus imp, seg, plotImage, plotImage1,   masked_imp, seg1, impd, spdimp,
       eximpd, exspimp;
   ImageProcessor ip, ip2, masked_ip, ip2_dup;

   long t1, t2;

   float[] noc, areac, perc, mAxisc, miAxisc, xCenc, yCenc, anglec, feretc, bXc, bYc,
       bWidthc, bHeightc; // cumulative
   float[] nos, areas, pers, mAxiss, miAxiss, xCens, yCens, angles, ferets, bXs, bYs,
       bWidths, bHeights; // sliding
   float[] circ, aspectratio, roundness, solidity, rty, rar, fmr, yvalue;
   double[] qualrows;
   int[] maxa, xval, yval, value, plants, starting_pt;
   float ar, pr, major, minor,sol,wd,ht;
   double ratio,comp,num1,den1,num2,den2,ral,el,
       conarea,hol,num3,den3,ltp,rbp,rect,x2;

   int flags = DOES_RGB+SUPPORTS_MASKING+KEEP_PREVIEW+NO_CHANGES;
   boolean previewing;

   double roi_width_factor = 0.6;
   int number_roi_image_steps = 2;
   double plant_spacing = 6; // Default value 6 inches provided
   double row_spacing = 30; // Default value 30 inches provided
   double dpi = 1; // Based on known number of pixel over known length

   int strokeWidth = 1;
   int offset = 0;
   int strokeWt = strokeWidth;
   int ovalsize = 5;

   double barWidthp = 8.0;  // distribution map bar width as a percent of image width
   int barWidth, outsize, insize;
   double sum_bHeights = 0.0;
   int no_plants = 0;
   int tot_particles = 0;

/*************************************************************************//
   public int setup(String arg, ImagePlus imp) { //setup method
```

```
      this.imp = imp;
      imp_name = imp.getShortTitle();
      return DOES_RGB+SUPPORTS_MASKING+KEEP_PREVIEW+NO_CHANGES;
   }

   public void run(ImageProcessor ip) {
      t1 = System.currentTimeMillis();

      segment_image(ip, thresh, previousImageName, chh);
      gaus_filter(masked_ip);
      starting_pt = plot_profile(masked_imp, ip2_dup, seg1, masked_ip, plen);

      IJ.log("\n\n************ Plant Stand Count Results ***************");
      IJ.log("starting point length"+starting_pt.length);
      tot_particles = roi_march(masked_ip, masked_imp, starting_pt);
//    extendCanvas(impd);

      t2 = System.currentTimeMillis();
      IJ.log("\nRoi width = " + roi_width_factor);
      IJ.log("Total image objects = " + tot_particles);
      IJ.log("Total plant stand count = " + no_plants);
      IJ.log("Total CPU time taken (ms) = " + (t2-t1));
      IJ.log("Total CPU time taken (s) = " + IJ.d2s((Math.round((t2-t1)/1000.0)),0));

   }
   /*************** Functions ***************/
   public ImagePlus segment_image(ImageProcessor ip, double thresh, String name,
       String chh){ //chh contains the String name of the method used to create the
       binary image

   int w = ip.getWidth(); // fetching the width of image
   int h = ip.getHeight(); // getting the height of image

   ImagePlus seg = NewImage.createByteImage("Segmented_Image_Thresh:" +
       IJ.d2s(thresh,0) + " of " + imp_name,
                 w, h, 1, NewImage.FILL_BLACK + NewImage.GRAY8);
      ImageProcessor ip2 = seg.getProcessor();
      int []RGB = new int[3];
         for (int i = 0; i < h; i++){
            for (int j = 0; j< w; j++){
               RGB = ip.getPixel(j,i,RGB);
               R = RGB[0];
               G = RGB[1];
               B = RGB[2];
//-------------------------------------------------------------------------------excess
    green segmentation
```

```
            if (chh =="ExG"){
               int ex = (2 * G) - R - B;
               if (ex > thresh){
                  ip2.set(j,i,255);
                  }//inner if
               else{
                  ip2.set(j,i,0);
               }//else
            }//outer if
//other segmentation methods
         }//inner for loop
      }//outer for loop
   ParticleAnalyzer pa;
   ResultsTable rt = new ResultsTable();
   masked_ip = masked_imp.getProcessor();

