AGRICULTURAL FIELD APPLICATIONS OF DIGITAL IMAGE PROCESSING USING

AN OPEN SOURCE IMAGEJ PLATFORM

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

By

Sunoj Shajahan

In Partial Fulfillment of the Requirements
for the Degree of
DOCTOR OF PHILOSOPHY

Major Program:
Agricultural and Biosystems Engineering

May 2019

Fargo, North Dakota

NORTH DAKOTA STATE UNIVERSITY

Graduate School

Title

AGRICULTURAL FIELD APPLICATIONS OF DIGITAL IMAGE PROCESSING

USING AN OPEN SOURCE IMAGEJ PLATFORM

By

Sunoj Shajahan

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

DOCTOR OF PHILOSOPHY

SUPERVISORY COMMITTEE:

Dr. Igathinathane Cannayen
Chair

Dr. Shafiqur Rahman

Dr. Zhulu Lin

Dr. Peter G. Oduor

Approved:

10/17/2019
Date

Dr. Kenneth Hellevang
Department Chair

ABSTRACT

Digital image processing is one of the potential technologies used in precision agriculture to gather information, such as seed emergence, plant health, and phenology from the digital images. Despite its potential, the rate of adoption is slow due to limited accessibility, unsuitability to specific issues, unaffordability, and high technical knowledge requirement from the clientele. Therefore, the development of open source image processing applications that are task-specific, easy-to-use, requiring fewer inputs, and rich with features will be beneficial to the users/farmers for adoption. The Fiji software, an open source free image processing ImageJ platform, was used in this application development study. A collection of four different agricultural field applications were selected to address the existing issues and develop image processing tools by applying novel approaches and simple mathematical principles.

First, an automated application, using a digital image and "pixel-march" method, performed multiple radial measurements of sunflower floral components. At least 32 measurements for ray florets and eight for the disc were required statistically for accurate dimensions. Second, the color calibration of digital images addressed the light intensity variations of images using standard calibration chart and derived color calibration matrix from selected color patches. Calibration using just three color patches: red, green, and blue was sufficient to obtain images of uniform intensity. Third, plant stand count and their spatial distribution from UAS images were determined with an accuracy of $\approx 96\%$, through pixel-profile identification method and plant cluster segmentation. Fourth, the

soybean phenological stages from the PhenoCam time-lapse imagery were analyzed and they matched with the manual visual observation. The green leaf index produced the minimum variations from its smoothed curve. The time of image capture and PhenoCam distances had significant effects on the vegetation indices analyzed. A simplified approach using kymograph was developed, which was quick and efficient for phenological observations. Based on the study, these tools can be equally applied to other scenarios, or new user-coded, user-friendly, image processing tools can be developed to address specific requirements. In conclusion, these successful results demonstrated the suitability and possibility of task-specific, open source, digital image processing tools development for agricultural field applications.

ACKNOWLEDGEMENTS

DEDICATION

This dissertation is dedicated to my parents *Mr. Shajahan Badrudeen* and

Mrs. Noorjahan Shajahan, and my elder brother *Mr. Sooraj Shajahan*.

TABLE OF CONTENTS

xi

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

2D . . . . Two-dimensional

3D . . . . Three-dimensional

API . . . . Application Program Interface

B of RGB . . . . Blue

BBH . . . . Bounding Box Height

BCC . . . . Blue Chromatic Coordinate

C of CMY . . . . Cyan

CCM . . . . Color Calibration Matrix

CCW . . . . Counterclockwise

CIVE . . . . Color Index for Vegetation

ColorCal . . . . Color Calibration Plugin

CPI . . . . Calibration Performance Index

CVI . . . . Color Vegetation Index

CW . . . . Clockwise

DAP . . . . Days After Planting

DGCI . . . . Dark Green Color Index

DII . . . . Difference Image Index

DSLR . . . . Digital Single-Lens Reflex

EW . . . . East-West

EXG . . . . Excess Green

EXGR . . . . Excess Green Minus excess Red

EXR . . . . Excess Red

FIJI . . . . Fiji Is ImageJ

FTP . . . . File Transfer Protocol

G of RGB . . . . Green

RGB . . . . Red, Green, and Blue — color space

RGBVI . . . . Red Green Blue Vegetation Index

RGRI . . . . Red Green Ratio Index

ROI . . . . Region of Interest

TGI . . . . Triangular Greenness Index

TIFF . . . . Tagged Image File Format

UAS . . . . Unmanned Aerial System

USA . . . . United States of America

VARI . . . . Visible Atmospherically Resistant Index

VI . . . . Vegetation Index

WDRVI . . . . Wide Dynamic Range Vegetation Index

X-Rite . . . . ColorChecker Classic target of 24 standard colors

Y of CMY . . . . Yellow

YUV . . . . Luminance (Y) are chrominance (U&V) — color space

LIST OF SYMBOLS

A_{cut} . . . . Area cutoff — 87th percentile of data

A_i . . . . Area of ith plant object for vigor analysis (pixel^2)

A_{min} and A_{max} . . . . Minimum and maximum area of plant for vigor analysis (pixel^2)

A_{norm} . . . . Normalized area for vigor analysis

A_{p} . . . . Area of profile using trapezoid formula (pixel^2)

A_{v} . . . . Area of valley in profile using Heron's formula (pixel^2)

AR . . . . Area ratio to decide segmentation

BBH_{cut} . . . . Bounding box height cutoff — 87th percentile of data

CVI_i . . . . Color vegetation index value at ith day

CVI_{Gmin} and CVI_{Gmax} . . . . Color vegetation index minimum and maximum among replications

CVI_{norm} . . . . Normalized color vegetation index

d . . . . Distance between two point coordinates (pixel)

$d_{1-N_{\text{rows}}}$ . . . . Center-to-center between first and last row (pixel)

d_i . . . . Spacing of ith distance (m)

$d_{\text{o},i}$ . . . . Observed plant spacing distance (m)

D_{180} . . . . Mean of 180 measurements on sunflower (m)

D_{2h} . . . . Mean of two measurements on sunflower head (m)

DI_{B} . . . . Blue channel difference image of individual pixels

DI_{G} . . . . Green channel difference image of individual pixels

DI_{R} . . . . Red channel difference image of individual pixels

DII_{A} . . . . Difference image index between calibrated dark and reference calibrated dark images

$L_\text{Deviation}$ · · · · · · · · · · · · · · · Curve length deviation of the raw and smoothened color vegetation index curves (%)

L_raw · · · · · · · · · · · · · · · · · Curve length of raw normalized color vegetation index curve (pixel)

L_row · · · · · · · · · · · · · · · · · Length of the row based on field ROI (m)

L_smooth · · · · · · · · · · · · · · · Curve length of the smoothened color vegetation index curve (pixel)

$\mathbf{M}_\text{CC}$ · · · · · · · · · · · · · · · · · A 3 × 3 color calibration matrix

MB · · · · · · · · · · · · · · · · · · · Megabyte

N_c · · · · · · · · · · · · · · · · · · · Number of data above the profile cutoff

N_I · · · · · · · · · · · · · · · · · · · Ideal number of plants in the given area

N_o · · · · · · · · · · · · · · · · · · · Observed plant count

N_plants · · · · · · · · · · · · · · · · · Number of plants in a qualified cluster

N_plugin · · · · · · · · · · · · · · · · · Number of plants that was counted by the plugin

N_PR · · · · · · · · · · · · · · · · · · Maximum possible number of ideally spaced plants in a row

N_rows · · · · · · · · · · · · · · · · · · Number of crop rows

N_T · · · · · · · · · · · · · · · · · · · Total number of data in the profile

N_visual · · · · · · · · · · · · · · · · · Number of plants that were visually counted

n · Number of measurements

N · Number of X-Rite color patches

p · Probability level of statistical significance

P_max · · · · · · · · · · · · · · · · · · Maximum value of the pixel profile

PR · · · · · · · · · · · · · · · · · · · Profile ratio

PR_cut · · · · · · · · · · · · · · · · · · Profile ratio cutoff value

r · Pearson's correlation coefficient

r_c, g_c, and b_c · · · · · · · · · · · · Chromatic coordinates of RGB color channels

R^2 · · · · · · · · · · · · · · · · · · · Coefficient of determination

R_gw · · · · · · · · · · · · · · · · · · A constant ratio for the X-Rite chart = 0.125

s · Semi-perimeter of a general triangle (pixel)

LIST OF APPENDIX TABLES

LIST OF APPENDIX FIGURES

1. GENERAL INTRODUCTION

This chapter provides the need for sensors in agriculture, advantages of imaging technology, the current status of imaging technology in agriculture, common image processing platforms with an emphasis on the need for open source platforms, and orients to the problem statement and objectives. Also, the dissertation organization is provided at the end to have an idea of the arrangement of chapters that deal with the formulated objectives.

1.1. Need for Sensors in Agriculture

Increasing population, decreasing arable area, escalating costs of inputs, and limited availability of resources have led to a movement of efficient agricultural practices. Of several possible approaches to improving the efficiency of agricultural systems, precision agriculture is one of the modern methods. Precision agriculture is a farm management practice aimed to maximize agricultural productivity and minimize farm inputs (e.g., seed, fertilizer, herbicide, irrigation) by applying the right amount of inputs at the right place at the right time (Mulla, 2013). More than 40 % farmers in the US have already adopted precision agriculture and this trend is expected to rise in the future (Schimmelpfennig, 2016). In precision agriculture, the fields are delineated into zones based on spatial variability in soil properties, field landscape, or plant growth for efficient management. The variability is captured using a variety of sensors that record a suite of measurements right from soil nutrients to crop yield. These timely measurements help farmers to make site-specific management decisions (Scharf et al., 2011).

1

At present, several sensors are being used to measure the in-field variability and they are classified based on the type of measurements as contact and non-contact. The former type is when a sensor is capable of recording measurements only when in contact with the object and they are sometimes destructive. Measurement of soil properties such as soil compaction and moisture, and leaf properties such as stomatal conductance and chlorophyll content are some of the examples.

The latter type is when a sensor is capable of recording measurements without being in contact with the object. This type of measurement is called "remotely sensed." It basically uses the amount of electromagnetic energy reflected, rather than absorbed or transmitted, from an object that indicates the object surface and other useful properties that can be measured. Such a near and far non-contact remote sensing method is drawing a great deal of research attention because of a wider range of scale and application.

Any remote sensing data can be categorized into several ways; and the most common classification is based on the region of the electromagnetic spectrum, such as visible, infrared, and microwave. Among the variety of sensors used in contactless remote sensing, a hand-held color digital camera is a commonplace and most simple optical sensor, which works in the visible range, and is extensively used because of its low cost, high resolution, and better efficiency when compared to other sensors (Bechar & Vigneault, 2016). Digital cameras readily produce the image of the objects of interest, which is considered as input data, and analyzed through "image processing" operations to derive meaningful results based on the application. Even though the input data (images) are obtained easily, the image processing stage is highly technical and challenging in producing

the deliverables (outputs). However, the image processing methodology can able to provide a variety of solutions and can be tailor-made to a specific application.

1.2. Advantages of Imaging Technology

At present, several of the agricultural management decisions are taken based on the manual field scouting through the visual appearance of the plant, as they directly indicate the plant growth and health conditions. Most of the plant conditions, such as foliage wilting, pest infestation, nutrient deficiency, and plant disease provide visible clues (A. Camargo & Smith, 2009). Those variations must be identified early to take necessary precautionary or management actions for efficient crop production. These clues are observed and quantified through visual inspection during field scouting by trained professionals (e.g, experienced farmers, agronomists, pathologists, crop inspectors) following a standard protocol. However, inspecting the entire field and recording the observations manually is a subjective, tedious, and time-consuming task. Also, direct inspection causes visual fatigue and measurements become more prone to error.

Even though the visual inspection is the basic and natural method, as well as serves as ground truth, the method has several limitations: (i) minor changes in leaf color, shape, and texture cannot be discriminated (Tillett, 1991); (ii) similar symptoms will be produced by different stresses (Barbedo, 2013); (iii) some stresses (e.g., foliage disease, nutrient deficiency) might need laboratory analysis (destructive) to confirm; (iv) some stresses require an actual severity estimates (e.g., pest count, area infested) to make decisions (Bock, Poole, Parker, & Gottwald, 2010); (v) changes in morphological traits such as canopy area, and plant volume cannot be measured visually (Golbach, Kootstra, Damjanovic, Otten, & van de Zedde, 2016); (vi) the human eye can only distinguish between 2.3 and

7.5 million colors depending on the individual's potential, while computers distinguishing up to 16.7 million (256 × 256 × 256) colors (Pointer & Attridge, 1998); and (vii) distinguishing among various shades of same color is highly challenging for humans.

To address these issues, modern agriculture moves towards an objective approach to overcome the aforementioned drawbacks (McCarthy, Hancock, & Raine, 2010). Advancements in digital imaging technology and the processing power of computers have led to the use of digital images for detecting the variations rapidly through image processing methods (Y. Zhao, Gong, Huang, & Liu, 2016). Therefore, the image processing methods can play a significant role as the approach is objective, quick, reproducible, easy to apply, enable real-time in-field measurements, support arriving management decisions, and allow for archiving results.

1.3. Current Status of Imaging Technology in Agriculture

In a study conducted among agricultural producers of Nebraska, Castle (2016) found that the technology adoption rate of imagery ranked 8 out of 10 technologies used in precision agriculture (fig. 1.1). However, there is a steady increase, but the adoption is at a slower pace because most of the farmers are not able to process the images on their own, which demands high technical knowledge.

Already, many image processing methods were developed and used in several agricultural activities. Out of many image processing methods developed, the following two significant agriculture field operations were successfully deployed. First, the automatic guidance systems in tractors, which were operated based on global positioning systems (GPS), and were now operated based on image processing algorithms as they are efficient, inexpensive, and accurate (Garcìa-Santillán, Montalvo, Guerrero, & Pajares, 2017).
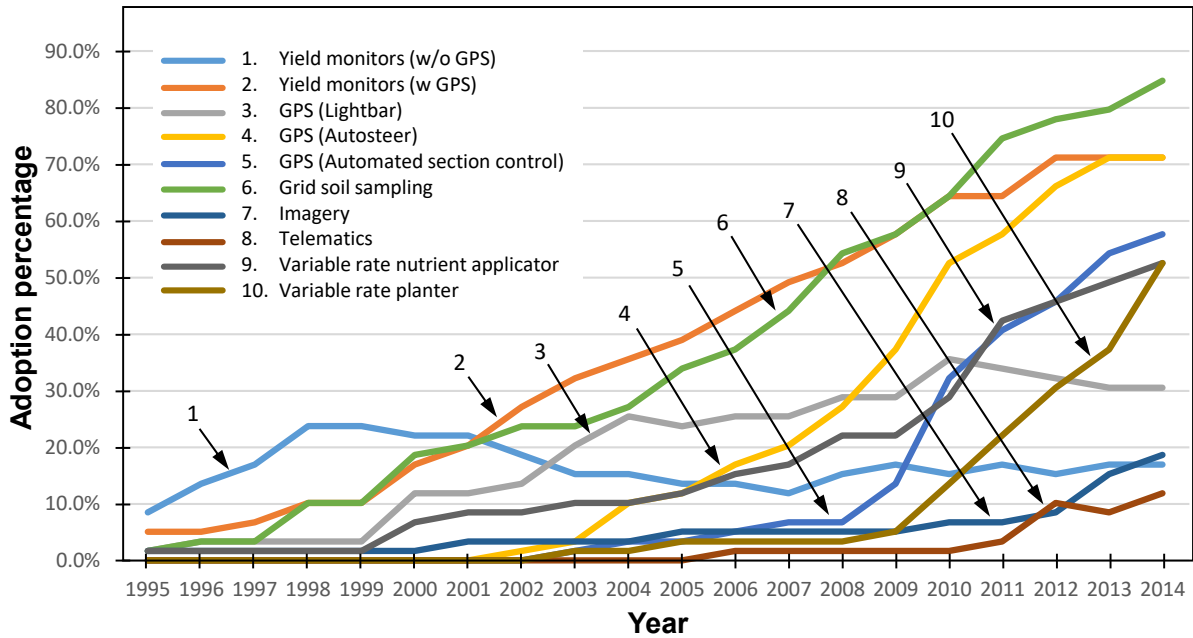
Figure 1.1. Adoption percentage of various technologies by agricultural producers in Nebraska between 1995 and 2014 (Castle, 2016).

Second, the fruit harvesting operation, which is seasonal, labor-intensive, and expensive, and is now replaced with automatic harvesters that perform operations, such as branch shaking, fruit harvesting, leaf pruning, and fruit sorting based on image processing (Amatya, Karkee, Zhang, & Whiting, 2017). Apart from these developments, there are several applications of image processing in agriculture that are being developed and validated under various test conditions.

A digital image used for image processing should possess good clarity, resolution, and focus. Acquiring images under a laboratory condition (more controlled environment) is favorable for processing, as necessary arrangements for acquiring well-focused images at high resolution can be employed in the laboratory setting. But, working with images acquired in field conditions is challenging due to the variations in lighting, improper focus, and the presence of artifacts (Barbedo, 2016). Therefore, processing such images require

5

a reliable image processing methodology that is flexible to work in a range of imaging conditions. For the past 25 years, the proportion of peer-reviewed journal publications on image processing applications under laboratory conditions was more than the field condition (fig. 1.2). On average, the contribution to field applications of image processing was <35 % of total publications in each year. This shows the potential research gap for developing flexible image processing methodologies for agricultural applications applied to field conditions.
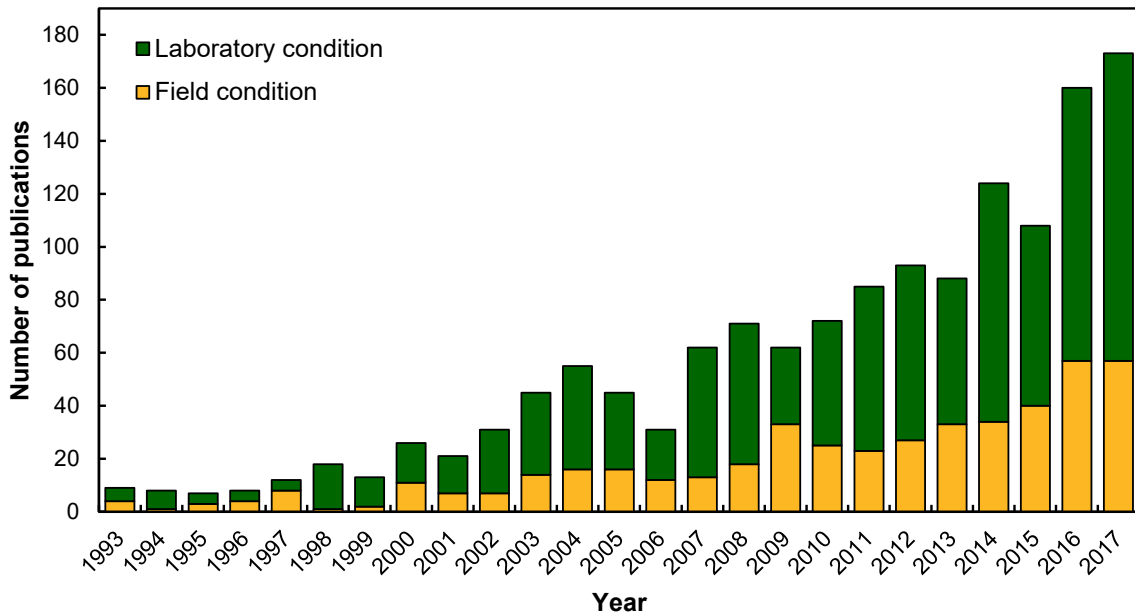
Figure 1.2. Number of peer-reviewed journal articles in past 25 years on image processing applications in agriculture performed in laboratory and field conditions. Data retrieved from Web of Science Core Collection through advanced search with following keywords: "TS = (Agri* NEAR/3 lab*) AND TS = (Image* OR Vision*)" for laboratory condition; and "TS = (Agri* NEAR/3 field*) AND TS = (Image* OR Vision*)" for field condition.

1.4. Image Processing Software — Commercial and Open Source Platforms

To process the images, a software capable of reading the images, extracting the digital information, and manipulating it mathematically is necessary. There were

≈75 image processing software packages available today (Wikipedia, 2018), out of which, most of them were commercial software, and were operated manually only by 'click-and-run' feature. This might be favorable for processing a few images with only limited operations, but to process a batch of images and perform customized operations in a sequence, a software with programming module will be necessary. Any image processing software with a programming module enables us to customize and automate the image processing operations, with the features inspired by the human interaction and simulating manual measurements, and program it to deliver meaningful outputs. For example, an image processing method developed using MATLAB (MathWorks®, 2017) counted aphids on soybean (*Glycine Max* L.) leaf and displayed whether the aphids population was below/above the economic threshold (Sunoj et al., 2017), which can be used for pesticide application decisions by the producers.

Commercial software such as MATLAB with specialized image processing tool-packs offers several user-friendly features for program development. This commercial software and specialized programs and were generally used in academic institutions and specialized industries and were generally out of the reach of the common user because of the high cost and license charges. However, an open source and no-cost image processing software system with programming capabilities will be highly demanding technically. Working with open source platform requires a deeper understanding of the operations and good programming skills to develop customized programs (plugins/tools) for specific user-defined applications (Landset, Khoshgoftaar, Richter, & Hasanin, 2015).

An internet search on "open source image processing software" would return several software systems; however, "ImageJ" and "OpenCV" would form the top hits. These image

7

processing systems with advanced capabilities match or surpass the available commercial software and are being used by many researchers worldwide. Between these two, OpenCV is an image processing library, where it has to be executed with an external integrated development environment (IDE) and does not support menu-driven manual operations. While ImageJ has an integral IDE, Java-based, public domain, platform-independent software, which has various image processing operations that can be accessed manually from toolbar and menu options and provides automatic running capabilities through macro and plugin coding (Rasband, 2018). The scripting languages are ImageJ Macro (IJM) language for macro development (Mutterer & Rasband, 2015), and Java language for plugin development (Bailer, 2006; Burger & Burge, 2016b). ImageJ also supports Javascript language for macro and plugin development.

ImageJ plugins are developed for exclusive applications and have been explored extensively in processing medical images (Y. Chen, Yu, & Xu, 2017; Khan, Nag, Igathinathane, Osuagwu, & Rubini, 2015), but only a few literature on its application in agriculture were available. Some of the earlier macros/plugins developed in agricultural applications include biomass particle size distribution determination (Igathinathane, Pordesimo, Columbus, Batchelor, & Sokhansanj, 2009c), biomass node and internode identification (Pothula, Igathinathane, & Kronberg, 2015), cashew whole and split classification (Sunoj, Igathinathane, & Jenicka, 2018a), rapid chickpea phenotyping (Sankaran, Wang, & Vandemark, 2016), biomass yield estimation (Ali, Streibig, Christensen, & Andreasen, 2011), and root dimensions measurement (Borianne, Subsol, Fallavier, Dardou, & Audebert, 2018). It was considered that the use of open source software packages is the future of agriculture because it brings innovation by collaborating

across multidisciplinary communities (computer science, mechatronics) to work together and contribute towards addressing the issues of the growing population (Ault, Krogmeier, & Buckmaster, 2018). With such open source software in combination with the user-developed plugins, the farmer can use their data, without the data/software privacy issues, analyze them, and make informed farm management decisions.

1.5. Problem Statement

This study focused on developing individual image processing tools (plugins) suitable for the above-mentioned applications in agriculture to replace the current manual, destructive, laborious, and subjective methods. The plugins will be developed using Fiji software (Rasband, 2018), an open source image processing software package. Fiji is a distribution of ImageJ that has core ImageJ plugins and extends additional functionality for creating and modifying the plugins (Schindelin et al., 2012). The plugins were developed in Java language utilizing appropriate APIs. The developed plugins in this study were tested on selected specific field crops; however, the underlying algorithms can be applied to any crop. Therefore, the developed plugins can be easily extended to other crops with minor modifications in the program.

Considering the advantages of image processing methods over manual methods, several areas in agriculture can be addressed efficiently through the application of image processing. Based on the field requirements for modern applications and problems already existing in current methodologies, a few specific issues (presented subsequently) were chosen and proposed to be solved in this research.

First, the sunflower floral components dimension has a direct influence on pollinators' visitation rate and this has been reported in many studies (Mallinger &

Prasifka, 2017a, 2017b; Portlas, Tetlie, Prischmann-Voldseth, Hulke, & Prasifka, 2018). In these studies, the dimensions were measured manually using a thread and ruler. This method is subjective, inefficient, and cannot measure geometrical parameters (morphology). Therefore, there exists a need for an objective method that can accurately measure the dimensions along with other geometrical parameters. This method will be useful for entomologists to study useful relationships between floral morphology and pollinator visitation rates. Results will also enable the farmers to predict the yield based on accurate head dimensions using the yield prediction equations (Kandel, 2018).

Second, the changes in ambient lighting, time, and camera directly influence the image color intensity and using such images directly to perform the phenological comparison and infer plant health will be misleading (Andresen, Tweedie, & Lougheed, 2018; Crimmins & Crimmins, 2008). Those images need to be calibrated before being used for phenological comparisons; however, there are no studies that scientifically reported an efficient calibration methodology. Therefore, an efficient calibration methodology to minimize the color intensity variations from the images obtained at varied lighting conditions that occur in the field is desirable. This method will help farmers/users to produce calibrated images that can be used for further interpretation and analysis.

Third, the plant stand count and spatial distribution are important to estimate the plant population and evaluate planter's performance on seed placement and spacing (Bruin & Pedersen, 2004; Lauer & Rankin, 2004). Nowadays, these two measures are being evaluated from UAS images employing commercial software (Lamb, 2016). The available software imposes several requirements and they are expected to work accurately only in the "best case" scenario with rows properly aligned and without overlapping plants (plants

cluster). But, it is practically challenging to manage field practices so that the stringent restrictions of the software can be fulfilled on the field. Therefore, a simple yet novel methodology using basic geometrical principles to perform stand count and spatial distribution analysis were developed and such application was expected to be user-friendly and helpful to the users/farmers.

Fourth, the timing of phenological stages of a plant is one of the essential information for making farm management decisions. A digital camera network called "PhenoCam" that produces time-lapse images at every 30 min is used to continuously track the phenological changes based on the variations in color vegetation indices (CVIs) (T. B. Brown et al., 2016; Hufkens et al., 2012). But, not many studies have attempted to use PhenoCam images for agricultural applications and the information on the effects time of image capture and camera distance on phenological trend is lacking. Therefore, a study on determining a set of the best CVIs that represented the natural smooth phenological trend and evaluating the effects of time of image capture and PhenoCam distance in CVIs would lead to consistent and efficient use of PhenoCam images for agricultural applications.

The hypotheses of this study are the developed image processing methodology and the associated specific user-defined plugins using open source ImageJ platform for agricultural field applications can (1) measure the sunflower floral dimensions simulating manual measurements, (2) calibrate the digital color images to minimize image color intensity variations and determine the least number of color patches required for calibration, (3) estimate the plant stand count accurately to match with visual count and evaluate the spatial distribution from the stand count data, and (4) determine the

phenological stages of soybean through PhenoCam images and estimate the effect of time of image capture and camera distance on phenological trend.

1.6. Statement of Objectives

The specific objectives of this study were to:

1. Develop an efficient method to measure the dimensions of the field sunflower floral components and derive models to obtain accurate dimensions simulating the manual measurements.

2. Develop a color calibration plugin to standardize the image intensity variations and test its performance with images obtained from different sources at varied lighting conditions in laboratory and field conditions.

3. Develop a simple and automated plant stand count and spatial distribution plugin that works in plant rows at any orientation and resolves plants cluster.

4. Derive various color vegetation indices using PhenoCam images, select the best indices that represent the phenological stages of soybean, compare with manual observations, and evaluate their effect on the time of image capture and the camera distance.

1.7. Dissertation Organization

The dissertation is generally organized into chapters that consist of (i) general introduction, (ii) review of literature, (iii) the individual objectives presented in the form of peer-reviewed journal articles as four research papers, (iv) general conclusions, (v) combined references, and (vi) appendices. The general introduction (this chapter) provides an orientation on the need for imaging technology, the importance of open source platforms for image processing, problem statement with hypotheses, and specific

objectives. The review of literature provides detailed background information on image processing, imaging platforms used, various applications of image processing in agriculture developed so far, details about importance of floral dimensions, issues in image intensity inconsistencies and available color calibration methodologies, manual plant stand counting and available commercial software and their limitations, and vegetation indices used in agriculture for monitoring plant growth. Each research paper presented here has its introduction with supporting literature review, materials and methods, results and discussion, and conclusions.

Among the individual studies presented, Paper 1 entitled: "Sunflower floral dimensions measurement using digital image processing" determined the efficient number of measurements required to best represent the floral dimensions (Objective 1); Paper 2 entitled: "Color calibration of digital images for agriculture and other applications" developed and tested a plugin to minimize the image intensity variations from the image acquired at varied lighting conditions in laboratory and field (Objective 2); Paper 3 entitled: "Profile-based plant stand count, plant vigor, and spatial distribution from UAS images" developed a plugin that used pixel profile and geometrical principles for row identification and plant cluster segmentation to count plants, assess growth, and determine spatial distribution from UAS images (Objective 3); and Paper 4 entitled: "PhenoCam application in agricultural cropping environment: Time and distance effect on color vegetation indices of soybean phenology" derived CVIs (existing and new) to study the phenology of soybean field crop, selected the best set of CVIs, compared with ground truth measurements, and evaluated the effect of time and distance on phenological trend (Objective 4).

A general conclusion summarizing the results derived from the chapters is then presented and is followed by suggestions for future work. The reference listing of all the citations in the chapters (including the papers) was presented as a combined unnumbered reference chapter. Appendices (A–D) present some of the data, pseudocode of the plugins developed, applications of PhenoCam, and PhenoCam images downloading procedure.

2. REVIEW OF LITERATURE

This chapter provides an overview of image processing applications in agriculture using various imaging devices and practical problems associated with the agricultural field applications. Research gap and importance of specific agricultural field applications such as sunflower floral dimensions measurement, image color calibration, plant stand count and spatial distribution analysis from UAS images, and application of PhenoCam in agricultural environment for soybean growth monitoring and analysis are explained in respective sections.

2.1. An Overview of Image Processing and its Applications in Agriculture

The concept of image processing originated to address the following two applications: (1) to enhance the quality of the image (e.g., image editing, computer graphics), and (2) to extract useful information from images and transmit data for automatic machine operations (e.g., machine vision, computer vision). The latter application falls in the branch of science where user-coded algorithms are developed to perform sequence of image processing operations and produce useful outputs (Gonzalez & Woods, 2018). The former application falls in the branch of arts to improve the aesthetics of an image. The science application of image processing begun in 1960s to process an image of moon's surface transmitted from Ranger-7 satellite to study the dimensions and morphology of craters. Most of the developments in image processing algorithms evolved from space research using remotely sensed satellite images and its applications infiltrated into many fields of science and agriculture is one among them. Till date, optical remote

sensing (using imagery) is the most used technology in agriculture sector (Khanal, Fulton, & Shearer, 2017).

The remote sensing sensors used in agriculture can be identified based on the combination of four categories (fig. 2.1): (1) platforms, (2) energy source, (3) number of spectral bands, and (4) region of electromagnetic spectrum. The output from any of these sensor combinations is an image, which is composed of pixel elements (pixels) with digital numbers (e.g., reflectance, emissivity, back scatter value) that can be manipulated to extract useful information from an image. The choice of sensors combination are affected by factors such as spatial resolution (e.g, leaf- or canopy-scale), temporal resolution (e.g., hourly, daily, weekly), and spectral resolution (e.g., monochromatic, trichromatic, multispectral, hyperspectral). There is always a trade-off among these factors and the selection of sensors combination solely depends on the application. For example, for identifying pests on a plant leaf, image with high spatial resolution (leaf-scale) is suitable (<5 mm/pixel resolution); for detecting nutrient deficiency, image with medium spatial resolution (canopy-scale) is suitable (<0.10 m/pixel resolution); and for classifying crop diversity for a county, state, or a country, image with coarse spatial resolution is suitable (e.g., MODIS imagery: 250–1000 m).
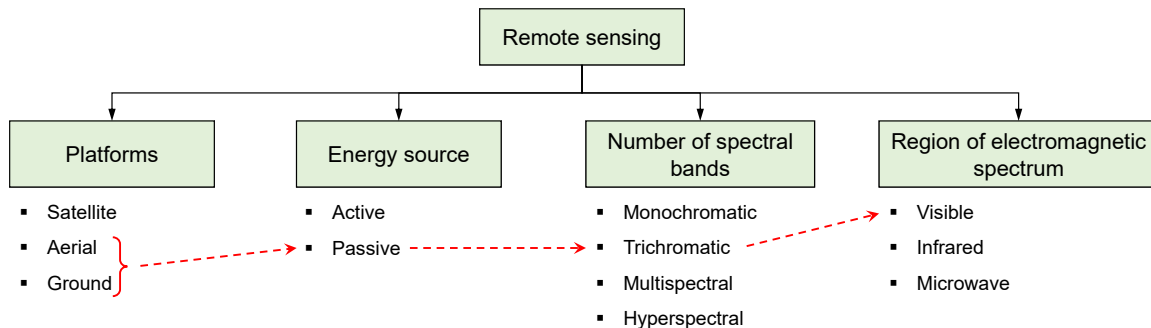
Figure 2.1. General classification of sensors used in the remote sensing area. Red dash arrows indicate the combination of categories used in this study.

16

Since the early 1970s, satellite images have been widely used in monitoring agricultural vegetation as satellites were equipped with wide range of spectroradiometers providing good spectral resolution. However, temporal and spatial resolutions were not suitable for agricultural applications because the temporal resolution for different satellites ranged from 1 d to 18 d, and spatial resolution ranged from 0.5 to 1000 m/pixel (Mulla, 2013). On the other hand, the satellite images are the only way to monitor and classify the land cover changes for a district, county, or a state that are being studied even now (Jenicka & Suruliandi, 2014, 2016). A review by Xie, Sha, and Yu (2008) provides a list of satellites launched for earth exploration, their spatial and temporal resolution, and the application of those images in various research fields.

Aerial imagery, obtained through unmanned and manned aerial vehicles, is used for field inspection of large fields with spatial resolution better than the satellite images. They are more suitable for canopy-scale decisions such as monitoring nutrient deficiencies (de Castro et al., 2018; Vega, Ramírez, Saiz, & Rosúa, 2015) and estimating plant stand count and plant density (Jin, Liu, Baret, Hemerlé, & Comar, 2017; Sankaran, Quirós, Knowles, & Knowles, 2017). Shi et al. (2016) reviewed the merits and demerits of different aerial systems, their operations, and various applications in the agronomic research. The images from 'passive' energy source were sufficient for many of these applications. Besides these, agricultural applications using 'active' energy source were also well explored using "light detection and ranging" (LiDAR) (Simard, Pinto, Fisher, & Baccini, 2011), and "radio detection and ranging" (RADAR) (Hirt, Filmer, & Featherstone, 2010) by producing digital elevation models and digital terrain models. The low-altitude aerial imagery is useful in obtaining the bird's eye view of the entire field and can be used for spatial mapping.

Similarly, the ground platforms (near-surface remote sensing) produce high resolution images that are suitable for making decisions on leaf-scale resolution (e.g., pest counting). The higher resolution images are obtained using cameras held near the plant canopy, by which every fine details can be captured. The images are captured manually during field scouting or mounted on front- or back-side of a tractor and the plants are monitored on-the-go (Deery, Jimenez-Berni, Jones, Sirault, & Furbank, 2014; Y. Jiang et al., 2018). The integration of computer vision and robotics enable real-time detection (by computer vision) and application (by robotics) for a subjective, quick, and efficient field management decisions. For such applications, ground platforms are being used by various agricultural technology industries (e.g., Climate FieldView, Blue River Technology) for plant specific managements. In particular, the 'See & Spray' technology developed by Blue River Technology (2018) uses computer vision and machine learning algorithms to identify weeds and they are immediately sprayed with herbicide on-the-go.

Therefore, working with aerial and ground platforms is highly demanding and has possibility for spatial mapping and real-time implementation. In this study few applications in agriculture, which are now performed manually and using commercial software, were identified from the reported literature and user-coded computer algorithms for performing specific applications were developed from scratch using an open source image processing software, ImageJ. The images used in this study are high resolution images acquired using a color digital camera, UAS, and PhenoCam in open field condition; hence, the sensor combination used in this study can be identified as 'aerial' and 'ground' platforms that uses 'passive' energy source with 'trichromatic' spectral bands that are sensitive to 'visible' region

of electromagnetic spectrum (red dash arrow; fig. 2.1). The details of the specific applications and their importance are discussed subsequently.

2.2. Sunflower Floral Dimensions Measurement

Sunflower (*Helianthus annuus* L.) is one of the major oilseed crops in the world that is a native of North America. The oil from sunflower seed is being used for different purposes such as food preparations due to the high amounts of unsaturated fatty acids (Elghandour et al., 2017), cosmetics as a base oil that prevents hair loss (Rele & Mohile, 2003), and biodiesel as due to high oil production (Demirbas, 2007). The seed production in sunflower is mainly by the process of pollination, by which pollens produced in stamen (male part) are transferred to pistil (female part) through pollinators (e.g., bees, butterflies) that visit flowers to consume nectar. Parker (1981) studied the effect of two kind of bee such as oligolectes (bees that visit specialized flowers) and polyleges (bees that visit diversified flowers) in pollination of sunflowers and found the oligolecets carried more pollen on their body hair and resulted in more seed production.

Some of the major reasons for pollinator attraction are based on flower size, shape, and floral display. A common tendency exists on pollinators to get attracted for large flowers than small flowers, unless small flowers are present as a bunch (Dafni & Kevan, 1997). A reasonable amount of literature are available on the effect of flower size on pollinator attraction on different species. Conner and Rush (1996) found a positive correlation between flower size and syrphid flies (pollinator) visitation rate in wild radish (*Raphanus raphanistrum*) species. Schmid-Hempel and Speiser (1988) classified the inflorescence size of *Epilobium angustifolium* species into large (4.3 ± 0.2 mm) and small (3.2 ± 0.1 mm) and their effect on pollinator visitation rate was studied. The results

showed, there was no significant difference on pollen export and pollen deposition between the inflorescence size categories; however, the large flower category attracted diverse pollinators. The size of the floral display has been shown to be important in attracting pollinators in other plants (Mitchell, Karron, Holmquist, & Bell, 2004) and is, therefore, likely a key variable for sunflowers as well.

To improve sunflowers as an agricultural crop and for other uses, researchers have tried to understand the attraction of pollinators. Greenleaf and Kremen (2006) compared the effect of wild bees and honey bees in pollination to produce hybrid sunflowers. The results showed that there was a 5-fold increase in the pollination efficiency with wild bees than honey bees. The bee visitation rates are also influenced by the time of the day. In a study conducted in the sunflower fields in University of Arkansas by Posey, Katayama, and Burleigh (1986), found that peak visitation rate was observed between 0800 and 0900 h with ≈58 bees per head per h. Therefore, farm management practices such as insecticide and herbicide applications during this time of the day should be avoided and can be applied between 1800 h and dark during which the pollinator visitation was almost zero. Various studies have suggested plant traits that are associated with pollinator visits to sunflower include pollen, floral coloration, floral odor, nectar rewards and size of the disk florets (Atlagić, Joksimović, Sakač, Miklič, & Dušanić, 2003; Mallinger & Prasifka, 2017b; Moyers, Owens, Baute, & Rieseberg, 2017).

Currently, the dimensions of the floral components are measured by manual methods, which are subjective, time-consuming, and inaccurate. For example, P. Li, Luo, Bernhardt, Kou, and Perner (2008) used digital vernier caliper to measure the dimensions of dorsal sepal, lateral sepal, and labellum on *Cypripedium plectrochilum* species, and

Kulbaba and Worley (2012) measured the floral dimensions on *Polemonium brandegeei* species. A mixture of in-field measurements and images of plant parts were performed using a ruler or micrometer (Mallinger & Prasifka, 2017a). The sizes of the sunflower disc, ray florets, and ultraviolet (UV) pigmentation regions were also manually measured from images using the 'line tool' of ImageJ (Moyers et al., 2017). These types of measurements are necessary for linking pollinator visitation rates to the floral traits that attract them, but manual field methods are time-consuming and often result in limited replication.

Apart from these applications, sunflower dimensions are measured in gardening contests (Giant Gardening, 2018). The measurements followed in the contest include head diameter, disc diameter, and disc perimeter as shown in fig. 2.2 using measuring tape, which is highly subjective. Predicting the sunflower yield through the equation developed by Kandel (2018) require precise head diameter as one of the factors along with plant population and seed size.

Head diameter	Disc diameter	Disc perimeter

Figure 2.2. Sunflower dimensions measurement strategy in the giant gardening contests (Giant Gardening, 2018).

Mirzabe, Chegini, and Khazaei (2018) measured length, width and area of the sunflower petals using an image processing technique in MATLAB; however, they used harvested sunflowers that made the measurements destructive. The studies discussed so far

did not consider the geometry of the flower on pollinator visitation rate as they are not possible with manual measurement methods. Also, limited measurements (two orthogonal measurements) were taken in all these studies that may not statistically represent the dimensions of the sunflower floral components. Therefore, a nondestructive, quick, objective, and accurate dimensions measurement method with the potential of measuring the geometry of the flower is highly desirable to correlate with pollinator visitation rate and estimate sunflower seed yield.

2.3. Color Calibration of Digital Images

The digital images obtained from the camera uses a camera sensor that consists of a grid of pixels with each pixel being sensitive to specific visible wavelengths (Bayer filter, fig. 2.3a). The sensitivity of the sensor was developed by mimicking human vision system, which is more sensitive to green. Therefore, half of the pixels are allotted with green (G) sensitivity and another half is shared among red (R) and blue (B) equally (Lee & Ortega, 2001). The number of grids in Bayer filter decides the image resolution. For example, the filter shown in fig. 2.3a produces an image of 64 pixels resolution (8×8 pixels). The color filter layer selectively allows the specific visible wavelength and the amount of photons hitting each pixel forms the intensity. As seen in the fig. 2.3a, each pixel contains single intensity value (R, G, or B), but the image we obtain contains three intensity values for each pixel (R, G, and B). The missing intensity values at each pixel location are interpolated by a process called demosaicking (Gunturk, Glotzbach, Altunbasak, Schafer, & Mersereau, 2005) that is performed in the 'Image processor' (fig. 2.3b). Therefore, a digital color image is represented in RGB form with three components per pixel in the range of 0 to 255.

a Bayer filter

8 pixels   8 pixels

Incoming light

Filter layer

Sensor array

Resulting pattern

b

Lens   Bayer filter   Sensor   Image processor   Sensor RAW   JPEG output
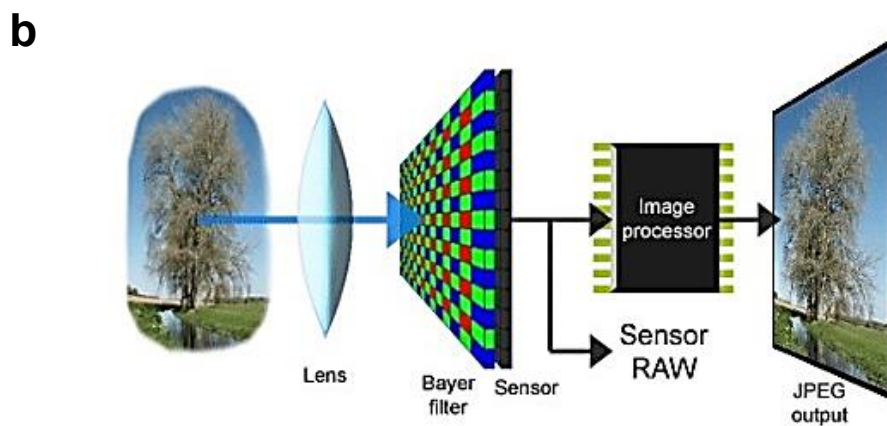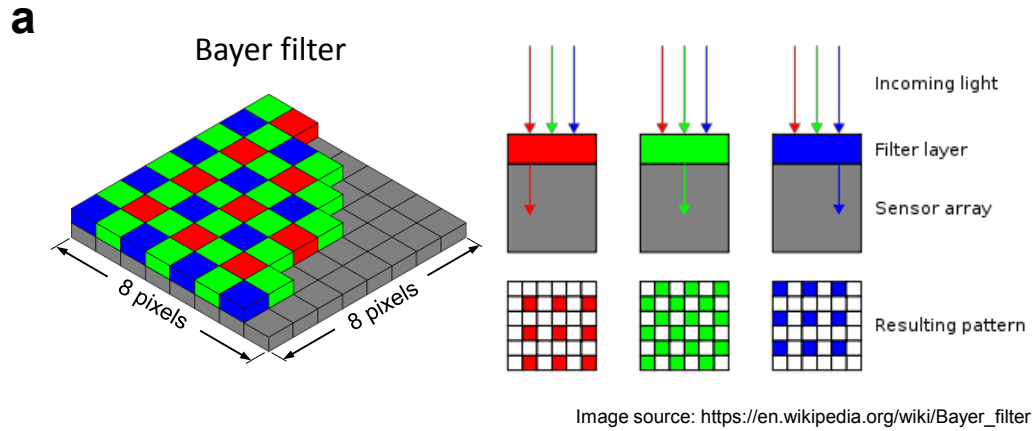
Figure 2.3. Working principle of a digital camera: (a) Bayer filter pattern of pixels arrangements showing selective visible band sensitivities, and (b) the image acquisition pipeline in a digital camera.

From fig. 2.3, it is obvious that the color presented in an image is directly generated by the camera sensor. Therefore, the image color is device-dependent, i.e., a scene captured using different cameras at the same time produce color variations due to the different camera sensors. Furthermore, the images captured on same day at different times, or same time on different days also produces color variations, despite no change in the object properties fig. 2.4. These variations are due to either ambient temperature, light irradiance, or cloud cover at that time. Obtaining image inferences directly from such images are

23

June 22, 2016, 11:00	June 22, 2016, 11:30	June 22 2016, 12:00	June 22, 2016, 12:30	June 22, 2016, 13:00

June 18, 2016, 12:00	June 19, 2016, 12:00	June 20, 2016, 12:00	June 21, 2016, 12:00	June 22, 2016, 12:00

Figure 2.4. Comparison of color variations from the images captured using single camera on same day at different times and same time on different days.

misleading. However, in image processing applications, the color response from the input image has to be corrected to obtain color values through a process called color calibration.

A common way of obtaining the true color practiced in professional photography is by using 'white balance' and 'gray balance' techniques, and they are followed in agriculture as well. White balance technique was used to consistently maintain image color for discriminating between plant and soil through different color channels (Philipp & Rath, 2002), estimating nitrogen status from digital images Y. Li, Chen, Walker, and Angus (2010), and guiding the automated rice planter (B. Chen, Tojo, & Watanabe, 2003). Grey balance technique was used in machine vision system to classify potatoes and sugar beet (Nieuwenhuizen, Tang, Hofstee, Müller, & Van Henten, 2007), in monitoring the changes in canopy phenology spatially and temporally (Richardson, Braswell, Hollinger, Jenkins, & Ollinger, 2009), and in unmanned aerial system to estimate green area index for wheat and rapeseed crops (Verger et al., 2014).