}// functions ends

/*************** Using gaussian filter ***************/

public void gaus_filter(ImageProcessor masked_ip){

   ip2_dup = masked_ip.duplicate(); //Here we show how we create a new image from
       a given ImageProcessor
   GaussianBlur gb = new GaussianBlur();
   // initialize the values of sigmaX and sigmaY
      gb.blurGaussian(ip2_dup,sigmaX,sigmaY,accuracy);
   } // function ends

/*************** Profile plot for row identification ***************/

public int[] plot_profile(ImagePlus masked_imp, ImageProcessor ip2_dup,ImagePlus
    seg1, ImageProcessor masked_ip, int plen){
   int w = ip2_dup.getWidth();
   int h = ip2_dup.getHeight();
   ProfilePlot profile = new ProfilePlot(seg1);
   // fetch the corresponding x-coordinate values at which profile-plot value is
       maximum
   return x_start_point;
}// function ends

/*************** ROI marching for off-linear row identification and plant stand
    count evaluation***************/

public int roi_march(ImageProcessor masked_ip, ImagePlus masked_imp,
    int[]marching_points){
```

```
int roi_x, roi_y, roi_top, roi_bot, roi_w_end, numberOfSlides, numberOfRows,
    lpix, miny, maxy;
boolean encounter;

int counterAll, counter;
long t1, t2;
ResultsTable rtOrig, rt;
TextWindow tw;
// number of rows can be found by the length of x_start_point
for (int j = 0; j < numberOfRows; j++) { // crop rows
   masked_imp.setRoi(roi_x, roi_top, roi_w, roi_bot-roi_top); // ROI march from
       bottom to top
      ipd.drawRect(roi_x, roi_top, roi_w, roi_bot-roi_top);
      // count the number of particles and fetch the centroid locations
      IJ.run(masked_imp, "Analyze Particles...", "display exclude overlay
          clear");
}

// labeling of plants
for (int i = 0; i < num; i++){
   TextRoi roi = new TextRoi(label, (double) bx+bw+1, (double) by+textHt-1,
       font);
   // add the labels and draw the bounding boxes on plants
   overlay.add(roi);

}

// Crop row identification
for (int i = 0; i < num-1; i++){
   cx1 = (int)xCenc[i];
   cy1 = (int)yCenc[i];
   cx2 = (int)xCenc[i+1];
   cy2 = (int)yCenc[i+1];

   if (cy2 < cy1)
      ipd.drawLine(cx1, cy1, cx2, cy2);
   else
      ipd.drawPixel(cx1, cy1);
}

// Generating a spatial distribution map

spdimp = masked_imp.duplicate();
spdimp.setTitle("Spatial distribution map of " + imp_name);
IJ.run(spdimp, "RGB Color", "");
```

```
      ImageProcessor spdip = spdimp.getProcessor();

      // Spacing classes cutoff
      double R1 = (plant_spacing * 0.5);
      double R2 = (plant_spacing * 1.5);
      double R3 = (plant_spacing * 2.5);
      double R4 = (plant_spacing * 3.5);

      for (int i = 0; i < num-1; i++){
      // testing conditions for finding doubles, single skip, double skips, multiple
          skips, and ideal plant spacing
      if(plant_distance < R1){
            spdip.setColor(Color.blue);
            }
         spdip.fillRect(cx2-barWidth/2, cy2, barWidth,(cy1-cy2));

      }

   }
   // used for getting passes from the slider
   public void setNPasses (int nPasses) {}


}
```

APPENDIX B. FUNCTION FOR FEATURE EXTRACTION

This appendix provides the pseudocode that can be used to extract shape features of different weed species. Comments are provided in order to ease the understanding of the codes.