Using only two colors (white and gray) will only alter the brightness of the image and will not render true color of the scene; however, these techniques are followed as a

simple approach and cannot be claimed as actual color calibration. The actual color calibration can be broadly classified into two categories: (i) device-based, and (ii) image-based calibration. Many researchers have explored device-based color calibration in which camera sensors are developed to automatically adjust to the color values based on the scene lighting (M. Brown, Majumder, & Yang, 2005; Porikli, 2003). Image-based color calibration is used to compensate for changes in the image due to illumination conditions. This is the most common method followed in professional photography to maintain the same color tone in all the scenes. It uses a standard color calibration chart to linearly map the pixel values obtained in the image to the standard color values of the calibration chart (Kagarlitsky, Moses, & Hel-Or, 2009). A relationship is developed between the color values obtained in the image to the standard color values, which can then be applied to image pixels (calibration). Commercially, there are several calibration charts available (ranging from 6 to 100 s of color patches); however, the 24 color patch calibration chart is widely used in many applications (McCamy, Marcus, & Davidson, 1976).

Many fields of study i.e., space research (Levin & Levin, 2004), medical research (Haeghen, Naeyaert, Lemahieu, & Philips, 2000), underwater biology (Akkaynak, Chan, Allen, & Hanlon, 2011), and agricultural research (Scharf & Lory, 2002) have had to solve color variation issues. However, comprehensive information on color calibration of images related to agricultural applications is limited. L. Brown, Datta, and Pankanti (2012) used hue, saturation, and intensity (HSI) color space and derived a parameter called color strength, which can be used to improve color correction. Akkaynak et al. (2014) developed a color calibration using XYZ color space and all 24 color patches in the calibration chart and converting it to XYZ color space, and later back-transforming to RGB color space.

Over the last decade, application of digital image processing in agriculture has been well researched. The major application in the agricultural research is mainly focused on color based identification/classification. The usage of unmanned aerial system, and PhenoCam systems in agriculture has taken a giant leap into the usage of digital color images to assess plant phenology and health. Since the images are taken in open atmospheric conditions, the brightness and darkness caused by the clouds and the time of the day result in wide color variations. Barbedo (2016) reported lighting variations during image acquisition in agricultural fields is the most common issue and it is inevitable. Thus, the color-based characteristics gathered from such digital images that are used for crop stress, nutrient deficiency, and crop productivity would lead to inaccuracies.

Researchers generally use various calibration boards with color patches of known color value in the agricultural field. A few studies on calibration boards used in agricultural applications include whiteboard in the test sites of Lageren Mountains in Zurich, Switzerland (Ahrends et al., 2008), gray painted board in the Bartlett test site (Richardson et al., 2009), five color boards, such as red, green, black, gray, and beige used for estimation of biomass and nitrogen levels in a corn field (Hunt, Cavigelli, Daughtry, Mcmurtrey, & Walthall, 2005). However, the color rendering efficiency of these color patches was not investigated, and the calibration accuracy and the methodology followed in those studies remain unknown.

2.4. Plant Stand Count and Spatial Distribution using UAS Images

Plant stand count is an important piece of information for farmers early in the growing season, since that can be an important factor when making crop management decisions (Bruin & Pedersen, 2004). When using unmanned aerial system (UAS) to assess

plant stand information, farmers have the option of using one of many cloud-based services that offer algorithms to perform stand counts on UAS imagery. Although that can be an easy route to go, such services can be expensive and might have some requirements to work properly, such as: (i) uniform plant-to-plant spacing, (ii) image resolution of or lower than 25 mm/pixel, (iii) consistent row spacing with parallel rows, (iv) uniform plant growth stages, and (v) no overlap between individual plants.

The stand count measure is important and is used to (i) estimate whether target plant population can be achieved, (ii) evaluate the planter performance on seed placement and spacing, (iii) derive insights on the emergence characteristics of the planted seeds, and (iv) assess damage due to frost and hail storms. Plant stand assessment is manually performed by scouting the fields during which either the row length (fig. 2.5a) or hula hoop method (fig. 2.5b) are employed (Stowe, 2018). In the row length method, a measuring tape of known length is placed in the row direction and the plant stand within the length is counted manually. The stand count is then scaled up using crop-specific constants (D. Berglund, 2007; DuPont, 2015; Kandel, 2010). While the hula hoop method is straightforward, by which the number of plants inside the hula hoop is manually counted and a rough estimate of plants per area is calculated. Then it is scaled up to the entire field area to estimate the approximate plant stand count.

Plant spatial variability in the field is also an important measure and reported to have a significant effect in the final crop yield (Lauer & Rankin, 2004; Nielsen, 2004). Spatial distribution is important for the effective utilization of resources, such as nutrients and sunlight. It was reported that the plants next to the skip yielded more than control, while the plants next to a double yielded less than control (Nafziger, 1996).

27
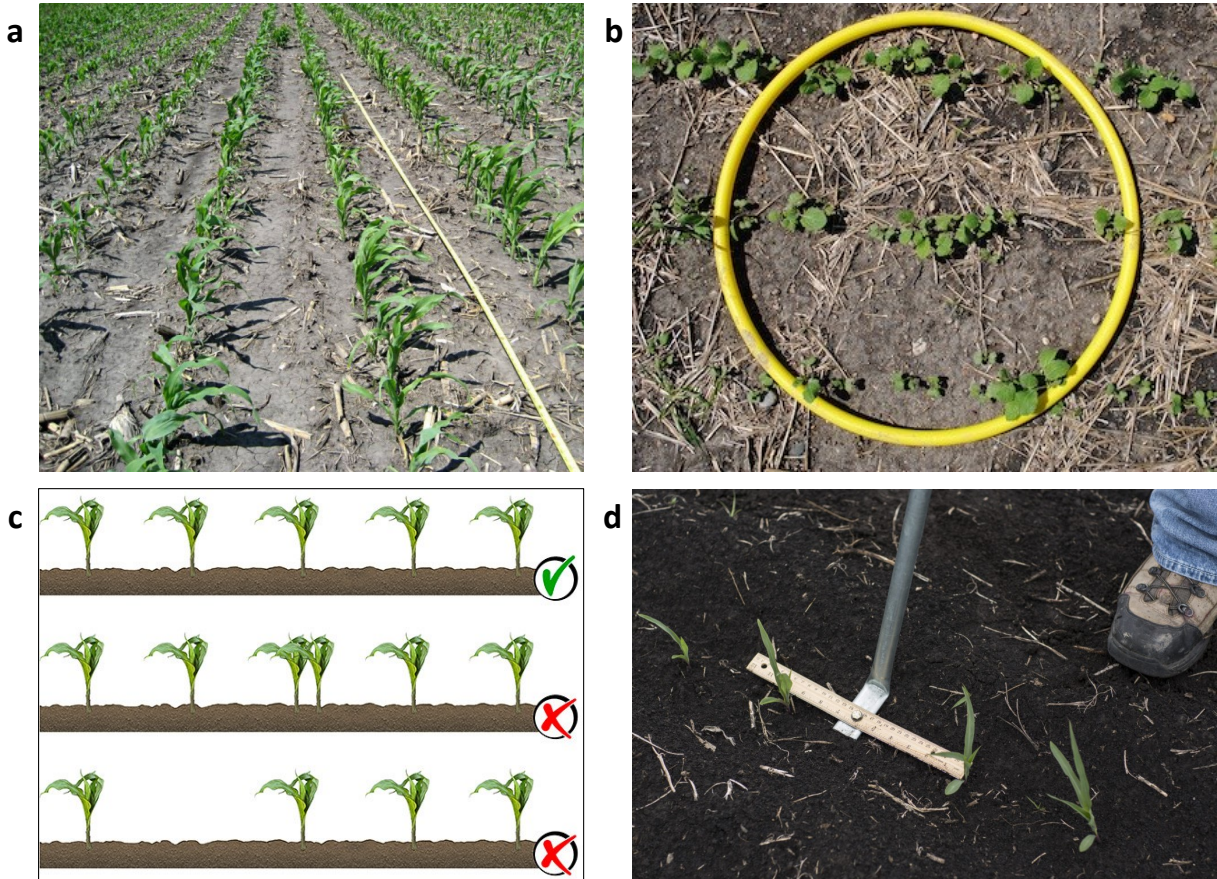
Figure 2.5. Plant stand count and plant spacing measurement in the field: (a) row length method for stand count (DuPont, 2015), (b) hula hoop method for counting plant stands (CanolaWatch, 2012), (c) possible plant spacing conditions in field with "ideal", "multiple", and "skip" (Väderstad, 2019), and (d) plant spacing uniformity measurement using ruler (Spelhaug, 2018).

Possible reasons for spacing variability (fig. 2.5c) between plants in a row, planter mechanism type and its efficiency, planting speed, soil moisture and temperature variability, and crop residue distribution (DuPont, 2015; Liu, Tollenaar, Stewart, & Deen, 2004). Generally, farmers scout through the field with a spacing ruler (fig. 2.5d) and check the uniformity in spacing at random locations. To achieve an effective plant population at uniform spacing, a farmer should have knowledge about his/her field history on stand count and spacing distribution.

It is obvious that counting the stands and measuring inter-plant distances manually through field scouting method is time-consuming, painstaking, and covers only a small portion of the field. Efforts were taken to develop automated plant stand counting using image processing through ground level imaging systems (D. S. Shrestha & Steward, 2003, 2005; Tang & Tian, 2008). However, these studies focused on counting only on a single crop row. Nowadays, field scouting method and ground level imaging method is being replaced by the use of UAS images as they are quick, easy, and covers the entire field.

Even though it is possible to manually count the plants from UAS images, which is prone to human error, automatic plant stand counting algorithms are developed through image processing. A support vector machine model was trained in MATLAB with 13 shape features to estimate the wheat crop emergence (Jin et al., 2017). Rapeseed stand count was estimated by simple correlation using 15 shape features of the plants in Trimble's eCognition developer software and achieved an R^2 of 0.84 (B. Zhao et al., 2018). A computer workflow in ArcGIS software to evaluate the sunflower stand count from UAS imagery attained R^2 ranging between 0.52 and 0.96 (Flores, Schatz, & Aberle, 2017). Stand count was performed in potato trials using ArcGIS from multispectral imagery and achieved a r of 0.83 between manual stand count and image-based plant count (Sankaran et al., 2017).

All these studies used a proprietary software, which are expensive, available only through educational institutions/agricultural consultants, needs high technical knowledge, and require license renewals, etc. Some of the software are cloud-based platforms, by which the user have to upload the image data to a server and the processing happens in the cloud. Such application is not encouraged by farmers, as they are concerned about data

security. In such case, an in-house and stand-alone application is expected to resolve these concerns as the data and the application can reside in the users' computer.

Until now, there is no software available that is easy to use, free of charge, and open source plant stand count solution that works across computer platforms, with few user inputs. Therefore, an algorithm using an open-source, free image processing software to produce a suite of results related to plant stand count and plants spatial distribution will be highly useful to users and farmers.

2.5. Monitoring Soybean Growth Stages using Remote Sensing Platforms

The launch of satellites to earth's orbit, since 1957, has initiated interest among researchers to launch satellites for better weather forecasting. During 1960s, a series of satellites were launched with advanced television cameras and radiometers on board. By then, a simple ratio of red and infrared bands was used for monitoring vegetation and termed as "vegetation index". Later, Kriegler, Malila, Nalepka, and Richardson (1969) developed a ratio between differences of red and infrared bands and sum of red and infrared bands that had shown good correlation to vegetation and termed as "normalized difference vegetation index" (NDVI). Soon after the launch of first earth resource satellite in 1972 (Landsat 1), several applications of NDVI are reported until now. Currently, there are around 1100 satellites revolving around Earth's orbit that still fascinates researchers to explore and develop new measures to monitor vegetation in a accurate way.

Typically, the vegetation indices (VI) are derived from the satellite spectral signals for environmental monitoring, forestry, agriculture, land use changes, and other related fields. In agriculture, these VI are used to measure and compare plant vigor, vegetation cover, growth dynamics, canopy structure, and photosynthetic activity on a spatial and

temporal scale (Mulla, 2013). Research have shown that 90 % of the vegetation information are obtained from red and near infrared bands of electromagnetic spectrum (Baret & Guyot, 1991). Hence, most of the VI developed to monitor vegetation uses a combination of red and near infrared bands. Some common examples are ratio vegetation index (Jordan, 1969), difference vegetation index (Richardson(AJ) & Wiegand, 1977), atmospherically resistant vegetation index (Kaufman & Tanre, 1992), and soil adjusted vegetation index (Huete, 1988). Xue and Su (2017) reviewed some of the significant VI used in agriculture. Currently, there are 519 VI available and a complete list can be accessed from `https://www.indexdatabase.de/`.

Remote sensing through satellite imagery was the major breakthrough for studying crop phenology. However, due to low image resolution, low temporal resolution, and obstructions caused by clouds, have led to the idea of developing near-surface remote sensing platforms. In 2007, a near-surface remote sensing network called "PhenoCam Network" was initiated by Richardson et al. (2007) through which several networked digital cameras (webcams) were installed across the United States to continuously track the vegetation changes at high temporal resolution. Currently, there are ≈ 526 cameras installed as of May 2018 that provides high resolution images at temporal resolution ranging between 30 min and 3 days (time-lapse images). All the installed digital cameras have panchromatic bands that produces digital color images with red (R), green (G), and blue (B) intensity values, and some cameras have infrared band.

The time-lapse images from PhenoCam network have been used to measure changes in landscapes (T. B. Brown et al., 2016), vegetation phenology (Hufkens et al., 2012; Richardson et al., 2018b), nutrient stress (Andresen et al., 2018), biomass content

and floral bloom (Crimmins & Crimmins, 2008), and patterns in plant leafing (Richardson et al., 2009). Until now, only four VI — green chromatic coordinates (GCC), red chromatic coordinates (RCC), blue chromatic coordinates (BCC), and excess green (ExG) are being used in the captured images from PhenoCam Network. These studies aimed in developing relationships between the above mentioned VIs and the satellite data, or manual observations of different phenophases (e.g., leaf unfolding, first flowering, full blooming, leaf falling).

However, other visible vegetation indices presented in table 2.1 were not used in any phenological studies from PhenoCam images. Also, all these studies were conducted on forest or grassland ecosystems, and did not explore much in agricultural crops. Therefore, this study aims to explore these VIs and develop a relationship between VIs and manual observations of vegetation growth stages.

Table 2.1. Vegetation indices using visible bands of electromagnetic spectrum developed and used in various studies.

Vegetation index	Mathematical expression	Reference
Green chromatic coordinates (GCC)	$GCC = G/(R + G + B)$	Richardson et al. (2007)
Red chromatic coordinates (RCC)	$RCC = R/(R + G + B)$	Richardson et al. (2007)
Blue chromatic coordinates (BCC)	$BCC = B/(R + G + B)$	Richardson et al. (2007)
Excess green (EXG)	$ExG = 2g_c - r_c - b_c$	Woebbecke, Meyer, Von Bargen, and Mortensen (1995)
Excess red (EXR)	$EXR = 1.4r_c - g_c$	G. E. Meyer (2011)
Excess green minus excess red (EXGR)	$EXGR = EXG - 1.4r_c + g_c$	N. J. Camargo (2003)
Normalized difference index (NDI)	$NDI = 128 \times ((G - R)/(G + R)) + 1$	Woebbecke, Meyer, Von Bargen, and Mortensen (1993)
Color index for vegetation (CIVE)	$CIVE = 0.441r_c - 0.881g_c + 0.385b_c + 18.78745$	Kataoka, Kaneko, Okamoto, and Hata (2003)
Vegitativen (VEG)	$VEG = g_c/r_c^{0.667}b_c^{0.333}$	Hague, Tillett, and Wheeler (2006)
Normalized green red difference index (NGRDI)	$NGRDI = (G - R)/(G + R)$	Tucker (1979)
Normalized green blue difference index (NGBDI)	$NGBDI = (G - R)/(G + B)$	Verrelst, Schaepman, Koetz, and Kneubühler (2008)
Modified green red vegetation index (MGRVI)	$MGRVI = (G^2 - R^2)/(G^2 + R^2)$	Bendig et al. (2015)
Red green blue vegetation index (RGBVI)	$RGBVI = (G^2 - BR)/(G^2 + BR)$	Bendig et al. (2015)
Visible atmospherically resistant index (VARI)	$VARI = (G - R)/(G + R - B)$	Gitelson, Kaufman, Stark, and Rundquist (2002)
Dark green color index (DGCI)	$DGCI = [((H - 60)/60) + (1 - S) + (1 - B')]/3$	Karcher and Richardson (2003)
Green leaf index (GLI)	$GLI = (2G - R - B)/(2G + R + B)$	Mounir, Michael, and Douglas (2001)
Red green ratio index (RGRI)	$RGRI = R/G$	Gamon and Surfus (1999)
Triangular greenness index (TGI)	$TGI = G - 0.39R - 0.61B$	Hunt, Daughtry, Eitel, and Long (2011)

Note: R, G, and B are red, green, and blue pixel values from RGB image; r_c, g_c, and b_c are the chromatic coordinates of RGB and is same as RCC, GCC, and BCC; H, S, and B' are hue, saturation, and brightness values from HSB image.

The PhenoCams were installed in 2015 on experimental plots of Northern Great Plains Research Laboratory (NGPRL), United States Department of Agriculture, Agricultural Research Service, Mandan, ND. Out of 576 PhenoCam sites, 63 sites are currently active and categorized into "Agriculture" as the primary vegetation type. Except Okalebo et al. (2017), no other work has been reported from these sites. In that study, a computer application was developed to detect two growth stages in corn such as green-up and tasseling in corn. Similar computer algorithms can be developed specific to the crop through the images from PhenoCam sites dedicated to agriculture vegetation. Such applications will develop interest among farmers to use PhenoCams as a tool to monitor vegetation continuously and take farm management decisions. The PhenoCam sites in NGPRL were sown with soybean (*Glycine max* L.) in the growing season of 2018, which was used to develop vegetation indices that indicate different growth stages as shown in fig. 2.6.

Soybeans are the main source of vegetable oil in the United States and largest source of proteins in animal feed. It is one of the important rotational crops as it improves the soil fertility. The productivity of soybean is affected by various abiotic factors (e.g., temperature, rainfall) and biotic factors (e.g., weeds, insects). Insect damage on soybean leaves impose stress to the plant, ultimately declines productivity. Defoliation is the major visible effect due to insect damage. Liang, Kirk, and Greene (2018) developed a computer algorithm to detect the amount of defoliation, and leaf area using digital color images obtained in field. They used leaf area and edge pixels to estimate the amount of defoliation and acheived an accuracy of up to R^2 of 0.96. An approach that can eradicate defoliation is to detect the insects on soybean leaves at earlier stage so that management decisions can be taken. Barbedo (2014) used image processing to identify white flies on soybean leaves.
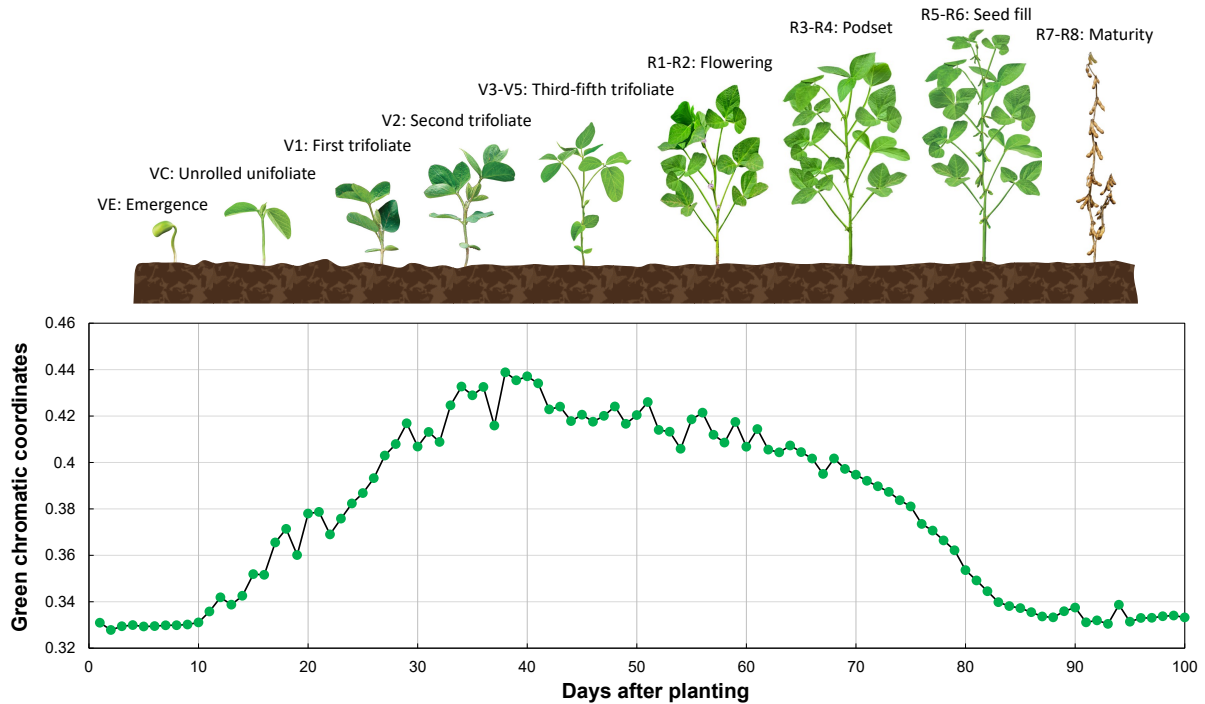
Figure 2.6. Growth stages of soybean (CropNutrition, 2018) and a typical green chromatic coordinate (GCC) curve at different days after planting of a crop in general.

Algorithms to identify and count aphids (*Aphis glycines*) on soybean leaves were developed using advanced shape analysis (Sunoj et al., 2017), and simple area comparison (Maharlooei, Sivarajan, Bajwa, Harmon, & Nowatzki, 2017).

The chlorophyll content of the soybean leaves are one of the indicators for root nodulation and nitrogen fixation status of the crop. An image processing algorithm was developed by Vollmann, Walter, Sato, and Schweiger (2011) to measure chlorophyll content of the leaves through digital images. The green channel and chlorophyll content from SPAD meter showed a strong negative correlation ($r = -0.94$). The relationship between chlorophyll content on nodulating and non-nodulating roots were also established. Root nodulation increased the chlorophyll content and non-nodulation decreased the chlorophyll content.

Concerted efforts have been taken to monitor soybean field through satellite platforms. Moderate-resolution Imaging Spectroradiometer (MODIS) satellite provides data at a spatial resolution of 250 m and temporal resolution of 1–2 d and this has been constantly used in remote sensing in agriculture (Mulla, 2013). Critical vegetative and reproductive stages of soybean were identified by Sakamoto et al. (2010) using a two-step filtering approach from the time-series MODIS data between 2003 and 2008 in three study sites of eastern Nebraska. A shape-model with Wide Dynamic Range Vegetation Index (WDRVI) data derived from MODIS was developed and validated with ground observations. The model estimated four key phenological stages of soybean (V1, R5, R6, and R7) with an error <7 d. The shape-model was improved in the saa me study sites by incorporating WDRVI with temperature and photoperiod data by Zeng et al. (2016) and reduced the error to <4 d. Zhong, Hu, Yu, Gong, and Biging (2016) used MODIS data to map soybean fields in Paraná state of Brazil by training a decision tree classifier and achieved an accuracy of 87.2 %.

Application of RapidEye satellite imagery was also explored by many researchers globally and it provides better spatial resolution ($\approx$5 m), temporal resolution ($\approx$daily), and enhanced spectral information due to the presence of red-edge band when compared to MODIS. Shang et al. (2014) used RapidEye images on corn, wheat, and soybean fields in southern Ontario, Canada to estimate plant area index (PAI) through NDVI and two-band enhanced vegetation index (EVI2). The PAI was most efficiently estimated from EVI2 with a high accuracy ($R^2 = 0.85$). RapidEye imagery was used to estimate leaf area index and biomass content in corn and soybean fields in Ontario, Canada between 2011 and 2013 (Kross, McNairn, Lapen, Sunohara, & Champagne, 2015). The vegetation indices such as

NDVI, red-edge NDVI, and green NDVI were tested and found that among other indices considered, NDVI performed well ($R^2 = 0.90$). Crnojević, Lugonja, Brkljač, and Brunet (2014) classified agricultural fields in a study area in northern Serbia into five vegetation categories (corn, wheat, soybean, sugar beet, and sunflower) using random forest classifier with an R^2 of 0.76.

3. SUNFLOWER FLORAL DIMENSION MEASUREMENTS USING DIGITAL IMAGE PROCESSING *

3.1. Abstract

Sunflower floral dimensions are essential for assessing pollinators attraction and estimating seed yields. Dimensions measured manually at present are subjective and time-intensive, therefore, an image processing method was developed as an alternative, which was objective, non-destructive, produces various outputs, and rapid. An ImageJ user-coded plugin with a field image acquisition method was developed to measure the dimensions of individual sunflower components, such as head, disc, and ray florets. Two measurement methods, direct (using the thresholded binary image) and wrapping-polygon (using a polygonal enclosure) were tested. The "pixel-march" method made multiple radial dimension measurements (diameter) on the ray florets binary image in a single computation. The effect of multiple measurements (2, 4, 8, 16, 32, 64, 128, and 180 along 0°–180° angles) was studied to determine an effective number of measurements, and user-friendly sunflower dimension prediction models from ImageJ's standard output parameters were developed. Results indicated that (i) a minimum of 32 measurements for the sunflower head and ray florets dimensions, but only eight measurements for the

sunflower disc, were necessary; (ii) wrapping-polygon method was efficient compared to direct; (iii) equivalent diameter (ED) and fitted ellipse minor axis (MinA) were well correlated ($r \geq 0.88$) to the accurate mean 180 measurements (D_{180}) for all sunflower components; (iv) linear models for predicting D_{180} using ED and MinA performed better ($R^2 > 0.99$) for head and disc than for ray florets ($R^2 > 0.76$); (v) user-friendly linear models using the mean of two manual measurements of the head (D_{2h}) for predicting D_{180} and area were good only for the head ($R^2 > 0.92$), and not suitable for disc ($R^2 \leq 0.62$) and ray florets ($R^2 \leq 0.43$); and (vi) the developed image processing method results were accurate, quick (≈ 11 s in Windows 10, Intel Core i5, and 8 GB RAM laptop), and have the potential to be adapted to other species.

3.2. Introduction

Cultivated sunflower (*Helianthus annuus* L.) is the world's fourth-ranked oilseed crop of economic importance (FAO, 2017). Sunflower yields are critically dependent on aspects of the sunflower head in at least two ways. First, head size as a component of yield is important as medium-sized sunflower heads are considered best for attaining a high yield (Škorić, 1992). Second, sunflowers depend on pollination by bees, which move pollen between male and female plants to produce hybrid seed, but bee visitation also increases yields of commercial hybrids that have moderate-to-high ability to self-pollinate (Mallinger & Prasifka, 2017b). Bees are attracted to sunflower heads by groupings of small central disk florets and larger outer ray florets (petals). The size of the floral display has been shown to be important in attracting pollinators to other plants (Mitchell et al., 2004) and is, therefore, likely a key variable for sunflowers.

Sunflowers are a component of the natural landscape in North America and are included in mixtures of seed for cover crops or plantings specifically to support pollinating insects. To improve sunflowers as an agricultural crop and for other uses, researchers have tried to understand the attraction of pollinators. Various studies have suggested plant traits that are associated with pollinator visits to sunflower include pollen, floral coloration, floral odor, nectar rewards and size of the disk florets (Atlagić et al., 2003; Mallinger & Prasifka, 2017a; Moyers et al., 2017; Pham-Delegue et al., 1990). Sunflower physical measurements usually rely on manual methods. A mixture of in-field measurements and images of plant parts were performed using a ruler or micrometer (Mallinger & Prasifka, 2017a). The sizes of the sunflower disc, ray florets, and UV pigmentation regions were also manually measured from images using the "line tool" of ImageJ (Moyers et al., 2017). These types of measurements are necessary for linking pollinator visitation rates to the floral traits that attract them, but manual field methods are time-consuming and often result in limited replication.

Researchers are now exploring more objective methods, such as image analysis, to replace the subjective and laborious manual methods in agricultural applications. Classifying plants based on their visual dissimilarities (Grinblat, Uzal, Larese, & Granitto, 2016; Hall, Dayoub, Perez, & McCool, 2018), inspecting and grading of fruits and vegetables (Pereira, Barbon, Valous, & Barbin, 2018; Subhashree, Sunoj, Xue, & Bora, 2017; Wan, Toudeshki, Tan, & Ehsani, 2018), classifying medicinal plant leaves (Prasad & Singh, 2017), estimating and classifying biomass (Pothula, Igathinathane, Kronberg, & Hendrickson, 2014; Schirrmann, Hamdorf, Garz, Ustyuzhanin, & Dammer, 2016) are some example applications for objective methods. Image processing methods can be applied to

field-scale applications that include detecting plant row count (de Souza, Lamparelli, Rocha, & Magalhães, 2017; Guerrero, Ruz, & Pajares, 2017), plant leaf disease (Prasad, Peddoju, & Ghosh, 2016; Singh & Misra, 2017; S. Zhang, Wu, You, & Zhang, 2017) pests (Barbedo, 2014; Maharlooei et al., 2017; Sunoj et al., 2017), and weeds (Bakhshipour & Jafari, 2018; dos Santos Ferreira, Freitas, da Silva, Pistori, & Folhes, 2017).

Given the expanse of image processing potential in agriculture, it can be extended to pollination research as literature in this field is lacking. As sunflower component dimensions can be correlated to pollinators attraction and seed yields estimations, as well as the current practice of manual measurements, is subjective, tedious, and limited to a few measurements, an alternative automatic dimensions measurement procedure for sunflowers through user-coded image processing tool was proposed and developed in this research. The proposed image processing approach will be objective, rapid, allow for multiple radial measurements that can be subjected to statistical analysis, and the results along with the images can be archived for later use. Other geometrical parameters, namely the area, perimeter, and shape of the sunflower components can also be readily determined. Furthermore, the proposed image processing approach will serve as a tool and the research will be beneficial to the sunflower breeders.

Therefore, the overall goal of this paper is to develop an image processing method to evaluate the floral dimensions of the sunflower head, disc, and ray florets. The specific objectives include (i) developing an image processing plugin (program) in ImageJ, (ii) determining the effective number of radial measurements that statistically best represents the mean dimension of individual components, and (iii) developing correlations

41

among various measured dimensions, determine their interrelationships, and produce prediction models.

3.3. Materials and Methods

Description of various aspects in this study that includes test material, image acquisition, preprocessing, image processing, dimensions and other geometrical measurements, image processing tool development procedure, statistical analysis of measurements, and friendly sunflower dimensions prediction model development are presented subsequently in appropriate sections.

3.3.1. Test Material — Sunflowers

Sunflowers (var. Peredovik) used in this study were from experimental plots of a cover crop-bee pollinator study conducted between June and August 2017, located 1 km southwest of Mandan, ND, USA (46.77° N, 100.90° W). Sunflowers of different sizes were randomly selected within the experimental area. The diameter of the whole sunflower (average of length and width — along two perpendicular dimensions) ranged from 90 mm to 221 mm.

3.3.2. Developed Background Board for Image Acquisition

Acquiring digital images in agricultural fields is challenging because images produced are prone to issues, such as image distortion, the presence of shadow, improper background, lack of contrast, and absence of reference scale for dimensional calibration. These issues can lead to difficulties in developing an image processing algorithm that can be generalized to all situations and images. Based on the preliminary observations, a sunflower image acquisition methodology to address these issues was developed (fig. 3.1).
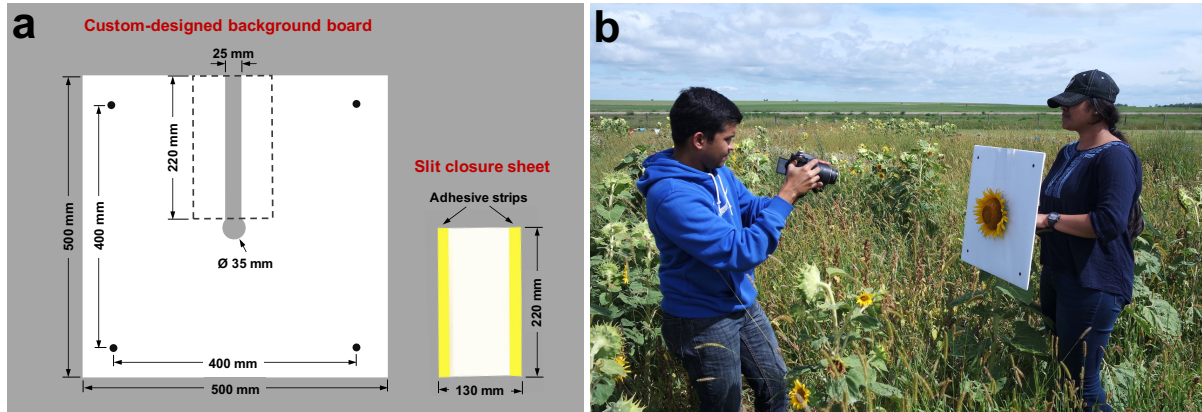
Figure 3.1. Image acquisition procedure followed for sunflower dimension measurement (a) custom-designed background board with slit closure sheet; and (b) capturing sunflower image in the field using the custom-designed background board and DSLR camera.

To provide a contrasting background and facilitate easy handling of sunflowers for image acquisition, a "custom-designed background board" (500 mm × 500 mm) was made using a white foam board (5 mm thick). A "keyhole" like slit (220 mm × 25 mm) was cut in the board, from the center to one edge with a circular hole (35 mm ∅) at the center (fig. 3.1a). The slit helps to slide the flower pedicel down, and the hole holds the pedicel and positions the flower at the center of the board. After positioning the flower, the slit will be visible and become an artifact of the captured image. To avoid this, a detachable "slit closure sheet" made using a white poster sheet (220 mm × 130 mm) with adhesive strips (using "Post-it" sticky notes) were attached in a way that the sticky surfaces were exposed, along the edges of the length to cover the slit (fig. 3.1a).

This arrangement ensured that only the sunflower head, eliminating stem and leaves, was presented with a clear white background for the image capture. Four dimensional markers (filled black circles of 13 mm ∅) were fixed at the corners of the board, to represent a square of side 400 mm. These markers serve as a reference scale for

43

dimension measurement calibration, and to rectify the image distortions using projective transformation (discussed in section 3.3.6). The developed imaging methodology was nondestructive and the sunflowers need not be clipped from the plant. The overall time required to capture images using the developed image acquisition setup was $\approx$30 s ($\approx$20 s to insert the board in sunflower pedicel, $\approx$5 s to fix the slit closure sheet, and $\approx$5 s to capture the images).

3.3.3. Digital Image Acquisition

A digital single-lens reflex (DSLR) camera (Model: Nikon D5100, Nikon Corp., Japan) was used to capture RGB images of sunflower in the field with following manual camera settings: lens $=$ 18–55 mm, shutter speed $=$ 1/300 s, aperture $= f/4$, and ISO $=$ 100. The camera was held perpendicular to the board at a distance of 1–2 m and three images (replications) were captured for each flower (fig. 3.1b). Sharp, well-focused images, without any camera shake, were captured. Commercial cameras, and smartphones available in the market are equipped with image stabilization technology that produces sharp images.

From preliminary observations, the images captured of sunflowers facing the sun (east) produced shadows and caused difficulties during image segmentation. Therefore, sunflower heads were slightly twisted so that they faced north during image capture. The background board was firm enough and could gently twist the sunflowers without causing damage either to the plants or the board. Images were captured in RGB color mode at the camera maximum resolution (4928 $\times$ 3264 pixels). The file sizes of the images were between 3 and 5 MB. A total of 126 images from 42 different sunflowers were obtained.

The captured RGB color images were directly used as inputs to the developed plugin without any intermediate operations such as cropping or zooming.

3.3.4. Overall Algorithm of the Plugin

All image processing operations were performed using Fiji (Ver. ImageJ 1.49v, Java 1.8), an open source software. Fiji is a distribution of ImageJ (Rasband, 2018; Schindelin et al., 2012) that provides a user-friendly environment for plugin development by coding in Java. A pseudocode of the developed plugin is presented in Appendix A. The acquired image was subjected to a sequence of image pre- and post-processing operations for measuring sunflower floral component dimensions (fig. 3.2).

3.3.5. Image Preprocessing

The input color image was first converted to an 8-bit grayscale image. A median filter of 9-pixel radius was then applied to smooth the grayscale variations before thresholding. The filtered image was thresholded using the standard "Minimum" thresholding method. The "Fill Holes" option was also applied to cover any holes that might have been generated during thresholding. These preprocessing operations were performed automatically through the plugin codes by calling the menu options "8-bit," "Median...," "Threshold...," and "Fill Holes" using `IJ.run()` command directly.

3.3.6. Image Projective Transformation

Image distortion (e.g., perspective, skew distortion) is inevitable while capturing images manually in a field setting, hence dimensions measured from such distorted image will be prone to errors. In this study, a projective transformation (fig. 3.3A) was performed to rectify the possible perspective distortion of the input image (Burger & Burge, 2016b). In fig. 3.3A, **I** represents the distorted field-captured input image and **I**′ the rectified

```
                              ┌─────────┐
                              │  Start  │
                              └─────────┘
                                   │
        ┌──────────────────────────▼──────────────────────────┐
        │                  Load input image                   │
        │       Digital image (RGB) acquired using custom-designed       │
        │       background board; Facing away from sunlight (north)      │
        └──────────────────────────┬──────────────────────────┘
                                   │
        ┌──────────────────────────▼──────────────────────────┐
        │                 Image preprocessing                 │
        │     RGB to grayscale image ▶ Median filtering ▶ 'Minimum'     │
        │     thresholding method ▶ Fill holes ▶ Analyze particles     │
        └──────────────────────────┬──────────────────────────┘
                                   │
        ┌──────────────────────────▼──────────────────────────┐
        │          Projective transformation of input image          │
        │      Mapping dimensional marker's centroid coordinates to target      │
        │       coordinates (3000 pixels); Bicubic interpolation       │
        └──────────────────────────┬──────────────────────────┘
                                   │
        ┌──────────────────────────▼──────────────────────────┐
        │            Segmentation of the transformed image            │
        │     Crop projective tranformed image to 3200 × 3200 pixels     │
        │              Head ▶ 'Default' thresholding method              │
        │              Disc ▶ 'Shanbhag' thresholding method              │
        │           Ray florets ▶ Images subtraction (Head−Disc)           │
        └──────────────────────────┬──────────────────────────┘
                                   │
        ┌──────────────────────────▼──────────────────────────┐
        │        Dimension measurement — 'Pixel-march' method         │
        │            Three objects: head, disc and ray florets            │
        │           Two methods: Direct and wrapping-polygon           │
        │           Number of measurements: 2–180 (8 variations)           │
        └──────────────────────────┬──────────────────────────┘
                                   │
        ┌──────────────────────────▼──────────────────────────┐
        │     Other geometrical measurements — Standard outputs      │
        │     Area, perimeter, equivalent diameter, fitted ellipse dimensions,     │
        │     bounding box dimensions, solidity, and circularity of sunflower     │
        │                       floral components                       │
        └──────────────────────────┬──────────────────────────┘
                                   │
                              ┌─────────┐
                              │   End   │
                              └─────────┘
```

Figure 3.2. ImageJ plugin development flowchart showing details of developed algorithm for sunflower floral components dimension measurement.

transformed image suitable for the measurements. The dimensional markers served as the reference points for rectifying the planar distortion of the image, and the spacing between them served as the reference scale for measurement calibration. In the plugin, the dimensional markers were automatically identified, based on their unique shape (circularity) from the input image, without user intervention.
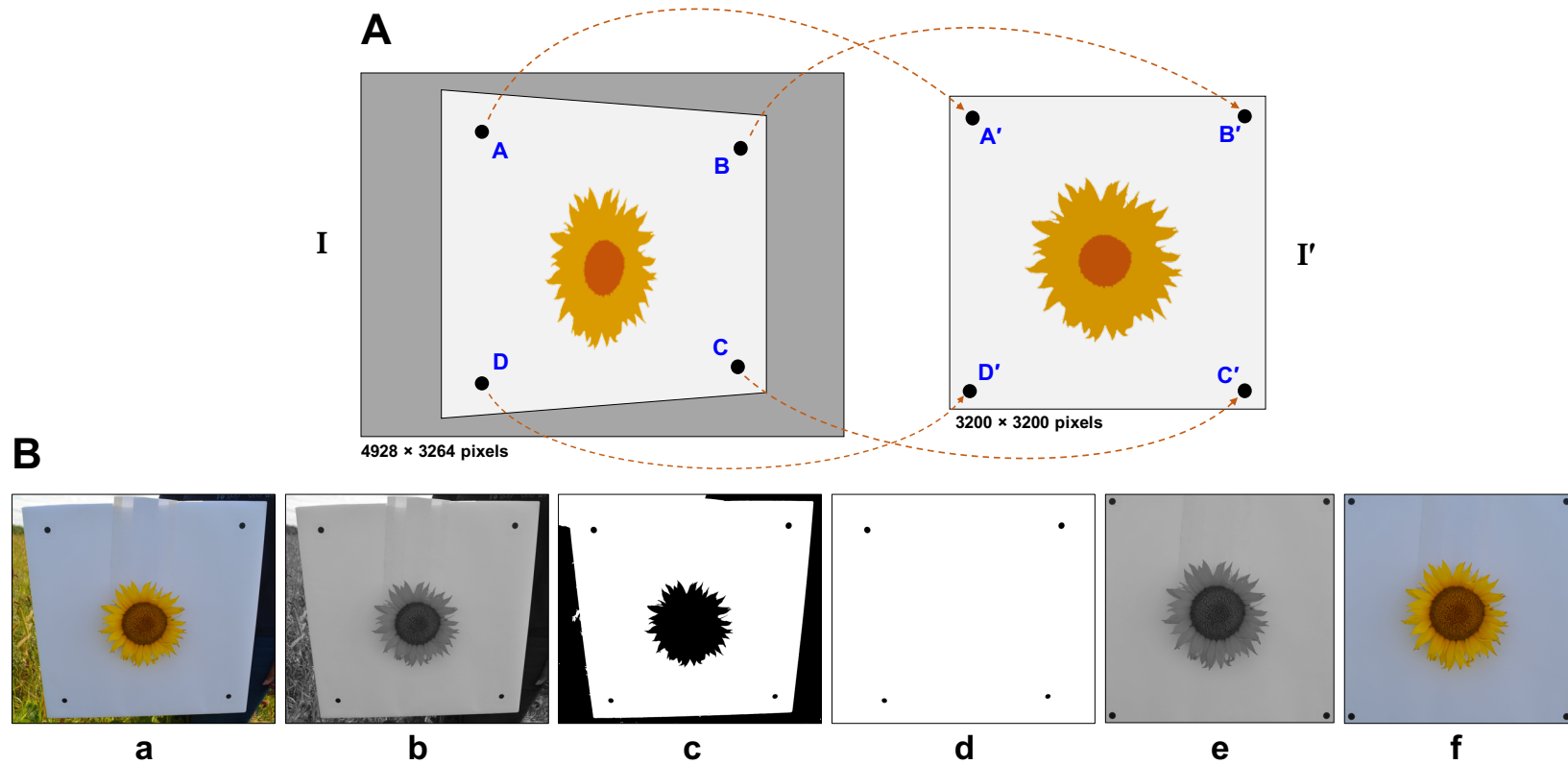
Figure 3.3. Demonstration of projective transformation for rectifying geometrical distortions: (A) Schematic representation of projective transformation of the distorted input image (I) to the rectified transformed image (I′) that was used for dimension measurements. (B) Illustration of projective transformation process: (a) distorted input RGB image, (b) grayscale image, (c) binary image, (d) identified dimensional markers, (e) projective transformed, clipped, and rectified grayscale image with padding used for further processing and measurement, and (f) illustration of RGB version of the transformed rectified image.

3.3.6.1. Identification of dimensional markers

The `ProjectiveMapping()` class, which does the projective transformation, requires the coordinates of the distorted image marker points (A, B, C, D) and the target points (A', B', C', D') as input (fig. 3.3A). The target points represent the user-defined desired dimension of the transformed image for further measurements, and a square of 3000 pixels size, suitable for the range of sunflower sizes considered, was fixed in this study. For this, the centroids of identified dimensional markers and the target points were used in the transformation mapping.

From the binary image, the dimensional markers were identified using area and circularity values from the `ParticleAnalyzer()` class outputs. Based on the preliminary observations on all captured images, an area >2000 pixels and circularity >0.85 was found suitable for identifying the circular dimensional markers. Positions of these identified markers were extracted from the standard outputs of centroid coordinates.

3.3.6.2. Image transformation and processing stages

The horizontal and vertical distances between the target points ($A'B' = B'C' = C'D' = D'A'$) were the set 3000 pixels (fig. 3.3A). The target points were further given a padding of 100 pixels on all sides to observe them as complete circles with an additional white margin for a better visualization of the image. Without providing margins, the target point centers would become the corners of the transformed image and the target points would only be seen as quarter circles; however, this padding did not affect the measurements. Therefore, the width and height of final projective transformed images were 3200×3200 pixels. As transformation creates new areas, bicubic interpolation was used to fill these areas with relevant image pixel values.

An actual illustration of the various stages of image processing is presented in fig. 3.3B. The raw input RGB image (fig. 3.3a) obtained from field was first converted into an 8-bit grayscale image (fig. 3.3b). The median filter (9-pixel radius) grayscale image was thresholded ("Minimum" method) with holes covered ("Fill Holes" option) to produce the binary image (fig. 3.3c). The `AnalyzeParticles()` with filters of area >2000 pixels and circularity >0.85 identified well only the dimensional markers (fig. 3.3d), as the circularity of the whole sunflowers were <0.2. Positions of the identified dimensional markers, which are equidistant originally at 400 mm square, appear to be distorted (unequal spacing among them). The projective transform rectified these distortions by mapping the identified dimensional markers to the target coordinates using the centroids, and this mapping also rectifies the whole image in the process. The transformed image was clipped at 3200×3200 pixels and was used for further image processing (fig. 3.3e). For illustration, the transforming and clipping of the color image (without thresholding) to the final image (fig. 3.3f) is also shown to visualize the process as if the image were taken without any perspective distortion. As seen in the process, the projective transformation only performed geometrical corrections and did not negatively affect the shape of the sunflower.