```
/*****************************************************************************
                    Functions for feature extraction
             Developed by Harsh Pathak & C. Igathinathane; ABEN, NDSU
*****************************************************************************/
// Importing the necessary libraries of Fiji (ImageJ)
import ij.plugin.*;
import java.io.*;
import ij.*;
import ij.io.*;
import ij.process.*;
import ij.gui.*;
import java.math.*;
import ij.ImagePlus;
import ij.process.ImageProcessor;
import ij.measure.ResultsTable;
import java.util.Vector;
import ij.measure.*;
import ij.plugin.filter.*;
/*****************************************************************************/
public class feature_extractor implements PlugIn, Measurements{
   // Data variables
   int R,G,B;
   float[] areas,perimeters,e_major,e_minor,RAR,solid,wid,hgt,fer,ferx,fery,ferang;
   String nameoffolder,nameoffile;
   ResultsTable rt;
   float ar, pr, major, minor,sol,wd,ht;
   double eqdia,perieqperi_ratio,eqperi,ratio,comp,num1,den1,num2,den2,ral,
el,ltp,conarea,hol,num3,den3,rbp,rect,x2,y2,ang,fer_dia,fir_no,sec_no,
fmi,fma,caf,num4,den4,mc,num5,num6,den5,den6,gc,ac,lhw;
/*****************************************************************************/
/*************** Fetching each image file from folder for extracting shape features
    ***************/
   public void run(String arg) {
```

```
   OpenDialog od = new OpenDialog("Select a file in source folder...", ""); //
       opening the dialog box for selecting the path
   selectfiles(od.getDirectory()); // function for fetching each file
      }

public void selectfiles(String dir) {
   String[] list = new File(dir).list();
   // ..
   // ..
   // opening each file or image from the selected directory
   for (int i=0; i<n; i++) {
      File f = new File(dir+list[i]);
      if (!f.isDirectory()) {
         ImagePlus img = new Opener().openImage(dir, list[i]);
         ImageProcessor ip = img.getProcessor();
         ..
         segmentation(img, ip);
      }
   }
}
/*************** Image segmentation ***************/
public void segmentation(ImagePlus img, ImageProcessor ip){
   int w = ip.getWidth(); // fetching the width of image
   int h = ip.getHeight(); // fetching the height of image
   // Creating a new ImagePlus
   ImagePlus seg = NewImage.createByteImage("Segmented_Image_Thresh:",
              w, h, 1, NewImage.FILL_BLACK + NewImage.GRAY8);

   ImageProcessor ip2 = seg.getProcessor();
   int []RGB = new int[3];
   for (int i = 0; i < h; i++){
      for (int j = 0; j< w; j++){
         RGB = ip.getPixel(j,i,RGB);
         R = RGB[0];
         G = RGB[1];
         B = RGB[2];
         int ex = (2 * G) - R - B;
         if (ex > 12){
            ip2.set(j,i,255);
               }//inner if
         else{
            ip2.set(j,i,0);
               }//else
         }//inner for loop
   }//outer for loop
```

```
      analyze_particles(seg); // function used to extract twenty-one shape features
   }
   /*************** Obtaining shape parameters ***************/
   public void analyze_particles(ImagePlus seg){
      ParticleAnalyzer pa;
      rt = Analyzer.getResultsTable();
      int measurements = Analyzer.getMeasurements();

      pa = new
          ParticleAnalyzer(options,Analyzer.getMeasurements(),rt,1500.0,10000000.0);

//    pa.showDialog();
      if (!pa.analyze(seg)){
         return;
      }
      for(int m = 0; m <no_particles ; m++){
      // features to be extracted
      }
      rt.show("Results");
      rt.updateResults();
   }
}
```

APPENDIX C. SCRIPTS FOR DATA VISUALIZATION THROUGH R PROGRAMMING

This appendix provides scripts for data visualization in R through ggplot2. The graphs shown in chapters 4 and appendix D were developed by using the following scripts.

```
/*****************************************************************************
    Script for data visualization using ggplot in R

     Developed by Harsh Pathak and C. Igathinathane; ABEN, NDSU
*****************************************************************************/
---------------------------------------------------------------------
title: "Data visualization for weed classification"
author: "Harsh Pathak and C. Igathinathane"
date: "04/12/2021"
output: pdf_document
---------------------------------------------------------------------

# Importing necessary libraries for data visualization
library(ggplot2)
library(ggExtra)
library(readxl)
library(ggthemes)
library(dplyr)
library(ggrepel)
library(rlang)
getwd()
setwd(path)

data <- read_excel("plantanalysis.xlsx")
p <- ggplot(data,aes(X, y, colour = Species, group = Species)) + geom_line() +
    geom_point() + theme_bw() + theme(legend.position = "desired position") + labs (x
    = "labels for x-axis", y = "label for y-axis") + theme(axis.title.x = position,
    axis.title.y = position)

p1 <- ggMarginal(p,..., type = 'density')
p1
```

APPENDIX D. PYTHON SCRIPTS FOR MACHINE LEARNING MODELS HYPERPARAMETERS TUNING

This appendix provides scripts for tuning the hyperparameters of the machine learning models. The optimal hyperparamters values for the machine models were extracted by using the following scripts.