3.3.7. Sunflower Head, Disc, and Ray Florets Segmentation

From the projective transformed grayscale image (fig. 3.3e), the sunflower head (whole flower), disc, and ray florets (petals) portions were segmented using ImageJ's inbuilt thresholding methods (fig. 3.4). Sunflower head (fig. 3.4a) was segmented by "Default" method and disc (fig. 3.4b) by "Shanbhag" method – ImageJ's inbuilt methods. These thresholding methods are dynamic that works solely based on the histogram pattern, and does not have a fixed threshold value. The ray florets were obtained by subtracting the

disc from the sunflower head segmented images using the `ImageCalculator()` class
(fig. 3.4c).
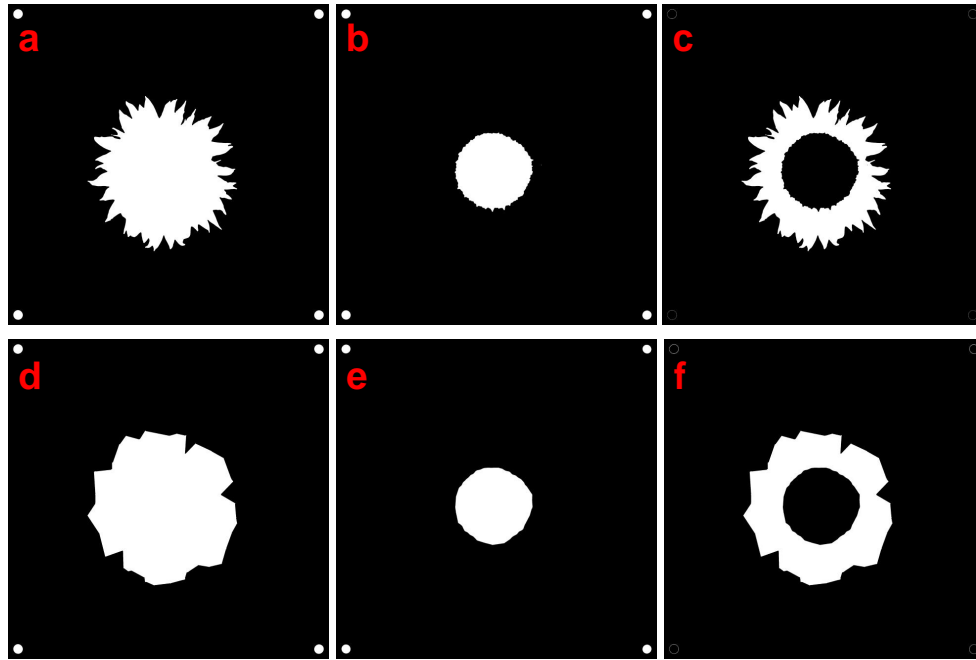
Figure 3.4. Segmented binary images of sunflower floral components (a) sunflower head "Default" thresholding method, (b) disc using "Shanbhag" thresholding method, (c) ray florets using image difference by subtracting disc from head, (d) wrapping-polygon of head, (e) wrapping-polygon of disc, and (f) wrapping-polygon of ray florets using image difference.

The binary images of sunflower head and disc were used to construct a
wrapping-polygon (discussed subsequently) of the head (fig. 3.4d) and disc (fig. 3.4e).
These two images served as inputs for creating the difference image of ray florets
(fig. 3.4f). The ray florets represent a composite binary image where the disc is black and
ray florets in white. Such composite images (fig. 3.4c & f) were used to measure the
dimensions of the sunflower head, disc, and ray florets in a single operation of
measurement that goes through different floral components from the centroid.

3.3.8. Direct and Wrapping-polygon Methods for Measurement

Based on the ray florets arrangement, the line of dimension measurements might fall on the sides or valleys of the ray florets (fig. 3.5), where the general expectation for measurement is "tip-to-tip." The multiple radial measurements will, however, minimize the discrepancy. To effectively address this, a new concept of "wrapping-polygon," which was different from the standard convex-hull was developed.
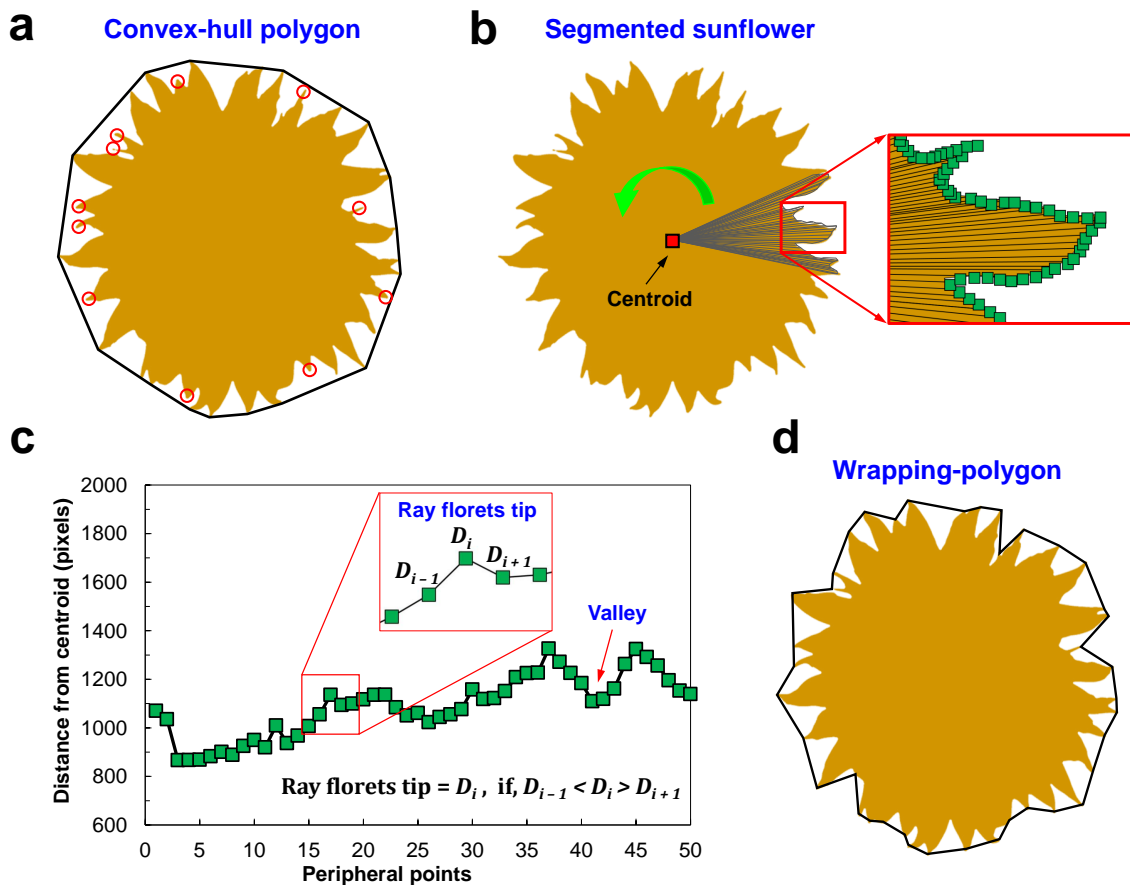
Figure 3.5. Development of wrapping-polygon from binary sunflower image (a) illustration of convex-hull polygon with missing ray florets tips (shown in red circles), (b) segmented sunflower head (shown in golden yellow) with connecting lines between the centroid and peripheral points, (c) plot between peripheral points and distance from centroid for identifying ray floret tips, and (d) wrapping-polygon (black solid line) enclosing the segmented head based on the identified tips.

The wrapping-polygon can be thought of as a "shrink-wrap" type of geometric enclosure that includes all ray floret tips developed to resemble the natural dimension measurement of sunflower floral components. A standard convex-hull polygon would have fewer vertices and would miss several ray floret tips (convex-hull polygon; fig. 3.5a), and measuring the dimensions along multiple angles would result in overestimation. The wrapping-polygon, however, provided the accurate tip-to-tip profile-based distance at any measurement angle.

The ray floret tips for the wrapping-polygon were determined by finding the tips from the ray florets edge profile (fig. 3.5b). An example of this process with 50 edge pixels is shown in fig. 3.5c. The `autoOutline()` method of `Wand()` class produced the coordinates of edge pixels. For each edge pixel, the distance (edge to centroid) was compared with the distances of adjacent pixels (one before and one after) and the tip was determined. To qualify as a peak, the ith pixel distance should be greater than that of $i-1$ and $i+1$ pixels. It was found that this simple method was good enough to identify all tips of the flower. These ray floret tips formed the vertices of the wrapping-polygon and the flood-fill option completed the object for further processing (fig. 3.5d) and dimension measurements (fig. 3.4d–f) .

3.3.9. Dimension Measurement — Standard and Pixel-march Methods

ImageJ offers standard dimension measurements that include bounding box width and height, Feret diameter, and major and minor axis length; along with other standard object properties that include area, perimeter, and centroid coordinates. Other derived properties such as equivalent diameter can be obtained from the object area through user codes. Of these standard measurements, the fitted ellipse major (IJ) and minor (KL) axes

(fig. 3.6a) best represent the length and width of an object. However, the major and minor axes are aligned randomly based on the object shape. Furthermore, in manual measurements, the length and width of the floral components are measured along the horizontal (east to west, EW) and vertical (north to south, NS) directions. Therefore, the standard major and minor axes based measurements, despite useful, may not always represent the generally expected manual measurements.

A "pixel-march" method developed by Igathinathane, Pordesimo, and Batchelor (2009b), which typically addressed the inability of standard measurement scenarios and worked irrespective of object orientation, was employed in this study for dimension measurements. The pixel- march operated on finding the boundary pixel whose color was different from that of the previous pixel while marching from a specified starting point (centroid). The two common physical dimensions along horizontal (length) and vertical (width) directions can be obtained using pixel-march by starting from the centroid and marching outwards at 0° (EW) and 90° (NS) in both directions, respectively (fig. 3.6a).

It can be observed from fig. 3.6b that using a single operation of pixel-march the sunflower disc and head dimensions were measured starting from the centroid and reaching the first boundary (e.g., D_1D_1') representing the disc, then reaching the second boundary (e.g., H_1H_1') representing the head along the same direction, and the difference (e.g., $H_1H_1' - D_1D_1'$) representing the ray florets dimensions. This pixel-march method of measurement has the potential of obtaining multiple dimensions along radial directions (fig. 3.6a), and such measurements better represent the mean dimension of the sunflower.

The number of measurements (n) required was taken as input and the pixel-march measurements proceeded along every angle interval ($\theta = 180°/n$). In this study, the

a

Major axis

Fitted ellipse (resembling circle)

Head

Centroid

Minor axis

Disc

Centroid

Major axis

Minor axis

$$D_2 = \frac{EW + NS}{2}$$

IJ = Fitted ellipse length
KL = Fitted ellipse width

EW = 'Pixel-march' length
NS = 'Pixel-march' width

b

Direct method

Wrapping-polygon method

Centroid

Centroid

$$\theta = \frac{180°}{\text{number of measurements } (n)}$$

$$\left. \begin{array}{l} \text{Head length}_i = H_i H_i' \\ \text{Disc length}_i = D_i D_i' \\ \text{Ray florets length}_i = H_i H_i' - D_i D_i' \end{array} \right\} \quad i = 1, 2, 3, \ldots, n$$

Figure 3.6. Dimension measurement methods for sunflower components
(a) illustration of fitted ellipse method available in ImageJ in sunflower head and disc, (b) multiple radial dimensions measurement by using "pixel-march" method with direct measurement method (fig. 3.4c) and wrapping polygon method (fig. 3.4f) difference images.

54

number of measurements considered for testing were 2, 4, 8, 16, 32, 64, 128, and 180. The two manual orthogonal measurements of the head, EW and NS can be simulated by selecting $n = 2$, which makes $\theta = 90°$ for pixel-march, and its mean was represented as $D_{2h} = (\text{EW+NS})/2$. This D_{2h} represents the manual measurement in the field. Similarly, the most elaborate dimension uses 180 measurements ($n = 180$; $\theta = 1°$) and its mean was represented as $D_{180} = \sum_{i=1}^{180} D_i / 180$.

3.3.10. Other Geometrical Parameters

Area, perimeter, solidity, and circularity were the other geometrical parameters of the sunflower floral components obtained from the standard outputs of ImageJ analysis. Equivalent diameter (ED) is a user-coded parameter derived from the object area ($\text{ED} = \sqrt{(4/\pi) \times \text{Area}}$) and it represents the diameter of a circle having the same area of the object. These parameters were evaluated through both direct and wrapping-polygon methods. These parameters were obtained by directly passing the segmented binary images of individual sunflower components (fig. 3.4a-f) through ImageJ's `AnalyzeParticles()` class.

3.3.11. Developed Image Processing Plugin

The front panel of the developed ImageJ plugin had two inputs and several output options (fig. 3.7). The developed plugin reads an input RGB image (e.g., fig. 3.1, field captured image), performs all the operations (sections 3.3.4 – 3.3.10) in the background, and produces the outputs. The minimal inputs necessary to run the plugin are (i) the known spacing of the dimensional markers the user employs, and (ii) the number of measurements desired. The output options generate additional images for inspection and archival purposes. When the plugin was run, after loading the color (RGB) field image, by

clicking the "OK" button, the measurement results were displayed as a pop-up window message. There is also an option to save the results for further statistical analysis, as a consolidated file that can be opened using a spreadsheet program.
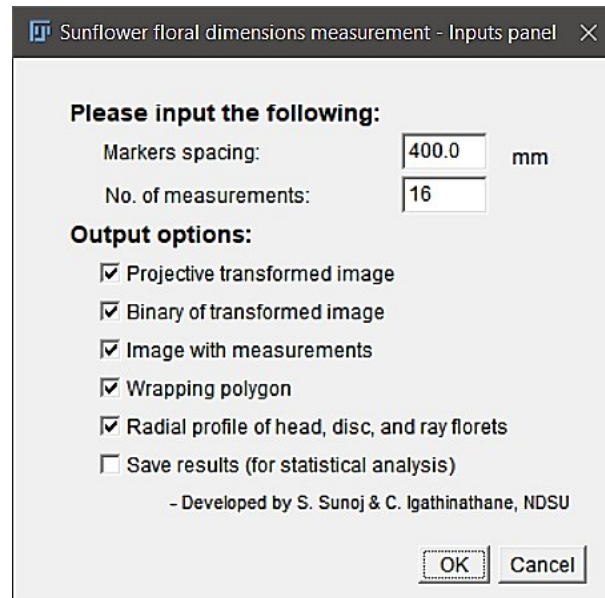
Figure 3.7. Front panel of the developed ImageJ plugin for sunflower floral dimension measurements.

All the calculations inside the plugin are performed in the basic pixel units, and it is necessary to convert them into physical units (e.g., mm) for dimensions interpretation. In this study, the markers spacing of 400 mm served as the reference measurement for units conversion. As the markers of the transformed image were set at 3000 pixel spacing, the image had a resolution ≈185 DPI, which was equivalent to a measurement accuracy of 0.133 mm/pixel. This transformation eliminates the inconsistencies in the image resolution caused due to variations in object-to-camera distance. The plugin took a CPU time of ≈11 s to perform the analysis in a Windows laptop (Windows 10, 2.20 GHz Intel Core i5-5200U processor, and 8 GB RAM).

3.3.12. Prediction Model Development

It will be desirable to develop user-friendly prediction models for sunflower components that use ImageJ standard outputs and simple measurements as inputs to obtain accurate dimension (through the developed image processing) and area measurements were developed. Correlation analysis determined the selection of independent variables for the models. The most accurate multiple dimension, D_{180}, was considered as the reference (control) and other measurement methods were compared; the least number of multiple dimension, D_{2h}, represents common manual measurement and serves as an independent variable to obtain the D_{2h}.

The independent variables that were correlated to D_{180} can be divided into three groups, namely linear dimensions (e.g., equivalent diameter, minor axis), geometrical parameters (e.g., area, perimeter), and D_{2h}. Here, the linear dimensions and geometrical parameters of head, disc, and ray florets only from direct method was considered as it was easily obtainable by the user after segmentation directly from ImageJ standard outputs (fig. 3.4a–f). The D_{2h}, in this study, was calculated using the two measurements from the wrapping-polygon method as it represented the tip-to-tip and simulated the field manual measurements. The linear regression model coefficients of determination (R^2) for D_{180} using these independent variables were also studied to observe the trend. It is also desirable to study how the area of the floral components was related to D_{2h}.

R program cor() function was used to determine the Pearson's correlation coefficient (R Core Team, 2018). Among the inputs, the most influential variables were determined based on the highest correlation coefficient, and were employed to develop linear regression models using lm() function.

3.3.13. Statistical Analysis

Analysis of variance was conducted to evaluate the effect of two measurement methods (direct and wrapping-polygon), and the number of measurements on dimensions. The deviation (%) in the measured dimension with respect to the control (D_{180}) was calculated as absolute deviation using equation 3.1.

$$\text{Absolute deviation (\%)} = \frac{|D_i - D_{180}|}{D_{180}} \times 100 \qquad (3.1)$$

where, D_i is the mean dimension of i measurements ($i = 2, 4, 8, 16, 32, 64, 128,$ and 180), and D_{180} is the mean dimension of 180 measurements.

Statistical comparison between methods will determine the most efficient measurement method, and the measurement comparisons will provide a minimum (efficient) number of measurements beyond which there is no need to consider based on statistical significance with the control. The R packages, such as `agricolae` (de Mendiburu, 2019) for Duncan's new multiple range test, and `ggplot2` (Wickham, 2016) for graphical visualization of the results were used in the analyses.

3.4. Results and Discussion

3.4.1. Effect of Number of Measurements on Mean Dimension

The developed image processing plugin allowed us to select different numbers of measurements and observe the effects on the mean dimension (fig. 3.8). A plot of mean dimension variations of 10 sunflower heads, selected as an example, and obtained through direct and wrapping polygon shows the existing variation among the number of dimensions (fig. 3.8a). Irrespective of the head size, variations were present in all sunflowers.
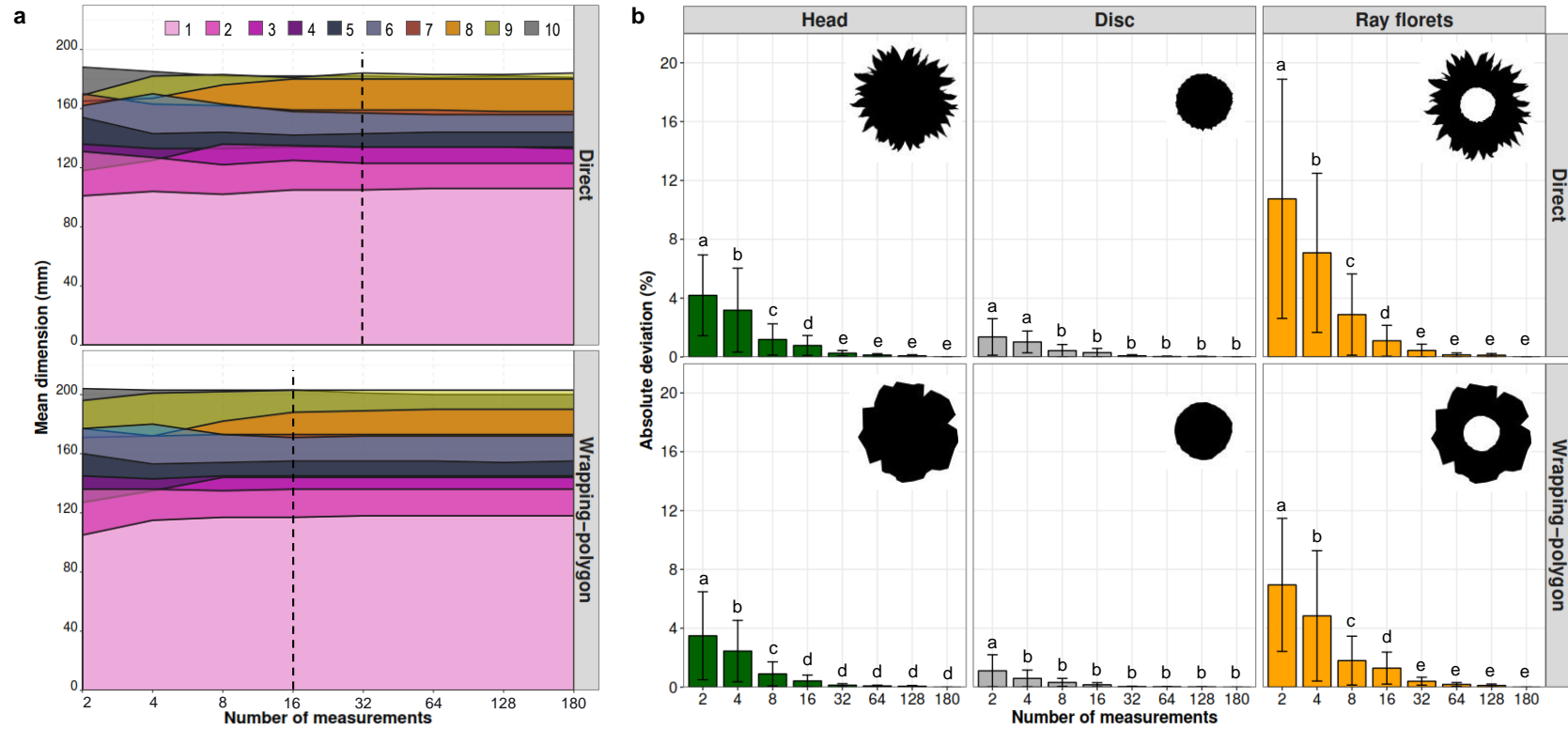
Figure 3.8. Effect of number of measurements on (a) mean dimensions of 10 sunflower heads, (b) absolute deviation for sunflower head, disc, and ray florets in direct and wrapping-polygon methods. (*Note:* Dissimilar letter groups (a–e) indicate statistically significant difference).

The means dimensions of the wrapping-polygon method were greater than direct method, but among-measurements variation of the wrapping-polygon were relatively less those of the direct method. Among individual flowers, the variations trend was random especially when the number of measurements was smaller (<32; fig. 3.8a). This random variation was caused by the ray florets arrangement, as the pixel-march based on the angle will be anywhere between tip-to-tip and valley-to-valley in the direct method (fig. 3.5). In case of the wrapping-polygon method, the valleys were filled, allowing only for joined tip-to-tip measurement and resulted in lesser variation. It is quite natural for the variations to flatten out with an increased number of measurements. The lower limits from which the variations were minimal were found to be 16 and 32 for the wrapping-polygon and direct methods, respectively, representing the efficient number of measurements (fig. 3.8a).

The absolute deviation (eq. 3.1) provides a clear visualization of the variations produced by the number of measurements, using the whole data, compared with the respective controls (D_{180}; fig. 3.8b). Although the nature of absolute deviation was different among sunflower components, it was evident that increasing the number of measurements decreased the absolute deviation values and gradually converged to 0 %. The absolute deviations for the wrapping-polygon method were comparatively lower than the direct method (fig. 3.8b).

Compared to the most accurate dimension (D_{180}), it was found that the two dimensions measured manually (D_{2h}) produced >3.5 % deviation in head, and >1.1 % in disc, and >6.9 % in ray florets, and these were significantly different. It can be observed that the more irregular the outer profile of head and ray florets produced the more deviation than the disc, which approaches a geometrical circle. Between head and ray

florets, the deviations of ray florets were higher because the dimensions of the head were significantly higher than that of the ray florets, despite both having the exact same irregular outer profile. Statistical mean comparison analysis revealed that a minimum of 32 measurements was required to best represent the head and ray florets dimensions, and eight measurements were enough for disc dimensions.

3.4.2. Effect of Methods on Mean Dimensions of Sunflower Components

The wrapping-polygon method resulted in higher mean dimensions compared to the direct method for the heads and ray florets, while both methods produced almost equal dimensions for the discs among the number of measurements (fig. 3.9). The mean dimensions of direct and wrapping-polygon method differed significantly for head ($0.004 < p < 0.01$), and ray florets ($3.3 \times 10^{-9} < p < 5.7 \times 10^{-5}$), but not with the disc. On average, the mean dimensions of wrapping polygon exceeded the direct method by 7.14 % for the head, 1.70 % for the disc, and 16.55 % for ray florets.

The mean dimensions of the head were normally distributed with uniform inter-quartile range (IQR) covering 128–180 mm in the direct method, and 130–192 mm in wrapping-polygon method. The mean dimensions of disc were right skewed with uniform IQR covering 76–128 mm in both methods. The mean dimensions of ray florets were normally distributed up to eight measurements, beyond which the data were slightly left skewed with the presence of few outliers. The IQR in the direct method was 36–68 mm for two measurements, and gradually narrowed to 46–60 mm for 180 measurements. Similarly, the IQR in wrapping-polygon method was 48–76 mm for two measurements and narrowed to 60–72 mm for 180 measurements.
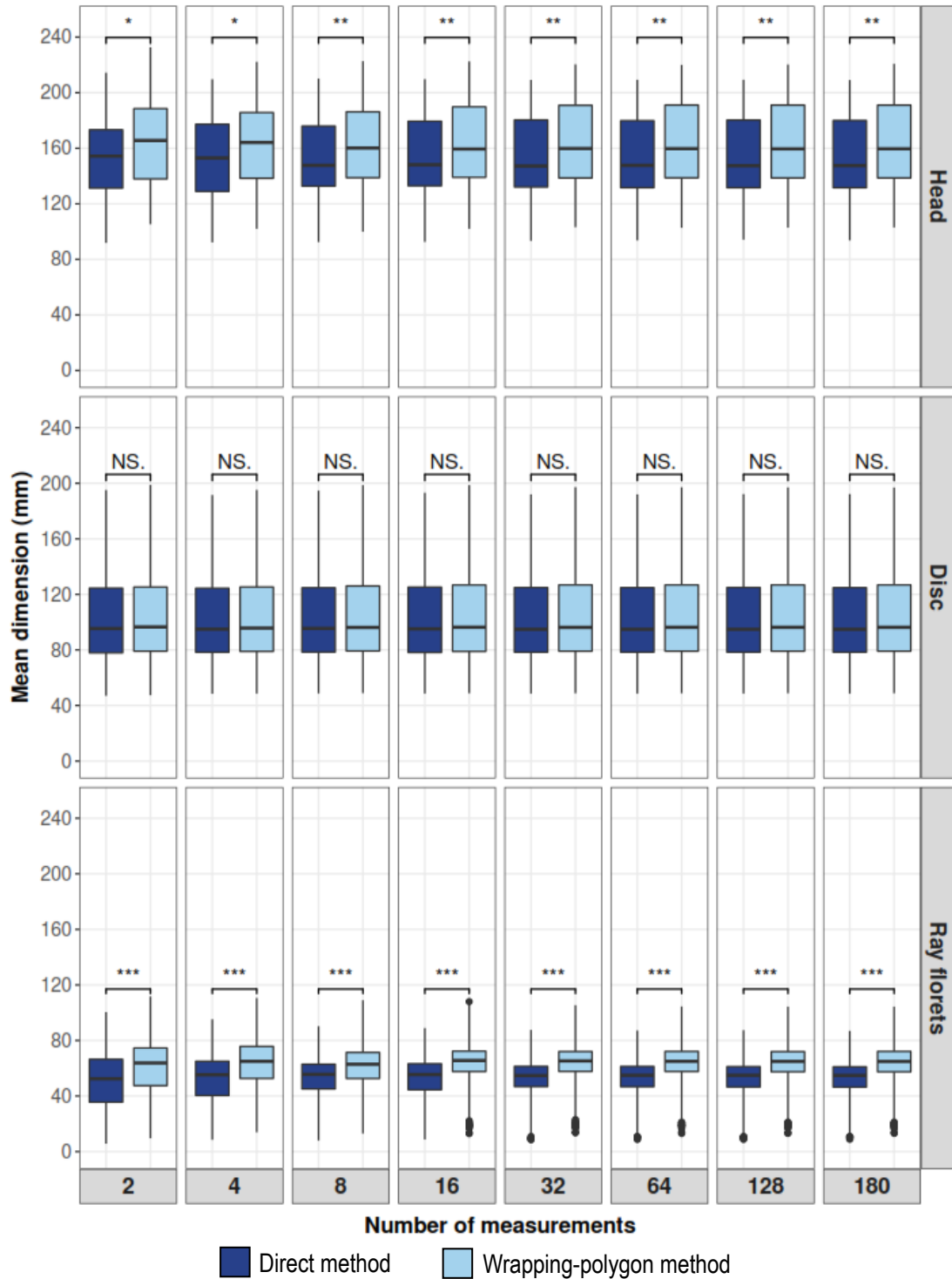
Figure 3.9. Comparison between direct and wrapping-polygon methods showing the effect of number of measurements on mean dimensions of sunflower components. (*Note*: Codes of statistical significance: * denotes $\alpha = 0.5$, ** denotes $\alpha = 0.01$, *** denotes $\alpha = 0.001$, and "NS." denotes not significant).

On closer observation, the overall range of mean dimension slightly narrowed as the number of measurements increased for both methods. For example, comparing the D_{180} to D_2 for direct method, the reductions in the range of values were 5.8 %, 3.1 %, and 17.5 % for the head, disc, and ray florets, respectively. This indicates that increasing the number of measurements produced more precise measurements.

3.4.3. Dimensions Comparison — Pixel-march *vs* ImageJ's Standard Outputs

To understand how the image-processing based sunflower components dimension measurements developed in this research matches with available measurements through standard outputs from ImageJ, a comparison was performed (fig. 3.10). The developed approach uses the pixel-march and measures dimensions through the two methods (direct and wrapping-polygon) operating on the binary images, while the ImageJ option also works on the preprocessed binary image and produces the standard outputs. The standard outputs were grouped as the enclosure-based (bounding box width, height, and Feret diameter) and area-based (major, minor axes, and equivalent diameter) methods.

Overall, comparing direct and wrapping-polygon methods, the mean dimensions from different measurement methods followed the same pattern among sunflower components, but significant differences ($p < 0.028$) were observed among the methods (5 significantly different groups). For head and disc, it is interesting to note that the accurate pixel-march method was statistically similar to area-based minor axis and equivalent diameter methods (fig. 3.10). Also, the mean dimensions from pixel-march method and area-based methods of wrapping-polygon, while only slightly higher, differed significantly ($2.5 \times 10^{-16} < p < 1.1 \times 10^{-12}$) from the direct method. The enclosure-based
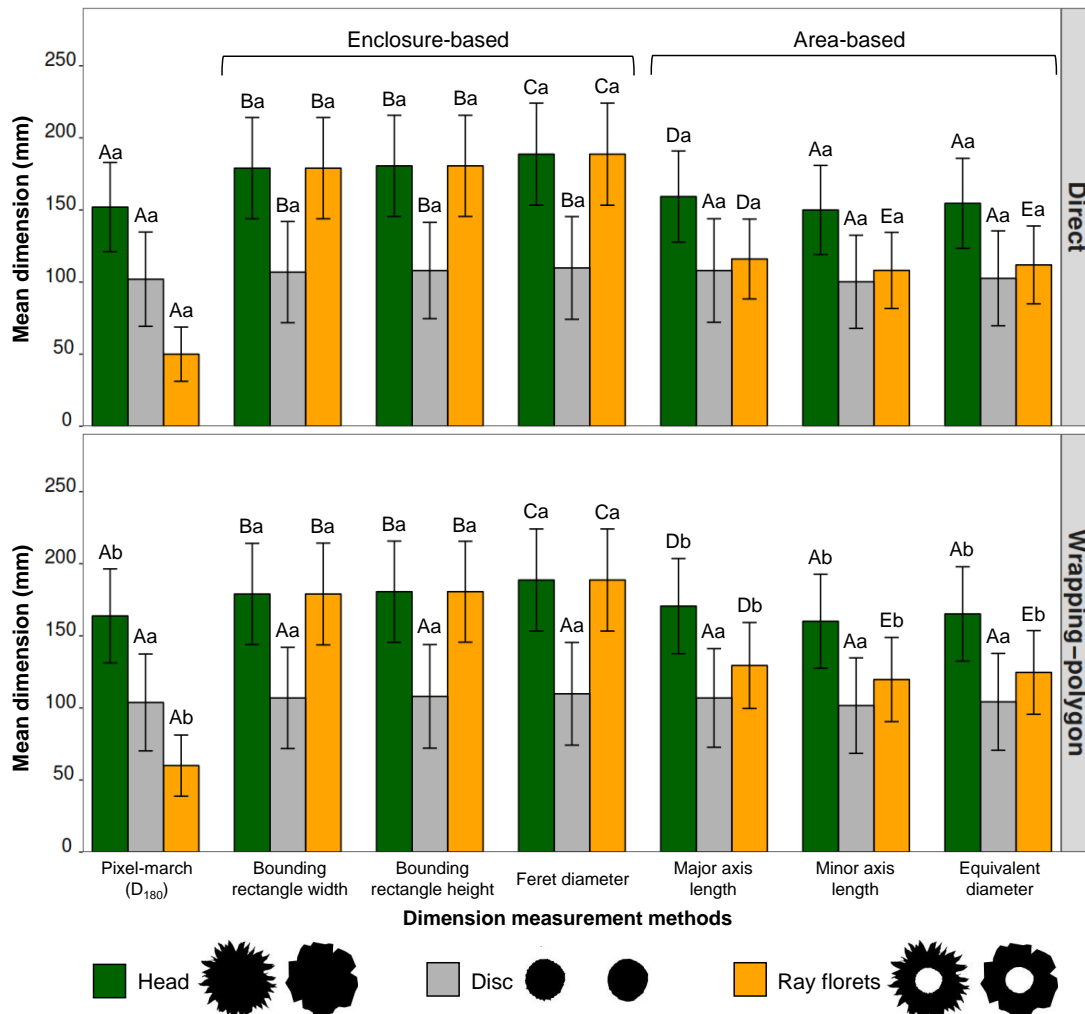
Figure 3.10. Mean dimensions of sunflower head, disc, and ray florets from different methods on direct and wrapping-polygon methods. (*Note:* **Bar heights and error bars indicate mean and standard deviation values, respectively, based on 42 samples and three replications. Dissimilar uppercase letter groups (A–E) denote significant differences among dimension measurement methods, and lowercase letter groups (a–b) denote differences between direct and wrapping-polygon methods).**

methods were not different between direct and wrapping polygon methods because the

enclosure (box, and circle) dimension remains the same for both methods.

Specifically, for sunflower head measurement (green bars; fig. 3.10), the

pixel-march method significantly differed ($p < 2.0 \times 10^{-16}$) from all enclosure-based

methods and major axis length ($p < 0.0002$), but not with minor axis length and

equivalent diameter. The mean dimensions of enclosure-based methods were higher than all the other methods because the object's enclosure box and circle are only based on the objects' extreme (horizontal and vertical) boundary points. As a result, the enclosure-based methods did not differ significantly between direct and wrapping-polygon methods.

For disc measurement (gray bars; fig. 3.10), although the differences in mean dimensions were small, the pixel-march method differed significantly with enclosure-based methods ($0.0013 < p < 0.0071$), but were on par with area-based methods. However, in the case of wrapping-polygon method, mean dimensions did not differ significantly among the measurement methods, while direct method produced two significantly different groups. This is because the wrapping-polygon around the disc reduced the boundary roughness and looked similar to a perfect circle.

For ray florets measurement (orange bars; fig. 3.10), only the pixel-march method was the most relevant and accurate because it starts from discs' boundary and radially marches outward till the ray florets boundary (fig. 3.6a). All the other measurement methods resulted in an over-estimated dimension and were significantly different ($p < 2.0 \times 10^{-16}$) from pixel-march dimension. This is because the measurement of ray florets length cannot be readily obtained from a segmented image using any of the ImageJ's standard methods; but as a simple approach, the difference between minor axis length or equivalent diameter of the head and the disc can be considered. The pixel-march method, on the other hand, offers profile data for studying the dimension variations at any chosen angular interval (profile plot), which provides information about floral bloom and ray florets arrangement.

3.4.4. Geometrical Parameters of Sunflower Components

Results of geometrical parameters of sunflower components obtained through direct and wrapping-polygon methods are presented and statistically compared (fig. 3.11). The area of sunflower head, and ray florets differed significantly ($2.0 \times 10^{-16} < p < 6.4 \times 10^{-14}$) between direct and wrapping-polygon methods, but not the disc. The wrapping-polygon method increased the area around the sunflower components as expected. On average, the increase in area was 14.1 % for head, 3.3 % for disc, and 23.5 % for ray florets.
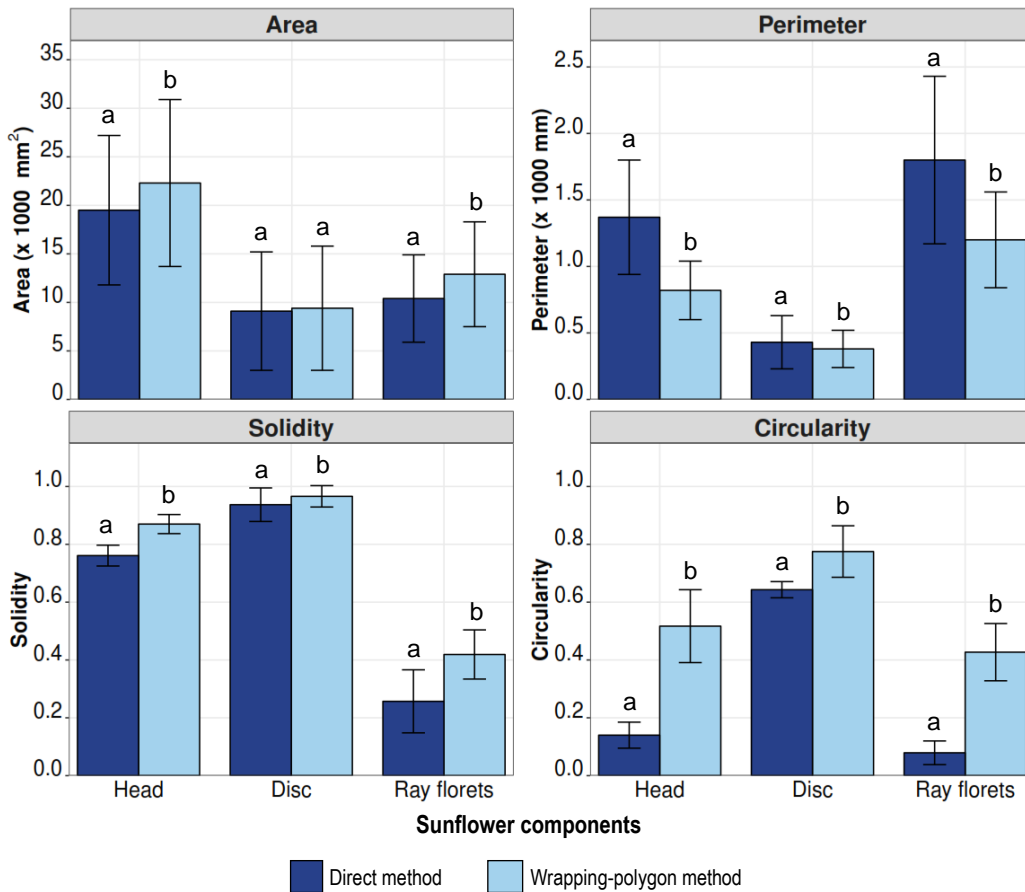
Figure 3.11. Other geometrical parameters of sunflower head, disc, and ray florets from direct and wrapping-polygon methods. (*Note:* **Bar heights and error bars indicate mean and standard deviation values, respectively, based on 42 samples and three replications. Dissimilar letter groups (a–b) denote significant differences between direct and wrapping-polygon methods).**

Perimeters of all the components differed significantly ($p < 2.0 \times 10^{-16}$) between the methods. The perimeter of ray florets includes the perimeters of head and disc, which was calculated as the sum of perimeters of head and disc. The wrapping-polygon around the sunflower components reduced the boundary roughness; hence its perimeter was significantly less than the direct method. The average decrease in perimeter due to wrapping-polygon method was 40.4 % for head, 11.1 % for disc, and 33.5 % for ray florets.

The solidity of sunflower components in direct method was significantly lower ($p < 2.0 \times 10^{-16}$) compared to the wrapping-polygon method because wrapping-polygon fills the space between ray florets thereby making them "more solid" and increasing the solidity values. Since the disc portion was removed in the composite images of ray florets (fig. 3.4c & f), the solidity of ray florets was the lowest compared to head, and disc. The average increase in solidity between direct and wrapping polygon methods was 14.3 % for head, 3.1 % for disc, and 38.9 % for ray florets.

Circularity, similar to area and solidity, was significantly higher ($p < 2.0 \times 10^{-16}$) for the wrapping-polygon method than the direct method. The average increase in circularity between direct and wrapping-polygon method was 269.8 % for head, 20.5 % for disc, and 442.1 % in ray florets.

3.4.5. Correlation and Regression Coefficients of D_{180} with Input Parameters

The Pearson correlation coefficient (r) and the coefficient of determination (R^2), indicating the performance of the linear regression models developed, using the respective input variables with the D_{180} measurement for the direct and wrapping-polygon methods are presented in table 3.1.

Table 3.1. Developed measurement method (D_{180}) correlation and linear model performance for sunflower components with other standard measurement methods and simulated manual measurements (D_{2h}).

Measurement method*	Pearson correlation coefficient (r)						Linear regression model coefficient of determination (R^2)					
	Direct method			Wrapping-polygon method			Direct method			Wrapping-polygon method		
	Head	Disc	Ray florets	Head	Disc	Ray florets	Head	Disc	Ray florets	Head	Disc	Ray florets
Equivalent diameter	1.00	1.00	0.88	1.00	1.00	0.89	1.00	1.00	0.78	0.99	1.00	0.79
Minor axis length	1.00	1.00	0.88	1.00	1.00	0.88	0.99	1.00	0.77	0.99	1.00	0.79
Major axis length	1.00	1.00	0.88	0.99	1.00	0.88	0.99	1.00	0.78	0.98	1.00	0.78
Bounding rectangle width	0.98	0.99	0.20	0.99	1.00	0.25	0.97	0.99	0.04	0.97	0.99	0.06
Feret diameter	0.98	0.99	0.18	0.99	0.99	0.23	0.96	0.98	0.03	0.97	0.98	0.05
Bounding rectangle height	0.97	0.99	0.20	0.98	1.00	0.25	0.95	0.99	0.04	0.96	0.99	0.06
Area	0.99	0.98	0.85	0.99	0.98	0.86	0.99	0.97	0.72	0.98	0.97	0.75
Perimeter	0.86	0.93	0.17	0.89	0.94	0.25	0.75	0.87	0.03	0.80	0.89	0.06
Solidity	0.23	−0.41	0.89	0.17	−0.44	0.85	0.05	0.17	0.79	0.03	0.19	0.72
Circularity	−0.44	−0.56	0.55	−0.49	−0.59	0.47	0.19	0.32	0.30	0.24	0.34	0.22
D_{2h}	0.96	0.79	0.21	0.97	0.78	0.25	0.93	0.62	0.05	0.94	0.60	0.07
Area and D_{2h}	0.96	0.75	0.62	0.96	0.74	0.65	0.92	0.56	0.39	0.93	0.55	0.43

Note: * All these ImageJ standard methods were based on the individual components' measurements from direct method and used as independent variables, D_{180} = mean diameter of 180 measurements (mm), and D_{2h} = mean of two orthogonal measurements from wrapping-polygon method of the sunflower head ([EW+NS]/2; fig. 3.6a) simulating the manual measurements (mm).

While studying the correlation strength between the variables, the R^2 of the linear model (selected based on preliminary analysis) directly provides a glimpse of the performance leading to a better selection of the prediction models. For sunflower head and disc, the dimension measurement methods showed a high correlation ($0.97 \leq r \leq 1.00$) with direct and wrapping-polygon methods. While for ray florets, area-based methods were highly correlated ($0.88 \leq r \leq 0.89$), but enclosure-based methods were not ($0.18 \leq r \leq 0.25$). These r values are indicative of the R^2 values of the linear regression models developed using the respective input variables. The prediction models for head and disc dimensions were excellent ($0.95 \leq R^2 \leq 1.00$), while for ray florets dimensions, area-based methods were fairly acceptable ($0.77 \leq R^2 \leq 0.79$); however, enclosure-based methods were unacceptable ($0.03 \leq R^2 \leq 0.06$).

Among the geometrical parameters, area and perimeter showed a high correlation for head and disc ($0.86 \leq r \leq 0.99$); while for ray florets, area and solidity showed a high correlation ($0.85 \leq r \leq 0.89$). The R^2 of regression models were excellent with area of head and disc ($0.97 \leq R^2 \leq 0.99$), and perimeter of disc ($0.87 \leq R^2 \leq 0.89$); however, R^2 values were not as good for the rest of the geometrical parameters.

In the spirit of predicting the mean dimensions (D_{180}) and area from two manual head measurements (D_{2h}), correlation and linear model R^2 were evaluated (table 3.1). The D_{2h} measurement was highly correlated ($0.96 \leq r \leq 0.97$) for the head, fairly ($0.74 \leq r \leq 0.79$) for the disc, and poorly ($0.21 \leq r \leq 0.65$) for the ray florets, and the corresponding regression models performed well only for head ($0.92 \leq R^2 \leq 0.94$) but not for disc and ray florets ($0.05 \leq R^2 \leq 0.62$). This shows that only head dimensions (D_{180})

and area ($R^2 \geq 0.92$), not others ($R^2 \leq 0.62$), can be accurately predicted from the manual measurements.

3.4.6. Dimension and Area Prediction Models from Selected Input Variables

Linear regression models ($y = a + bx$) developed from the highly influential variables (table 3.1) along with their evaluated performance parameters (R^2) are presented in table 3.2. The developed linear models for D_{180} with equivalent diameter (ED) as the input variable showed the highest R^2 values ($R^2 \geq 0.99$) for head and disc using direct and wrapping-polygon methods. Since the R^2 values for predicting ray florets dimensions were lower ($R^2 \leq 0.77$), these models may not provide accurate predictions.

Among the dimension measurement methods considered in this study, ED was not a standard output from ImageJ, but was available in image processing platforms as MATLAB and OpenCV. However, it can be easily derived from the object area as indicated earlier (section 3.3.10). Even so, a model directly developed from the ImageJ's standard output is useful and user-friendly. Therefore, the next influential input variable was minor axis length (MinA; table 3.1). The model developed for D_{2h} using MinA as the input and the prediction models R^2 values are not significantly different from ED's (table 3.2). This indicates that either of the models (ED or MinA) can be used interchangeably.

As observed before, the models can predict only the head dimensions and area accurately with the manual measurements as inputs. Therefore, these models can be used in a practical sense for predicting head dimensions and area (table 3.2).

Table 3.2. Selected sunflower components user-friendly prediction models for dimension and area with direct and wrapping polygon methods and their performance.

Component and method	$D_{180} = a + b \times ED$; (R^2)	$D_{180} = a + b \times MinA$; (R^2)	$D_{180} = a + b \times D_{2h}$; (R^2)	Area $= a + b \times D_{2h}$; (R^2)
Head - direct	$-1.218 + 0.991\,ED_{h,D}$; (0.997)	$2.139 + 0.999\,MinA_{h,D}$; (0.993)	$6.155 + 0.881\,D_{2h}$; (0.934)	$-16614.118 + 219.082\,D_{2h}$; (0.923)
Disc - direct	$-0.126 + 0.996\,ED_{d,D}$; (1.0)	$0.767 + 1.011\,MinA_{d,D}$; (0.998)	$-24.002 + 0.761\,D_{2h}$; (0.621)	$-13437.793 + 137.102\,D_{2h}$; (0.568)
Ray florets - direct	$-18.868 + 0.614\,ED_{r,D}$; (0.778)	$-17.489 + 0.623\,MinA_{r,D}$; (0.768)	$30.163 + 0.122\,D_{2h}$; (0.046)	$-3176.291 + 82.646\,D_{2h}$; (0.389)
Head - wrapping polygon	$2.811 + 1.043\,ED_{h,D}$; (0.990)	$6.802 + 1.046\,MinA_{h,D}$; (0.991)	$9.391 + 0.942\,D_{2h}$; (0.942)	$-18340.674 + 246.924\,D_{2h}$; (0.931)
Disc - wrapping polygon	$-0.988 + 1.022\,ED_{d,D}$; (0.999)	$-0.008 + 1.036\,MinA_{d,D}$; (0.996)	$-23.672 + 0.774\,D_{2h}$; (0.604)	$-13856.601 + 141.523\,D_{2h}$; (0.554)
Ray florets - wrapping polygon	$-18.46 + 0.700\,ED_{r,D}$; (0.794)	$-16.651 + 0.708\,MinA_{r,D}$; (0.788)	$33.062 + 0.159\,D_{2h}$; (0.067)	$-4484.072 + 105.484\,D_{2h}$; (0.427)

Note: D_{180} = mean diameter of 180 measurements (mm), ED = equivalent diameter (mm); MinA = minor axis dimension of the fitted ellipse (mm); D_{2h} = mean of two orthogonal measurements from wrapping-polygon ([EW+NS]/2; fig. 3.6a) simulating the manual measurements (mm); Area = area derived from image processing standard output (mm^2); and $ED_{h,D}, ED_{d,D}, ED_{r,D}, MinA_{h,D}, MinA_{d,D}$, and $MinA_{r,D}$ = equivalent diameter as well as minor axis specific to the head, disc, and ray florets from the direct method, respectively.