```
/*****************************************************************************
    Python scripts for tuning the machine learning hyperparameters

     Developed by Harsh Pathak and C. Igathinathane; ABEN, NDSU
*****************************************************************************/
-----------------------------------------------------------------
title: "Machine learning models hyperparameters tuning using GridSearchCV"
author: "Harsh Pathak and C. Igathinathane"
date: "01/7/2021"
-----------------------------------------------------------------

// Importing necessary libraries
//---------------------------------------------------------------
import numpy as np
import pandas as pd
from sklearn.externals import joblib
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
//---------------------------------------------------------------
// Reading the dataset
dataset = pd.read_excel("path")
print("Shape of dataset: " + str(dataset.shape)) // shape of dataset
print(dataset.head())
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
```

```
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.2,
    random_state = 100) // splitting the dataset into training and testing
// Scaling the dataset
sc = StandardScaler()
X_train[:,20:21] = sc.fit_transform(X_train[:,20:21])
X_test[:,20:21] = sc.transform(X_test[:,20:21])

// Optimal values of hyperparameters for three non-parametric machine learning
    models namely kNN, RF, and SVM

// kNN
hyperparameters_knn = dict(n_neighbors = k values, p = [1,2])
knnclassifier = KNeighborsClassifier()
grid_classifier_kNN = GridSearchCV(hyperparamters)
// train the model and find hyperparameters
// RF
parameter_grid = {
       'n_estimators' : n_estimators, // number of trees
       'max_depth' : max_depth,
       'max_features' : max_features
     }
rfclassifier = RandomForestClassifier()
grid_classifier_RF = GridSearchCV(parameter_grid)
// train the model and find hyperparameters
print(grid_classifier_RF.best_params_)

//SVM
hyperparameters_svm = dict(C = c,gamma = g, kernel =ker)
svmclassifier = SVC()
grid_classifier_SVM = GridSearchCV(hyperparamters)
grid_classifier_SVM.fit(datasets, labels) // train the model and find hyperparameters
print(grid_classifier_SVM.best_params_)
```

APPENDIX E. PYTHON SCRIPTS FOR MACHINE LEARNING CLASSIFIERS

This appendix provides scripts for training and testing machine learning models in python. The different machine learning models used to classify weed species were developed by using the following scripts.

```
/*****************************************************************************
    Python scripts for training and testing machine learning

     Developed by Harsh Pathak and C. Igathinathane; ABEN, NDSU
*****************************************************************************/
-----------------------------------------------------------------
title: "Machine learning models hyperparameters tuning using GridSearchCV"
author: "Harsh Pathak and C. Igathinathane"
date: "01/7/2021"
-----------------------------------------------------------------

// Importing necessary libraries
//---------------------------------------------------------------
import numpy as np
import pandas as pd
from sklearn.externals import joblib
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
//---------------------------------------------------------------
// Reading the dataset
dataset = pd.read_excel("path")
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.2,
    random_state = 100) // splitting the dataset into training and testing
// Scaling the dataset
sc = StandardScaler()
```

```
classifier = KNeighborsClassifier(tuned hyperparamters values) // the classifier
    classes can be changed along with the corresponding hyperparamters
classifier.fit(X_train,y_train)
y_pred = classifier.predict(X_test)
print(accuracy_score())
print(classification_report()) // same procedure can be followed for all the other
    machine learning models
```

APPENDIX F. PYTHON SCRIPTS FOR TRAINING AND TESTING MACHINE LEARNING

This appendix provides scripts for training and testing machine learning models in python. The different machine learning models used to classify weed species were developed by using the following scripts.