However, the coefficients of the independent variable (D_{2h}) from other models (disc and ray florets) provides an indication of the relative proportion of disc and ray florets. The disc dimension was ≈0.76 times the head, and ray florets was ≈0.12 times the head in direct method. Similarly from the area models, considering the coefficient of head area (219.08) to be 100 %, the indication of relative disc area was ≈62.6 % and ray florets was ≈37.7 %. Therefore, in order to get accurate dimensions, it is recommended to use ImageJ's equivalent diameter and minor axis length as the inputs in the derived models (table 3.2), while the accurate area can come directly from the ImageJ standard outputs.

3.4.7. Advantages and Limitations

The image processing methodology in general and the developed plugin has the following advantages: (1) Performs any number of radial measurements; (2) Provides information about the boundary profile of head, disc, and ray florets from the diameter variations at any selected fine intervals (radial profile; fig. 3.12a); (3) Images captured and processed can be archived for later use; (4) Provides the projective transformed image as one of its outputs, which can be used for further processing or analysis; (5) Objective dimension measurement can be achieved; (6) Produces the mean dimensions of the head, disc, and ray florets from direct and wrapping-polygon methods in single computation; and (7) Provides several other geometrical parameters (area, perimeter, solidity, and circularity), which helps in understanding the floral morphology.

Limitations of the proposed imaging approach and the developed plugin include: (1) At least two people are required for image acquisition (fig. 3.1b) in the present protocol; (2) Images captured with sunflower heads not perpendicular to the camera resulted in poor segmentation, due to unfilled disc with the loss of clear disc border
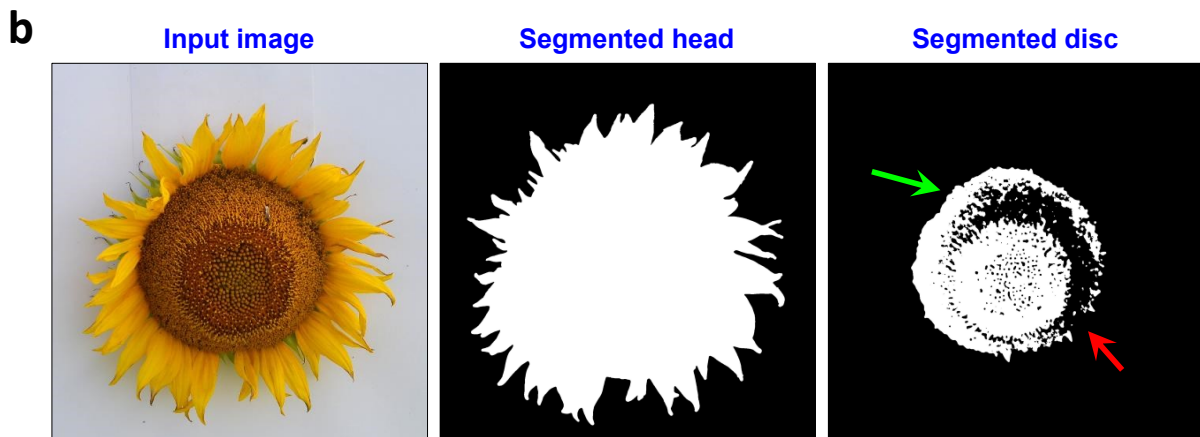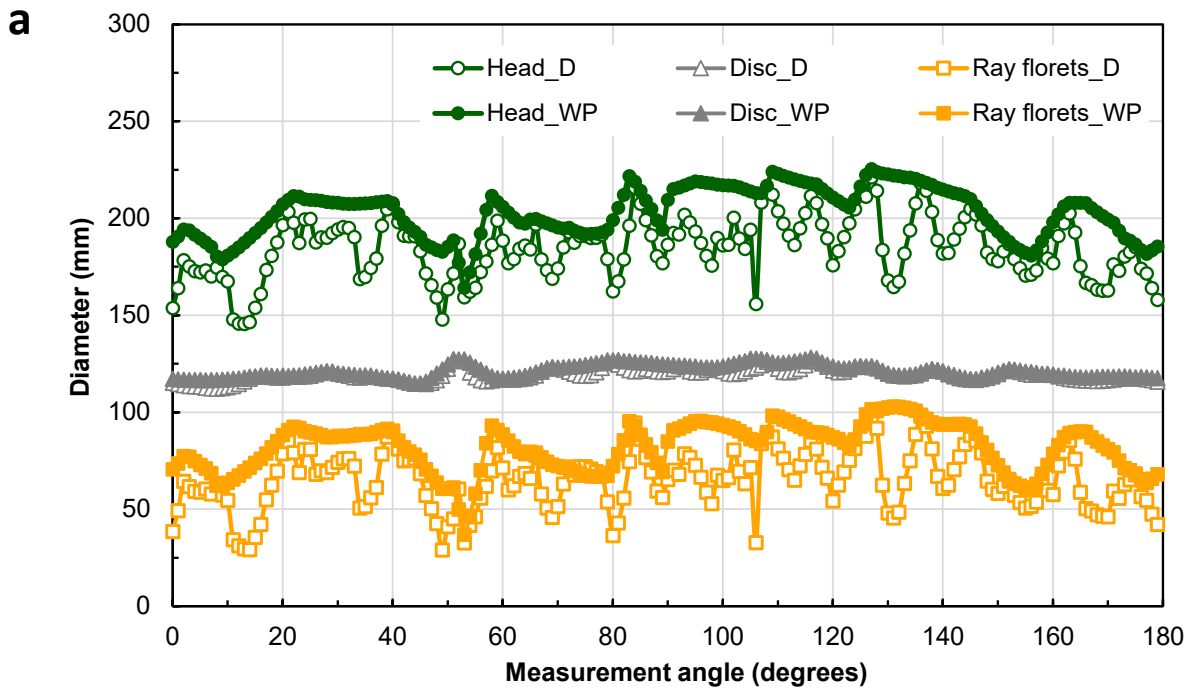
Figure 3.12. Advantage and limitation of the developed plugin (a) boundary profile plot showing diameter variations at $1°$ angle intervals for sunflower head, disc, and ray florets in direct (D) and wrapping-polygon (WP) methods that is advantageous to visualize and study the entire profile; (b) distorted head shape and improperly segmented binary image of disc due to excessive tilt of the sunflower. Green arrow indicates closed disc boundary; red arrow indicates unfilled open boundary that leads to incorrect disc filling.

between disc and ray florets, and misrepresented head shape (fig. 3.12b); (3) Shadows caused issues in sunflower head segmentation and result in overestimation; and (4) The developed plugin could handle only one flower at a time.

3.4.8. Suggestions for Future Work

Utilizing the present research as a tool to measure sunflower dimensions and studying the correlation between dimensions and pollinators' visitation rates will generate useful relationships that are worth investigating. For example, the relative effects of disc area versus the area of ray florets (petals) on pollinator visitation rates have not been investigated in cultivated sunflower and are not generally well understood in related plants. Some additional features can be implemented in the plugin such as counting/estimating the number of disc florets and ray florets as the number of these, in addition to total area might have a potential impact on pollinators' visitation rates. Because floret-feeding causes premature senescence (and darkening) of disc florets, it is also possible that image processing comparing colors could be used to rapidly score damage from floret- and seed-feeding caterpillars, which varies significantly among sunflower inbred lines (Prasifka & Hulke, 2016).

The proposed imaging protocol and the developed plugin can also be easily extended to other flowers of the Asteraceae family by adjusting the segmentation method and plugin inputs. From production agricultural point of view, the image processing application has the potential for use in counting the number of seeds in the disc as the count provides accurate seed yield estimates. The relationship between disc dimension and oil yield can also be studied. Furthermore, with the advancements in processing power of

recent smartphones, a mobile application, based on the scientific recommendation from the plugin, will be a great tool as it provides instant outputs and can be readily used in the field.

3.5. Conclusions

This study developed and tested an image processing ImageJ plugin to accurately measure sunflower components (head, disc, and ray florets) dimensions. A custom-designed background board and slit-enclosure sheet facilitated capturing sunflower images non-destructively at the field without clipping. The pixel-march method performed multiple radial measurements as per the user's input and produced the dimensions results. Two measurement methods, direct and wrapping-polygon, were tested with the latter considered efficient as it produced minimum absolute deviation and made the tip-to-tip measurements possible. The recommended number of measurements that best represented sunflower head and ray florets was 32, and for the disc was eight.

Dimensions of the sunflower head and disc obtained from pixel-march method were statistically similar to equivalent diameter (ED) as well as minor axis (MinA) methods of ImageJ. However, accurate dimensions of ray florets can be obtained only through this developed method as there were no standard methods available in ImageJ or in any other image processing software. The plugin took ≈ 11 s (Windows 10, Intel Core i5, and 8 GB RAM) to perform 180 measurements on all sunflower components for both methods.

Linear prediction models developed for dimensions of sunflower components based on the 180 measurements using ED and MinA produced the best performance for head and disc ($R^2 > 0.99$) than for ray florets ($R^2 > 0.76$). Linear prediction models for 180 measurement dimension and area using two orthogonal manual measurements performed better only for sunflower head ($R^2 > 0.92$) and are not suitable for disc and ray florets. The

developed plugin can be readily utilized as a tool for entomologists as useful relationships

between floral morphology and insect behavior, including pollinator visitation rates are

well established.

4. COLOR CALIBRATION OF DIGITAL IMAGES FOR AGRICULTURE AND OTHER APPLICATIONS *

4.1. Abstract

Image processing in agriculture relies primarily on correlating color changes within images' region of interest to specific quality attributes (e.g., plant phenology, plant health, crop stress, maturity). Changes in lighting conditions during image acquisition affect image color, even though there is no change in quality, and produces misleading inference when used without calibration. The focus of this study was to develop a method for calibrating images to make them homogeneous to improve phenological comparisons. The method was developed with synthetic images and validated with actual plant images in laboratory and field conditions using a standard ColorChecker (X-Rite) chart. Six different color schemes were tested to determine the effect of patch order, and minimum number of patches required for efficient calibration. A user-coded ImageJ plugin named "ColorCal" was developed in Fiji software for color calibration that derived and applied a $[3 \times 3]$ color calibration matrix, based on selected color patches and standard values. Modified total error and calibration performance index (CPI) were developed to evaluate calibration performance. Calibration using any 12 color patches taken in any order gave equal

performance ($0.14 \leq$ CPI ≤ 0.26). Calibration performance using only commonly followed neutral color patches (e.g., white, gray) was poor ($0.26 \leq$ CPI ≤ 1.0). Using red (R), green (G), and blue (B) color patches was recommended as it produced visually similar images, the performance was comparable with 24 color patches ($0.21 \leq$ CPI ≤ 0.24) and was simple and practical. The developed plugin took ≈ 7 s for calibration (Windows laptop, Intel Core i5, and 8 GB RAM). Determining phenological and other applications using the plugin was more reliable than using the raw images.

4.2. Introduction

Color of the digital image is an important feature in many image analysis applications; however, the calibration of images by users is still a great challenge. An image's true color is vital in obtaining the physical, chemical, and surface characteristics of the object from the image. Images captured using different cameras at the same time produce color variations due to the different spectral responses of the camera sensor, or due to the different digitizers of the cameras. Furthermore, images obtained using the same camera at different times of the day produce color variations, due to either the ambient temperature or the light irradiance at that time, despite no change in the object characteristics (Andresen et al., 2018). Such variations in color tend to critically impact the object's characteristics evaluation when color is considered as the major criterion in the image analysis. The only way to overcome this issue of color variations is to calibrate the pixels color values in the captured image.

Color calibration is broadly classified into two categories: (i) device-based, and (ii) image-based. Many researchers have explored device-based color calibration in which camera sensors were developed to automatically adjust to the color values based on the

scene lighting (M. Brown et al., 2005; Porikli, 2003). Image-based color calibration is used to compensate for changes in the image due to illumination conditions. This is the most common method in professional photography to maintain the same color tone in all scenes. It used a standard color calibration chart to linearly map the pixel values obtained in the image to the standard color values of the calibration chart (Kagarlitsky et al., 2009). A relationship was developed between the color values obtained in the image to the standard color values, which can then be applied to image pixels for standardization (calibration). Commercially, there were several calibration charts available (ranging from 6 to 100s of color patches); however, the 24 color patch calibration chart was widely used in many applications (McCamy et al., 1976).

Many fields of study, such as space (Levin & Levin, 2004), medical (Haeghen et al., 2000), underwater biology (Akkaynak et al., 2011), and agriculture (Scharf & Lory, 2002) have had to solve the color variation issues while processing images. In some studies using satellite and aerial images, radiometric calibration was performed by linearly relating the known reflectance of ground control points (e.g., road, concrete roof, ocean, sand) to the reflectance of corresponding points in the image (Xu, Hou, & Tokola, 2012). This simplified calibration was performed for individual channels (red, green, blue, and infrared) to obtain the calibrated reflectance. However, comprehensive information on color calibration of images related to agricultural applications is limited. L. Brown et al. (2012) used HSI color space and derived a parameter called color strength, which was used to improve color correction. Akkaynak et al. (2014) developed a color calibration using all 24 color patches in the calibration chart by transforming to XYZ color space, and later back-transforming to RGB (red, green, and blue) color space in underwater images.

Over the last decade, application of digital image processing in agriculture has been well researched. The major application in agricultural research was mainly focused on color-based identification/classification (Beeri & Peled, 2009). The usage of unmanned aerial vehicles and phenocam systems in agriculture has increased the use of digital color images to assess plant phenology and health (O'Connell, Bradter, & Benton, 2015; van Iersel, Straatsma, Addink, & Middelkoop, 2018). Because the images are taken in open atmospheric conditions, the brightness and darkness caused by clouds and the time of the day result in wide color variations. Barbedo (2016) reported lighting variation during image acquisition in agricultural fields is the most common issue in image segmentation and feature extraction and it is inevitable. Thus, color-based characteristics gathered from such digital images to assess crop stress, nutrient deficiency, and crop productivity would lead to inaccuracies due to inconsistent lighting conditions.

Researchers generally use various calibration boards with color patches of known color value in agricultural fields. A few studies on calibration boards used in phenological comparisons include white, gray, and black patches in estimating the leaf emergence in a beech forest ecosystem (Ahrends et al., 2008); gray painted board in estimating spring green-up and autumn senescence in a deciduous forest ecosystem (Richardson et al., 2009); and five color boards, such as red, green, black, gray, and beige in estimating biomass and nitrogen levels in a cornfield (Hunt et al., 2005). However, the color rendering efficiency of these color patches was not investigated, and the calibration accuracy and the methodology followed in those studies remain unknown.

The overall research goal of this study was to produce calibrated images from the images acquired under varying lighting conditions. Such calibrated images will serve as

better input images for further analysis and interpretation of various plant characteristics (e.g., plant health, phenology). However, there is no standardized method for image-based color calibration. Therefore, this research was undertaken with the following objectives: (i) develop an open source user-coded ImageJ plugin for color calibration of digital images; (ii) develop calibration performance measures to evaluate the calibrated images; and (iii) determine and recommend a minimum number of color patches for efficient color calibration.

4.3. Materials and Methods

Several processing stages to obtain the calibrated image suitable for further analysis are summarized in figure 4.1. In the development process, a synthetic image (two leaves) printed from FreePNGimg (2018) were used to evaluate the effect of color patch order and to determine the minimum number of color patches for calibration. Reproducibility and simplicity were the reasons for using the synthetic image in the method development process. Actual plant images acquired in a laboratory environment (cut cilantro under controlled lighting), and in field conditions (actual plants in the field under natural lighting) were used to validate the method developed.

Images were taken with a DSLR camera (Model: Nikon D5100, Nikon Corp., Japan) with the following camera settings: exposure mode = manual, shutter speed = $1/250\,\text{s}$, aperture = $f/7.1$, ISO = 250, focal length = $18\,\text{mm}$ along with a wireless shutter release remote for hands-free imaging and avoiding camera shake. To calibrate, it is important to include a standard color calibration chart of known color values in the image. We used the standard 24-patch ColorChecker classic target (X-Rite Inc., Grand Rapids, MI, USA) in this study.

```
            ┌─────────┐
            │  Start  │
            └─────────┘
                 │
                 ▼
┌────────────────────────────────────────────────┐
│       Load input image with calibration chart.  │
└────────────────────────────────────────────────┘
                 │
                 ▼
┌────────────────────────────────────────────────────────┐
│ Mark corners of calibration chart using polygonal       │
│ selection tool.                                          │
└────────────────────────────────────────────────────────┘
                 │
                 ▼
┌────────────────────────────────────────────────────────┐
│ Choose color schemes/color patch number for calibration.│
└────────────────────────────────────────────────────────┘
                 │
                 ▼
┌────────────────────────────────────────────────────────┐
│ Derive color calibration matrix (CCM) based on chosen   │
│ color patches and their respective standard values.     │
└────────────────────────────────────────────────────────┘
                 │
                 ▼
┌────────────────────────────────────────────────────────┐
│ Apply CCM to individual pixel in input image to         │
│ produce a calibrated image.                             │
└────────────────────────────────────────────────────────┘
                 │
                 ▼
┌────────────────────────────────────────────────────────┐
│ Evaluate performance measures such as total error       │
│ (eₘ), difference image index (DII), and calibration     │
│ performance index (CPI).                                │
└────────────────────────────────────────────────────────┘
                 │
                 ▼
            ┌─────────┐
            │   End   │
            └─────────┘
```

The last process box reads: Evaluate performance measures such as total error (e_m), difference image index (DII), and calibration performance index (CPI).

Figure 4.1. Overall process flowchart of digital image color calibration.

4.3.1. Image Acquisition

4.3.1.1. Synthetic image from internet

An image of the front and back side of two different leaves was selected from the FreePNGimg (2018) stock pictures. A color printout of the leaves image along with the original ColorChecker classic target beside was arranged on a white poster sheet for imaging under different lighting conditions, and these served as the synthetic image. Two benchtop fluorescent lights (65 W, 4300 lm, Model: Designers Edge L-2006, Coleman Cable Inc., Waukegan, IL, USA) were used for lighting, and the lighting conditions were varied by choosing one or two lights at a time, as well as by arbitrarily altering the distance between the object and the lights. The combinations of the number and distance of lights

used produced the necessary lighting levels ranging from dark (dim) to bright (well-lit) conditions for image capture.

The camera was placed on a tripod at a height of 900 mm facing down, and three images (replication) were captured at each lighting condition. This arrangement ensured crisp images without disturbing the object position for each image while varying only the lighting condition.

4.3.1.2. Actual plant images from laboratory

Laboratory images of the actual plants were acquired using bunches of cut cilantro (*Coriandrum sativum* L.). The sample was purchased fresh locally and stored in the refrigerator until used for imaging. Before image acquisition, the cilantro bunches were rolled on a paper towel to remove moisture. Bunches were then arranged in a recyclable paper cup so that the top portion of the bunch represented the canopy of the standing plant viewed from the top. The color calibration chart was placed alongside and the images were captured. The camera settings and lighting conditions were maintained the same as in synthetic image acquisition.

4.3.1.3. Actual plant images from field

Field images were acquired using experimental plots at the Northern Great Plains Research Laboratory, Mandan, ND (46.81° N, 100.92° W). The camera was fitted on a tripod facing north at a distance of $\approx$2.7 m ($\approx$9 ft) from the field plot boundary. The color calibration chart was placed near the plot edge and the images were captured.

In general, the lighting variations in the image are due to the intensity of ambient lighting that hits the camera sensor. The lighting conditions were simulated by adjusting camera aperture setting (f-stop). Typically in a professional DSLR (digital single lens

reflex) camera, the aperture setting varies from $f/3.5$ to $f/22$. This range indicates the size of aperture opening, with $f/3.5$ represents wide opening (more light to sensor — lighter image), and $f/22$ represents narrow opening (less light to sensor — darker image).

However, the two extreme ranges were not necessary for our application as it produced an overexposed image at $f/3.5$, and an underexposed image at $f/22$. Hence, the aperture range between $f/6.3$ and $f/16$ were used, and kept the rest of the camera settings constant (shutter speed $= 1/250$ s, ISO $= 250$, and focal length $= 18$ mm).

4.3.2. Color Calibration Process

Images obtained at varying lighting conditions produced variations in the pixel color intensities and resulted in color casts (unwanted tint on the whole or part of the image). The color calibration eliminated such color casts and reduced the variability among the calibrated images. The calibrated images represented images acquired at a uniform lighting condition, which are suitable for further color-based analysis.

The color calibration chart included in each input image was used to map the average RGB values of each color patch (of the color chart in the image) to the standard RGB values of the corresponding color patch as supplied by X-Rite (X-Rite, 2018). The calibration essentially means to bring the RGB values of color patches in the input image to the standard RGB values. A color calibration matrix (CCM) was derived based on this mapping and when applied to the entire image produced the calibrated image.

In matrix notation, the process of image color calibration is expressed as:

$$\mathbf{S}_{\text{RGB}} = \mathbf{M}_{\text{CC}} \times \mathbf{I}_{\text{RGB}} \qquad\qquad (4.1)$$

where $\mathbf{S}_{\mathrm{RGB}}$ is a $3 \times N$ matrix of standard RGB values from N color patches obtained from

X-Rite, $\mathbf{M}_{\mathrm{CC}}$ is the 3×3 CCM to be derived, and $\mathbf{I}_{\mathrm{RGB}}$ is a $3 \times N$ matrix of average RGB

values from the corresponding N color patches in the input image.

Rearranging eq. 4.1 by multiplying the transpose of $\mathbf{I}_{\mathrm{RGB}}$ on both sides, and taking

inverse determined $\mathbf{M}_{\mathrm{CC}}$ as follows:

$$[\mathbf{S}_{\mathrm{RGB}} \times \mathbf{I}_{\mathrm{RGB}}^{\mathrm{T}}] = \mathbf{M}_{\mathrm{CC}} \times [\mathbf{I}_{\mathrm{RGB}} \times \mathbf{I}_{\mathrm{RGB}}^{\mathrm{T}}]$$

$$\mathbf{M}_{\mathrm{CC}} = [\mathbf{S}_{\mathrm{RGB}} \times \mathbf{I}_{\mathrm{RGB}}^{\mathrm{T}}] \times [\mathbf{I}_{\mathrm{RGB}} \times \mathbf{I}_{\mathrm{RGB}}^{\mathrm{T}}]^{-1} \qquad (4.2)$$

where $\mathbf{I}_{\mathrm{RGB}}^{\mathrm{T}}$ is the transpose of $\mathbf{I}_{\mathrm{RGB}}$. The above equations (eqs. 4.1 and 4.2) work for any

number of color patches (1, 2, 3, ... N) and the resulting $\mathbf{M}_{\mathrm{CC}}$ is always a 3×3 matrix.

Irrespective of the number of color patches considered, the $\mathbf{M}_{\mathrm{CC}}$ need to determine only

9 coefficients (3×3). This leads to underdetermined ($N < 3$) and overdetermined ($N > 3$)

systems of equations equation 4.2 and for which the least squares approximation solution

approach was followed. Thus, the derived $\mathbf{M}_{\mathrm{CC}}$ when multiplied to the individual pixel

values (RGB) in the input image produced the calibrated image. Unlike the simplified

radiometric calibration with linear interpolation of individual channels (Xu et al., 2012),

this calibration procedure with 3×3 CCM will have interaction among the RGB channels

starting from a single color patch.

4.3.3. Color Calibration Plugin (ColorCal) Development

A user-coded color calibration plugin named "ColorCal" was developed in Fiji (Ver.

ImageJ 1.52d, Java 1.6, lifeline version, Rasband (2018)), an open source image

processing software. Fiji is a distribution of ImageJ that has core ImageJ applications and

extends additional functionality for plugin development (Schindelin et al., 2012).

The ColorCal plugin had 925 lines of Java codes was divided into two major sections: (i) semi-automatic color patch selection from the color chart for average RGB values extraction, and (ii) CCM derivation and image calibration.

4.3.3.1. Semi-automatic color patch selection

The primary step in a color calibration process extracts the average RGB values of individual color patches of the calibration chart included in the input images. If the camera and calibration chart are placed in a fixed position in a scene, the color values will be extracted by either overlaying a region of interest (ROI) mask on the input image or through the coordinates of ROI of each color patch, which is known and can be hard-coded into the program. However, maintaining an accurate fixed location of calibration chart with respect to the camera is not always feasible.

In our experiments, as the calibration chart was placed at different positions among the images, a semi-automatic method was developed for color patches ROI extraction with the least inputs from the users. Using this method, the user selects only the four overall corner points of the calibration chart (**A**, **B**, **C**, **D**; fig. 4.2) in the input image using ImageJ's polygonal selections tool, and the ROI for each color patch with its average RGB values ($\mathbf{I}_{RGB}$; eq. 4.2) were extracted for determining the $\mathbf{M}_{CC}$. The detailed derivation of this developed method is discussed subsequently. It is also possible to automate the color patch selection through feature descriptors (e.g., SURF, SIFT, FAST) and machine learning algorithms; however, that requires additional software to implement the algorithm and will be complicated to integrate in this ImageJ plugin development.

After loading the acquired image into ImageJ, the user should select only the corner

four points (A, B, C, D; fig. 4.2) of the calibration chart using ImageJ's polygonal

selections tool. These points do not always form the corners of a rectangle because of

perspective distortions (rectangle $\rightarrow$ quadrilateral) during image acquisition; however, the

plugin handled this perspective distortion. Such distortions actually produce a

quadrilateral, instead of a rectangle, with unequal sides ($\mathsf{AB} \neq \mathsf{CD}$, and $\mathsf{AD} \neq \mathsf{BC}$).

Because of the unequal sides, the distances between intermediate points, representing the

color patches, on four sides were also not equal (e.g., $\mathsf{AE} \neq \mathsf{BE'}$), but were proportional to

their respective overall side lengths (AD and BC). These distortions, however, were

minimal as efforts were made to arrange the color chart perpendicular to the camera in the

image. Furthermore, the developed method handles the possible distortion by

appropriately identifying the starting corners of each color patch considering the

proportionate distortion of the overall dimensions of the chart in the image directly using

simple geometrical principles subsequently described. Using this methodology, the

locations of individual color patches were derived from the minimal inputs (A, B, C, D;

fig. 4.2) and the average color values were extracted for further analysis.

The ColorChecker classic calibration chart consisted of 24 color patches arranged in

a 4×6 array with each $40\,\mathrm{mm} \times 40\,\mathrm{mm}$ square (fig. 4.2). All these color patches were

separated by a uniform gap (black stripes). Based on proportionality, the ratio of gap and

patch size for the calibration chart image and actual were assumed to be equal because of

minimal distortion and this ratio was determined as:

$$\left(\frac{g_{w,i}}{w_{\mathrm{cp},i}} = \frac{g_{wa}}{w_{\mathrm{cpa}}} \right) = \left(\frac{g_{h,\mathrm{L}}}{h_{\mathrm{cp,L}}} = \frac{g_{h,\mathrm{R}}}{h_{\mathrm{cp,R}}} = \frac{g_{ha}}{h_{\mathrm{cpa}}} \right) = R_{\mathrm{gw}} = \frac{5\,\mathrm{mm}}{40\,\mathrm{mm}} = 0.125 \qquad (4.3)$$
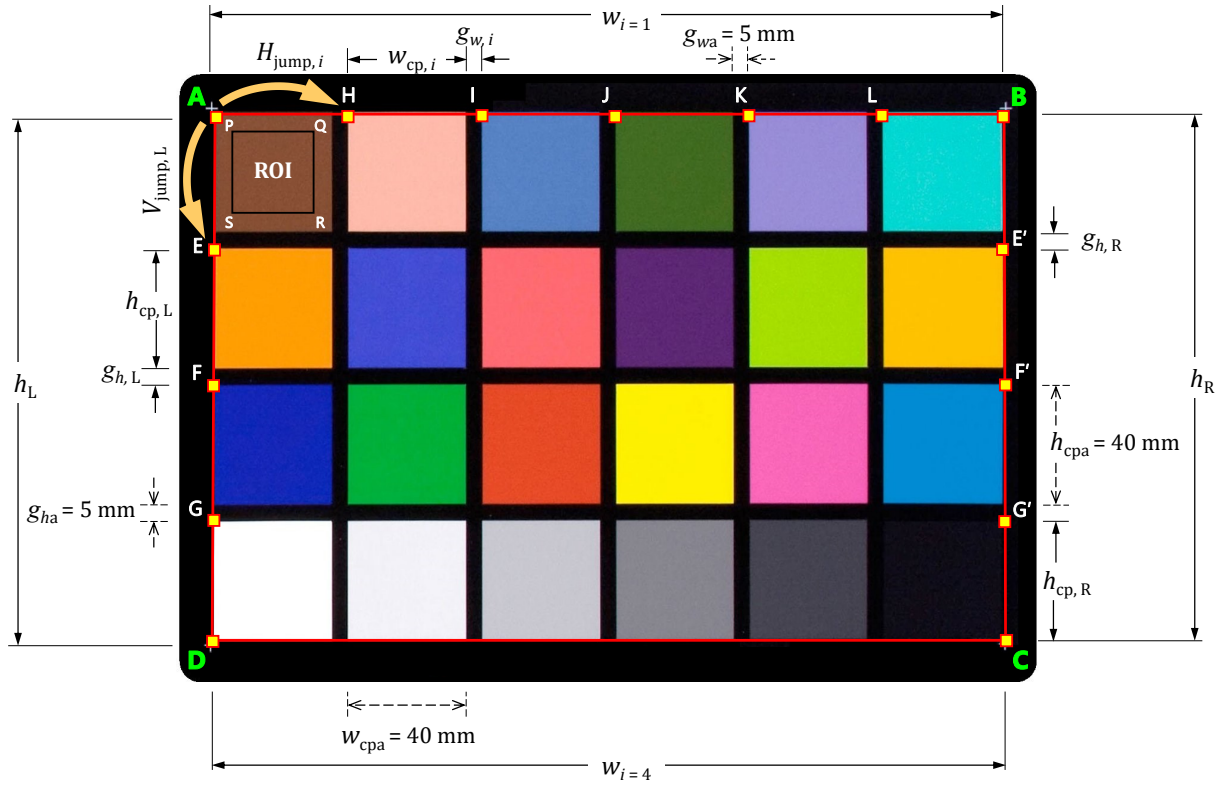
Figure 4.2. Illustration of the semi-automatic color patch selection from user input polygon coordinates (A, B, C, D). The color patch (P, Q, R, S) is the final size of the ROI applied to all patches that was used in color value extraction. Refer to the section text for the definition of the symbols.

where $g_{w,i}$ and $w_{cp,i}$ are the widths of gap and patches in the ith row; g_{wa} and w_{cpa} are the actual width of gap and patch; $g_{h,L}$, $h_{cp, L}$, $g_{h,R}$, and $h_{cp, R}$ are the heights of gap and patch on left and right side, respectively; g_{ha}, and h_{cpa} are the respective actual height of the gap and patches; and the ratio R_{gw} is a constant for the calibration chart.

      Overall, the process of fixing the region of interest (ROI) for all the color patches in the image (fig. 4.2) from the four user input coordinates (A, B, C, D) follows these steps: (1) the top-left corners of color patches along the left side, along AD, were established, (2) similarly the corners of the right side, along BC, were established, (3) connecting the corresponding end corners of the established left and right edges the corners of patches

88

along the horizontal directions (4 rows), all the corners of the patches were determined, and (4) ROIs of reduced size were fixed based on the corner coordinates for each color patch with necessary shift (fig. 4.2) to extract the color information. This procedure addresses the issue of possible shape deviation of the color chart to a "quadrilateral" in the image from the actual "rectangular" shape of the chart.

Based on the corner coordinates (A, B, C, D) of the color chart, the distances (h_L, and h_R) and the slopes (m_L and m_R) of left and right sides were calculated using distance and slope formula, respectively. Because of possible distortions, the $h_L \neq h_R$, and the distances between intermediate points were also not equal (e.g., $AE \neq BE'$). However, they will be proportional to their respective side lengths. The height of color patches on left and right ($h_{cp,\,L}$ and $h_{cp,\,R}$) were found by:

$$h_{cp,\,L} = \frac{h_L}{4 + 3R_{gw}} = \frac{h_L}{4.375} \tag{4.4}$$

$$h_{cp,\,R} = \frac{h_R}{4 + 3R_{gw}} = \frac{h_R}{4.375} \tag{4.5}$$

As seen in figure 4.2, the coordinate E is $V_{jump,\,L}$ (eq. 4.6) distance from A. Therefore, the factor $V_{jump,\,L}$ (eq. 4.7) on left side was calculated using equation 4.4 and simplifying for $g_{h,L}$ in equation 4.3, with similar approach along BC for $V_{jump,\,R}$, as:

$$V_{jump,\,L} = g_{h,L} + h_{cp,\,L} = 0.257 \times h_{cp,\,L} \tag{4.6}$$

$$V_{jump,\,R} = g_{h,R} + h_{cp,\,R} = 0.257 \times h_{cp,\,R} \tag{4.7}$$

The line **AD** may deviate from the vertical and this interferes with the simple positioning of the row starting coordinates (**E**, **F**, and **G**). Therefore, the inclination of the line segment **AD** having the slope m_L was used and the coordinates were determined geometrically from the intersection of the line **AD** and a circle with center at **A** having the radius of $V_{jump, L}$. By solving line and circle equations analytically, the coordinates of **E** (**E**$_x$, **E**$_y$) were derived as:

$$E_x = \frac{V_{jump, L}}{\sqrt{1 + m_L^2}} \tag{4.8}$$

$$E_y = m_L \times \frac{V_{jump, L}}{\sqrt{1 + m_L^2}} \tag{4.9}$$

Using equations 4.8 and 4.9, other points on **AD** (**F** and **G**) were obtained using the $V_{jump, L}$ from their previous points **E** and **F**, respectively, assuming $V_{jump, L}$ is constant along **AD**. Likewise, **E′**, **F′**, and **G′** were obtained starting from **B** with the calculated $h_{cp, R}$ (eq. 4.5), $V_{jump, R}$ (eq. 4.7), and m_R when substituted correspondingly in equations 4.8 and 4.9. These end points when connected by a line segment (e.g., **EE′**, **FF′**, and **GG′**), thus established the rows of the calibration chart.

Moving through each row ($i = 1$ to 4), the top-left corners of individual color patches (**H**, **I**, **J**, **K**, and **L**) were obtained after evaluating the corresponding color patch widths and jumps. The width of each color patch at each row was calculated as:

$$w_{cp,i} = \frac{w_i}{6 + 5R_{gw}} \tag{4.10}$$

where $w_{\text{cp},i}$ is width of the color patch at ith row ($i = 1, 2, 3, 4$) of calibration chart; w_i is the width of ith row (AB, EE′, FF′, and GG′). Similar to $V_{\text{jump, L}}$, the top-left corner H is $H_{\text{jump},\,i}$ distance from A. Therefore, the factor $H_{\text{jump},i}$ was calculated using equation 4.10 and simplifying for $g_{w,i}$ in equation 4.3 and we obtain:

$$H_{\text{jump},i} = g_{w,i} + w_i = 0.170 \times w_i \tag{4.11}$$

The point H is $H_{\text{jump},i}$ distance from A, along the line segment AB with slope m_i. Therefore, similar to equations 4.8 and 4.9, the coordinates of H was determined as:

$$H_x = \frac{H_{\text{jump},\,i}}{\sqrt{1 + m_i{}^2}} \tag{4.12}$$

$$H_y = m_i \times \frac{H_{\text{jump},\,i}}{\sqrt{1 + m_i{}^2}} \tag{4.13}$$

Similarly, marching through each row, the top-left corners were established from its previous points (e.g., H, I, J, K), thus, establishing all the locations of color patches. It was again assumed that for a given row (i) the $H_{\text{jump},\,i}$ as a constant.

An ROI (e.g., PQRS; fig. 4.2) was overlaid on each color patch for color values extraction. The ROI was positioned on the center of the color patch based on the size of each color patch, and the ROI size was fixed as 70 % of the color patch size (based on preliminary observations to avoid the edges).

The algorithm started from the top-left color patch and advanced column-wise through each row and extracted the average RGB values. These values were used in the CCM derivation.

4.3.3.2. Color calibration matrix derivation in plugin

Matrix manipulations such as transpose and matrix multiplication were available through appropriate methods in the `RealMatrix` class from Apache Commons Mathematics (Ver. 3.6) package (ASF, 2018). The matrix inverse was calculated using the singular value decomposition method available in Java class `SingularValueDecomposition` and interface `DecompositionSolver`. These methods from the Apache Commons Mathematics were useful in performing the necessary matrix operations of equation 4.2.

The average RGB values of each extracted color patch ($\mathbf{I}_{RGB}$) and the standard RGB values as supplied by the X-Rite ($\mathbf{S}_{RGB}$; X-Rite (2018)) were stored in respective `RealMatrix` objects (eq. 4.1). These two matrices were solved to obtain the CCM ($\mathbf{M}_{CC}$) as shown in equation 4.2. Each pixel RGB values $[1 \times 3]$ of the input image when multiplied with the derived CCM $[3 \times 3]$ returned the calibrated RGB values $[1 \times 3]$ of the corresponding pixel.

Sometimes, while multiplying CCM with each pixel values of the input image, the resulting calibrated pixel value fell beyond the range of 0 to 255. Such pixel values produced a false-colored image and were rectified. Reassigning such "out of the range" values to the respective limiting pixel values as 0 and 255, with appropriate plugin code, solved this issue.

4.3.4. Minimum Number of Color Patches for Calibration

Usage of all 24 color patches of the standard color chart for color calibration is expected to produce good calibration. Even though the analysis CPU times were similar (3.27±0.16 s for CCM; and 7.28±0.25 s for overall) among the number of patches ($N = 1$ to 24), in other practical applications (e.g., unmanned aerial vehicle images, online color-based sorting system) usage of all 24 color patches for calibration becomes tedious.
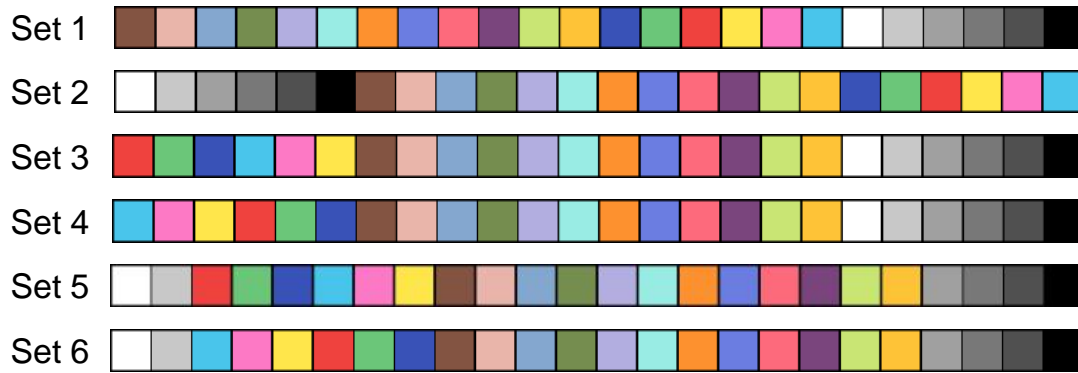
Figure 4.3. Order of color schemes used to determine the minimum number of color patches required for efficient color calibration.

Therefore, in the present study, the minimum number of color patches required for color calibration was determined. For this purpose, six different color schemes ("Set 1–6"; fig. 4.3) were used with the numbers of color patches considered were progressively increased one by one, and their calibration performances were calculated and compared. Choosing a combination of color scheme and the number of color patches results in a unique group of colors available for image calibration.

Out of six color schemes, two color schemes ("Set 1" and "Set 2"; fig. 4.3) suggested by Akkaynak et al. (2014) were considered, and also four new color schemes ("Set 3–6"; fig. 4.3) were developed. The color scheme Set 1 was the standard color scheme provided by the X-Rite (X-Rite, 2018). The color scheme Set 2 started with the six neutral colors in the 4th row of the calibration chart (fig. 4.2) followed by the other 18 color patches in the regular order.

The basis for considering new color schemes was as follows: The color scheme "Set 3" started with the additive colors (R, G, and B), followed by subtractive colors (cyan: C, magenta: M, and yellow: Y), then followed by the regular color patch order. The combination of different ratios of additive colors (R, G, and B) produced a wide color

93

gamut and has been used in many image processing applications, camera sensors, televisions, and monitors. Similarly, the use of subtractive colors (C, M, and Y) produced a wide range of colors as used in commercial printers to produce the same color printed output as seen in the display monitor. Hence, the first 6 colors of Set 3 and "Set 4" were based on the combination of RGB, and CMY color patches, followed by the regular color patch order of the chart.

The first two color patches of "Set 5" and "Set 6" were based on the previous studies on phenocam network (Ahrends et al., 2008; Richardson et al., 2009), white and gray patches were considered primarily, which was same as first two color patches in Set 2. The effect of RGB and CMY color patches after white and gray color patch was observed in Set 5 and Set 6, respectively. The calibrated images produced by different color schemes were evaluated by different developed performance measures to determine the suitable color scheme and the minimum number of color patches required for efficient color calibration.

4.3.5. Calibration Performance Measurement

To evaluate the performance of the calibration process, the resulting calibrated image quality had to be accessed based on the uniformity in color values among calibrated images. Manual observation cannot discern the amount of color differences among the calibrated images. Therefore, along with the available performance measure (total error; Akkaynak et al. (2014)), a new measure named "calibration performance index" (CPI) was developed, using a "difference image index" (DII) by comparing two images, for evaluating the calibration performance.

4.3.5.1. Modified total error

The calibration process alters the RGB values of color patch in the input image ($\mathbf{I}_{RGB}$; eq. 4.1) closer to the standard RGB values ($\mathbf{S}_{RGB}$; eq. 4.1). The total error is a measure of closeness that the calibrated image has attained with respect to the standard reference RGB values of the chart. The modified expression of total error (Akkaynak et al., 2014) after normalization is given as:

$$e_m = \frac{\sum_{i=1}^{N=24} \sqrt{(R_s - R_i)^2 + (G_s - G_i)^2 + (B_s - B_i)^2}}{N \times (MD_R + MD_G + MD_B)} \tag{4.14}$$

where e_m is the modified total error (range between 0.0 and 1.0); N is the number of color patches in the ColorChecker and is fixed at 24; R_s, G_s, and B_s are the standard reference values of the individual color patch in ColorChecker (supplied by X-Rite); R_i, G_i, and B_i are the R, G, and B values of corresponding color patch in the calibrated image; and MD_R ($255 - 8 = 247$), MD_G ($255 - 52 = 203$), and MD_B ($255 - 31 = 224$) are the maximum difference of ColorChecker standard reference values obtained from 255 and the minimum value of the 24 patches, for R, G, and B, respectively.

A perfect calibration, where RGB values of all color patches in the calibrated image exactly match the standard RGB values, produced e_m of 0.0, and obtaining e_m of 1.0 is unlikely for actual images as the denominator of equation 4.14 is a constant and values might not represent most natural images. Attaining $e_m = 0.0$ is practically impossible; however, any lower value (close to 0.0) is desirable.

4.3.5.2. Calibration performance index

The CPI is a developed measure of how well the two input images of varied lighting conditions (dark, and light) are calibrated. The CPI was developed to test the calibration performance of different color schemes and the number of color patches selected for calibrating the input images. For this purpose, two synthetic images of extreme practical lighting variation (dark and light) were used as the fixed input, as the results from these included any other intermediate lighting variations.

As images have to be compared for CPI, a method of comparing two images and expressing the difference numerically was developed using a DII. For comparing two images, a "difference image" was first obtained using the `ImageCalculator()` class of ImageJ, which was the "bound checked" RGB values (0 – 255) of pixelwise difference between two images. From this difference image the DII was evaluated as:

$$\text{DII} = \frac{\sum_{i=0}^{W} \sum_{j=0}^{H} \left(\text{DI}_{\text{R},i,j} + \text{DI}_{\text{G},i,j} + \text{DI}_{\text{B},i,j} \right)}{W \times H \times 3 \times 255} \tag{4.15}$$

where DII, the difference image index, is a normalized value ranging between 0 and 1, where 0 indicates no difference between the two images, and 1 indicates the images are totally opposite; i and j are the pixel coordinates in an image; W and H are width and height of the difference image; and DI_{R}, DI_{G}, and DI_{B} are difference image's R, G, and B values of individual pixels.

For CPI measure, the plugin took 2 input images (dark and light), and generated 4 output images (2 calibrated images with N patches, and 2 reference calibrated images with 24 patches) and evaluated DII between images. The reference calibrated images were

obtained using the maximum number of patches ($N = 24$) from Set 1 as they were

reported to be the best (Ilie & Welch, 2005; Joshi, Wilburn, Vaish, Levoy, & Horowitz,

2005). The CPI was calculated as a summation of the DIIs obtained between images among

the calibrated images as:

$$CPI = W_1 \cdot DII_A + W_2 \cdot DII_B + W_3 \cdot DII_C \tag{4.16}$$

where W_1, W_2 and W_3 of equation 4.16 are the respective weights attached to DII values of

DII_A (between calibrated dark and reference calibrated dark images), DII_B (between

calibrated light and reference calibrated light images), and DII_C (between calibrated dark

and light images), respectively.

From all the combinations of color patch order and number of patches, the

maximum value of CPI was evaluated. The maximum CPI value was used to normalize the

CPI values to have a range from 0.0 to 1.0. Thus, the best calibration method will have the

lowest CPI value and a poor calibration a value towards 1.0.

4.3.6. Features of the ColorCal Plugin

The flowchart of the ColorCal plugin capable of performing (i) calibration,

(ii) comparison, (iii) calibration and comparison, and (iv) comparison with reference,

based on user inputs through options, is illustrated in figure 4.4A. The plugin is operated

from the front panel by selecting the appropriate options (fig. 4.4B). The options allow the

users to select the type of analysis ("Calibration," "Comparison," "Calibration and

comparison," and "Comparison with reference"), color schemes (Set1 – Set6 and

"User-defined") through calibration options, and output options ("Display color calibration

matrix" and "Display calibrated image"). Along with the six pre-defined color schemes

(fig. 4.3), the User-defined color scheme option allows the user to enter any number of color patches in any order in the text box provided (using numbers 1 to 24 separated by commas) for experimentation and testing. While the "Run" button executes the process and produces the outputs, the "Help" button displays helpful information about the running of the plugin. The pseudocode of the ColorCal plugin is provided in Appendix B.4.

As the plugin output, the user can have either the CCM or the calibrated image or both as output by choosing appropriate output options. All the inputs choices will output an appropriate calibration performance measure in the textual form.

If the user chooses the Calibration option, then the plugin takes one input image and calibrates it with the color scheme and outputs e_m performance measure (eq. 4.14) and displays the calibrated image if the output option was chosen. With "Comparison" option, the plugin takes two input images, creates the difference image and outputs the DII performance measure (eq. 4.15). The comparison can be performed with any two input images, either with raw or calibrated. When both Calibration and Comparison options are chosen, the plugin takes two input images, calibrates both images with the specified color scheme, and produces the DII between the two calibrated images (fig. 4.4).

A smaller DII (darker difference image) indicates better calibration performance; however, this will be misleading, especially when two poorly calibrated but similar images will have the smallest DII. To address this issue, the "Comparison with reference" option (fig. 4.4) was included in the plugin that took two input images, produced the calibrated images with the specified color patches (N), also generated the reference calibrated images ($N = 24$ color patches), compared between the calibrated and reference calibrated images,
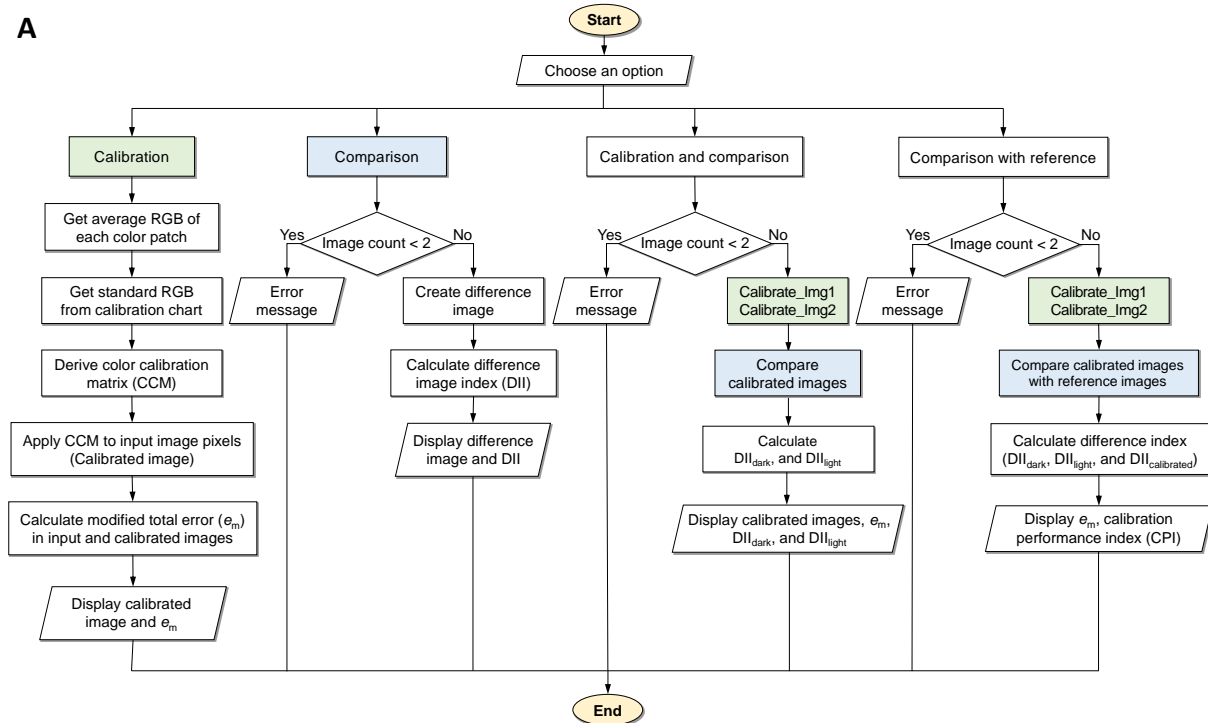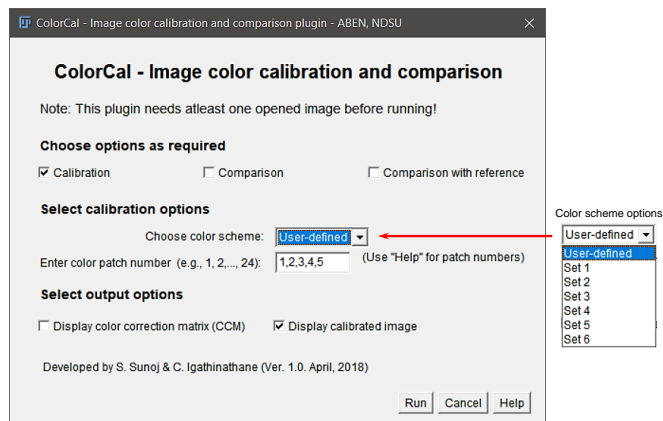
Figure 4.4. Features of the ColorCal plugin; (A) flowchart showing various operations performed by the plugin with the given user inputs; and **(B)** front panel of the developed plugin with user input and output options.

99

and produced the CPI (eq. 4.16). Thus, CPI is an overall performance measure which does not suffer the misleading comparison produced by DII.

4.3.7. Plugin Validation using Actual Plant Images

The ColorCal plugin was developed using the synthetic image and the calibration process was evaluated to determine the best performing color scheme and the efficient number of color patches. The best color scheme and the minimum color patches, selected based on performance measures, were validated with the actual plant images from laboratory and field conditions. The intensity variation among the calibrated images was also studied to verify the image calibration process.

4.4. Results and Discussion

4.4.1. Plugin Outputs and Interpretations

The input synthetic images obtained from dark and light conditions, their corresponding calibrated images from the developed plugin; and the difference image between dark and light input images, and dark and light calibrated images are presented in figure 4.5. The shown calibrated images were obtained by choosing Set 1 option from the plugin front panel, which selected all 24 color patches for calibration (fig. 4.4B). Previous studies (Ilie & Welch, 2005; Joshi et al., 2005) suggested that using 24 color patches in the calibration chart results in the best calibrated image, and is followed in professional photography as well. An example of CCMs derived from dark and light input images (eq. 4.17), obtained through display CCM option in the front panel (fig. 4.4), shows the dominance of the principal diagonal elements with the dark and light image CCMs:

$$\mathbf{M}_{\mathrm{CC\_dark}} = \begin{bmatrix} 1.06 & 0.09 & 0.02 \\ 0.07 & 1.16 & -0.09 \\ 0.09 & 0.08 & 1.02 \end{bmatrix} ; \qquad \mathbf{M}_{\mathrm{CC\_light}} = \begin{bmatrix} 0.87 & -0.07 & -0.23 \\ -0.03 & 0.79 & -0.14 \\ -0.06 & -0.13 & 0.83 \end{bmatrix} \qquad (4.17)$$

As seen in figure 4.5, leaf color in the dark input image almost looked black due to the dark lighting and caused e_{m} of 0.26; however, the calibration process brought the leaf color to green and reduced the e_{m} to 0.06. Similarly, the e_{m} was reduced from 0.06 to 0.02 in the light condition. The difference images shown in the third column were obtained by choosing the Comparison option after loading the two input images (dark and light), and the two corresponding calibrated images, respectively.

The difference images served as an illustration of the intensity difference between the images; however, the difference images worked only when both images were the same with matching location, which was obtained without changing the camera position. It can be seen that the darker the difference image, the greater was the similarity between input images, and vice versa. The difference image between the two input images was bright (DII = 0.38) because of a wide difference in input lighting conditions, but the difference image between the two calibrated images was dark (DII = 0.13) because the difference was less between the calibrated images. This demonstrates the process of calibration in bringing the input images towards a common intensity.

The calibration process can be better visualized with the 3D scatter plot shown in figure 4.6. Each red marker denotes the normalized RGB values of each color patch in the calibration chart (standard values), and the blue marker denotes the normalized average

	Dark	Light	Difference image
Input image	Modified total error = 0.26	Modified total error = 0.06	Difference image index = 0.38
Calibrated image	Modified total error = 0.06	Modified total error = 0.02	Difference image index = 0.13
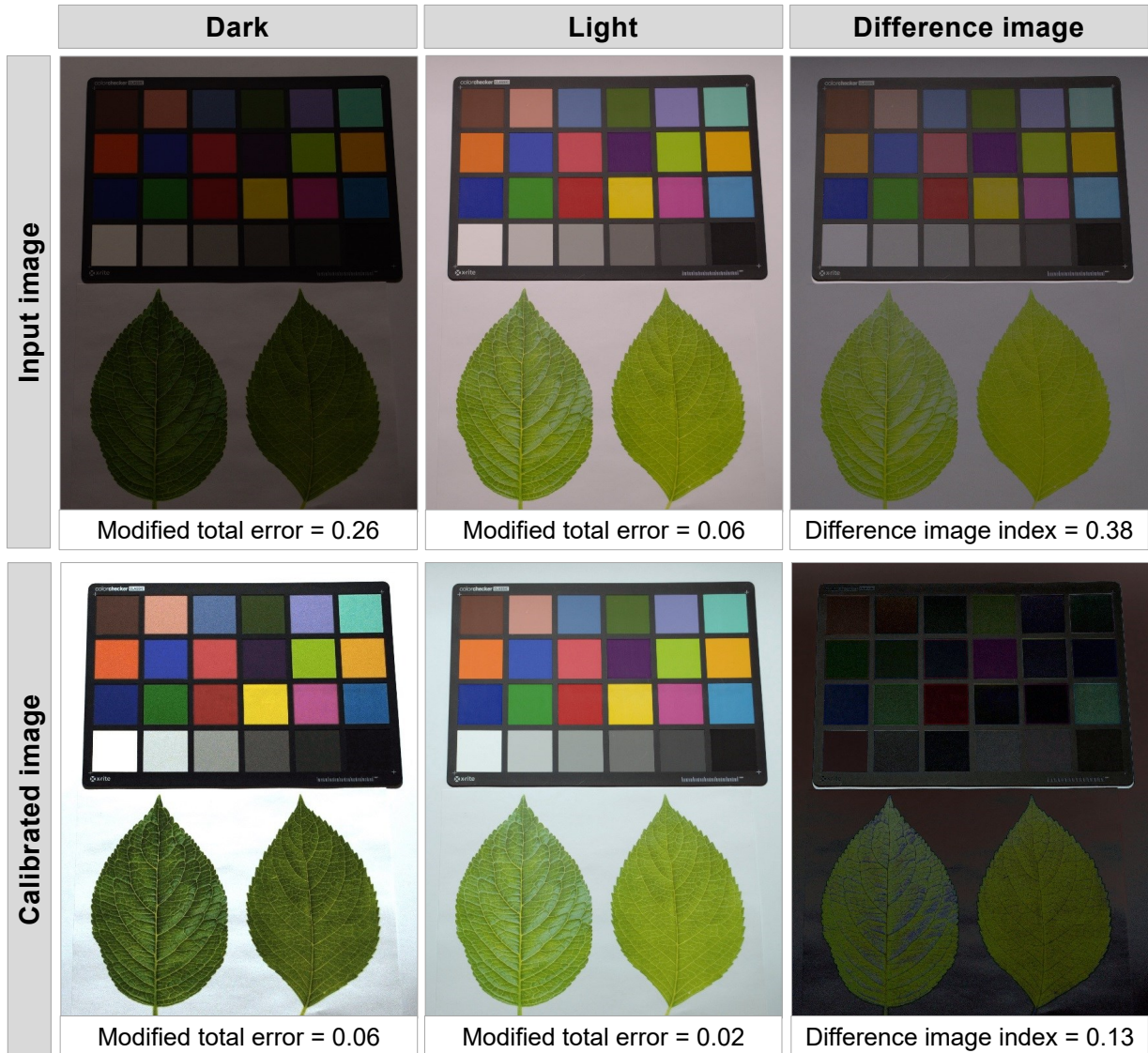
Figure 4.5. Synthetic input images acquired in laboratory at dark and light conditions, respective calibrated images (using "Set 1"), and the difference image between two input images and two calibrated images.
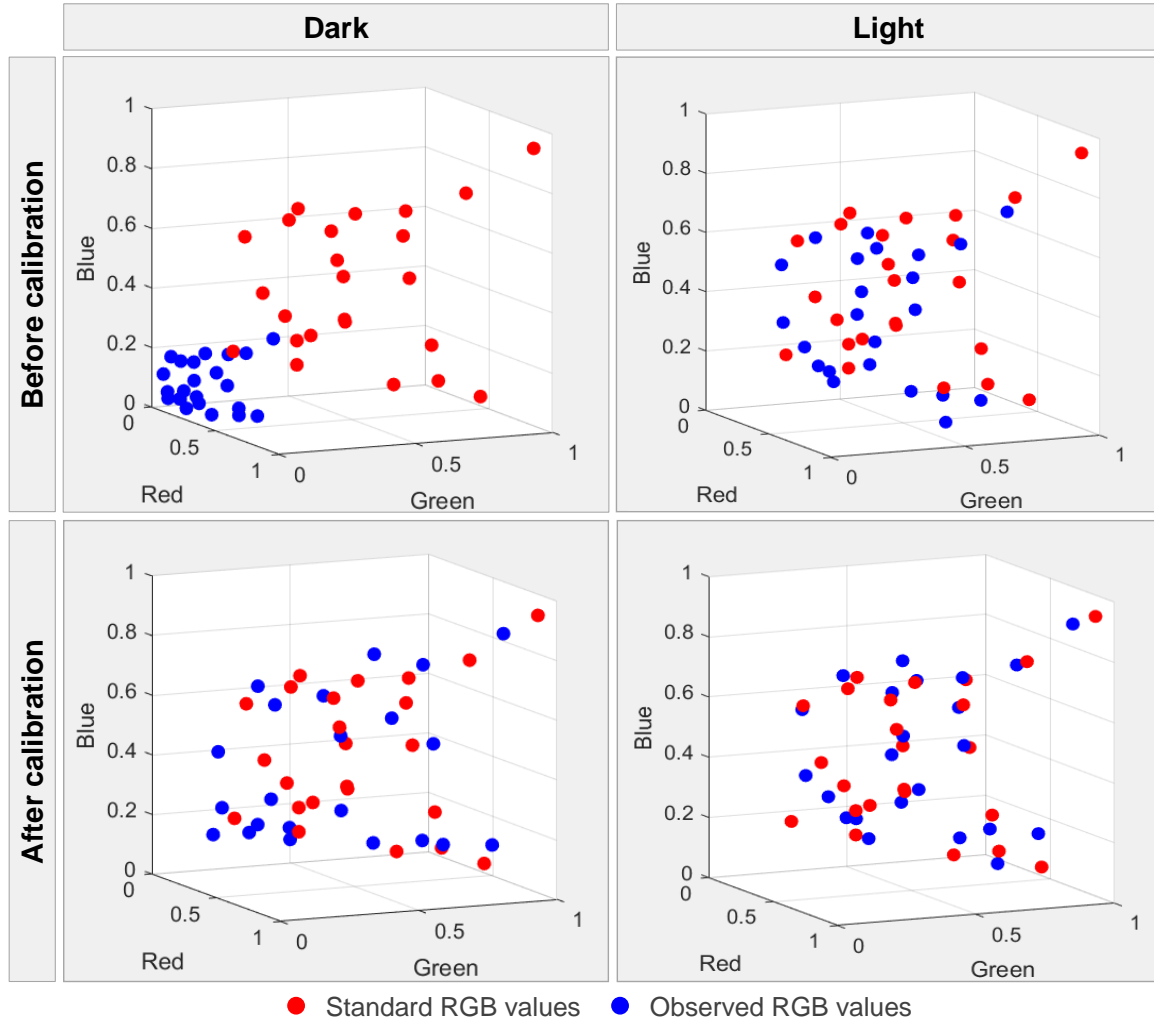
Figure 4.6. Scatter plot of normalized RGB values of 24 color patches in the image before and after calibration in dark and light conditions.

RGB values of each color patch in the input image. Before calibration, the RGB values in dark condition were closer to 0, while the RGB values of color patches in light condition were well scattered similar to the standard RGB values. After calibration, the RGB values of color patches in dark condition were closer to the standard values, and in light condition already distributed input values got adjusted and most of the calibrated RGB values overlapped with the standard values.

103

4.4.2. Selection of the Minimum Number of Color Patches

The variations in e_m (eq. 4.14) for calibrated images obtained from dark and light input images are shown in figure 4.7. There was no difference in e_m beyond 13 color patches in the calibrated light image, while the calibrated dark image required >18 color patches. This was also observed in the 3D scatter plot (fig. 4.6), where the light image had well distributed normalized RGB values of each color patches before calibration and the e_m (0.06) was less than dark image (0.26) as they were well separated before calibration. Previous studies (Akkaynak et al., 2014; Alsam & Finlayson, 2008) also followed a similar pattern for color schemes Set 1 and Set 2, where the magnitude of e_m was lesser in the light calibrated image than the dark calibrated image.

The color scheme Set 2 required more color patches (>9 patches) for producing the calibrated image with minimum error, because the first 6 colors (neutral shades) were achromatic, and did not contribute efficiently to the color calibration (fig. 4.7). Other than Set 2, all other color schemes (Set 1, 3-6) attained the minimum value after the first few color patches (≤6). The color schemes Set 3 and Set 4 were evidently seen as efficient with only 3 color patches producing the minimum error in both dark and light calibrated images compared to other color schemes. In the dark calibrated image, Set 4 produced the minimum error with 3 color patches, while in the light calibrated image Set 3 produced the minimum error with 3 color patches. However, the calibrated image produced by Set 4 using the first 3 color patch produced a greenish color cast and was not suitable compared to the calibrated image by Set 3. Therefore, color scheme Set 3 was found efficient in rendering the colors of the image based on e_m.
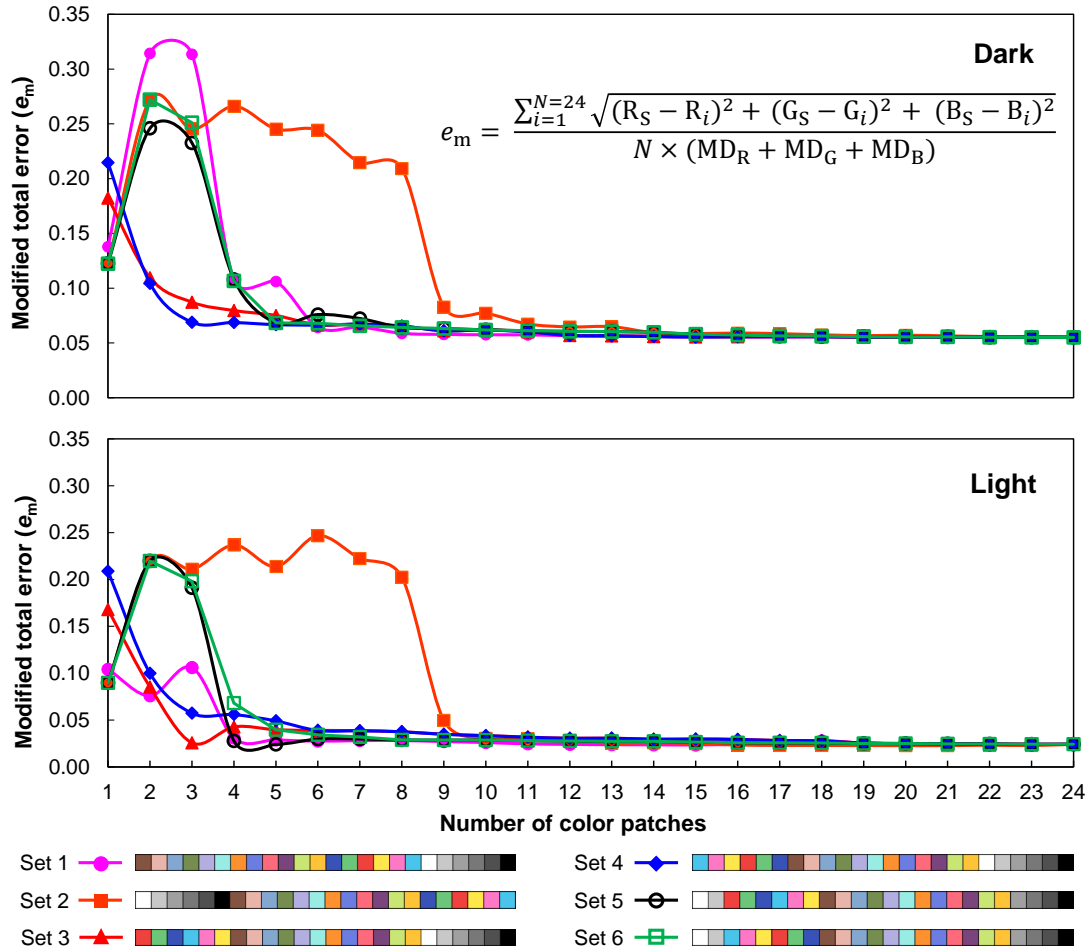
Figure 4.7. Variations in modified total error in calibrated images by different color schemes in dark and light conditions (data in Appendix B.1).

4.4.2.1. Comparison of dark and light calibrated images

The CPI was also used in the selection of color scheme and number of patches to be considered in calibration. We found that images calibrated using the color scheme Set 1 with all 24 color patches produced the minimum e_m (fig. 4.7) and considered as the reference image for CPI calculation in both dark, and light conditions. Additionally, weight factors as used in equation 4.16 is important in deciding the contribution of differences. Even though the difference between the two calibrated images was kept to a minimum,

the similarity between the calibrated image (with less number of color patches) and the reference image (using all 24 color patches) needs to be considered. Therefore, more weights were considered for W_1 and W_2, and slightly less weight for W_3. The weights were fixed, after preliminary trials, as 0.35 for each W_1 and W_2, and 0.30 for W_1 in equation 4.16 to evaluate CPI for each color patch.

The plot of CPI values and number of color patches of different color schemes is shown in figure 4.8. The CPI showed a pattern similar to the e_m performance measure (fig. 4.7). The CPI values from all color schemes converged and attained a low CPI when half of the color patches (12) were used. Overall, below 9 color patches, all color schemes showed distinct variations in the CPI values. It was interesting to note that the order of color patches did not have any effect on calibration, i.e., using R, G, B or B, G, R produced identical calibrated images.

The color schemes Set 1, Set 2, Set 5, and Set 6 showed lower CPI with a single color patch, but the calibrated images had increased color cast (dark skin and white). This was also observed when only 2 color patches were included for calibration. By including the 3rd color patch, other than the schemes Set 3 and Set 4, the calibration with the rest of the schemes resulted in a color cast that raised the CPI values (fig. 4.8). The grayish color cast with color schemes Set 2, Set 5, and Set 6 was because of the achromatic color patches (white and gray), and with color scheme Set 1 was because of skin tones in the starting patches that were inefficient in overall image calibration.

The variations of CPI values for all color schemes with 3, 6, 12, and 24 color patches, indicated in figure 4.8, are presented in table 4.1. In all image samples, the CPI values from Set 3 and Set 4 with 3 color patches were the lowest (Set 3: 0.205–0.244;
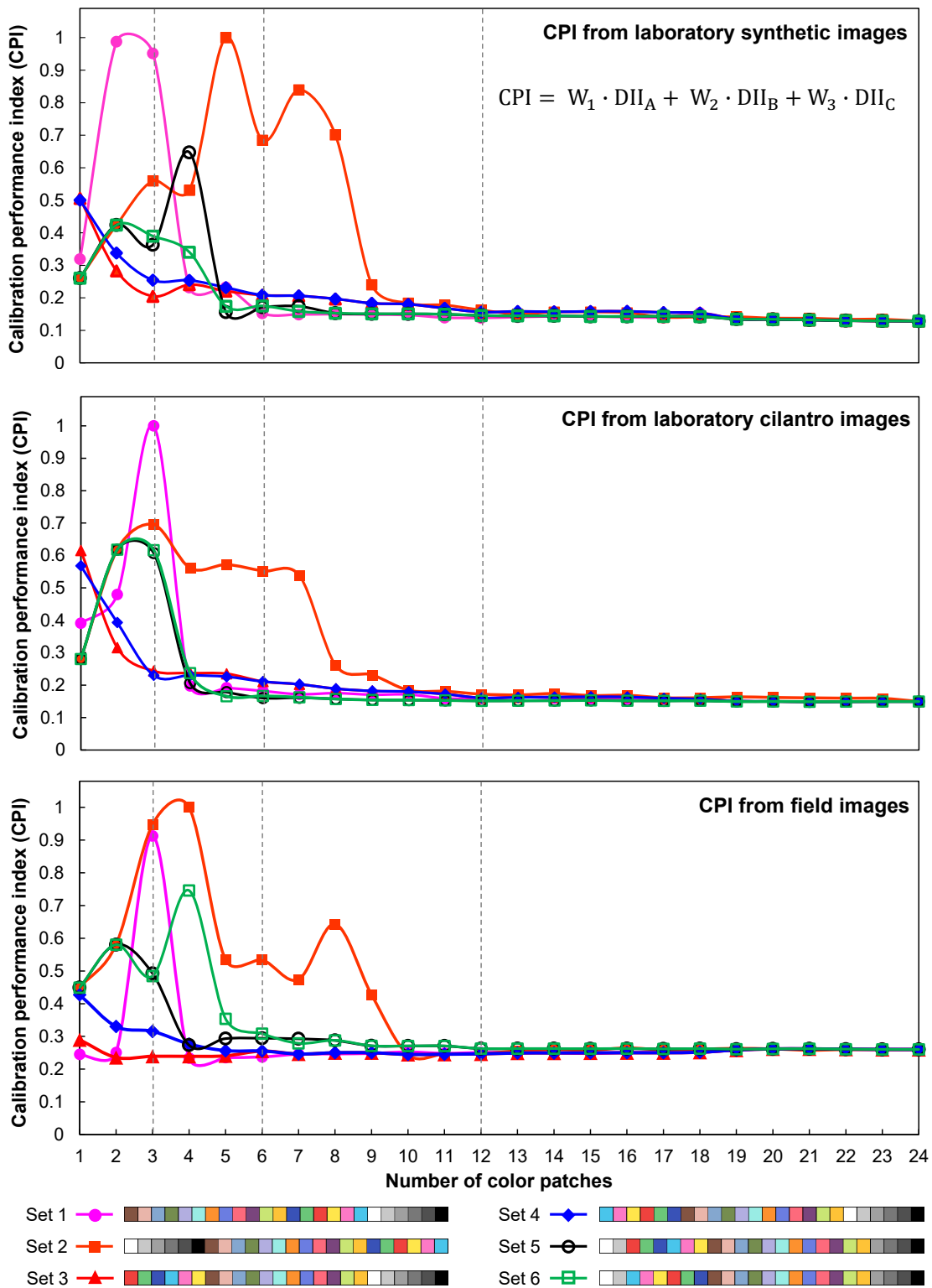
Figure 4.8. Variations in calibration performance index (CPI) in calibrated images with the number of color patches by different color schemes (data in Appendix B.2).

Set 4: 0.230–0.315) compared to other color schemes. The calibrated images produced from Set 3 and Set 4 with 3 color patches were visually similar. The next lower CPI values were obtained with 6 color patches in all color schemes except Set 2. Including additional color patches further reduced the CPI values; however, considering practical applications and simplicity reasons for using color patches in a field setting, using a minimum number of color patches is desirable. Therefore, including 3 color patches — R, G, and B were recommended for an efficient color calibration as it produced calibrated images that were visually similar and had low CPI values.

Table 4.1. Calibration performance index variations for different color schemes with selected number of color patches.

Image samples	Color scheme[*]	Number of color patches			
		3	6	12	24
Synthetic images	Set 1	0.952	0.152	0.138	0.128
	Set 2	0.560	0.685	0.162	0.128
	Set 3[†]	0.205	0.208	0.156	0.128
	Set 4[‡]	0.254	0.208	0.156	0.128
	Set 5	0.363	0.171	0.146	0.128
	Set 6	0.390	0.175	0.146	0.128
Cilantro images	Set 1	1.000	0.182	0.159	0.150
	Set 2	0.695	0.552	0.172	0.150
	Set 3[†]	0.244	0.211	0.161	0.150
	Set 4[‡]	0.230	0.211	0.161	0.150
	Set 5	0.608	0.160	0.152	0.150
	Set 6	0.617	0.167	0.152	0.150
Field images	Set 1	0.913	0.237	0.251	0.260
	Set 2	0.948	0.534	0.247	0.260
	Set 3[†]	0.239	0.255	0.247	0.260
	Set 4[‡]	0.315	0.255	0.247	0.260
	Set 5	0.492	0.295	0.263	0.260
	Set 6	0.485	0.308	0.263	0.260

Note: These selected calibration performance index (CPI) vales were obtained from the data provided in Appendix B and are also indicated in figure 4.8.

[*] Refer to figure 4.3 for details on the color schemes; however, the schemes with lowest CPI are reproduced for ready reference.

[†] Set 3:

[‡] Set 4:

4.4.3. Color Calibration Plugin Validation

4.4.3.1. Laboratory conditions — Cilantro cut plant

The color calibration method with 3 color patches (RGB) was validated using cut cilantro plant samples in the laboratory under varying lighting conditions (fig. 4.9).

It was observed that with actual samples having three-dimensional shapes, shadows occurred in artificial or natural lighting. However, the shadows have not interfered in extracting of an average color value from the region of interest (ROI) and in the process of image calibration. The calibration performance of the cilantro images in dark and light conditions had a low CPI value of 0.244 (table 4.1), which was good. The DII (eq. 4.15) between dark calibrated image and its dark reference image was 0.12 (DII_{dark}), and the difference index between light calibrated image, and its light reference image was 0.13 (DII_{light}). However, the dark and light calibrated images produced slightly higher difference index of 0.52 ($DII_{calibrated}$). This is expected when two images obtained under widely varying lighting conditions are calibrated and compared.

The calibration process raised the whole image intensity value from 92 to 129 in dark condition (ROI intensity from 31 to 47) and lowered from 124 to 115 in light condition (ROI intensity from 60 to 51). The intensity value of the whole dark calibrated image was higher than light calibrated image because the white background used had become saturated in the calibration process and increased the whole image intensity. However, comparing the plant canopy and the selected ROI, the color looks more uniform between the dark and light calibrated images than the input images. Therefore, comparison and analysis using the calibrated images would be definitely better than using the raw input images, which had a wider color difference.

109
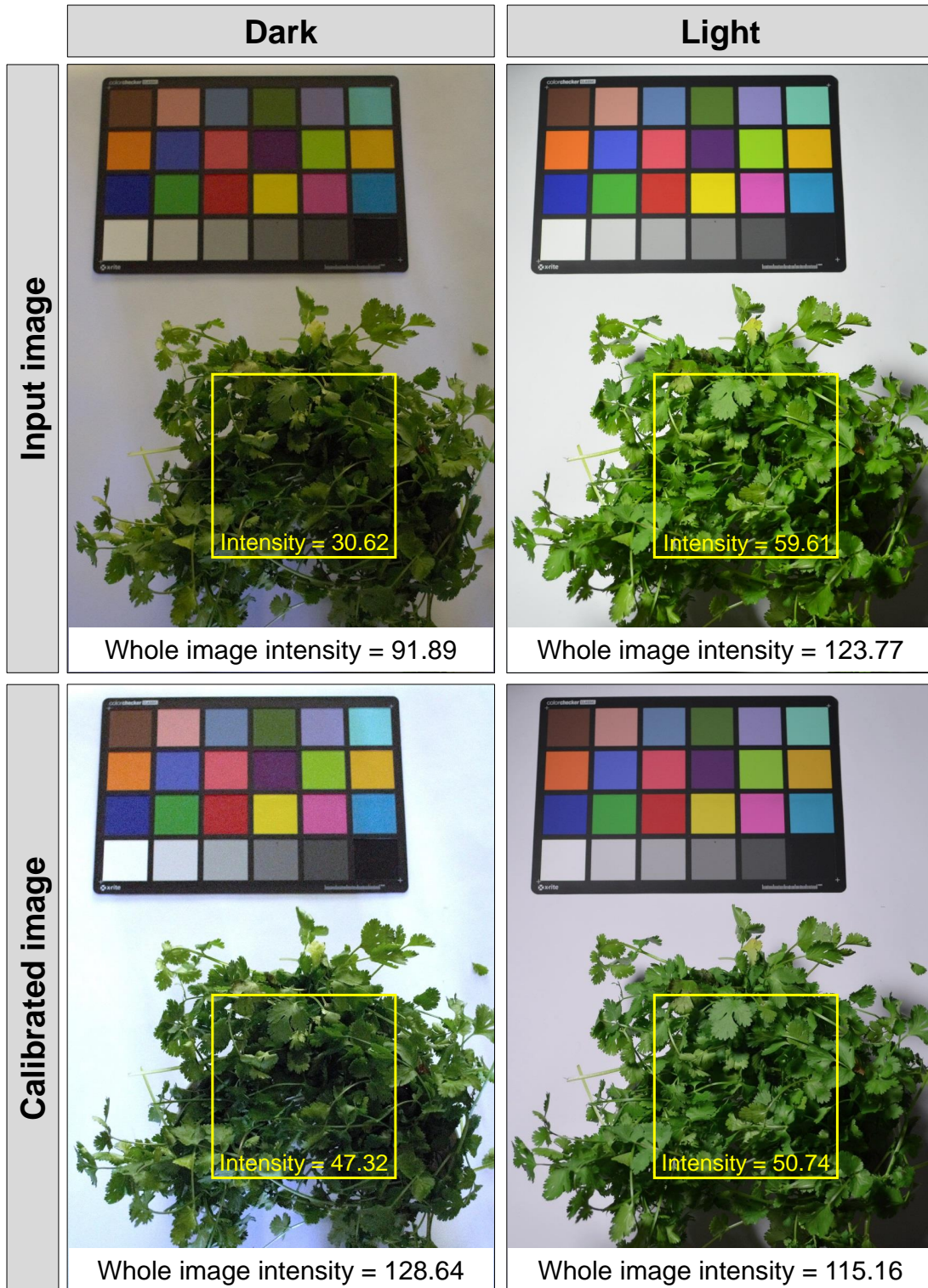
Figure 4.9. Cilantro cut plant images acquired in the laboratory at dark and light conditions as inputs and their respective calibrated image obtained using only R, G, and B color patches. The calibration performance index was 0.244.

4.4.3.2. Field conditions — alfalfa and spring wheat

The acquired field images consisted of two crops: alfalfa (*Medicago sativa* L.) in the late bud stage that looked green (left), and spring wheat (*Triticum aestivum* L.) in a mature stage (right) that looked brown (fig. 4.10). These images represented a typical field condition for studying the image calibration at two growth stages of field crops. Varying lighting conditions were captured by adjusting the aperture setting from $f/6.3$ to $f/16$ that showed a gradual darkening of the image indicated by decreasing image intensities representing the typical lighting variations in field conditions.

In general, in color image processing, a simple "white balance" method is widely followed (Lam, 2005; Richardson et al., 2009), which makes the image white regions look white and correspondingly updating all image pixels. This method calculated a factor that brings an object/section/color-patch of the image from discolored white to perfect white (R = G = B = 255). The factor was uniformly applied to all image pixels to produce the white balanced image. This process simply altered all the pixel values only based on white, which is similar to the satellite and aerial images radiometric simplified calibration using linear interpolation of known ground control points reflectance (Xu et al., 2012). The processed images, however, still suffered similar intensity variations as the input images (fig. 4.10 - White balance). Interesting to observe that the white patch (bottom-left patch of the color chart) can be distinctly seen as perfect white in all white balanced images.

The calibrated images using the developed plugin with R, G, and B color patches showed an overall homogeneity across the images (3×3 calibration; fig. 4.10), which demonstrated the good calibration performance of the ColorCal plugin. The DII between

	Input image	White balance	3 × 3 calibration
f/6.3			
f/7.1			
f/8			
f/9			
f/10			
f/11			
f/13			
f/14			
f/16			
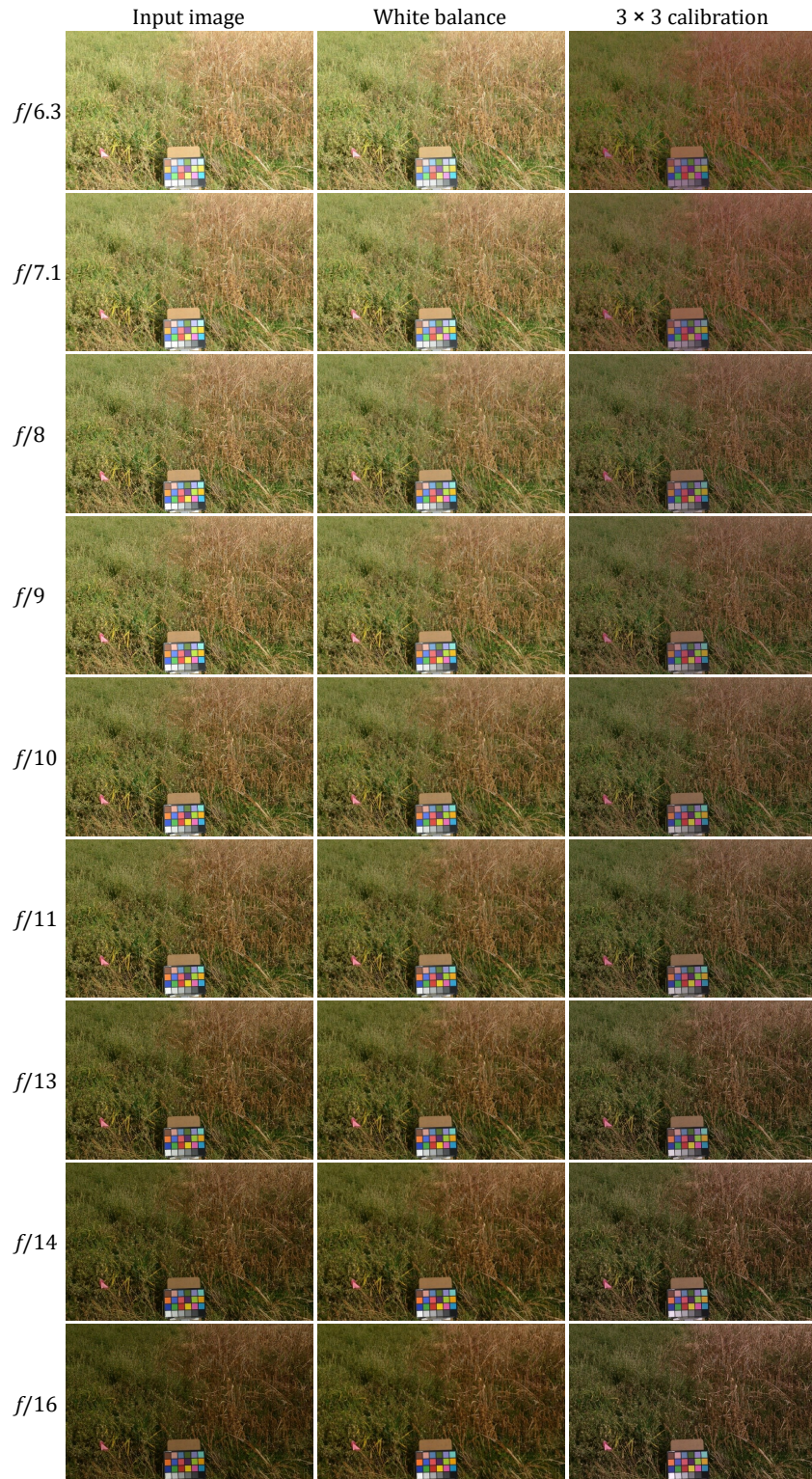
Figure 4.10. Comparison of input, white balanced, and the calibrated images in the field (alfalfa and spring wheat). The calibrated images were obtained using the three R, G, and B color patches.

the calibrated image and their respective reference image ranged between 0.053 ($f/16$)

and 0.280 ($f/6.3$), with an average of 0.154 and standard deviation of 0.073. The overall

CPI evaluated by assuming the image from $f/6.3$ as light, and $f/16$ as dark conditions was

0.239 (table 4.1). In the field, the CPI value for 3 color patches (0.239) was better than

24 color patches (0.260) indicating that for field images 3 color patches (RGB) were

sufficient. Further analyses with other input images may be required to establish this trend.

The plugin took ≈ 7 s (CPU time) per input image of resolution 0.68 mm/pixel to produce

the respective calibrated image in a laptop PC (Windows 10, 2.20 GHz Intel Core i5-5200U

processor, and 8 GB RAM).

All the calibrated images appear to fall toward the darker intensity, but this did not

impact the comparison and analysis of the images for any practical applications because

they tend to be relatively uniform independent of lighting conditions. However, the actual

variation in the crop phenology (growth stages) were actually captured by the raw images,

while the images calibrated with ColorCal plugin had eliminated the lighting variation and

standardized the visual tone or overall intensity.

The intensity variation plot among these images helped to visualize the effect of

white balance and calibration (3 color patches) on the input images from field conditions

(fig. 4.11) and allowed us to perform further analysis. The steeper negative slopes for the

input and white balance images indicated a decrease in intensity values from light to dark.

Ideally, the goal of the calibration process is to make the intensity variations flat

(zero slope), while maintaining a desirable intensity. A hypothetical flat line with a lower

intensity value (e.g., <30) indicates an underexposed image, and that with a higher

intensity value (e.g., >160) indicates an overexposed calibrated image, and these are not

Images processed	Alfalfa (green, left)	Spring wheat (brown, right)	Overall image (both)
Input	$y = -12.04x + 154.28$ ($R^2 = 0.996$)	$y = -11.96x + 157.74$ ($R^2 = 0.997$)	$y = -11.67x + 152.02$ ($R^2 = 0.996$)
White balanced	$y = -10.14x + 143.91$ ($R^2 = 0.991$)	$y = -9.88x + 145.98$ ($R^2 = 0.989$)	$y = -9.72x + 140.8$ ($R^2 = 0.989$)
Calibrated	$y = -2.69x + 83.55$ ($R^2 = 0.976$)	$y = -2.31x + 84.21$ ($R^2 = 0.996$)	$y = -2.42x + 81.38$ ($R^2 = 0.990$)

Figure 4.11. Intensity variation of input, white balanced, and the calibrated images in field (alfalfa and spring wheat). The calibrated image was obtained using the three R, G, and B color patches and the fitted linear models (data in Appendix B.3).

desirable. The slopes of the fitted linear models for input and white balanced images were −11.7 and −9.8, respectively. The slope for calibrated images was only −2.42, indicating less variation in intensity among calibrated images.

When the left (alfalfa) and the right (spring wheat) image portions were considered separately (Inset; fig. 4.11), similar variations in slopes of linear regressions were found (Models; fig. 4.11). Slopes for alfalfa and spring wheat ranged between −11.96 and −12.04 for input images, −9.88 and −10.14 for white balanced images, and −2.31 and −2.69 for calibrated images. Although the calibrated images were slightly darker than input and white balanced images, this indicated a more homogeneous "color feel" (intensity). The brightness of calibrated images can be improved through further image processing operations to increase the intensity uniformly, without changing the color value (hue) of the pixels. This forms a potential aspect of future research.

4.4.4. Suggestions for Future Work

Future work will consider studying the calibration performances with higher order CCM by including the square terms and interaction terms. Also, chromatic adaptation techniques, such as white balance, gray balance, or RGB equalization before or after color calibration to improve the brightness of the calibrated images, and comparing the results using the developed performance measures should be considered. The color calibration should be explored in other color spaces, such as $L^*a^*b^*$, HSI, and XYZ because these color spaces cover a wide gamut of colors and might produce better results. The performance of the developed method should be compared with the calibration methods offered by commercial software (e.g., Adobe Lightroom, Aperture). Finally, the calibration

performance should be tested in images with a wide range of natural illumination variations (brightness and color) in the phenocam network systems.

4.5. Conclusions

The artificial lighting and camera aperture (f-stop) settings produced a wide range of lighting variation in the laboratory environment and field setting, respectively, suitable for digital color image calibration study. The user-developed plugin performed well in all conditions (laboratory and field conditions) and took ≈ 7 s/image to produce calibrated images by deriving a color calibration matrix (CCM). The developed calibration performance index (CPI) was useful in observing the effect of the number of color patches and deciding the minimum number of color patches for efficient color calibration. The CPI values converged when only 12 color patches (out of 24) were used. On selected color schemes, 3 to 6 color patches gave a comparable performance to the 24 patches. We found that the order of selected color patches in a color scheme did not affect the calibration performance. The calibration using only the 3 patches namely red, green, and blue (R, G, and B; $[3 \times 3]$ CCM) produced visually homogenous intensity images with low CPI (0.21–0.24), while the commonly followed neutral color white as well as gray patches produced less performance with high CPI (0.26–1.0). Considering practical application and simplicity, 3 color patches: R, G, and B are recommended for an efficient color image calibration. Future work should consider calibrating using CCM derived from higher order terms (square and interaction terms), and testing with real-time field images with natural lighting variations.

5. PROFILE-BASED PLANT STAND COUNT, PLANT VIGOR, AND SPATIAL DISTRIBUTION FROM UAS IMAGES *

5.1. Abstract

Plant stand count is one of the important measures to determine the attainment of the target plant population, obtain emergence characteristics of the seeds, and evaluate the planter's performance. Unfortunately, the available commercial software packages are expensive and rigid, and the literature mostly followed the "black-box" approach. Therefore, a user-coded ImageJ plugin from scratch, using simple geometrical principles, that performs automatic row orientation, row identification, plant cluster segmentation, and plant stand counting from UAS images was developed. A novel approach of using pixel profile was developed to address the objectives. Identified plants' clusters were segmented by the combined approach of applying pixel profile, bounding box height, and the area in series. Two set of sunflower field images ("Set I" and "Set II") obtained from two locations (Carrington and Prosper Counties, ND, USA), having different UAS flight altitudes (12.2 m and 23 m), field areas (46 m^2 and 24 m^2), image resolutions (0.003 31 m/pixel and 0.006 35 m/pixel), and growth stages (V2–V4 and V4–V6) were used for plugin validation. The plugin was able to accurately identify rows up to 100 m and expected to work >100 m,

as long as the rows are straight. The plugin achieved a plant stand count accuracy of 95.9±2.5 % for Set I and 84.9±3.9 % for Set II images. For the spatial distribution analysis, a new index termed ideal spacing uniformity index (ISUI) was found to be the best to evaluate plant spatial distribution. The plant spacing distribution was compared with ideal spacing and achieved an accuracy of >76.5 %. The CPU time for analysis were 31.7±1.2 s and 4.2±0.6 s (Intel Core i5 processor, 8 GB RAM). Application of the free open source ImageJ software and the user-coded plugin ensures "easy availability" and "data security," a major concern nowadays, as the software and plugin reside in the user's system. As the plugin was developed using a simple geometrical approach, with suitable modifications the plugin can be potentially used for estimating plant stand count in other row crops (e.g., corn, soybean, cotton).

5.2. Introduction

Plant stand count is an important measure in the early growing season for making better management decisions (Bruin & Pedersen, 2004). Nowadays, using unmanned aerial system (UAS) images for assessing plant stand count is gaining traction and commercial applications are available that offer cloud-based services to perform stand counting through image analysis algorithms. Although that can be an easy route to take, such services are expensive, crop-specific, and impose a set of restrictions and requirements. Also, there is no simple, open source, free, and unified application that works on all platforms (operating system) with minimum inputs and restrictions. Therefore, we propose to develop a new algorithm based on simple profile and geometry principles using ImageJ (Rasband, 2018), an open source, free, image processing software to produce a suite of results related to plant stand count, spatial distribution, and vigor.

118

In order to function properly, the existing commercial applications impose several requirements including: (i) uniform plant-to-plant spacing, (ii) image resolution of $\leq 25\,\text{mm/pixel}$ (iii) consistent row spacing with parallel rows, (iv) uniform plant growth stages, and (v) no overlap between individual plants. Besides, they also involve subscription charges ($\approx$\$99/month).

The plant stand count measure is used to (i) determine whether target plant population is achieved as tested/verified based on seed emergence or seedling establishment, (ii) evaluate planter performance on seed placement and spacing, (iii) obtain insights on the emergence characteristics of the planted seeds, and (iv) assess damage due to frost, wind, and hail storms. Similarly, plant spacing variability in the field is also an important measure and reported to have a significant effect on the final crop yield (Lauer & Rankin, 2004; Nielsen, 2004). Possible reasons for spacing variability between plants in a row, planter mechanism type, and its efficiency, planting speed, soil moisture and temperature variability, and crop residue distribution (DuPont, 2015; Liu et al., 2004).