```
/*****************************************************************************
    Python scripts for training and testing machine learning

     Developed by Harsh Pathak and C. Igathinathane; ABEN, NDSU
*****************************************************************************/
-----------------------------------------------------------------
title: "Machine learning models hyperparameters tuning using GridSearchCV"
author: "Harsh Pathak and C. Igathinathane"
date: "01/7/2021"
-----------------------------------------------------------------

// Importing necessary libraries
//---------------------------------------------------------------
import numpy as np
import pandas as pd
from sklearn.externals import joblib
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
//---------------------------------------------------------------
// Reading the dataset
dataset = pd.read_excel("path")
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.2,
    random_state = 100) // splitting the dataset into training and testing
// Scaling the dataset
sc = StandardScaler()
```

```
classifier = KNeighborsClassifier(tuned hyperparamters values) // the model classes
    can be changed along with the hyperparamters
classifier.fit(X_train,y_train)
y_pred = classifier.predict(X_test)
print(accuracy_score())
print(classification_report()) // same procedure can be followed for all the other
    machine learning models
```

APPENDIX G. COMMON WEEDS OF NORTH DAKOTA AND THEIR IDENTIFICATION FEATURES

This appendix provides information about the different types of weeds (studied and others) of North Dakota (ND) and their identification features. Weeds are unwanted plants that grow automatically in a field and compete with the main crop for nutrients, sunlight, and water and reduce crop yield. Weeds can be identified or classified based on their biological morphology that includes shape and structure. Different types of weeds commonly available in ND, and their identification features are discussed. It is to be noted that computer vision algorithms can be developed to classify and identify weeds using these shape features. Weed identification can help to efficiently perform the weed management process by reducing the cost of herbicides and their environmental impacts and increasing the crop yield. The various weed species of ND and their identification features are briefly presented in the subsequent sections.