To evaluate plant stand count and plant-to-plant spacing uniformity field scouting was commonly performed. Two methods of assessing the stand count through field scouting are the row-length method and the hula hoop method (Stowe, 2018). The latter method is common for all crops, while the former method has minor variations in the scale-up calculations based on crops (D. Berglund, 2007; DuPont, 2015; Kandel, 2010). It is obvious that the field scouting method is time-consuming, painstaking, and covers only a small portion of the field. Efforts were taken to develop automated plant stand counting using image processing through ground level imaging systems (D. S. Shrestha & Steward, 2003, 2005; Tang & Tian, 2008). However, these studies focused on counting only on a

119

single crop row. Nowadays, field scouting method and ground-level imaging method is being replaced by the use of UAS images as they are quick, easy, and covers the entire field.

Even though it is possible to visually count plants from UAS images, which would be a tedious task and prone to human error, automatic plant stand counting algorithms have been developed through image analysis. Wheat (*Triticum aestivum* L.) stand count per unit area (plant density) at the emergence stage was estimated with an error of $\approx 9\,\mathrm{plants/m^2}$ using a support vector machine model with 13 shape features in MATLAB (Jin et al., 2017). Rapeseed (*Brassica napus* L.) stand count was estimated by simple correlation (r) using 15 shape features of the plants in Trimble's eCognition developer software and achieved an R^2 of 0.85 from a model developed using only three shape features (B. Zhao et al., 2018). A computer workflow in ArcGIS software to evaluate the sunflower (*Helianthus annuus* L.) stand count from UAS imagery attained R^2 ranging between 0.52 and 0.96 (Flores et al., 2017). Potato (*Solanum tuberosum* L.) stand count was performed using ArcGIS from multispectral imagery and achieved a r of 0.83 between manual stand count and image-based stand count (Sankaran et al., 2017).

Most of the reported studies on plant stand count used proprietary software because of their built-in image analysis tools and capabilities. This, of course, is an advantage, however, it comes with a price and lack of flexibility. Developing such image processing algorithms in an open source image processing platforms allow farmers to adopt the technology, rather than purchasing commercially available software or expensive annual subscriptions. ImageJ is an open source, free image processing system through which user-coded plugins can be developed for any specific applications using Java programming language (Burger & Burge, 2016a; Rasband, 2018). Several specific agricultural

applications using such ImageJ plugins have been developed (Borianne et al., 2018; Igathinathane et al., 2009a, 2009c; Pothula et al., 2015; Sunoj et al., 2018a; Sunoj, Igathinathane, Saliendra, Hendrickson, & Archer, 2018b; Sunoj et al., 2018). The development of such plugins helps protect "data security" as the analysis can be performed by the users on their system, unlike commercial software where the users have to upload their image to get the analysis results.

The present research aims to develop a plant stand counting algorithm from UAS images through a user-developed ImageJ plugin rather than a black-box approach. Unlike the reported research that used proprietary and often undisclosed methods of commercial software, the present research proposed to develop simple geometrical approaches from scratch to obtain necessary solutions.

The novel idea of the proposed method is to use the pixel profile (plot of the number of pixels along vertical) of the image combined with geometrical features, commonly available routine in any image processing systems, to identify the crop rows and individual plants in a row irrespective of the field orientation in the image. Furthermore, it is desirable to determine the planting direction automatically in a field image and plant stand distribution. Therefore, the specific objectives of this research were to:

1. Determine the planting pattern in the field and crop row orientation for images of rotated fields.

2. Identify crop rows based on pixel profile for user-defined zones in the field.

3. Estimate plant stand count along each row after resolving the plant clusters.

4. Evaluate plant vigor based on the plants' projected areas.

5. Determine plant spacing, map their spatial distribution to evaluate skips and doubles and evaluate various distribution indices.

5.3. Materials and Methods

A user-coded plugin was developed that used the UAS's stitched-image as the input and performed analyses using a direct simplified geometrical approach. This direct approach means working with image objects' pixels through algorithm development for the various intended analyses to address the objectives. It was assumed that the user has the stitched UAS image from other sources and used as the input in the developed plugin. The details of the plugin development and methodology developed in this study are discussed subsequently. To our knowledge, such an application of open source user-coded plugin development for the aforementioned objectives has not been reported in the literature.

5.3.1. Experimental Sites

Sunflower field research trials at the Carrington Research Extension Center (CREC), Carrington, ND ($47.52175°$ N, $-99.11488°$ W) and Prosper, ND, USA ($46.96667°$ N, $-97.06667°$ W) were used in this study. In CREC, the trial was planted on June 1, 2018, using a 7200 John Deere MaxEmerge 2 planter, with a row spacing of 0.76 m (30 in) and plant spacing of 0.22 m (8.5 in). The overall dimensions of the field study were 48 m × 24 m. The Prosper field trials used extra-large confectionary sunflower seeds and the experimental plots were planted using an Almaco SeedPro precision planter (Almaco, Ames, IA, USA) on June 1, 2017 (Sidhu, Wiesenborn, Johnson, Monono, & Eriksmoen, 2019). The individual plots were of 7.6 m × 3.0 m size with a row spacing of 0.76 m (30 in) and plant spacing of 0.27 m (10.6 in). The standard recommended agronomic practices were followed based on the experimental trials in these sites.

5.3.2. Image Acquisition – UAS Images

In this study, the images from CREC and Prosper were referred to as "Set I" and "Set II" images, respectively. The Set I forms the major image input data ($n = 25$) in this study and is predominantly used in the plugin development and validation. However, the Set II forms only the minor image data ($n = 5$) and was used as an additional test case for the developed plugin.

Aerial images were captured 15 days after planting (June 15, 2018) using a DJI Phantom 4 Pro quadcopter (Shenzhen, Guangdong, China), equipped with a 20.1 MP color digital camera, flown at 12.2 m ($\approx$40 ft) above ground level with a front and side overlap of 80 % each. The camera model and imaging settings were: camera make and model $=$ DJI FC6310, F-stop $= f /5.6$, exposure time $= 1/400$ s, and focal length $= 9$ mm. The acquired images were in JPEG (Joint Photographic Experts Group) format and stitched using Pix4Dmapper Pro (Pix4D, Switzerland). The stitched image was exported in TIFF (tagged image file format) and later converted to JPEG using ArcGIS (ESRI, Redlands, California) for this study. The final image dimensions were 14 685×7350 pixels representing a field area of $\approx$1150 m^2 and the file size was $\approx$270 MB. Representative image sections ($n = 25$) of each $\approx$46 m^2 (2940×1470 pixels) were cropped from the stitched image and used in the plugin development. The image resolution was $\approx$3.31 mm/pixel.

The Prosper trials also used a similar DJI Phantom 4 quadcopter (Shenzhen, Guangdong, China) and the images were captured on 21 days after planting (June 21, 2017). The flight altitude was maintained at 23 m (75 ft) above ground level. The plants were between the V4 and V6 growth stages at the time of imaging. The Agisoft PhotoScan software (Version 1.4.2., St. Petersburg, Russia) was used to stitch the images. The

resulting image resolution was ≈6.35 mm/pixel. The images were converted to JPEG and a few representative image sections ($n = 5$) of each ≈21 m^2 (1350×450 pixels) were used for testing the plugin.

5.3.3. Image Analysis Plugin Development

The user-coded plugin developed for the analysis utilized the open source free image processing software Fiji (Ver. ImageJ 1.52n, Java 1.6.0_24 [64-bit], Rasband (2018); Schindelin et al. (2012)). The plugin was coded in Java language and consisted of about 2230 lines and brief pseudocode is provided in Appendix C.2. for better understanding. The plugin comprised of several Java methods (standard and developed) for performing the outlined individual tasks based on the research objectives (fig. 5.1A).

The user first loads the image into ImageJ by "Drag and Drop" or by Open, and selects the field boundary or any region of interest (ROI) using the "Polygonal selection" tool from the ImageJ toolbar. Then the loaded plugin when compiled generates the front panel (fig. 5.1B) with various input and output options. The input options were of general nature and correspond to the row crop being analyzed. Users can select any or all from the list of output options that generated different images of the analysis and run the plugin using the Run button. In the background, the plugin performed various tasks in sequence (fig. 5.1A) and produced the output images as chosen by the user. The details of the individual operations are discussed subsequently.
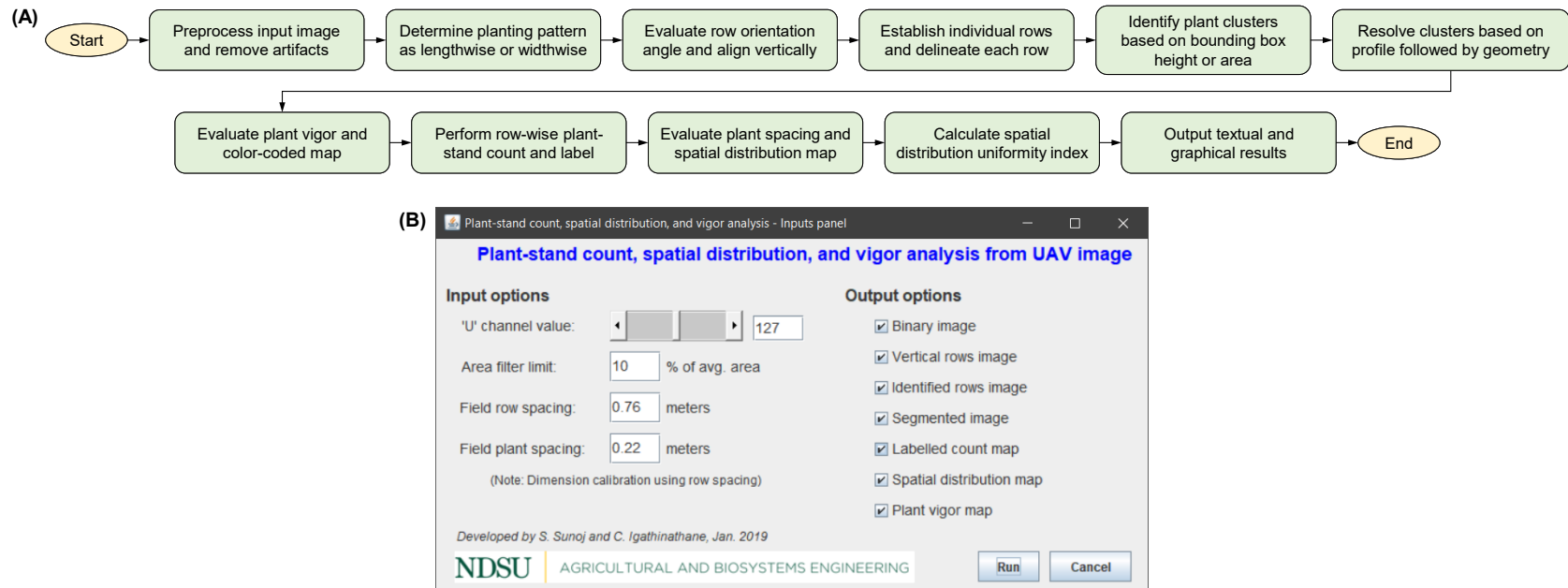
Figure 5.1. Process flow diagram and front panel of the developed plant stand count plugin with special emphasis on plant cluster segmentation: (A) sequence of image preprocessing and processing operations performed, and (B) front panel of the developed row identification and plant stand counting ImageJ plugin.

5.3.4. Image Preprocessing

Conversion of a digital color image to binary by thresholding is the primary step in several image processing analysis. For thresholding plants from a color UAS image (fig. 5.2A), many thresholding methods were tested out of which excess green and vegetativen were reported to be accurate (Torres-Sánchez, Peña, de Castro, & López-Granados, 2014).
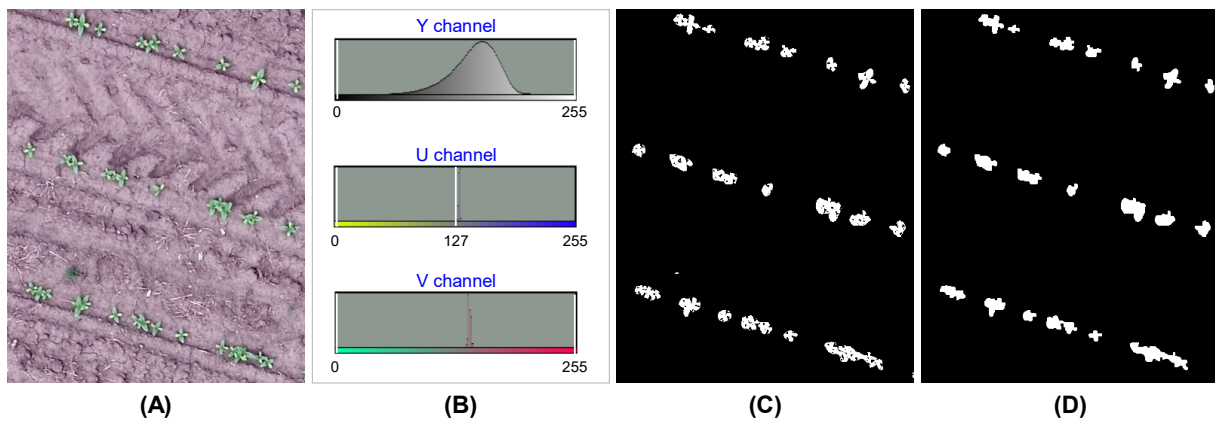
Figure 5.2. Plant thresholding and artifacts removal from UAS image: (A) a section of color UAS imagery (12.2 m altitude); (B) histograms of individual channels of YUV color space with selected threshold range only for U channel (0–127); (C) binary image from selected threshold representing plants in white and soil in black; and (D) artifacts removal and segmentation by applying series of 'Fill Holes', 'Open', and 'Close' operations along with an area filter of objects ≤10 % of average area.

Both these thresholding methods were based on RGB color space, which is sensitive to illumination variations. To standardize such variations, color calibration is a vital process (Sunoj et al., 2018b). However, it requires a standard color calibration chart included in the image to perform the calibration. In such scenarios, working with the YUV color space (Y is luminance channel, U and V are chrominance channels) is proven to be reliable as it is insensitive to illumination variations (Yuntang, 2012).

5.3.4.1. Color thresholding using YUV color space

Color thresholding is the digital extraction of the plants from the background by fixing a threshold value for the individual color channel. The `Color Threshold...` menu in ImageJ displays the histograms of the individual channel on different color spaces. Most of the image analysis software provide YUV color space built into the system. Otherwise, the following conversion formula (eqs. 5.1 and 5.3) can be used to transform the RGB color space to YUV color space (Cheng, Jiang, Sun, & Wang, 2001; Ford & Roberts, 1998).

$$Y = 0.299\,R + 0.587\,G + 0.114\,B \tag{5.1}$$

$$U = -0.147\,R - 0.289\,G + 0.436\,B \tag{5.2}$$

$$V = 0.615\,R - 0.515\,G + 0.100\,B \tag{5.3}$$

where Y is the luminance channel indicating brightness, U is the first chrominance channel spanning between yellow and blue, V is the second chrominance channel spanning between green and red (fig. 5.2B), R, G, and B are red, green, and blue intensity values, respectively.

The `Color Threshold...` menu in ImageJ displayed the histograms of individual channels of the image based on several included color spaces. In this study, we found that the plants from the UAS image can be thresholded by adjusting only the U channel (yellow to blue) of the YUV color space. An optimum threshold U value ranged between 0 and 127 successfully selected the plants from the background (fig. 5.2B). The foot of the U channel histogram peak indicated the yellow range limit. This range is expected to segment the plants (yellowish-green) in all cases as it chooses all the yellowish chrominance range (U channel) of the entire brightness range (Y channel).

5.3.4.2. Image artifacts removal

The thresholding operation produced a binary image (fig. 5.2C) with plants

represented with a pixel value of 255 (white), while background soil with 0 (black). It can

be seen that the plant objects were porous (with holes) and morphological operations, such

as `Fill Holes`, `Open`, and `Close` were applied to the rectify the objects. Other noises and

small artifacts were removed by applying an area filter as a cutoff percentage of the average

area, which utilized the `Analyze Particles...` method. This area cutoff percentage value

was obtained as one of the user inputs (fig. 5.1B). In this study, we used a value of $\leq 10\,\%$.

This binary preprocessed image served as the input to various image processing analyses.

5.3.5. Planting Pattern Determination

The commonly followed planting pattern is oriented along the longest dimension of

the field because the farm vehicles will have a minimum number of turns. However, in

some fields, the planting pattern is perpendicular to the field length because of the existing

alleyways or headlands (Poncet, Fulton, Port, McDonald, & Pate, 2016). Generally, the

decision about planting pattern is based on maximum field efficiency and machinery

operation convenience (Grisso, Jasa, & Rolofson, 2002; Grisso, Kocher, Adamchuk, Jasa, &

Schroeder, 2004). It was reported that the planting pattern has a significant impact on

weed control and yield potential of the crop (Borger, Hashem, & Pathan, 2010). Planting

perpendicular to the direction of sunlight, i.e., north to south planting suppressed the weed

growth by obstructing the light (A. Shrestha & Fidelibus, 2005). Considering these

scenarios, the planting pattern can be classified as "lengthwise" and "widthwise." It will be

useful to detect the planting pattern automatically from the UAS images but has not been

reported so far. Also, a knowledge of the planting pattern will facilitate the crop row identification (discussed subsequently).

5.3.5.1. Input image rotation and ROI extraction

A novel approach of planting pattern determination using a simple pixel profile method, which is mapping the number of object pixels along a line of interest, from the binary image was developed (fig. 5.3). From the user-defined ROI (ABCD; fig. 5.3A) after loading the image, the field angle (θ_f) was determined by calling the `Measure` command using `IJ.run()`. Using θ_f, the input image was rotated at an angle of $\theta_v = \theta_f - 90°$ to make the field ROI vertical to obtain profile (fig. 5.3B). If the θ_f is acute (I quadrant), then θ_v will be negative, which indicates the counterclockwise (CCW) rotation, otherwise the clockwise (CW).

5.3.5.2. Plants profile evaluation in ROI

The rotation operation modifies the dimensions of the rotated image by enlarging the image canvas to accommodate the rotation. A profile ROI was overlaid at the center of the rotated image. The width of profile ROI was defined as 10 % of the image canvas width, and the height of profile ROI was the canvas height. Pixel profile (fig. 5.3C) was obtained from the profile ROI using `getProfile()` method from the `ProfilePlot()` class and analyzed to identify the planting pattern.

From the plot of the profile (fig. 5.3C), it was clear that the lengthwise planting pattern formed distinct peaks in a cyclic pattern and the widthwise planting formed multiple random peaks in an irregular pattern. If the profile ROI width was increased beyond 10 %, the nature of the plots was expected to be similar. In this study, 10 % of image width was selected for the planting pattern determination to minimize the
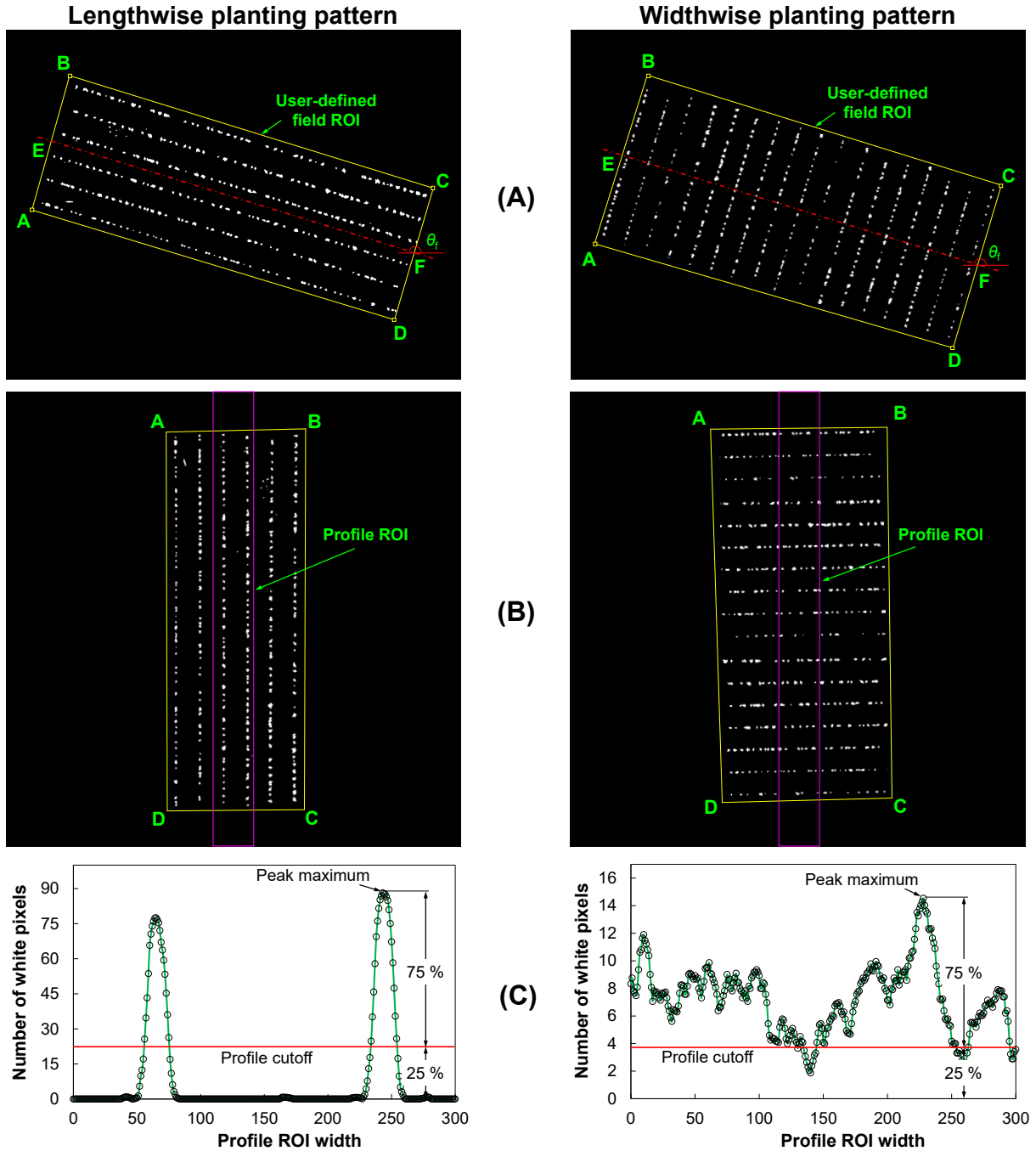
Figure 5.3. Planting pattern determination approach by profile analysis: (A) binary images of lengthwise and widthwise planting patterns, (B) binary images rotated at an angle of $\theta_v = \theta_f - 90°$ to make them vertical, and (C) profile plot obtained from the selected ROI of the rotated images showing cyclic and random peaks.

computation and was found sufficient. It becomes evident from this analysis that the differences in profile patterns can be effectively utilized to determine the planting pattern.

5.3.5.3. Profile ratio cutoff for planting pattern

It was established that observing the cyclic peaks of a profile plot determines the planting pattern (fig. 5.3). However, it is necessary to derive a numerical value that automatically differentiates the two profile patterns. On the initial look, the lengthwise planting pattern will have fewer profile data points above while the widthwise pattern will have more data points above an arbitrarily chosen cutoff of 25 % (fig. 5.3C), selected to eliminate the data points created by plants that were off-rows. However, considering only the number of data points above the cutoff might be misleading because it is influenced by the profile ROI width. Therefore, a non-misleading profile ratio (PR) was developed (eq. 5.4) using the number of data above the profile cutoff (N_c) to the total number of data in the profile (N_T).

$$\text{PR} = \frac{N_c}{N_T} = \begin{cases} \text{Lengthwise planting,} & \text{PR} \leq \text{PR}_{cut} \\ \\ \text{Widthwise planting,} & \text{otherwise} \end{cases} \tag{5.4}$$

Analyzing a range of PR values from the two planting patterns images at different θ_f will determine the best PR cutoff (PR_{cut}) value to identify the planting patterns. A low PR value with respect to the PR_{cut} will represent the lengthwise pattern compared to the widthwise pattern.

5.3.6. Crop Row Orientation Estimation

Aligning the crop rows vertically, irrespective of the planting pattern, was essential for identifying rows. It is not always possible to have vertical crop rows in the input UAS

image. In such cases, either the user can preprocess the image manually to make the rows vertical or automatically through image processing. Therefore, in this study, an automatic method was developed to identify the crop row orientation (angle) using a novel profile-based approach. The best vertical rows orientation by rotating the input image ensures accurate and easier rows identification.

5.3.6.1. Determining the image initial direction of rotation

To have a computationally efficient method for aligning the rows vertically, it is first necessary to determine the direction of rotation as clockwise (CW) or counter clockwise (CCW). Obtaining the sense of rotation beforehand minimizes the number of rotation iterations (discussed subsequently). The logic of determining the rotation direction was based on two parameters: field angle (θ_f, measured from +ve x-axis (fig. 5.4A), and the planting pattern (lengthwise or widthwise). Field angle was obtained based on the "user-defined field ROI" delineated by the user (fig. 5.4A) and the planting pattern was obtained based on the profile values (eq. 5.4). If the field ROI was not delineated by the user, then the entire image will be considered as the ROI automatically. Using these two parameters, the direction of rotation (CW or CCW) was found based on the following cases (eq. 5.5):

$$\text{Rotation direction} = \begin{cases} \text{CW}, & \text{lengthwise \& } \theta_f \geq 90°\text{(or)} \\ & \text{widthwise \& } \theta_f < 90° \\ \text{CCW}, & \text{lengthwise \& } \theta_f \leq 90°\text{(or)} \\ & \text{widthwise \& } \theta_f > 90° \end{cases} \tag{5.5}$$

5.3.6.2. Method of crop row orientation evaluation

For determining the row orientation, most of the studies reported so far adopted Hough transform (Jin et al., 2017; Varela et al., 2018). Hough transform detects the collinear points in an image and establishes a straight line joining the points (Hough, 1962). There are two major limitations in this procedure: (i) the search of the collinear points is performed in parameter space, which is computationally intensive (Ji & Qi, 2011), and (ii) there will be false collinear points, which may not represent a crop row (Tang & Tian, 2008). Therefore, to overcome the aforementioned limitations a simple methodology of determining the row orientation using plant profile is proposed. A demonstration of this methodology with an example binary image of a field inclined at θ_f is presented in figure 5.4.

A representative simple rectangular test ROI of size 50 % of image size was fixed at the center and clipped off (fig. 5.4A). This test ROI image was subjected to rotations of 12 test angles at intervals of 10° (Iteration-I) in the range from $\theta_f - 10°$ to $\theta_f + 100°$ to encompass the 90° sweep with two additional angles on both sides (fig. 5.4B). The test rotation direction was decided based on the planting pattern and θ_f using the equation 5.5. In the example (fig. 5.4), with $\theta_f = 132°$ given the lengthwise planting pattern and $\theta_f > 90°$ the CW rotation was used with 12 test angles in the range of 142° down to 32°. After each rotation, the profile was obtained for the rotated image using `getProfile()`, which was analyzed for further fine rotation to obtain the best alignment of the rows.
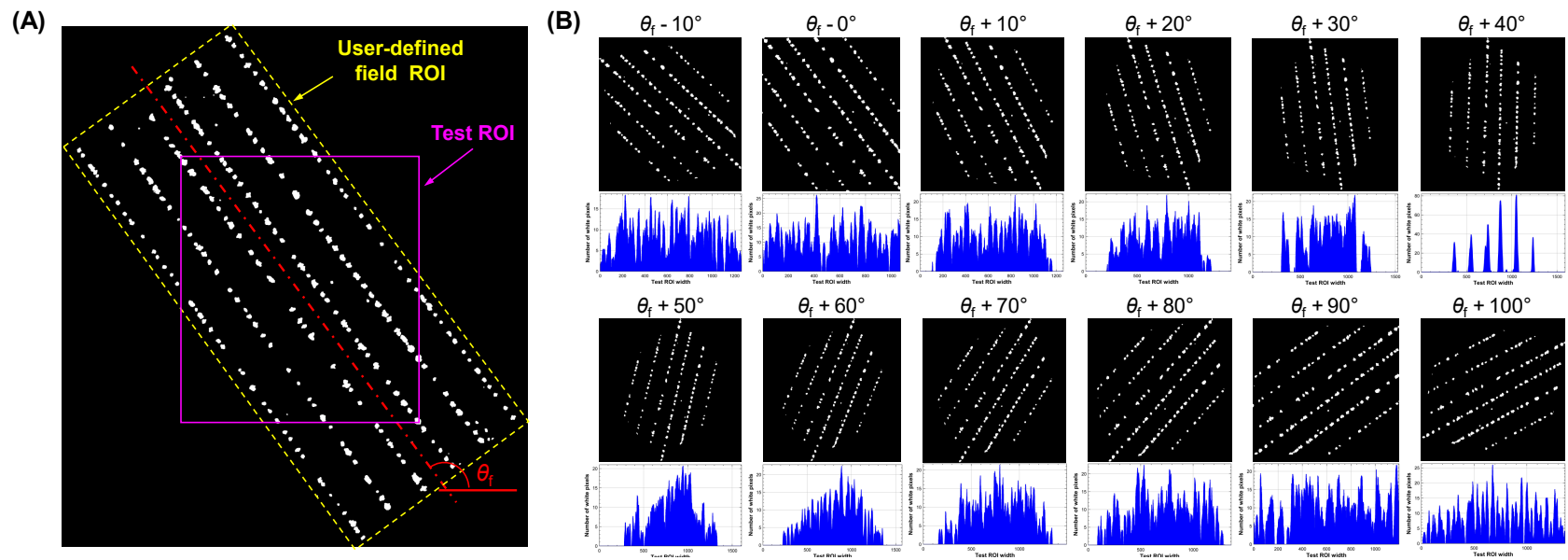
Figure 5.4. Optimum rotation angle determination strategy for making vertical crop rows in a field inclined at θ_f: (A) a representative "Test ROI" (50 % of image size) positioned at the center of the binary image; and (B) series of CW rotations performed on the test ROI from at 10° steps from $\theta_f - 10°$ to $\theta_f + 100°$ (Iteration-I) along with their respective profile plots.

The combination of rotation direction and 12 test angles captured all the possible row orientation variations profiles. The pattern of the profile plots was unique for each rotated image (fig. 5.4B). Following the earlier observed phenomenon (fig. 5.4B), distinct peaks in the profile plot at $\theta_f + 40°$ indicated the possible vertical row orientation (fig. 5.4B) after 40° CW rotation of the image. It can be seen that this rotation (40° CW) is not the optimized result and further refinement only will produce the best angle, also an automatic method to determine the optimized angle should be developed.

5.3.6.3. Optimized angle of image rotation for vertical rows

To automatically identify an optimum angle of rotation, a simple method from the profile that is unique among the rotation angles will be necessary. For this purpose, six descriptive statistics measures such as, mean, median, maximum value (P_{max}), standard deviation (σ), skewness, and kurtosis obtained from the profile values using `DescriptiveStatistics()` class, and a new measure defined as σP_{max} was also derived to test and determine the best indicator for optimum rotation angle.

With the optimum rotation angle from Iteration-I, a finer angle search with 1° step for 20 times were performed between the one step before ($\theta_f + 30°$) and after ($\theta_f + 50°$) the optimum (fig. 5.4B) in the Iteration-II. Similarly the Iteration-III was continued with progressive refinement. At each iteration, the step angle was reduced by 1/10th of the previous step angle. Each iteration will refine the result by fixing one additional decimal to the previous result at stay in the "optimum zone" after the Iteration-I. This type of searching was similar to the "decimal search technique" followed for determining the optimum moisture diffusivity of grains in drying (Igathinathane, 1997). For the automatic optimum search angle determination, the descriptive statistics measures were again

135

calculated and used from the respective profile plots. Based on the number of iterations, this method performed the search to the desired level of accuracy. However, in this study, only three iterations were performed to obtain the fine search angle up to one decimal degree accuracy (0.1°). Finally, the optimized accurate rotation angle determined was applied to the image (fig. 5.5A) to produce vertical crop rows and used for row identification and further analyses.

5.3.7. Crop Row Identification using Pixel Profile Plot

To evaluate the plant stand count along each row, it is imperative to identify the individual crop row locations and perform counting within each row (Flores et al., 2017). Reported studies on the crop row identification methods from an image can be broadly grouped into the following five categories: least-square fitting method, Hough transform method, vanishing point method, stereo-vision method, and a simplified profile-based method (G. Jiang, Wang, & Liu, 2015). Out of the available methods, row identification using vertical profile is common (Søgaard & Olsen, 2003). In the profile-based approach, selecting a profile cutoff value is one of the critical problems. This cutoff value was often hard-coded or assumed as an average value worked best only for the specific image and other cutoff had to be selected for other images. Therefore, in the present study, we have developed an automatic cutoff determination applied to the given image for the profile method that identified crop rows from the UAS image (fig. 5.5).

5.3.7.1. Profile smoothing and their spacing

The obtained profile data plot had irregular edges that lead to unnecessary peaks in the analysis. Therefore, the profile was smoothed using Apache Commons Math™ `LoessInterpolator()` class (Cleveland, 1979) with the following parameters:
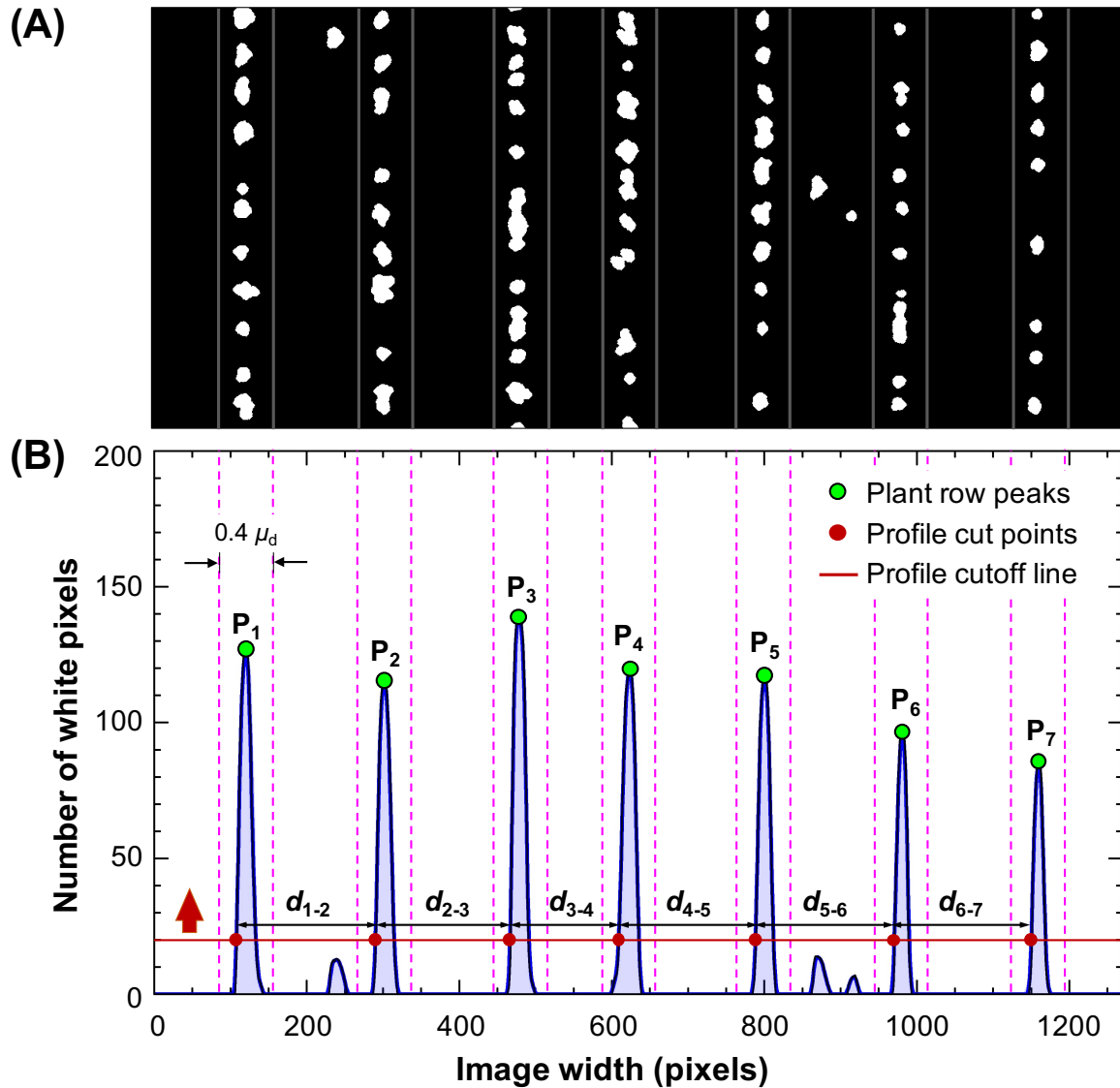
Figure 5.5. Crop rows identification from binary image by profile analysis:
(A) a section of a binary image with vertical rows and row edge lines derived based on the identified profile peaks; and (B) row identification from the profile plot of the binary image. P_1, P_2, ..., P_7 are the crop row peaks; d_{1-2}, d_{2-3}, d_{3-4}, ..., d_{i-i+1} are the distances between subsequent profile points of intersection with the cutoff line.

bandwidth $= 10$, robustness iterations $= 2$, and accuracy $= 1 \times 10^{-12}$. It was evident from the smoothed profile plot (fig. 5.5B) that the crop rows can be clearly identified by the major peaks and were also evenly-spaced. Therefore, it was hypothesized that the crop rows can be identified from the reduced standard deviation of the distances between the points of intersection of the left edge of the peaks and the profile cutoff line ($d_{1\text{-}2}$, $d_{2\text{-}3}$, $d_{3\text{-}4}$, ..., $d_{\text{i-i+1}}$; fig. 5.5B), as uniformly spaced distances carry similar data thus have reduced standard deviation.

5.3.7.2. Automatic profile cutoff for peak determination

The profile cutoff value influences the number of peaks above the level and their distribution along image width, hence there exists an optimum value where the distribution of the peaks is uniform. In the plugin, the optimum cutoff value was automatically identified by incrementing the value from 0 to the global maximum peak with unit increment.

At every increment, the "left edge" of the each row peak profile was obtained, and if the profile data point **P[i]** satisfied the condition "**P[i-1]** < cutoff < **P[i+1]**", based on the adjacent points (fig. 5.5B). The number of times this condition was satisfied indicated the number of crop rows (N_{rows}) included at that cutoff. For different cutoffs ranging from 0 to maximum peak, the distances between the profile cut points ($d_{1\text{-}2}$, $d_{2\text{-}3}$, $d_{3\text{-}4}$, ..., $d_{\text{i-i+1}}$), the mean (μ_d), and the standard deviation (σ_d) were calculated to identify the optimum cutoff. The cutoff with minimum σ_d (when peaks are equally spaced) in combination with low cutoff (where more crop rows included) was automatically determined as the optimum cutoff value.

It can be observed that evenly spaced rows of plants produced the corresponding number of major peaks that are evenly spaced that in turn produced the minimum σ_d; on the other hand, the random location of plants, lower cutoff values (0–10; fig. 5.5B), or improper orientation of the field image with respect to the horizontal would produce an excessive number of peaks along with increased σ_d. It was interesting to note that incrementing the cutoff value beyond a certain limit reduced the number of profile cut points but increased σ_d (cutoff > 90; fig. 5.5B). Similarly, at lower cutoff values (≤ 15), the number of peaks were identified more but, this increased the σ_d (peaks identified = 10; fig. 5.5B).

5.3.7.3. Peaks identification and crop rows ROI layout

After selecting the optimum cutoff value, the peaks ($\mathbf{P_1}$, $\mathbf{P_2}$, ..., $\mathbf{P_7}$) of each row were determined as the maximum value between two consecutive profile data points of intersection with the cutoff line (fig. 5.5B). It can also be visualized that a range of values around the optimum cutoff line would have also produced the same set of peaks. Based on the preliminary analysis, the width of crop row ROI was arbitrarily fixed at $\pm 20\,\%$ of μ_d ($0.4\mu_d$) aligning the peak in the middle (fig. 5.5). The regions outside of these crop row ROI were excluded from the analysis, and this method efficiently avoided any weeds or other unintended plants that emerged between the rows.

5.3.8. Image Dimension Calibration

The dimension calibration procedure determines a scale factor based on the reference object dimensions included in the image to derive the plant spacing and area in physical units (e.g., m, m^2). To have an accurate scale factor, it is desirable to have a large reference dimension such as the whole study area as it could improve measurement

accuracy and avoid image geometrical distortions (Igathinathane et al., 2009b; Sunoj et al., 2018). In this study, as the vertical reference measurement varied among field images, the maximum horizontal reference measurement (row spacing) was considered for the calibration. This reference measurement was taken as the center-to-center distance (pixels) between the first and last crop row. Knowing the actual row spacing the scale factor was calculated as:

$$\text{SF} = \frac{S_{\text{rows}}(N_{\text{rows}} - 1)}{d_{1-N_{\text{rows}}}} \tag{5.6}$$

where, SF is the scale factor (m/pixel), S_{rows} is the row spacing an user-input (m; fig. 5.1B), N_{rows} is the number of rows determined by the row identification algorithm, and $d_{1-N_{\text{rows}}}$ is the center-to-center between first and last row measured in the image (pixels).

The numerator in equation 5.6 is the maximum horizontal reference measurement (m), while the denominator is the corresponding measurement (pixels) in the image. The scale factor thus calculated for the UAS images used was ≈0.003 31 m/pixel. The scale factor when multiplied with any measured distances in the image in pixels will be converted into the physical unit (m).

5.3.9. Clusters Segmentation by Profile Analysis and Geometry

Plant stand count, once the plants are well separated (singulated) and the rows ROI correctly identified, is a straightforward process. However, resolving the two or more touching plants, termed "cluster," is quite challenging. The occurrence of clusters will become more likely as plants grow and adjacent plants' leaves touch. Clusters, when not

resolved, usually produce underestimated stand count, as a cluster will be identified and counted as a single plant.

Most of the reported studies on stand count analysis did not deal with the cluster segmentation as it would possibly increase the algorithm's complexity or lack of readymade tools. Although watershed segmentation, the most common method, is successful in resolving overlapping objects in some studies (Igathinathane & Ulusoy, 2016; Sunoj et al., 2017); however, it resulted in over-segmentation especially when the object has a complex geometry as plants.

Therefore, a two-stage segmentation approach was developed and tested in this study. The first stage was based on the sideways pixel profile shape of the clusters, derived by rotating the identified cluster related ROI. The second stage was based on the cluster standard geometry, such as bounding box height and area. This novel approach is expected to resolve any cluster irrespective of its geometrical complexity. The details of the developed segmentation approach are discussed subsequently.

5.3.9.1. Cluster identification and profile pattern analysis

To resolve the plant clusters, they have to be identified first. The bounding box heights (BBH) of all the objects were first assessed using ImageJ's `ParticleAnalyzer()` and a cutoff value (BBH_{cut}) of 87th percentile (fixed after preliminary analysis) of this data was used to identify clusters. Only the qualified objects were identified as a cluster and subjected to the first stage of segmentation. The identified clusters were cropped and rotated 90° CCW to obtain their sideways profile (fig. 5.6). To avoid the profile roughness, the raw profile was again smoothed using `LoessInterpolator()`. The identified cluster can be categorized as a "double" or a "multiple" based on their characteristics.
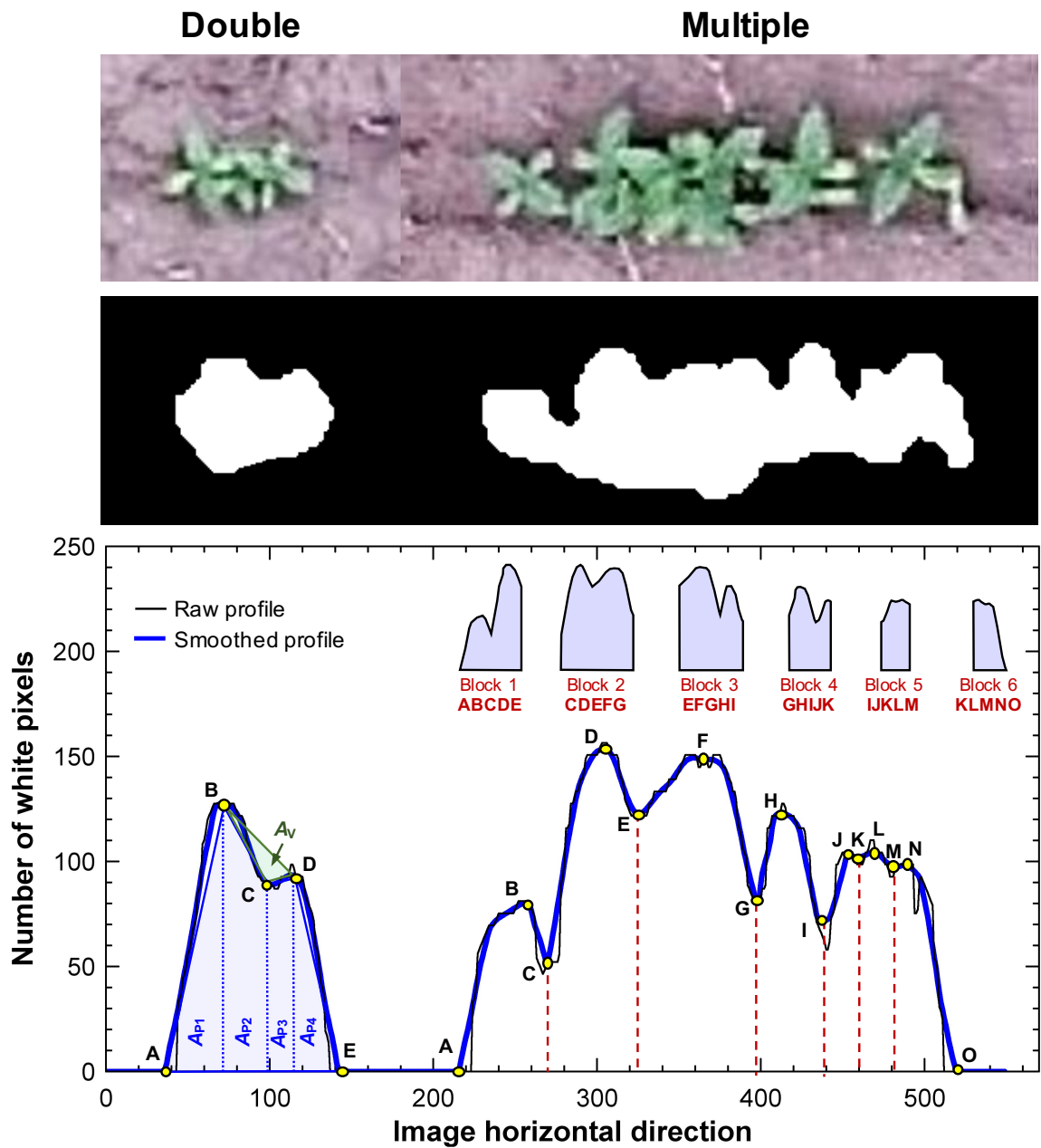
Figure 5.6. Demonstration of the plants' cluster segmentation strategy based on profile area (A_P; trapezoid formula) and valley area (A_V; Heron's formula). A "double" cluster is the simplest case (peak-valley-peak pattern), while the "multiple" cluster (several peaks and valleys) which can be analyzed as a series of doubles.

A double is a simple case and can be observed from the sideways profile pattern. A double has a characteristic profile pattern with two major peaks and a valley enclosed between the peaks (peak-valley-peak pattern) — characteristics of two plant projections coming together. In the case of a multiple, the number of peaks and valleys in a profile will be high, but the profile can be partitioned into individual blocks representing a double having peak-valley-peak (e.g., Blocks 1–6; fig. 5.6). The peaks in the profile indicate individual plant's center (more pixels) and the valley indicates the overlapping portion between plants (fewer pixels).

5.3.9.2. Plants cluster profile analysis

Using the smoothed plants cluster sideways profile, the profile start and end points along with possible peaks and valleys of each profile were assessed by calculating slopes (dy/dx) using the consecutive data points along the profile. Here, dy will be the varying difference between the consecutive profile values, but the dx difference will always be one. A profile start point was determined when the profile value changed from zero to a positive value and an endpoint when the value changed from a positive value to zero. The peak and valley points were identified when dy/dx changed from a positive to a negative slope and a negative to a positive slope, respectively.

5.3.9.3. Profile and valley areas estimation

With the identified peaks and valleys of the plant cluster sideways profile, a segmentation strategy was developed (fig. 5.6) that compared the valley area (A_V) with respect to the profile area (A_P). To make computation efficient, for identifying a qualified segmenting valley, it was sufficient to use an approximation for A_P (method of trapezoid) and A_V (method of triangle, Heron's formula) rather than the whole area formed by the

143

curve. A double is the most simplified form or a multiple that can be divided and analyzed into a series of doubles (fig. 5.6).

A peak-valley-peak pattern of a double was sectioned into two triangles at the ends (A_{P1} and A_{P4}) and two trapezoids at the mid-region (A_{P2} and A_{P3}; fig. 5.6). The area of the individual sections was calculated using the generalized trapezoid area:

$$A_{Px} = \frac{1}{2}w(h_1 + h_2) \tag{5.7}$$

where A_{Px} is the area of one peak section (pixel^2), w is the distance between the parallel sides of the trapezoid (pixel), and h_1 and h_2 are the parallel sides length (pixel).

This generalized trapezoid formula (eq. 5.7) can be equally applicable to the triangles (A_{P1} and A_{P4}) by substituting either one of the side length h_1 or h_2 as 0. The total profile area of the double was obtained by adding the four individual sections as $A_P = A_{P1} + A_{P2} + A_{P3} + A_{P4}$. By appropriate single shifting of the four sections of the profile, the multiple's profile area (A_P) can be obtained similar to the doubles.

The valley and two peak points on either sides, when connected, formed a triangle (fig. 5.6), and its area was calculated using Heron's formula:

$$A_v = \sqrt{s(s-a)(s-b)(s-c)} \tag{5.8}$$

where A_v is the area of the valley ($\triangle$ BCD, pixel^2; fig. 5.6), s is the semi-perimeter $(a + b + c)/2$, and a, b, and c are the sides length of $\triangle$ BCD, namely BC, CD, and BD (pixel), respectively. The sides length were determined by distance formula $(d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2})$.

5.3.9.4. Segmentation location selection based on area ratio

An area ratio (AR) was developed to identify the segmentation region of the cluster based on A_V and A_P (eq. 5.9). The AR indicated whether the valley point is a potential segmentation location by comparing it with a selected cutoff value (AR_{cut}) or not. Smaller AR values might denote the orientation effect of the same plant, and higher AR values denote touching of two plants thus a qualified segmentation location. To obtain a range of AR values for segmentation, representative plant clusters were obtained from the input images and analyzed.

$$\text{AR} = \frac{A_V}{A_P} = \begin{cases} \text{Segmentation,} & \text{AR} \geq \text{AR}_{cut} \\ \text{No segmentation,} & \text{otherwise} \end{cases} \tag{5.9}$$

At the qualified segmentation location, a one-pixel thick black vertical line was drawn in the binary image to digitally segment the plants. The segmented cluster was rotated 90° CW back and inserted into the binary image at the same location where it was cropped, using the `insert()` method from the `ImageProcessor` class.

5.3.10. Segmentation based on Bounding Box Height and Area

It was not always expected to have clear peak-valley-peak patterns in the sideways profile. Sometimes, when the overlap between the plants was high and the plants were found side by side perpendicular to the row length, the sideways profile did not produce a clear valley in the profile for segmentation. Therefore, the second level of segmentation based on cluster standard geometry, such as BBH and area (*A*) was included in the algorithm to resolve the unsegmented clusters.

The image obtained after the profile-based segmentation was used for this process. The clusters were again identified using the same cutoff value (87th percentile of BBH) as the first stage of segmentation. It should be noted that some of the clusters that were segmented by profile will not be identified, as it might have already undergone segmentation. Along with BBH_{cut}, area cutoff (A_{cut}) as the 87th percentile of area data was also used for cluster identification, as the area represented the cluster better.

At this segmentation stage, three scenarios occur in the identification and segmentation of clusters (eq. 5.10). Possible number of plants (N_{plants}) in the qualified clusters were calculated by dividing the object's BBH or A by its respective cutoff value (BBH_{cut} or A_{cut}), and N_{plants} was used to segment the qualified clusters.

$$N_{plants} = \begin{cases} A/A_{cut}, & (BBH > BBH_{cut}) \text{ and } (A > A_{cut}) \\ BBH/BBH_{cut}, & (BBH > BBH_{cut}) \text{ and } (A < A_{cut}) \\ A/A_{cut}, & (BBH < BBH_{cut}) \text{ and } (A > A_{cut}) \end{cases} \qquad (5.10)$$

The N_{plants} is often a fractional value, thus it was rounded to the next greatest integer. After determining N_{plants} in each cluster, the cutting height (H_{cut}) was determined by dividing BBH of the cluster by N_{plants} to obtain the H_{cut}, i.e., $H_{cut} = BBH/N_{plants}$, assuming that the cluster contained N_{plants} of equal size. Since the motivation was to count the plants in each row, this approach of chopping the clusters was valid; however, for deriving any shape features, this approach may not be valid as it misrepresents the shape of individual plants.

With the H_{cut} value, the clusters were cut into individual plants of equal height by drawing a black line of one-pixel thickness at each H_{cut} locations starting from top to bottom of the bounding box. The clusters qualified in the first and second cases (eq. 5.10) were cut horizontally, while the last that qualified based on the excess area was cut vertically based on the N_{plants}.

5.3.11. Plant Vigor Determination and Mapping

Physiologically plants with good vigor produce more vegetation and healthy leaves (foliage). Therefore, the total area of the foliage is an excellent indicator of the plant's vigor. In this image-based study, the projected area, which can be considered directly proportional to the actual foliage area, was considered as an indicator of the plant vigor. Also, a color-coded plant vigor map will be helpful as it provides an insight into the crop growth and health. This information will be useful for making site-specific farm management decisions.

5.3.11.1. Plant vigor determination

The area (eq. 5.6; mm^2) and the location of individual plants (centroids) after segmentation were assessed using ImageJ's `ParticleAnalyzer()`. Plants vigor was classified into three categories, namely, class I, class II, and class III, based on mean area (μ_A) and standard deviation (σ_A) values:

$$\text{Plant vigor} = \begin{cases} \text{Class I,} & A < \mu_A - \sigma_A \\ \text{Class II,} & \mu_A - \sigma_A \leq A \leq \mu_A + \sigma_A \\ \text{Class III,} & A > \mu_A + \sigma_A \end{cases} \qquad (5.11)$$

The class I was considered under-performing plants, class II as optimum, and class III represent over-performing plants. It is expected that with normal growth the plant area data will have more plants in class II, followed by class III, and a limited portion in class I. There is a possibility that the plants in class III might be produced due to some unsegmented clusters as well.

5.3.11.2. Color-coded plant vigor map generation

To produce a color-coded vigor map, the plant area data were first normalized to produce a smooth distribution of area values (gradient) as follows:

$$A_{\text{norm}} = \frac{A_i - A_{\text{max}}}{A_{\text{max}} - A_{\text{min}}} \qquad (5.12)$$

where A_{norm} is the normalized area with values between 0 and 1, A_i is the area of ith object, A_{max} and A_{min} are the maximum and minimum areas (pixel^2), respectively.

The binary image of the whole plot is then converted into a grayscale image based on the area values and the color-coded map was created from the grayscale image using ImageJ's `Lookup Tables` (LUT). At each plant's centroid, a `Wand()` was applied in the binary image and a filling color was obtained by multiplying the A_{norm} with 255 to produce a color between black (0) and white (255), which produced the grayscale image. A custom color scheme was developed spanning from red, green, and yellow and loaded into ImageJ's LUT schemes database. The custom color scheme was applied to produce the color-coded image, and an appropriate legend was also included representing the color codes in physical units (mm^2).

5.3.12. Plant Stand Counting and Labeling

The total number of plants in the segmented binary image can be analyzed in a single run of `ParticleAnalyzer()`. However, it is required to count the plants along the rows so that the distances between subsequent plants can be calculated. Therefore, the row ROI was overlaid on the segmented binary image and the number of plants in each row was analyzed. The plants were counted from the top to bottom and were labeled.

The counted plants marked at the centroid (obtained from the `Analyze Particles...` measurements) with a red circular marker. The markers in the labeled RGB image were drawn using `fillOval()` method. The counter labels were generated using `TextRoi()` class and placed alongside a x-shift from the centroid based on a percentage of the image width. The labeled image was used for validating the plugin count with reference to visual manual count evaluated row-wise and the total.

5.3.13. Spatial Distribution Analysis and Mapping

The centroids obtained from the plant stand counting were also useful for calculating the inter-plant distances in each row. Using the distance formula, the distance between the two centroid points was calculated in pixels, which was converted into physical dimensions (m) using the scale factor (eq. 5.6), and used for plant spacing and distribution analysis. This analysis also included the determination of several performance indices, distribution uniformity, and plant stand spatial distribution mapping.

5.3.13.1. Existing planter performance indices

Following the existing method to evaluate planter performance (Kachman & Smith, 1995), the plant spacing data was categorized into five classes using the ideal plant spacing (S_I) obtained as an user-input (fig. 5.1B). The following five classes (**I–V**) are defined as:

0–0.5S_I (**I**), 0.5S_I–1.5S_I (**II**), 1.5S_I–2.5S_I (**III**), 2.5S_I–3.5S_I (**IV**), and >3.5S_I (**V**). The S_I followed in this study was 0.22 m, that produced the classes range of 0–0.11 m, 0.11–0.33 m, 0.33–0.55 m, 0.55–0.77 m, and >0.77 m. These classes were named as multiple, ideal, single skip, double skip, and triple skip, respectively.

Mapping the obtained spacings into each class, the following four performance indices (Kachman & Smith, 1995) were calculated as:

$$\text{Ideal index} = \frac{n_{II}}{N} \tag{5.13}$$

$$\text{Skip index} = \frac{n_{III} + n_{IV} + n_V}{N} \tag{5.14}$$

$$\text{Multiple index} = \frac{n_I}{N} \tag{5.15}$$

$$\text{Precision} = \frac{\sigma_{II}}{S_I} \tag{5.16}$$

where n_I, n_{II}, n_{III}, n_{IV}, and n_V are number of spacings in **I**, **II**, **III**, **IV**, and **V** classes; N is the total number of spacings in the image; σ_{II} is the standard deviation of spacing (m) in class **II**; and S_I is the ideal plant spacing (0.22 m).

5.3.13.2. Existing distribution uniformity indices

As indices related to spatial distribution applicable to plant spacing were not reported, similar measures from other fields of study were considered and new indices were developed in this research. The distribution uniformity, a common measure in the evaluation of a sprinkler irrigation system (volume of water collected at grid points), was adapted to the plant stand spatial distribution. The two common irrigation-based distribution uniformity measures modified using plant spacing data are (i) lower quartile

distribution uniformity, and (ii) Christiansen's uniformity coefficients are described subsequently.

The lower quartile distribution uniformity (Dudek & Fernandez, 2015) is the most simple and commonly used measure:

$$\text{DU} = \frac{\mu_{\text{LQ}}}{\mu_{\text{S}}} \times 100 \qquad (5.17)$$

where DU is the lower quartile distribution uniformity (%), μ_{LQ} is the average value of lowest 25 % spacing data(m), and μ_{S} is the overall average of all spacing data (m). Irrigation industries have classified the DU into the following four categories: DU <77 % is very poor, 77–83 % is poor, 83–90 % is good, and >90 % is excellent.

The Christiansen's uniformity coefficient (Christiansen, 1942) considered the absolute deviation of all the spacing data to the overall average spacing data, while the DU considered only the average of lower quartile data (eq. 5.17). The Christiansen's uniformity coefficient was calculated as:

$$\text{UC} = 1 - \frac{\sum_{i=1}^{N} |d_i - \mu_{\text{S}}|}{\mu_{\text{S}}} \times 100 \qquad (5.18)$$

where UC is the Christiansen's uniformity coefficient (%), N is the number of spacing data, d_i is the ith spacing distance (m), and μ_{S} is the overall average of all spacing distances (m).

5.3.13.3. Developed ideal spacing uniformity index

The ranges of value from these spacing distributions (DU and UC) are not expected to be in a similar range that applies to irrigation applications. These existing measures of uniformity did not incorporate the ideal plant spacing to evaluate the performance of the

planter. This is important as seeds spaced equidistant deviating from the ideal spacing will be evaluated to be 100 % uniform (eqs. 5.17 and 5.18), but may not be the desirable outcome from the planting equipment perspective. Therefore, we have developed a uniformity index that takes into account of ideal spacing and spatial distribution for the entire field and was calculated as:

$$\text{ISUI} = \frac{\frac{1}{N_\text{I}}(N_\text{I}S_\text{I}^2)}{\frac{1}{N_\text{o}}\sum_{i=1}^{N_\text{o}} d_{\text{o},i}^2} \times 100 \tag{5.19}$$

where ISUI is the ideal spacing uniformity index (%); $N_\text{I} = N_\text{PR} \times N_\text{rows}$, is the ideal number of plants in the given area with the number of identified rows (N_rows); $N_\text{PR} = (L_\text{row}/S_\text{I}) + 1$, is the maximum possible number of ideally spaced plants in a row; L_row is the length (m) of the row based on field ROI (fig. 5.3A); S_I is the ideal spacing (e.g., 0.22 m); N_o is the observed plant count; and $d_{\text{o},i}$ is the observed plant spacing distance (m). In case of more multiples, N_O exceeds N_I. In such cases, the N_I in the numerator is replaced with N_O. An increased ISUI value indicates increased number of plants in the ideal spacing (more uniform), and lower ISUI value indicates the increased deviation from the ideal spacing.

5.3.13.4. Plant stand spatial distribution mapping

A color-coded plant stand spatial distribution map based on the five identified classes (5 colors) was also generated for better visualization of the entire field. The distance between the plants was color-coded as rectangles for this visualization. The `fillRect()` method was used to produce the colored rectangles of width same as row ROI and height as the centroid-to-centroid distance between two plants. The different filling color was automatically selected from the predefined colors indicating each class

(blue: multiple, green: ideal, yellow: single skip, orange: double skip, and red: triple skip) in the program. This plant spatial distribution map provides a good insight into the seeds' emergence and planter's performance.

5.3.14. Validation of the Plugin

The stitched UAS sunflower image (Set I), which was huge and cannot be suitable for visual-manual counting, was partitioned into 25 equal sections and used for the validation. A few sections were rotated manually to produce different field angles (θ_f) and used as the input to test the plugin for optimum row orientation method. For validating the stand count, the plants in the image sections were visually counted (N_{visual}) after enlarging the image and used as the reference to compare with the plugin count (N_{plugin}). With these values, error and accuracy were calculated as:

$$\text{Error}\,(\%) = \frac{N_{plugin} - N_{visual}}{N_{visual}} \times 100 \qquad (5.20)$$

$$\text{Accuracy}\,(\%) = 100 - |\text{Error}| \qquad (5.21)$$

5.4. Results and Discussion

The stitched images (Set I and II) were used as the primary input, and the test images (sub-images) generated after necessary sectioning and rotation were employed in the testing and validation. The various results of row orientation, plant clusters segmentation, and plant stand count in the form of determined constants and performance evaluation, illustration images along with tabular and graphical outputs were presented and discussed.

5.4.1. Profile Ratio Results for Planting Pattern Determination

The profile ratio (PR) values tested with sample images ($n = 5$) of lengthwise and widthwise planting pattern between 0° and 160° is presented (Appendix C.1). It was clear that the lengthwise planting pattern produced lower profile ratio values compared to the widthwise planting pattern. The PR values of lengthwise planting pattern ranged from 0.12–0.23, while the widthwise planting pattern ranged from 0.79–0.89, displaying a narrow range of variation at different angles. The overall average and standard deviation of PR value for lengthwise planting pattern were 0.17±0.03 and the widthwise planting pattern was 0.84±0.02.

As the PR values had a clear separation in the range of values, a single cutoff value, $PR_{cut} = 0.51$ (rounded), which lies in the middle of two ranges were fixed for planting pattern determination. Any other cutoff value between the two ranges would have worked, but this PR_{cut} value produced the maximum separation between two planting patterns. A planting pattern with PR ≤ 0.51 was identified as lengthwise and otherwise as a widthwise planting pattern. This PR_{cut} value worked well for all images tested at different θ_f and the individual section images that were used for plugin validation achieving 100 % accuracy.

5.4.2. Optimum Angle Determination from Descriptive Statistics

The developed strategy to make the rows vertical from any given image used seven simple descriptive statistics measures to determine the optimum angle of rotation. The strategy was demonstration using an example image with $\theta_f = 113°$ (fig. 5.7). Among these measures, σP_{max} was the best indicator as the these values only showed a prominent peak to determine the optimum rotation angle.
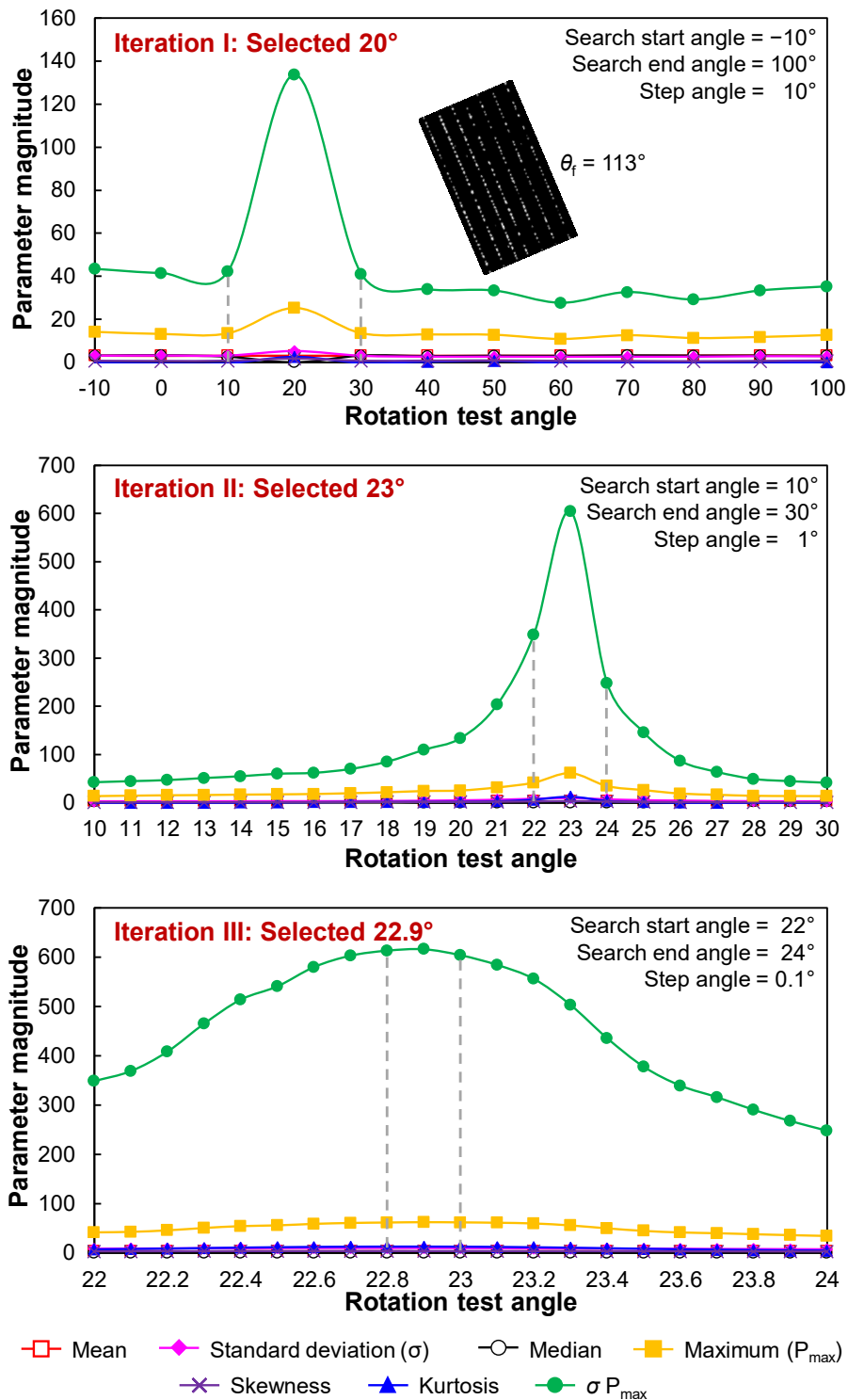
Figure 5.7. Variation in descriptive statistics measures for optimum angle of rotation in an example field with $\theta_f = 113°$ (Figure inset) subjected to different rotation test angles between $\theta_f - 10°$ and $\theta_f + 100°$ for optimum angle of rotation determination.

The iterations were necessary to obtain a precise angle for rotation in an efficient manner. The example field image had lengthwise planting pattern with $\theta_f = 113°$ ($\theta_f > 90°$); hence, the rotation direction was CW. For a CW rotation, the search started from 10° towards CCW (−ve direction) and subsequently rotating at CW direction (+ve direction) with 10° steps. Thus, in the first iteration, a series of rotation operation were performed between 123° down to 13°, rotating CW at 10° steps. The optimum rotation angle was was observed to be 20° (i.e., $\theta_f + 20°$) based on σP_{max} curve's peak, which gives an indication that the exact angle lies close to 20°.

Proceeding further, the second iteration within a reduced step angle of 1° (1/10th) on limits from one before and one after the optimum rotation angle (10°–30°) revealed that the next optimum angle lies around 23°. Similarly, the final angle search between 22° and 24° performed with a step angle of 0.1° the optimum angle was determined as 22.9°. It was found not necessary to perform beyond three iterations as there was no visible change in the orientation and will not affect the results of further analysis.

This method of iterations is efficient, as a direct search method with a fine step angle of 0.1°, from −10° to 100°, would have performed 1100 rotations compared to the 54 rotations needed with the method of iterations. Therefore, the three-step iteration procedure is ≈20 times faster and computationally efficient.

5.4.3. Row Identification Method Performance

Aligning the rows vertically facilitated in identifying rows by simple profile analysis. The developed row identification method automatically determined the profile cutoff value, based on the logic of minimizing the standard deviation among profile peaks

distances (fig. 5.5B) and identified the rows accurately for all images ($n = 25$) validated. This resulted in a row identification accuracy of 100 % for all the tested images.

The row length of these validation images was about 10 m, and to evaluate the capability of the method for longer row lengths other whole stitched images were also tested. With the UAS stitched image of the sunflower field of about 48 m and corn field image of about 100 m row lengths, the method worked accurately in both the instances. Even though the method was developed using one crop (sunflower), the method was capable of accurately identifying the rows in other crops (e.g., corn) images as well. This proves the applicability and flexibility of the developed method and confirms that as long as the crop rows were straight the profile method will not have issues in the row identification. Unlike other methods reported, the developed method has the advantage of automatically identifying the profile cutoff value. However, the limitation of the method, such as curvilinear rows (circular planting), irregular fields can be addressed in future research.

5.4.4. Cluster Segmentation Performance Evaluation

To demonstrate the segmentation process, a few extracted sample clusters and the process of obtaining the segmented plants are illustrated in figure 5.8. A color montage image (fig. 5.8A) used was a collection of singulated plants and clusters of varying shapes extracted from different parts of the tested images. The first three plants shown are single plants (fig. 5.8A), presented for demonstration, while the other plants get qualified as a cluster based on BBH_{cut} for profile-based segmentation.
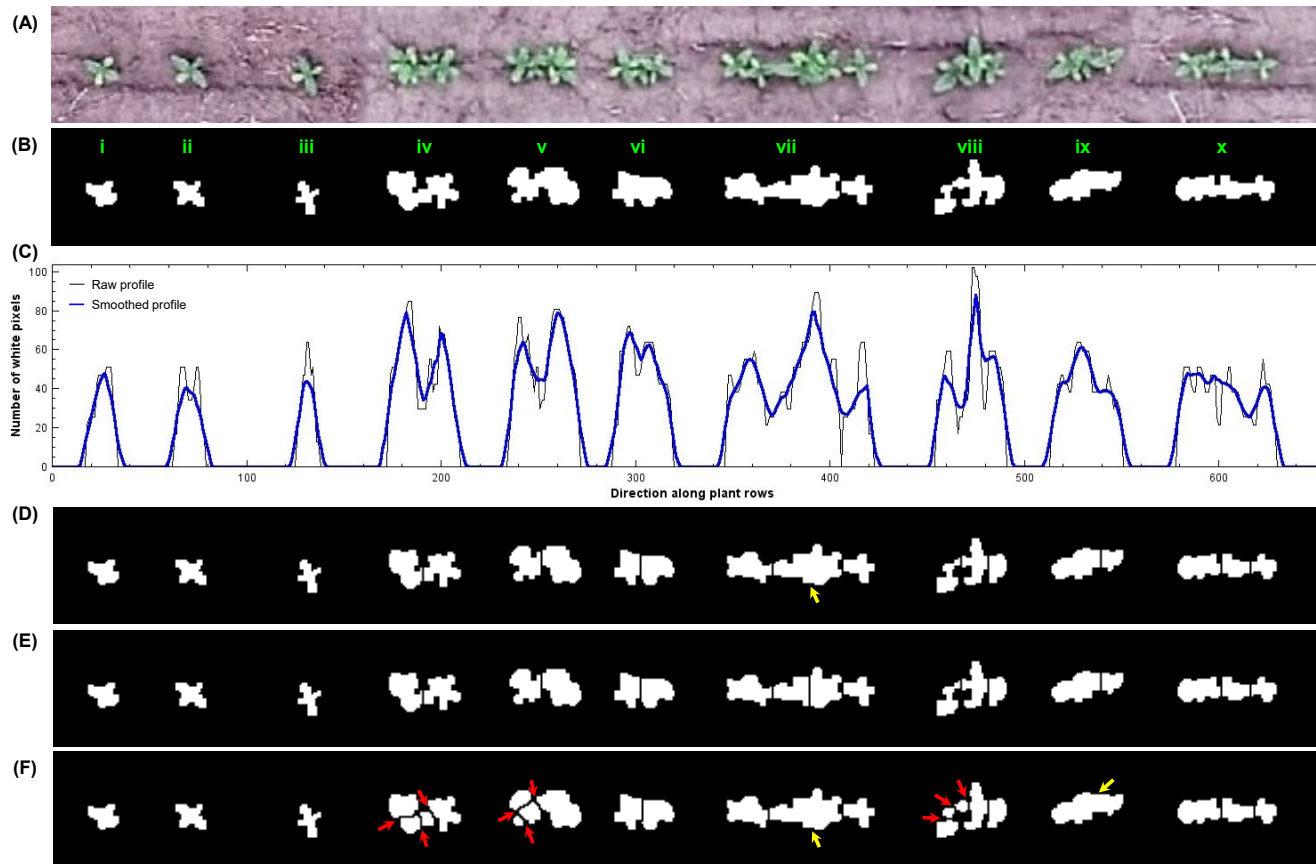
Figure 5.8. Demonstration of the developed profile- and geometry-based cluster segmentation procedure using a montage image: (A) digital color image montage image of individual plants and clusters; (B) binary image obtained after YUV thresholding and morphological operations; (C) raw and smoothed profile of the binary image; (D) segmented image obtained based on the profile-based procedure; (E) segmented image after the geometry based segmentation, which segmented the cluster vii by bounding box and area criteria; and (F) watershed segmented image for comparison. Note: Yellow arrows indicate under-segmentation and red arrows indicate over-segmentation.

The segmentation performance of most of the algorithms (Park, Huang, Ji, & Ding, 2013; Zafari, Eerola, Sampo, Kälviäinen, & Haario, 2015) depends on the shape of the objects in the binary image; therefore, a good thresholding method that retains the object's shape is necessary. The binary image produced by YUV thresholding (fig. 5.8B) retained the shape of each plant considerably, which was suitable for obtaining the sideways profile. The raw profile produced more undulations (surface roughness), which was minimized by smoothing operation (fig. 5.8C) making it appropriate for profile analysis. It is interesting to note that the single plants (i, ii, and iii; fig. 5.8B), forms peaks and valleys in the raw profile, but after smoothing, the peaks and valleys got suppressed. While the other clusters form prominent valleys even after smoothing.

5.4.4.1. Area ratio cutoff value selection

Quantifying the valley area (A_V; eq. 5.8) in the smoothed profile with respect to the area under peaks profile (A_P; eq. 5.7), the potential segmentation location was determined. The AR values (eq. 5.9) were obtained from a few sample cluster images. To evaluate, whether the AR values qualify for segmentation, the range of AR values produced from the profile was visually examined with its respective cluster image to arrive at an AR_{cut} value. It was observed that the clusters with AR values >0.04 potentially qualified for segmentation; hence, the AR_{cut} was fixed as 0.04 in this study. It should be noted that the AR_{cut} value will be influenced by the smoothness of the profile. For a bandwidth value of 10 (smoothness interpolator parameter fixed in this study; section 5.3.9.1), the AR_{cut} worked well and it is expected to work with any other images with the same bandwidth value. The segmented image was obtained by drawing a one-pixel thick black line at the qualified valley locations on the plant clusters (fig. 5.8D).

5.4.4.2. Profile-based and watershed segmentation comparison

As mentioned earlier, when the overlap between the plants was more, the sideways profile may not always produce a clear valley for segmentation. This can be noted in the cluster marked "vii," where the central portion consisted of two plants that were very close together (doubles) and resulted in under-segmentation, due to the lack of valley between those two plants, based on the profile-based approach (yellow arrow; fig. 5.8D).

The second level of segmentation (geometry-based) tackled such scenarios and segmented the clusters that did not have a clear profile (fig. 5.8E). This second level of segmentation simply cuts the cluster based on BBH or A of the cluster (section 5.3.10), it is beneficial in resolving the clusters unresolved by the profile-based approach. Even though the simple geometry-based method (BBH and A) produces the results in stand count applications the logic of segmentation will not be as rigorous as the profile-based approach and may not be useful in any shape analysis applications as the segmented sections may not belong to the same plant.

On basic comparison with watershed segmentation, it was found that it resulted in over-segmentation on few clusters (red arrows; fig. 5.8F). In total, for the 22 identifiable plants (fig. 5.8A), the watershed segmentation produced 25 segments that will be labeled as plants in the analysis; however, the combined approach (fig. 5.8E) performed better and identified 21 plants. This shows that simply applying watershed will produce over-segmentation, and without other processing will not be helpful in plant stand count applications.

5.4.4.3. Geometry-based segmentation

The geometry-based segmentation of plant clusters that are not resolved by the profile-based method is illustrated (fig. 5.9) to further demonstrate the different conditions of segmentation involved (eq. 5.10). When a cluster was big (area) and long (BBH) it will be qualified in the criteria of both and the cluster will be divided into N_{plants} number of segments based on the area (fig. 5.9-top row). When the clusters were small and elongated, representing late emergence, which will not be identified as a cluster based on the area but will be on BBH and segmented into N_{plants} based on BBH (fig. 5.9-middle row).
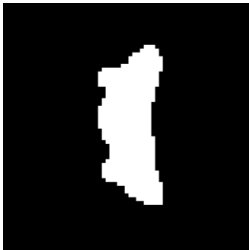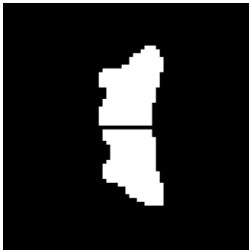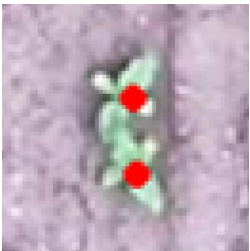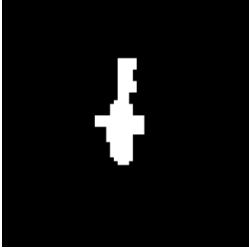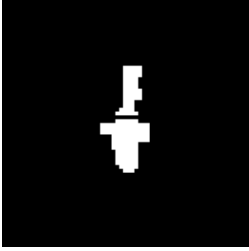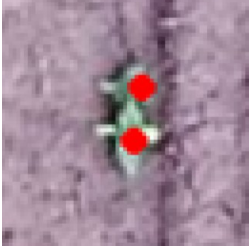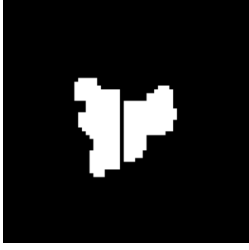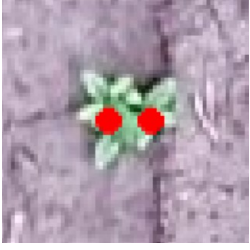
Criteria	RGB cluster	Binary image	Segmented image	Labeled image
Bounding box and area				
Only bounding box				
Only area				

Figure 5.9. Detailed illustration of geometry-based segmentation for unsegmented clusters from profile-based segmentation. Note: The row direction is from north to south.

Sometimes, when two or more seeds were placed sideways perpendicular to the row direction and the plants had normal growth, the plants based on the area will be identified as a cluster but will not be based on BBH (fig. 5.9-bottom row). Such clusters were segmented into N_{plants} based on the area but divided vertically, and this is unique to the geometry-based segmentation.

Practically, the occurrence of clusters in a field directly evaluated, which were not resolved by the profile-based method, by the first condition (BBH & A; eq. 5.10) will be more followed by the third condition (only A) and the second (only BBH). The clusters qualified based on the second condition (only BBH) were predominantly due to late emerged plants that lost its boundary pixels during thresholding and morphological operations.

The proportions of segmentation performed in this study based on the combined profile and geometry were 52.1 % (1153 out of 2213) of the segmentation based on profile and the remaining 47.9 % based on geometry (table 5.1). Among the three conditions in geometry-based segmentation (eq. 5.10), 26.6 % was based on first condition (BBH and A), 6.6 % was based on second (only BBH) and 14.6 % was based on third condition (Only A). However, with the "Set II" images, the plants were well grown and they produced clear peak and valley in the sideways profile. Therefore, a majority of the segmentation was based on the profile (70.8 %) and the remaining (29.2 %) were based on geometry.

5.4.5. Plant Vigor Classification Results

Based on the classes defined in eq. 5.11, the individual plants were classified into the three classes (I, II, and III; table 5.2). As expected from the plant vigor data, in the Set I image, the majority of the plants were in class II (69.29±2.44 %) and an almost equal

Table 5.1. Number of identified clusters and plants after segmentation from each segmentation criteria for different section images from "Set I" and "Set II" images.

Image ID	Total	Total	Profile		BBH and area		Only BBH		Only area	
	N_{cls}	N_{plt}	N_{cls}	N_{plt}	N_{cls}	N_{plt}	N_{cls}	N_{plt}	N_{cls}	N_{plt}
Set I image (V2–V4)**										
1 (0-0).jpg	48	93	28	53	12	24	2	4	6	12
2 (2937-0).jpg	45	88	29	55	8	17	0	0	8	16
3 (5874-0).jpg	40	77	26	47	8	18	0	0	6	12
4 (8811-0).jpg	45	85	27	49	10	20	3	6	5	10
5 (11748-0).jpg	45	95	30	65	5	10	0	0	10	20
6 (0-1470).jpg	48	81	23	28	18	39	4	8	3	6
7 (2937-1470).jpg	54	95	26	39	16	32	7	14	5	10
8 (5874-1470).jpg	45	88	28	54	6	12	3	6	8	16
9 (8811-1470).jpg	57	97	30	42	17	35	5	10	5	10
10 (11748-1470).jpg	49	88	27	44	8	16	7	14	7	14
11 (0-2940).jpg	46	85	24	40	12	25	3	6	7	14
12 (2937-2940).jpg	42	74	25	40	7	14	3	6	7	14
13 (5874-2940).jpg	45	79	24	37	9	18	6	12	6	12
14 (8811-2940).jpg	47	83	27	41	16	34	1	2	3	6
15 (11748-2940).jpg	43	74	26	40	10	20	2	4	5	10
16 (0-4410).jpg	51	97	29	53	12	24	1	2	9	18
17 (2937-4410).jpg	46	79	27	41	14	28	1	2	4	8
18 (5874-4410).jpg	47	88	27	45	13	29	2	4	5	10
19 (8811-4410).jpg	55	103	27	43	18	40	3	6	7	14
20 (11748-4410).jpg	50	89	24	34	13	29	6	12	7	14
21 (0-5880).jpg	44	88	27	54	6	12	2	4	9	18
22 (2937-5880).jpg	50	87	27	40	12	25	4	8	7	14
23 (5874-5880).jpg	47	90	29	54	7	14	2	4	9	18
24 (8811-5880).jpg	47	96	30	62	9	18	3	6	5	10
25 (11748-5880).jpg	64	114	34	53	18	37	3	6	9	18
Sum	1200	2213	681	1153	284	590	73	146	162	324
Average	48	88.5	27.2	46.1	11.4	23.6	2.9	5.8	6.5	12.9
Set II image (V4–V6)**										
15.jpg	11	29	11	29	0	0	0	0	0	0
19.jpg	13	30	10	24	2	4	0	0	1	2
20.jpg	18	31	10	15	5	10	1	2	2	4
32.jpg	15	32	9	20	3	6	0	0	3	6
50.jpg	14	27	9	17	2	4	1	2	2	4
Sum	71	149	49	105	12	24	2	4	8	16
Average	14.2	29.8	9.8	21	2.4	4.8	0.4	0.8	1.6	3.2

Note: BBH - bounding box height, * Set I image represents sunflower at V2–V4 stage, UAS altitude = 12.2 m (≈40 ft), S_I = 0.22 m, ** Set II image represents sunflower at V4–V6 stage, UAS altitude = 23 m (≈75 ft), S_I = 0.27 m, N_{cls} - number of identified plant clusters, and N_{plt} - number of plants identified in the clusters.

percentage of plants were in class I (16.20±3.23 %) and class III (15.51±1.85 %). Similar grouping into classes was observed in Set II images as well, with 64.25 %, 18.90 %, and 18.05 % in class II, I, and III, respectively. It is evident that the plant vigor data closely followed the normal distribution with one interval of standard deviation (SD), as in a perfect normal distribution where the percentage of values in class II will be 68.26 %, while the other two classes will equally contain 15.87 % each.

The SD values of three vigor classes ranged between 1.85 and 3.23 % (table 5.2). This small variation indicated that the plant vigor among the image sections considered for validation was more or less uniform. The percentage of plants in class I represented under-performing plants, which might be caused due to late emergence or the artifacts that were produced during segmentation. Similarly, the plants in class III represented over-performing plants that might be due to rapid foliage growth.

To have an overall representation of the plant vigor classes across the field, a color-coded plant vigor map was produced as one of the graphical outputs from the plugin (fig. 5.10). The spots on the map indicated in red denote under-performing plants (class I), green denotes optimally grown plants (class II), and yellow denotes over-performing plants (class III). In an ideal condition, it is expected to have more green and yellow portions in the image, and less red portions.

Since we divided the plant vigor data based on μ_A and σ_A, the plugin always provided three classes irrespective of the normal growth. The classification at any stage can be correctly interpreted as under-, normal-, and over-performing plants. However, based on field research if the normal performing plant area at different growth stages is made

164

Table 5.2. Plant vigor results of individual image sections from "Set I" and "Set II" images of sunflower fields.

Image ID	Plant vigor class		
	Class I (%)	Class II (%)	Class III (%)
Set I image (V2–V4)*			
1 (0-0).jpg	18.41	66.43	17.33
2 (2937-0).jpg	17.60	70.79	15.36
3 (5874-0).jpg	16.80	69.92	15.23
4 (8811-0).jpg	18.93	68.31	15.64
5 (11748-0).jpg	19.48	71.91	15.73
6 (0-1470).jpg	14.68	71.43	16.27
7 (2937-1470).jpg	17.81	66.40	16.19
8 (5874-1470).jpg	15.04	69.17	16.54
9 (8811-1470).jpg	16.22	71.43	17.37
10 (11748-1470).jpg	16.87	70.68	16.87
11 (0-2940).jpg	17.67	75.35	13.02
12 (2937-2940).jpg	9.48	68.97	11.21
13 (5874-2940).jpg	17.33	67.56	16.44
14 (8811-2940).jpg	18.88	73.39	13.30
15 (11748-2940).jpg	9.17	68.12	12.66
16 (0-4410).jpg	16.54	68.38	16.54
17 (2937-4410).jpg	16.47	69.80	14.12
18 (5874-4410).jpg	15.47	70.19	15.09
19 (8811-4410).jpg	17.80	65.15	17.42
20 (11748-4410).jpg	16.79	68.70	16.79
21 (0-5880).jpg	16.85	70.04	14.61
22 (2937-5880).jpg	15.44	70.27	15.44
23 (5874-5880).jpg	21.13	64.15	18.11
24 (8811-5880).jpg	6.59	68.50	12.09
25 (11748-5880).jpg	17.43	67.11	18.42
Average	16.20	69.29	15.51
SD	3.23	2.44	1.85
Set II image (V4–V6)**			
15.jpg	18.27	66.35	16.35
19.jpg	19.19	64.65	19.19
20.jpg	17.71	70.83	12.50
32.jpg	21.15	55.77	24.04
50.jpg	18.18	63.64	18.18
Average	18.90	64.25	18.05
SD	1.22	4.90	3.77

Note: The plant vigor classes are classified based on plants' projected area using equation 5.11.
* Set I image represents sunflower at V2–V4 stage, UAS altitude = 12.2 m (≈40 ft), S_I = 0.22 m,
** Set II image represents sunflower at V4–V6 stage, UAS altitude = 23 m (≈75 ft), S_I = 0.27 m, and SD is the standard deviation.
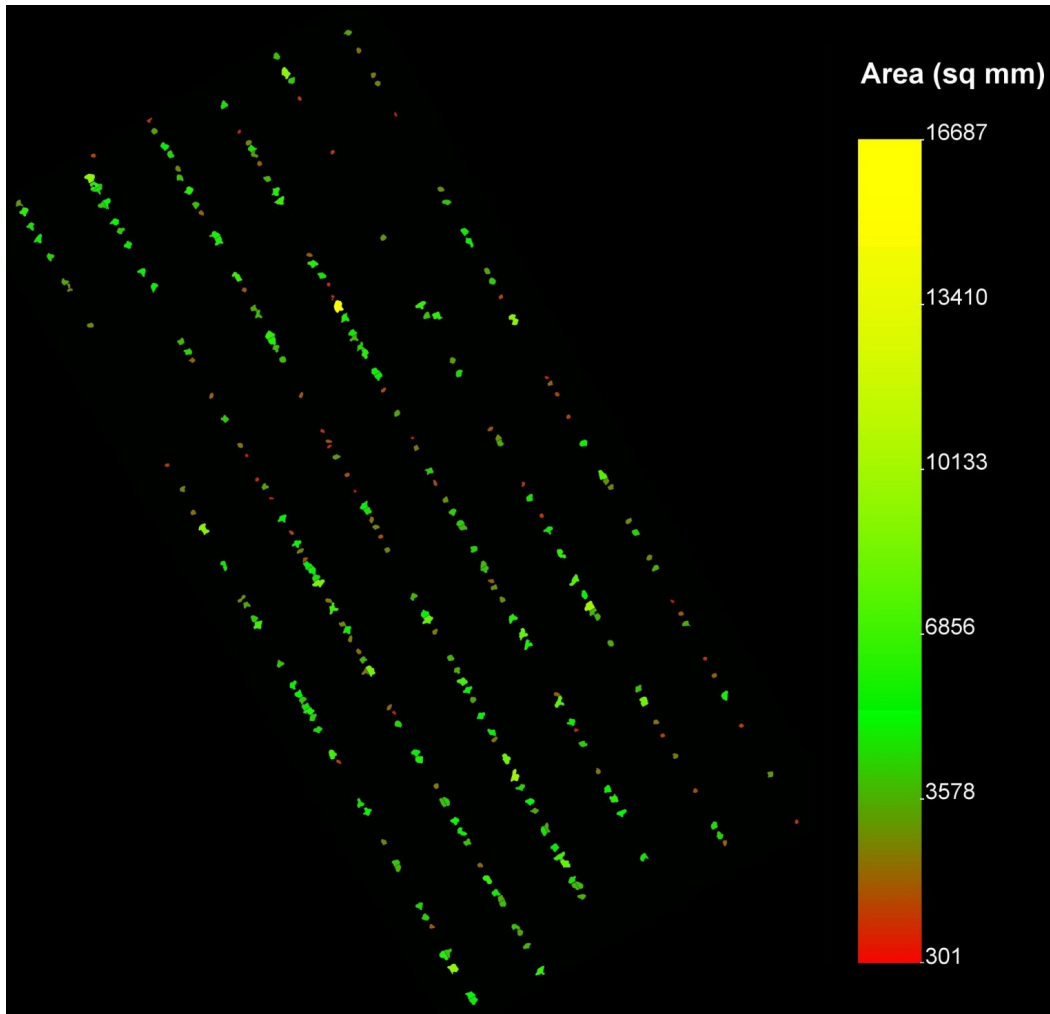
Figure 5.10. Color-coded plant vigor map classified into three classes: under-performing (red), optimally grown (green), and over-performing (yellow) in UAS image.

available then it is possible to make reference-based classification by coding the algorithm easily into this plugin.

5.4.6. Plant Stand Count Results — Row-wise and Overall

To assess the performance of the plugin, the plants were visually counted on the image by zooming in, which served as a reference, and compared with the plugin count. The plugin also had the potential of producing a row-wise stand count that was compared

with a row-wise visual count. The plant stand count graphical output from the plugin provides the labeled plants count numbered sequentially (top to bottom) in rows from left to right. (fig. 5.11). Table 5.3 presents the row-wise and overall plant stand count results from visual and plugin and the results indicated the promising performance of the plugin.

It can be observed that the average row-wise visual count (39.91±5.39) and row-wise plugin count (40.22±4.04) were very close and comparable (table 5.3). The theoretical population per row (N_{PR}), calculated based on L_{row} and S_I (0.22 m) was 43.60±0.8 (data not shown) plants per row. The evaluated plugin and visual count achieved ≥91.5 % of the N_{PR}.

From the results of an overall plant stand count for each image, the number of plants contained in each plot on an average was 254±25 by visual count and 256±19 by plugin count. Using the N_{PR} (43.60±0.8) and N_{rows} in each image, an ideal number of plants in the given area (N_I) was calculated as 279±23 plants. Comparing this with the plugin count, the seed emergence percentage can be approximately calculated as ≥91.8 %. Even though it is desirable to achieve the ideal population, it is practically impossible to obtain this as it depends on various field related factors (e.g., soil health, weather, seed quality).