G.1. Common Lambsquarters

Common lambsquarters (*Chenopodium album L.*) is widely spread in agricultural fields and gardens. Leaves of the common lambsquarters are alternate, irregularly toothed, and vary in shape from roughly triangular to diamond-shaped or lanced. Leaves of common lambsquarters have grayish-white powdery coating at the early growth stage.

Figure G1. Seedlings and matured stage of common lambsquarters weed.

G.2. Common Purslane

Common purslane (*Portulaca oleracea*) is a highly variable weedy plant in the purslane family (*Portulaca*) with a wide distribution. It grows readily in moist soil even after being cutoff and thus needs to be removed from the garden after hoeing. Its leaves are present nearly opposite or alternate to each other or in clusters at the end of the branches. Their leaves are thick and fleshy, oval to spoon-shaped, rounded, or nearly truncated at the tips. They usually do not have petioles and are attached to the stems.

Figure G2. Common purslane weed imagery.

G.3. Horseweed

In recent years, horseweed (*Conyza canadensis*) has failed to respond to glyphosate (for example, Roundup) in agricultural fields and landscapes. Cotyledons are less long and egg-shaped, while the true leaves are larger, densely-haired, and pointed ovals. It has sessile to short petioles.

Figure G3. Horsweed imagery at two different stages.

G.4. Redroot Pigweed

Redroot pigweed (*Amaranthus retroflexus*) is an abundant seed producer and found all across the United States (USA) in horticultural nurseries, agronomic crops, landscapes roadsides, and in pastures and forages. Leaves are round to oval in shape, have prominent veins, and have pubescence, especially at the lower surface near the veins and stems. Petioles of redroot pigweed are as long as or shorter than the blades.

Figure G4. Redroot pigweed imagery at two different stages.

G.5. Waterhemp

Waterhemp (*Amaranthus rudis*) is a highly variable and aggressive weed found in horticultural nurseries, agronomic crops, landscapes roadsides, and in pastures and forages. The stems do not have any hair and are smooth, and their color varies from green to red. Leaves of waterhemp vary from ovate or lanceolate to spatulate or oblong and are dark green. Petioles are up to 5 cm long.

Figure G5. Waterhemp imagery at two different stages (*Source*: `https://extension.psu.edu/invasive-pigweeds-palmer-amaranth-and-waterhemp`).

G.6. Common Ragweed

Common ragweed (*Ambrosia artemisiifolia*) is a widespread, commonly-found weed in agricultural fields. It can be easily identified due to its distinct leaves. Its cotyledons are dark green, thick, and spatula-shaped. The first true leaves have hairs on both surfaces and have a fern-like structure. Leaves are once or twice compound. Lower leaves of common ragweed have longer petioles, while the upper leaves seem to have no petioles.

Figure G6. Common ragweed imagery at two different stages (*Source*: `https://blogs.cornell.edu/weedid/field-crops/common-ragweed-ambrosia-artemisiifolia/`).

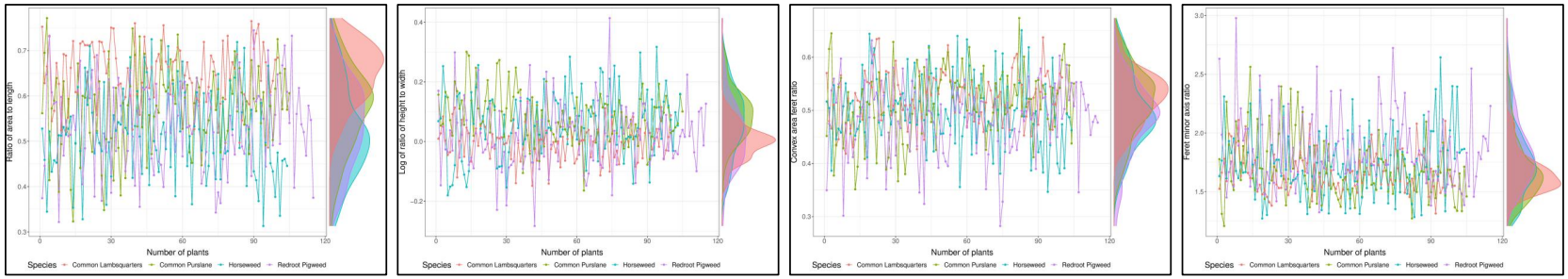
G.7. Kochia

Kochia (*Kochia scoparia L.*) is another weed species commonly found in agricultural fields. The leaves are linear to narrowly ovate. Young leaves of Kochia are also covered with hairs, especially along the leaf margins giving it a silver tint. Furthermore, they have short-petioles. These plants are highly branched and are bushy. Their stems turn red with age.

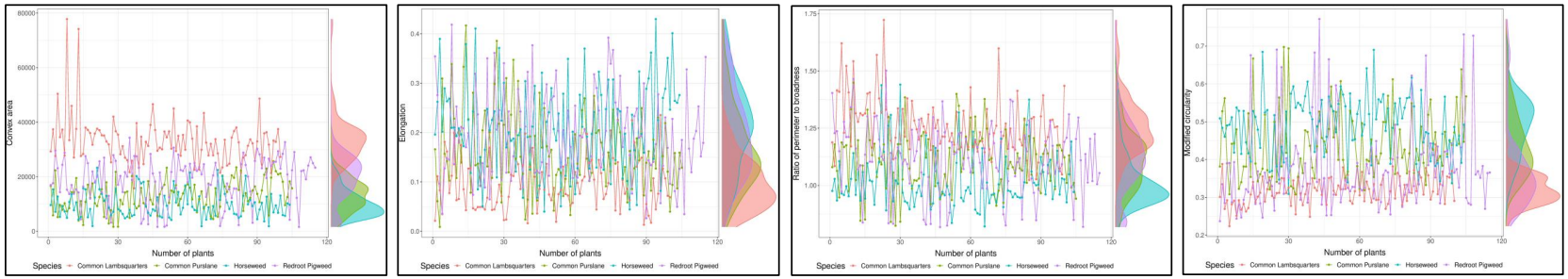
Figure G7. Kochia imagery at two different stages (*Source*: https://weedid.missouri.edu/weedinfo.cfm?weed_id=148).

APPENDIX H. GRAPHICAL REPRESENTATION OF SHAPE FEATURES USED FOR WEED CLASSIFICATION

Twenty one shape features were extracted for each weed species RGB imagery. These features were used to develop handcrafted algorithm and machine learning model for classifying four different weed species (common lambsquarters, common purslane, horseweed, and redroot pigweed). This appendix provides illustrations of the distribution of feature values of different weed species.



(A) (B) (C) (D)



(E) (F) (G) (H)

Figure H1. Graphical representation of weed shape features (1st set): (A) Ratio of area to length, (B) Log of ratio of length to width, (C) Convex area feret ratio, (D) Feret minor axis ratio, (E) Convex area, (F) Elongation, (G) Ratio of perimeter to broadness, and (H) Modified circularity.

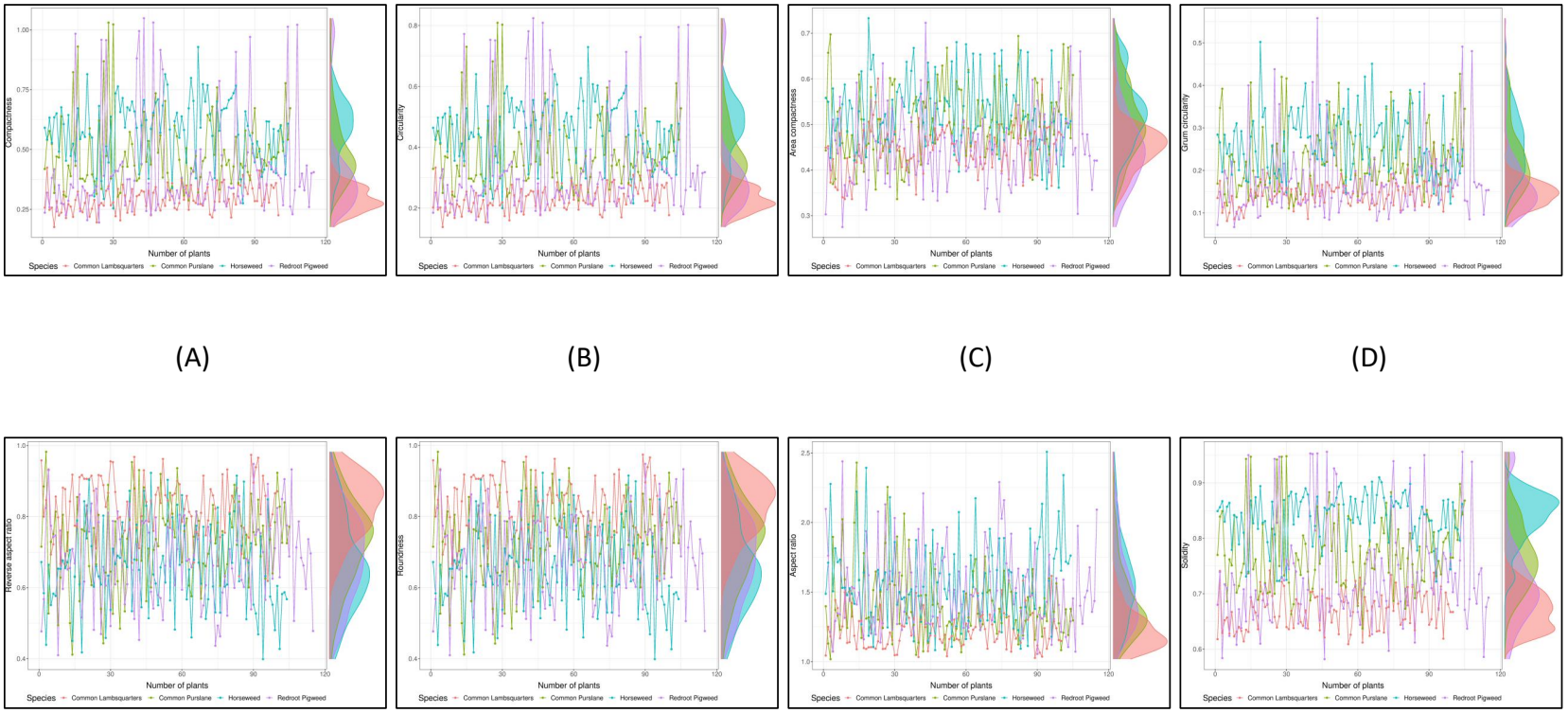


Figure H2. Graphical representation of weed shape features (2nd set): (A) Compactness, (B) Circularity, (C) Area compactness, (D) Grum circularity, (E) Reverse aspect ratio, (F) Roundness, (G) Aspect ratio, and (H) Solidity.