The plugin counts were either under- and over-estimated compared to the overall visual count, which was evident from the error (%) observed (table 5.3). The average absolute error was only 1.29±4.66 %. The accuracy calculated for individual images were >91.19 % with an overall average accuracy of 95.85±2.48 % for Set I images. The low SD value of 2.48 %, which was 2.59 % only of the average value, indicated a good and uniform performance of plant stand count in all the section images.
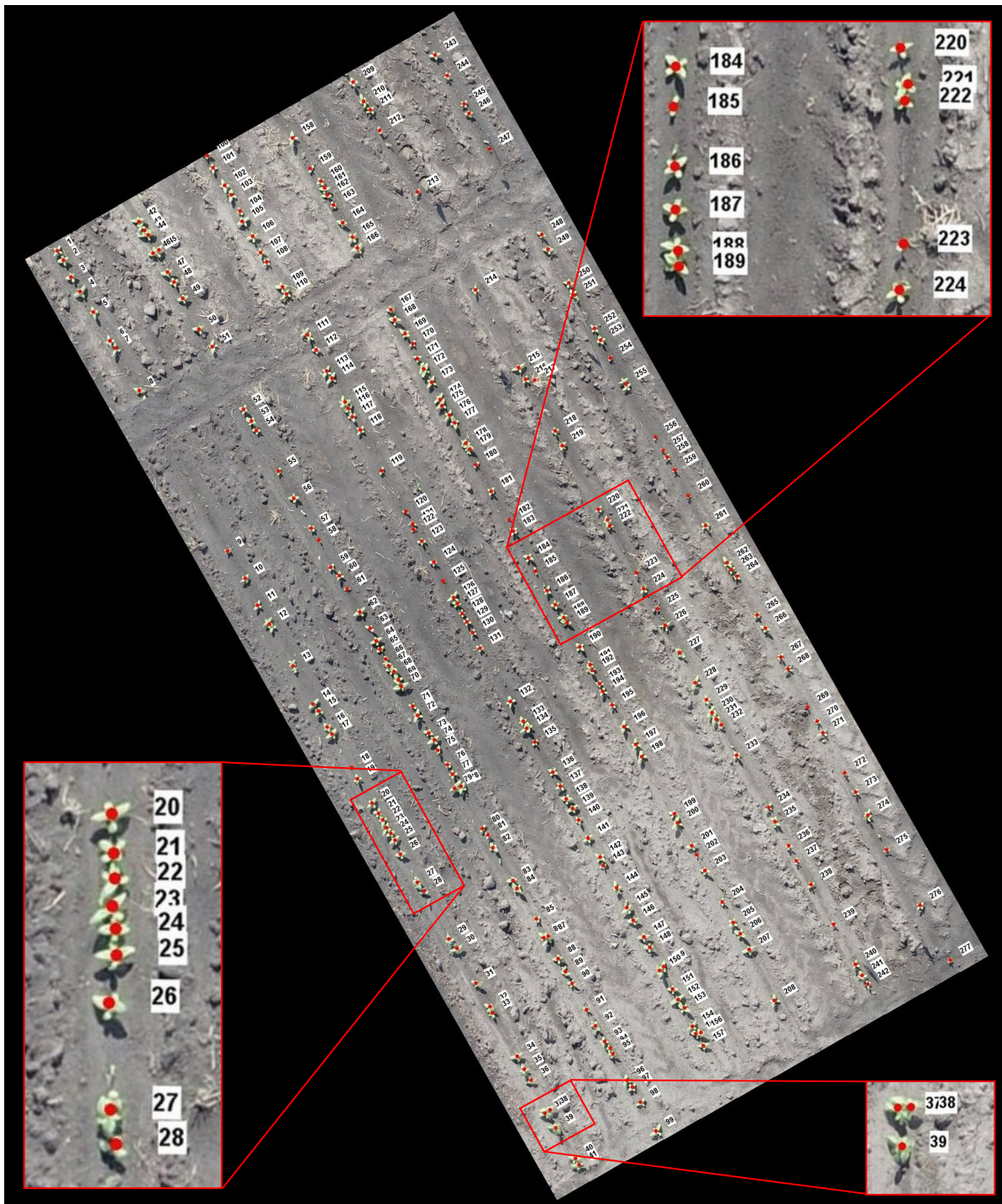
Figure 5.11. Plant stand count sequentially labeled along the rows in UAS image from top to bottom and from left to right (Insets: zoomed portions to show the plants, markers, and labels clearly).

Table 5.3. Plant stand count results of individual image sections from "Set I" and "Set II" images of sunflower fields.

Image ID	Plant stand count (row-wise and overall average)					
	Visual count mean ± SD (row-wise)	Plugin count mean±SD (row-wise)	Visual count (overall)	Plugin count (overall)	Error[a] (%)	Accuracy[a] (%)
Set I image (V2–V4)*						
1 (0-0).jpg	48.67±9.98	46.17±10.98	292	277	−5.14	94.86
2 (2937-0).jpg	46.00±9.64	44.50±10.13	276	267	−3.26	96.74
3 (5874-0).jpg	43.67±11.38	42.67±12.37	262	256	−2.29	97.71
4 (8811-0).jpg	41.17±9.94	40.50±11.38	247	243	−1.62	98.38
5 (11748-0).jpg	47.33±9.43	44.50±11.90	284	267	−5.99	94.01
6 (0-1470).jpg	34.00±2.73	36.00±3.61	238	252	5.88	94.12
7 (2937-1470).jpg	32.43±3.96	35.29±7.50	227	247	8.81	91.19
8 (5874-1470).jpg	36.71±3.33	38.00±3.74	257	266	3.50	96.50
9 (8811-1470).jpg	37.00±4.60	37.00±4.47	259	259	0.00	100.00
10 (11748-1470).jpg	32.71±3.88	35.57±6.27	229	249	8.73	91.27
11 (0-2940).jpg	34.67±4.50	35.83±7.14	208	215	3.37	96.63
12 (2937-2940).jpg	36.67±3.04	38.67±2.94	220	232	5.45	94.55
13 (5874-2940).jpg	35.83±4.6	37.50±5.82	215	225	4.65	95.35
14 (8811-2940).jpg	37.67±2.92	38.83±4.83	226	233	3.10	96.90
15 (11748-2940).jpg	38.00±4.2	38.17±4.88	228	229	0.44	99.56
16 (0-4410).jpg	39.00±3.66	38.86±4.95	273	272	−0.37	99.63
17 (2937-4410).jpg	35.43±4.56	36.43±5.32	248	255	2.82	97.18
18 (5874-4410).jpg	38.29±2.12	37.86±3.29	268	265	−1.12	98.88
19 (8811-4410).jpg	34.71±5.15	37.71±8.81	243	264	8.64	91.36
20 (11748-4410).jpg	35.29±4.71	37.43±8.20	247	262	6.07	93.93
21 (0-5880).jpg	47.33±10.98	44.50±11.04	284	267	−5.99	94.01
22 (2937-5880).jpg	42.00±8.98	43.17±10.98	252	259	2.78	97.22
23 (5874-5880).jpg	47.00±12.45	44.17±14.62	282	265	−6.03	93.97
24 (8811-5880).jpg	47.33±13.71	45.50±15.41	284	273	−3.87	96.13
25 (11748-5880).jpg	48.83±7.40	50.67±11.54	293	304	3.75	96.25
Average	39.91±5.39	40.22±4.04	253.68	256.12	1.29[b]	95.85
SD	6.47±3.44	8.08±3.64	25.07	18.79	2.48[b]	2.48
Set II image (V4–V6)**						
15.jpg	25.50±3.35	26.00±5.48	119	104	−12.61	87.39
19.jpg	25.75±2.68	24.75±4.57	122	99	−18.85	81.15
20.jpg	24.50±2.69	24.00±2.45	105	96	−8.57	91.43
32.jpg	25.75±5.40	26.00±9.63	125	104	−16.80	83.20
50.jpg	26.75±0.83	22.00±2.00	108	88	−18.52	81.48
Average	24.79±2.03	23.58±2.55	115.80	98.20	15.07[b]	84.93
SD	3.47±1.71	5.07±2.36	7.88	5.95	3.94[b]	3.94

Note: * Set I image represents sunflower at V2–V4 stage, UAS altitude = 12.2 m (≈40 ft), $S_I = 0.22$ m, ** Set II image represents sunflower at V4–V6 stage, UAS altitude = 23 m (≈75 ft), $S_I = 0.27$ m, SD is the standard deviation, [a] Error and accuracy are calculated based on plugin and visual count using equations 5.20 and 5.21, and [b] are the average and SD values based on absolute error values.

Unlike the Set I images, the accuracy of the plugin on Set II images was 84.93±3.94 %, which was less but comparable with the reported studies (Jin et al., 2017; Sankaran et al., 2017). The possible reasons for lower accuracy than Set I images may be due to (i) low resolution (≈0.006 35 m/pixel; almost twice of Set I), (ii) presence of more clusters leading to inappropriate selection of BBH_{cut} and A_{cut} value, (iii) canopy closure due to later growth stage (V4–V6), and (iv) inappropriate AR_{cut} value for this set of images. Slightly adjusting the above-mentioned factors will potentially improve the plugin accuracy for Set II images as well. Also, it is expected that the plugin will perform well in a similar way (Set I) even for different images with different crops, provided the factors are adjusted appropriately.

These results showed the potential of the developed plugin in producing an accurate plant stand count. The possible reason for such high accuracy was because of an efficient segmentation approach, which probably reduced the over- and under-segmentation issues (fig. 5.8E). It is interesting to point out that an open source user-coded ImageJ plugin developed from scratch using a simplified geometrical approach can achieve an accuracy ≥95 % is encouraging. This proved the capability and flexibility of ImageJ (open source platform) for developing image processing applications for precision agriculture.

5.4.7. Plant Spatial Distribution and Uniformity Results

The plugin produced a color-coded spatial distribution map (fig. 5.11) with the five classes defined by Kachman and Smith (1995). This map provided a visual clue on the planter's performance on seed placement and spacing. It is commonly understood that the multiples are solely due to issues in the planter's metering mechanism as well as the size of the seed that allowed more seeds to be packed in a single cell. However, the skips may be

due to either planter issue of unfilled cells or the seeds' viability. The map can also be used to infer the possible blockage issues in any specific seed tube.

In general, any row with more number of green strips is considered good, while blue and red are considered inefficient. For example, row 3 (fig. 5.12A) produced the maximum ideal plants (32, other rows <30), row 2 produced more multiples (21, other rows <16), and row 5 produced maximum skips (12, other rows <9). In this test image (fig. 5.12A), the spatial distribution based on the numbers are: multiple 30.1 %, ideal 55.1 %, single skip 10.7 %, double skip 2.6 %, and triple skip 1.5 %. Since producers are interested in having the ideal number of seeds on the ground, the skips (single, double, and multiple) are of an increased concern than the multiples. An overall reduction of the ideal spacings in the field can be seen as an indicator to perform corrective measures such as checking the seed tubes or the mechanism as well as in extreme cases recalibration of the planter.

The plugin also produced the plant spacing histogram as one of the graphical outputs that indicated the number of plants in each spacing (fig. 5.12B). Ideally, it is expected to have all the plants falling in the ideal spacing - S_I; however, attaining that is practically challenging. Both graphical outputs were straightforward and equally informative, as the figure 5.12A indicated 'where' the spacing was defective and figure 5.12B provided an indication of 'which' spacing needed adjustment. The spacing histogram demonstrated that only around 10 plants were placed exactly at S_I (0.22 m) and more plants were placed at 0.12 m. With this insight, a farmer, if necessary, can calibrate the planter to ensure proper spacing in the next planting. Also, the number of multiples (**I**) were high, which has to be reduced, while number of skips (**III**, **IV**, and **V**) were less and quite acceptable (fig. 5.12B).
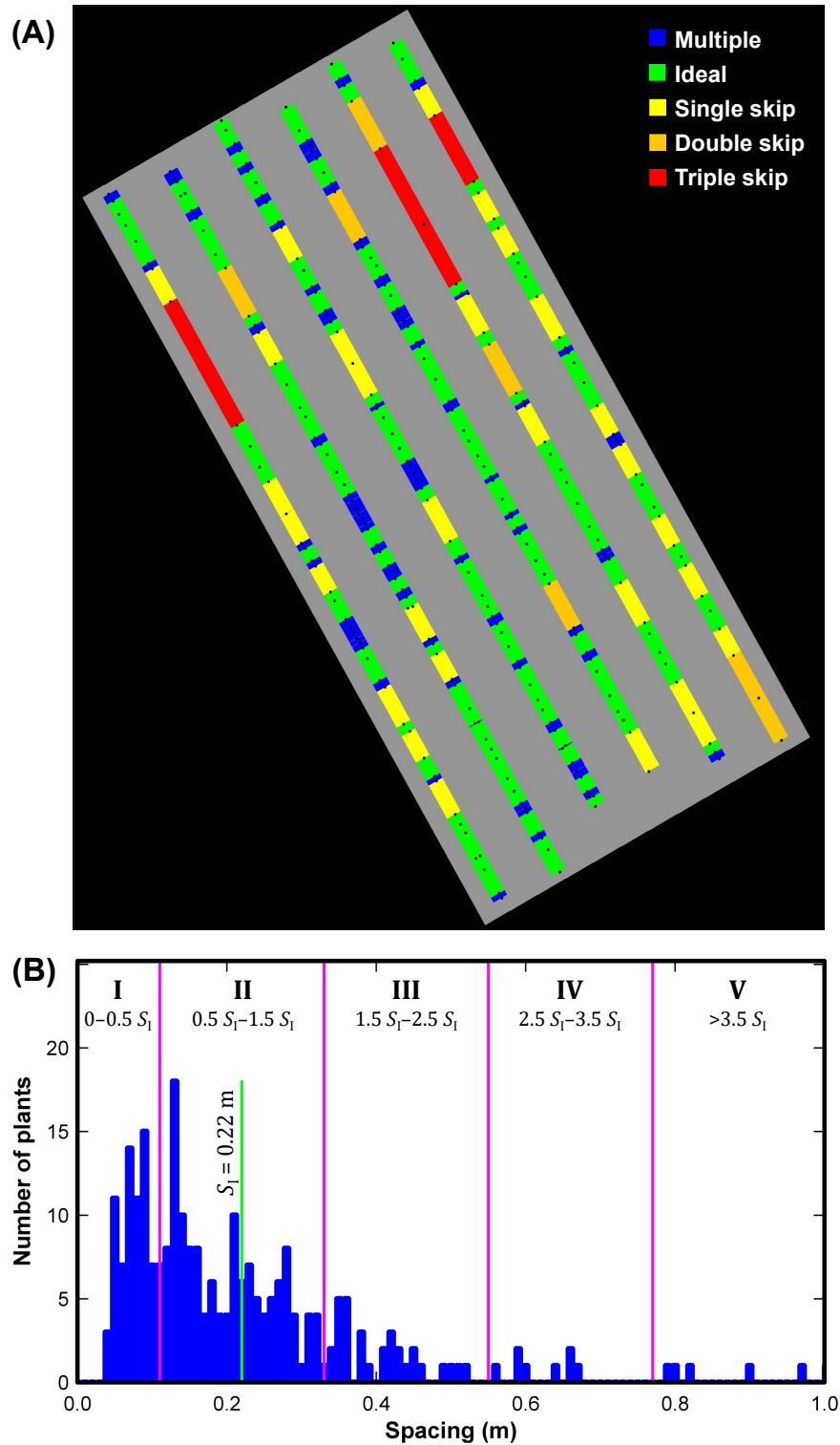
Figure 5.12. Plant spatial distribution analysis plugin outputs in UAS image: (A) Color-coded plant spatial distribution map, (B) Plant spacing distribution histogram. S_I is the ideal spacing.

The results of various spatial distribution uniformity indices (existing and developed) are presented in table 5.4. The overall ideal index values for the Set I image ranged between 0.43 and 0.63 with an average and SD of 0.52 and 0.05, respectively. Low SD value (9.6 % of the average) of the ideal index indicated that for any section of the field, the planter maintained uniform spacing. However, the skip index (0.22±0.05) and multiple index (0.26±0.06) had high variation among the sections (≥23.1 % of average) and this is logical given the reduced values of these averages. The observed precision value was 0.27±0.02, which confirms to the reported practical precision value of 0.29 (Kachman & Smith, 1995). A plot of precision × 100 (to depict the variation) has shown only a little variation (fig. 5.13). This indicates that the performance of the planter used in this study was comparable with the other planters.
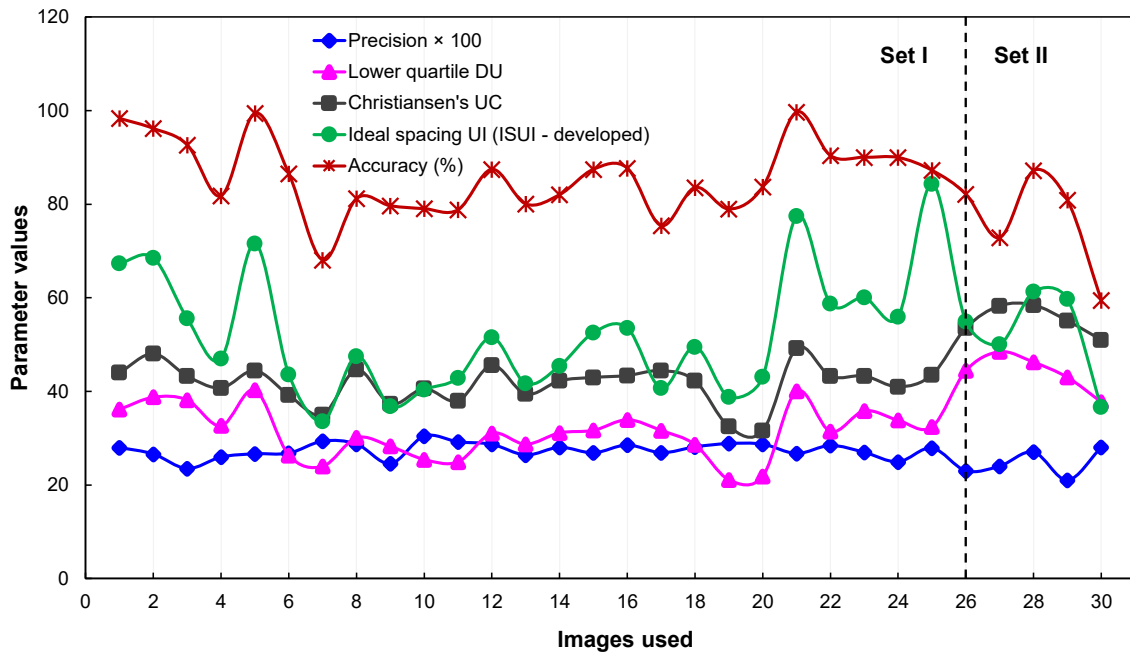
Figure 5.13. Plant spatial distribution parameters variation among sample images from Set I and Set II images of sunflower fields.

Table 5.4. Plant spatial distribution analysis results of individual image sections from "Set I" and "Set II" images.

Image ID	Ideal index	Skip index	Multiple index	Precision Precision	Lower quartile DU	Christiansen's UC	Ideal spacing UI	Row-wise plant spacing mean±SD (m)	Error[a] (%)	Accuracy[a] (%)
Set I image (V2–V4)*										
1 (0-0).jpg	0.54	0.15	0.31	0.28	36.16	43.98	67.37	0.216±0.054	−1.69	98.31
2 (2937-0).jpg	0.60	0.18	0.21	0.27	38.76	48.04	68.52	0.228±0.049	3.83	96.17
3 (5874-0).jpg	0.63	0.16	0.21	0.23	38.16	43.26	55.56	0.236±0.057	7.35	92.65
4 (8811-0).jpg	0.50	0.23	0.27	0.26	32.63	40.74	46.97	0.260±0.085	18.22	81.78
5 (11748-0).jpg	0.56	0.15	0.29	0.27	40.29	44.44	71.58	0.219±0.067	−0.57	99.43
6 (0-1470).jpg	0.49	0.24	0.27	0.27	26.33	39.19	43.63	0.249±0.016	13.48	86.52
7 (2937-1470).jpg	0.47	0.30	0.24	0.29	23.96	35.03	33.61	0.291±0.063	31.93	68.07
8 (5874-1470).jpg	0.50	0.27	0.22	0.29	30.07	44.64	47.46	0.261±0.031	18.81	81.19
9 (8811-1470).jpg	0.48	0.27	0.25	0.25	28.24	37.31	36.79	0.265±0.043	20.35	79.65
10 (11748-1470).jpg	0.48	0.25	0.27	0.30	25.41	40.59	40.37	0.266±0.060	20.97	79.03
11 (0-2940).jpg	0.46	0.26	0.27	0.29	24.91	37.97	42.86	0.267±0.049	21.20	78.80
12 (2937-2940).jpg	0.54	0.25	0.21	0.29	31.04	45.58	51.52	0.248±0.015	12.55	87.45
13 (5874-2940).jpg	0.47	0.28	0.24	0.26	28.69	39.42	41.70	0.264±0.033	19.96	80.04
14 (8811-2940).jpg	0.53	0.25	0.21	0.28	31.13	42.27	45.44	0.259±0.038	17.97	82.03
15 (11748-2940).jpg	0.51	0.25	0.24	0.27	31.66	42.97	52.54	0.248±0.032	12.54	87.46
16 (0-4410).jpg	0.51	0.25	0.24	0.29	33.92	43.39	53.54	0.247±0.036	12.31	87.69
17 (2937-4410).jpg	0.56	0.26	0.17	0.27	31.63	44.43	40.69	0.274±0.049	24.58	75.42
18 (5874-4410).jpg	0.48	0.29	0.23	0.28	28.61	42.26	49.47	0.256±0.031	16.44	83.56
19 (8811-4410).jpg	0.43	0.26	0.31	0.29	21.06	32.46	38.76	0.266±0.060	21.04	78.96
20 (11748-4410).jpg	0.44	0.25	0.31	0.29	21.83	31.60	43.08	0.256±0.057	16.34	83.66
21 (0-5880).jpg	0.57	0.16	0.27	0.27	40.07	49.23	77.43	0.219±0.063	−0.32	99.68
22 (2937-5880).jpg	0.52	0.20	0.28	0.28	31.41	43.24	58.72	0.241±0.074	9.62	90.38
23 (5874-5880).jpg	0.55	0.17	0.28	0.27	35.84	43.30	60.07	0.242±0.093	9.99	90.01
24 (8811-5880).jpg	0.57	0.15	0.28	0.25	33.82	40.98	55.90	0.242±0.108	10.04	89.96
25 (11748-5880).jpg	0.53	0.12	0.34	0.28	32.41	43.53	84.35	0.192±0.042	−12.77	87.23
Average	0.52	0.22	0.26	0.27	31.12	41.59	52.32	0.249±0.021	14.19[b]	85.80
SD	0.05	0.05	0.06	0.02	2.27	3.17	12.91	0.052±0.022	7.62[b]	7.62
Set II image (V4–V6)**										
15.jpg	0.65	0.23	0.11	0.23	44.42	53.53	54.97	0.318±0.057	17.85	82.15
19.jpg	0.70	0.25	0.04	0.24	48.49	58.27	50.07	0.343±0.054	27.14	72.86
20.jpg	0.69	0.18	0.12	0.27	46.26	58.43	61.29	0.305±0.029	12.84	87.16
32.jpg	0.58	0.24	0.17	0.21	43.01	55.08	59.71	0.322±0.089	19.13	80.87
50.jpg	0.49	0.38	0.12	0.28	37.74	51.00	36.61	0.379±0.047	40.56	59.44
Average	0.62	0.25	0.11	0.24	43.98	55.26	52.53	0.334±0.026	23.51[b]	76.49
SD	0.08	0.06	0.04	0.02	3.62	2.84	8.87	0.055±0.019	9.68[b]	9.68

Note: Ideal, skip, multiple index and precision were calculated using equations 5.13 and 5.16; DU - distribution uniformity (eq. 5.17), UC - uniformity coefficient (eq. 5.18), UI - uniformity index (eq. 5.19), and SD - standard deviation. [a] Error and accuracy were calculated by comparing row-wise plant spacing with the ideal spacing ($S_I = 0.22$ m for "Set I" images, and $S_I = 0.27$ m for "Set II" images). [b] average and SD values were calculated based on absolute values, * Set I image represents sunflower at V2–V4 stage, UAS altitude = 12.2 m ($\approx$40 ft), $S_I = 0.22$ m, and ** Set II image represents sunflower at V4–V6 stage, UAS altitude = 23 m ($\approx$75 ft), $S_I = 0.27$ m.

The uniformity indices adapted from sprinkler irrigation systems (table 5.4), such as lower quartile DU and Christiansen's UC (eqs. 5.17 and 5.18) showed consistent results among the section images (31.12 and 41.59), which was corroborated by low SD values (2.27 and 3.17). The SD values of DU and UC were only 7.3 % and 7.6 % of their respective average values. It can be observed that both the trends of DU and UC were similar and DU was consistently lower than UC, which indicated that any one of these can be used. Based

on these observations, any of these uniformity indices, derived from the specialization of irrigation, can be extended to the plant stand spatial distribution analysis.

The ideal spacing uniformity index - ISUI, developed in this study, showed a wide range between 33.61 % and 84.35 % unlike DU or UC (fig. 5.13). The wide range was obvious because it is sensitive to the deviations from the ideal spacing ($S_I = 0.22$ m) and penalizes these deviations, unlike DU and UC. The ISUI produced increased values only when the spacings are uniform and coincided with the ideal spacing; however, the DU and UC produced increased values with uniform spacing even though it deviated from the ideal spacing. Thus, an increased value of DU and UC may not represent a desirable plant spacing distribution, but an increased ISUI is always desirable. Both ISUI and UC had somewhat similar variation but high values of UC may not always coincide with high values of ISUI, indicating uniformity but deviation from the ideal spacing. It is also interesting to observe that the ISUI followed the trend of accuracy, which is desirable because both were related to the ideal spacing (fig. 5.13). Therefore, the ISUI will serve as the best parameter to evaluate precision planters and plant spatial distribution in modern precision agriculture of the present and the future.

In addition, it should be noted that these spatial distribution parameter values may not solely indicate the planter's performance, as the evaluation was based on only the emerged plants. However, it is interesting to note that the range of values was well within the range of coefficient of precision (CP3) values between 17.1 and 83.2 reported (Panning, Kocher, Smith, & Kachman, 2000), which is also an indicator of seed spacing uniformity. This demonstrates that the developed indices, especially the developed ISUI,

can be potentially employed in future studies of UAS-based plant spatial distribution analysis, which are quick, efficient, and meaningful.

Row-wise plant spacing for all section images is expected to be close to the S_I (0.22 m). The row-wise average spacing values among the section images ranged between 0.192–0.291 m. The mean value was majorly affected because of the presence of skips. As the skips are more, the number of plants will be less, which in turn increased the mean value. Any slight deviation from the S_I can be observed in the error value. For example, section 7 produced the maximum error (31.93 %), whose skip index was the maximum (0.30) among different section images. However, in the case of maximum multiples, there was no considerable effect on the plant spacing error. The average accuracy of row-wise plant spacing was about 85.80±7.62 %, which was fairly good and reasonable in a practical plant emergence scenario.

5.4.8. Plugin Run Performance and Future Scope

The plugin was developed and analyzed in a Windows laptop (Windows 10, 2.20 GHz Intel Core i5-5200U processor, and 8 GB RAM). The plugin took a CPU time of 31.73±1.23 s for a field area of 46.03±3.07 m^2 for Set I images and 4.17±0.56 s for a field area of 23.76±1.25 m^2 for Set II images. Even though the field area in Set II images was only half of Set I images, the analysis time reduced by ≈86.9 %. This was due to the lower image resolution of Set II images (0.006 35 m/pixel) compared to Set I images (0.003 31 m/pixel). This indicates that the CPU time is expected to increase with the resolution for the same field area as Set I images. The plugin was also tested for a major portion (20 of 25 representative sections) of the Set I stitched image for a field area of

995 m^2 and took a CPU time of 916.28 ± 12.06 s ($\approx$15 min) for performing all the operations and producing the labeled image (figure C1).

The automatic row identification is a vital operation in the developed plant stand count plugin. Currently, the plugin algorithm addresses straight row planted crops, which is commonly practiced in modern agriculture. However curvilinear planting is also followed in limited situations of irregular and curved boundary fields. Therefore, in the future, the plugin can be improved to tackle situations with irregular boundaries and curvilinearly planted rows.

With further programming, the planting pattern can be determined automatically based on profile and used in the program instead of obtaining it as user input. Furthermore, using plant spacing information, the plant spatial distribution (row-wise and overall) can be determined from which the planter performance on seed placement and spacing can be evaluated.

As the plugin was developed based on the profile and geometrical principles of the objects in the image, the plugin is expected to perform equally well with other row crops (e.g., soybeans, corn, cotton) between V2 and V6 stages. Unlike the black-box approaches generally followed and reported, the open source user-coded plugin approach does not require any image set for training and can be applied for other crops by suitably modifying the cutoff value and constants.

5.5. Conclusions

This study developed and demonstrated the potential of a user-coded open source ImageJ plugin, starting from scratch using a simple geometrical approach that addressed UAS image-based analysis research needs, such as row orientation correction, row

177

identification, plant cluster segmentation, and plant stand counting. For thresholding, a new method of using the YUV color channel by adjusting only the U channel was found to produce the binary images used in further analysis. Unlike the reported studies, a simple method developed using profile analysis to identify the crop rows orientation and align the rows vertically was found efficient. The developed row identification method using pixel profile of vertical rows accurately identified the rows up to a row length of $\approx$100 m. It is expected to work accurately even >100 m, as long as the rows are straight, which is possible with modern planters used in precision agriculture.

A novel segmentation approach was developed to resolve the plant clusters, which used the plant cluster's sideways profile to identify the overlapping portion for the segmentation followed by simple geometry applications using bounding box height and area. This combination approach of strategies resolved all clusters of sunflower plants in V2 to V6 stages successfully. The AR_{cut} value of 0.04 worked well for the tested images. The AR_{cut} value is influenced by the image resolution. For images with higher resolution, a higher AR_{cut} value should be used to capture the valleys better. The proportions of segmentation performed based on the combined profile and geometry were 52.1 % and 47.9 %, respectively for V2–V4 growth stages, and 70.8 % and 29.2 % for V4–V6 stage.

Test images from the two set of UAS stitched images from sunflower fields (Set I and Set II) from different field locations (Carrington and Prosper, ND), UAS flight altitudes (12.2 m and 23 m), and growth stages (V2–V4 and V4–V6) were used for the plugin validation. The plugin performed the stand count with an overall accuracy of 95.9 % for Set I images and 84.9 % for Set II images. For the spatial distribution analysis, a new index termed ideal spacing uniformity index (ISUI), which utilized the ideal seed spacing, was

developed and analyzed along with six different existing measures and found to be the best to evaluate plant spatial distribution. The plant spacing distribution was compared with ideal spacing and achieved an accuracy of 85.8 % and 76.49 % for Set I and Set II images, respectively. The graphical outputs such as the spatial distribution map and the histogram helped in visualizing the "location" and "count" of ideal, skip, and multiple, respectively. The overall analysis took a CPU time of 31.7 s for a field area of 46 m^2 for Set I images and 4.2 s for a field area of 23.76 m^2 for Set II images.

This study results proved the potential of an open source user-coded ImageJ plugin could match the capability of commercial professional software and extend their capability in terms of applicability, flexibility, and accuracy. The plugin can be suitably modified to handle other crops at different growth stages and attain good accuracy as observed in this study (>95 %). Most importantly, being an "open source" application, the processing software (ImageJ), the developed image processing application (ImageJ plugin), and the image data can reside in the user's system where the analysis can be performed in-house or even offline, at any time, thus ensuring "data security," apart from free and easily installed and run in any system.

6. PHENOCAM APPLICATION IN AGRICULTURAL CROPPING ENVIRONMENT: TIME AND DISTANCE EFFECT ON COLOR VEGETATION INDICES OF SOYBEAN PHENOLOGY *

6.1. Abstract

A PhenoCam is a near-surface remote sensing system traditionally used for monitoring phenological changes in forested landscapes. But, limited studies are applying PhenoCam images to crop phenology in agricultural ecosystems and clear guidelines on phenological analysis of the images are also lacking. Therefore, a study was conducted from two PhenoCam fields near Mandan, ND, USA on vegetation phenology of soybean (*Glycine max* L.) during the 2018 growing season to formulate image analysis guidelines. The phenological stages of soybean were also recorded through a manual visual assessment weekly to compare with PhenoCam images. PhenoCam images captured between 10:00 h and 14:00 h at 30 min intervals were used in this analysis. A user-coded ImageJ plugin was developed to extract the color values from a specified region of interest (ROI) from these images. A total of 15 color vegetation indices (CVIs), 13 existing and two new, were

---

considered that best represented the observed phenological trend from RGB, HSB′, and Lab color spaces. Of the CVIs analyzed, green leaf index (GLI), green chromatic coordinates (GCC), excess green (EXG), and a color index of vegetation (CIVE) were selected based on the reduced deviation from their loess smoothed phenological curve. The time of image capture produced at least two groups across the time (10:00–14:00). The camera distance also had significant differences among distance on some of the selected CVIs but not on all. Furthermore, a simplified, straightforward, and no-code method of analyzing the phenological trend using the "kymograph" was successfully developed. This simplified kymograph method showed promising results, characterized distinct phenology stages, and operated $\approx 73.3\%$ faster than the regular multiple image-based plugin method.

6.2. Introduction

A PhenoCam is a near-surface remote sensing tool configured to produce time-lapse images (e.g., $\approx 30\,\text{min}$) from which phenological transitions with respect to time were observed and recorded. Phenology is defined as the seasonal activity of plants and animals, which is a sensitive indicator of the impacts of climate change (T. B. Brown et al., 2016; Keenan et al., 2014). As of now, the phenological transition trends in plants, such as the timing of initial growth, peak greenness, or flowering and fruit production, have been instrumental in making management decisions (Browning, Karl, Morin, Richardson, & Tweedie, 2017). Such phenology transitions in plants are displayed in terms of changes in color, typically greenness. Color digital images from PhenoCams are used to obtain, from appropriate relationships, the greenness that indirectly indicates the amount of photosynthetically active green leaves and their growth conditions such as peak bloom and senescence (T. B. Brown et al., 2016).

181

Methods to monitor plant phenology can be classified broadly into two categories: (i) manual and (ii) instrument-based monitoring. The manual method involves direct human observation of only the major events such as flowering and budburst (Richardson, Bailey, Denny, Martin, & O'Keefe, 2006). However, in the manual method, the visual assessment is time-consuming, inaccurate, and can cover only a limited portion (sample) of the field. Instrument-based monitoring is otherwise called as "remote sensing," wherein, for example, a light-sensitive sensor is used to record the reflectance at different wavelengths that indicate the status of the plant. Satellite remote sensing is the most common remote sensing method. Although it covers a wide area, the lack of spatial and temporal resolution makes it less suitable for continuous monitoring of plant phenology (White et al., 2009; X. Zhang, Friedl, & Schaaf, 2006). Information about the classes of application of PhenoCam and PhenoCam network is presented in Appendix D.1.

Typically, the imagery obtained from PhenoCam comprises of three color channels – red (R), green (G), and blue (B), which are used to derive the various color vegetation indices (CVIs) and correlate them to different phenology stages. Research studies in several literature correlated CVIs to phenology transition dates in forest ecosystems (Richardson et al., 2007; Sonnentag et al., 2012). However, the application of PhenoCam to monitor phenology changes in an agricultural crop as well as the use of various CVIs in crop phenology monitoring, even though PhenoCams are available in many agricultural fields are not available. Therefore, it is desirable to use PhenoCam images from two agricultural fields with soybean (*Glycine max* L.) as the test crop and evaluate the various CVIs and correlate to phenological stages of the crop.

Digital cameras with a variety of configurations, end application, size, quality, and imaging sensors are being used in different test sites within the PhenoCam network (Sunoj, Igathinathane, & Hendrickson, 2016). Using the RGB pixel values from the images, CVIs are derived by mathematical expressions of pixel values from two or more color channels to highlight features of vegetation (Xue & Su, 2017). These CVIs provide quantitative and qualitative information of vegetation and are indicative of parameters such as plant cover, vigor, and crop stress. Color CVIs are the simplest form as they accentuate a particular color such as the greenness, which is intuitive to humans (G. Meyer & Neto, 2008; Woebbecke et al., 1995).

The PhenoCam network has widely used only green chromatic coordinates (GCC) and excess green (EXG) for monitoring canopy greenness. It was reported that these two CVIs produced fluctuations due to scene illuminations (Andresen et al., 2018). This was because the PhenoCam images are captured in open atmospheric conditions, the brightness and darkness caused by clouds and the time of the day result in such color variations. Lighting variation during image acquisition in agricultural fields is the most common issue in monitoring phenology and it is inevitable (Barbedo, 2016). Various color CVIs have been developed and used in various agricultural remote sensing applications (G. Meyer & Neto, 2008; Torres-Sánchez et al., 2014). However, the applicability of these CVIs in the PhenoCam network has not been evaluated so far. Therefore, in this study, the variations of CVIs from RGB color space and other spaces were evaluated and it was hypothesized that some CVIs (existing or developed) will have minimum variations on images captured on each day at finer time intervals ($\approx$30 min).

PhenoCam being a fixed location imaging device the subject's (plant) color might vary as a function of distance (Richardson et al., 2009). In the PhenoCam image, a region of interest (ROI) was usually identified in the field of view and the color information from the ROI was extracted for analysis. The image in the ROI represents objects at an increased distance from the bottom to the top. It is hypothesized that this variation in the distance is negligible and the ROI information can be used in the analysis irrespective of the pixel's location. However, this information was not reported and therefore deserved to be studied.

Another technical and common issue with PhenoCam data is the necessity to handle a very high volume of information in the form of images (about 16 useful images/day collected over ≈ 140 days). Even though this can be analyzed using a computer with decent speed and RAM, it will be desirable to simplify the image handling process and yet perform a representative analysis (Richardson et al., 2018b). In this research, a kymograph (a montage of thin ROI across time span) was tested for its suitability to simulate the phenological trend (Goldstein, 2011). As the kymograph is a single flat image representing the whole growing season, it will be easy to handle and subject to different analyses, and if successful would serve as a simplified method of PhenoCam analysis.

Given the background, the overall goal of this research was to use the PhenoCam images and evaluate the variations produced by different CVIs (from 10:00 to 14:00 each day), validate them against observed variation, and determine a set of best CVIs, which produced the least variation, suitable for PhenoCam application. This study used PhenoCam images from two research fields (Mandan-H5 and Mandan-I2) with soybean crop from the 2018 growing season. The recommended set of CVIs from this research are expected to serve as a better indicator and can be applied in future PhenoCam network

crop-based studies to obtain better phenological comparisons. Therefore, the specific

objectives formulated for this research are as follows:

1. Evaluate the variations of existing RGB-based CVIs and develop new other color
   spaces-based CVIs with reference to the observed phenological trend of soybean crop
   (30 min interval for ≈140 days; 2018 growing season).

2. Determine a set of best CVIs that produced minimum variation and validate them
   against the actual field phenological trend.

3. Evaluate the selected CVIs' variation with respect to the distance from the PhenoCam.

4. Develop a simplified approach to obtain phenological curve from kymograph using
   profile analysis and test its suitability for phenological studies.

6.3. Materials and Methods

6.3.1. Site and Management Description

Two experiment fields, Mandan-H5 (46.775° N, 100.951° W) and Mandan-I2

(46.761° N, 100.923° W), of the Area IV SCD Cooperative Research Farm, Mandan, North

Dakota, USA were selected for the study (fig. 6.1). The area of these fields were ≈19.8 ha

and ≈22.1 ha, respectively. Soybean (Mycogen seeds 5B024, The Dow Chemical Company,

Midland, MI) was planted using a John Deere 1890 planter at ≈543 631 seeds/ha

(220 000 seeds/ac) in both the fields on May 29 and 31, 2018, respectively. Row and plant

spacing were maintained as ≈0.19 m and ≈0.06 m. The soybean crops were harvested on

October 18 and 17, 2018, respectively.

6.3.2. PhenoCam and Images Collection for Phenological Curve

Both fields were installed with a NetCam color digital camera (NetCam SC IR,

StarDot Technologies, Buena Park, CA, USA). The digital cameras were placed inside a

Figure 6.1. Location of PhenoCam study experiment fields in Area IV SCD Cooperative Research Farm, Mandan, ND, USA, PhenoCam setup used in this study, phenology visual measurements in the field, and an overview of the two PhenoCam fields – Mandan-H5 ($\approx$19.8 ha) and Mandan-I2 ($\approx$22.1 ha). Ten yellow points in each field indicate the locations for visual assessment of phenology (ground truth) and the yellow triangle represents the approximate field of view area ($\approx$0.27 ha) for each PhenoCam. *Note*: Top panel aerial image was obtained from Google Maps for showing the two fields location and details.

186

weather-proof enclosure for continuous monitoring and were configured based on the instructions provided by the PhenoCam network (Richardson, Seyednasrollah, Hufkens, & Milliman, 2017b) and illustrated in figure 6.2. Some of the critical aspects on camera installation are: (1) the camera should be set to automatic exposure setting, (2) camera should be mounted on a rigid tower or building overlooking the vegetation canopy, (3) camera should be inclined downward at 20°–40° and partly cover some portion of the sky, and (4) camera should be pointed towards north to minimize lens flare and direct exposure to sun (Richardson, Klosterman, & Toomey, 2013b).
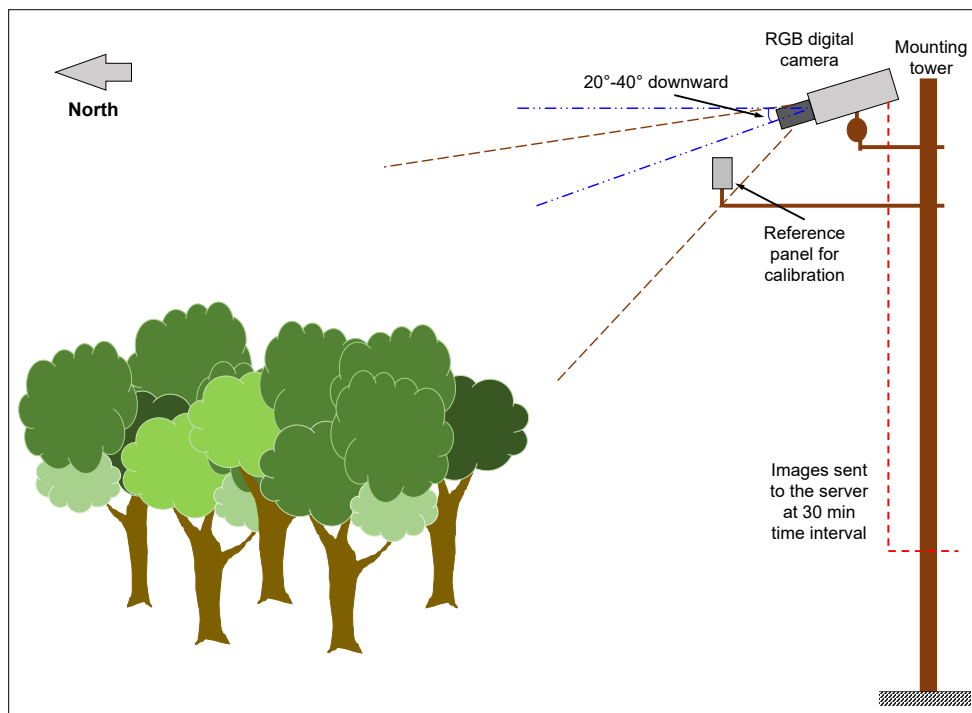
Figure 6.2. An illustration of the PhenoCam network system showing various components (Sunoj et al., 2016).

The cameras installed in PhenoCam network tower captured images automatically at time intervals of 30 min, transferred via file transfer protocol (FTP), and stored on the PhenoCam server. The images were downloaded from the PhenoCam website

(`https://phenocam.sr.unh.edu/webcam/gallery/`) for each field separately

("mandanh5" and "mandani2") by specifying the dates and time span.

For this study, the dates specified were the planting and harvest dates for Mandan-H5 (May 29–Oct. 18, 2018), and Mandan-I2 (May 31–Oct. 17, 2018), respectively. The total number of growing days between planting and harvest dates were 143 d for Mandan-H5, and 139 d for Mandan-I2. Within these days, a time span specified from 10:00 to 14:00 on each day was used for images collection, even though the PhenoCam captured the images from 8:00 to 20:30 continuously. This time span is commonly followed in PhenoCam studies (Sonnentag et al., 2012), however, the usable images with better lighting fall in a reduced span. For the study's time span (10:00 to 14:00), with the images captured at 30 min interval generated nine images per day. Therefore, the total number of images collected for the entire growing period in Mandan-H5 was 1287 images (143 × 9) and Mandan-I2 was 1251 images (139 × 9).

The detailed procedure to download the PhenoCam images for a specific field from the website, on required days and time span is provided in Appendix D.2. Along with PhenoCam images, a few data files (spreadsheet) that contain several parameters, such as daily mean RGB values, mean GCC, 5–95th percentile RGB values, 3-day mean RGB, etc. were also make available. These data are also accessible through the PhenoCam website and can be used for other calculations. In this study, for comparing among different CVIs, the daily mean RGB values from these data were used.

6.3.3. Visual Assessment of Phenological Stages

Visual assessment was performed weekly on each field to record the phenological stages (fig. 6.1). Ten random locations were identified in each field and marked with flags

for consistently recording phenological trends throughout the growing season. At each location, the phenological observation was performed and five samples were considered as a replication. Soybean phenology stages were identified according to D. R. Berglund, McWilliams, and Endres (2015). The phenology observation was started after the emergence of seedlings and all the locations in both fields were assessed on the same day. The vegetative and reproductive stages were indicated with the letters V and R, respectively. Some of the important phenology stages in soybean recorded include VC (cotyledon stage), V1 (first trifoliate), V2 (second trifoliate), V3 (third trifoliate), R1 (first flowering), R2 (full bloom), R3 and R4 (pod development), R5 and R6 (seed development), and R7 and R8 (plant maturation).

6.3.4. Image Color Value Extraction — ImageJ Plugin Development

To process the downloaded images, a user-coded plugin was developed using Fiji (Ver. ImageJ 1.52n, Java 1.6.0_24 [64-bit], Rasband (2018); Schindelin et al. (2012)). The plugin was coded in Java language and comprised of several methods for performing individual tasks. The process flowchart (fig. 6.3A) shows the sequence of tasks and the front panel of the developed plugin (fig. 6.3B) shows the options available for ROI specification and color value extraction from PhenoCam images and stores the results in a file for reading and further processing.

The images should be opened and ROI should be specified before running the plugin. To open a series of images in a folder as a convenient stack, the user should navigate to the folder and drag and drop the entire folder in the ImageJ panel. This action pops up an "Open Folder" (with Convert to RGB and Use Virtual Stack options — can be checked/unchecked as desired) dialog and clicking on the "Yes" button opens all the

A

Start

Before plugin execution

Open images as a stack by loading the images folder

Draw a polygonal ROI in any one image

During plugin execution

Calculate sub-ROI coordinates

For i = 1; i < n_images; i++

Set sub-ROIs and get RGB pixel values

Calculate RGB mean

Convert RGB mean to HSB′ and Lab values

Save RGB, HSB′, and Lab values to a CSV file

End

B

Phenocam images color values extraction plugin - ABEN, NDSU

Color values extraction from phenocam images

Note: Images should be open before running the plugin by droppping the images folder into ImageJ panel

Choose required color spaces:

☑ RGB values

☑ HSB values

☑ Lab values

Note: Color value outputs generated will be saved as a CSV file in the "Documents" folder.

An example phenocam image

Choose ROI options:

○ Single ROI

● Multiple ROI   Rows: 3   Columns: 3

Note: Make a polygonal ROI **ABCD** in the opened image by choosing the vertices and adjusting.

Developed by S. Sunoj and C. Igathinathane, Apr. 2019

NDSU | AGRICULTURAL AND BIOSYSTEMS ENGINEERING

Run   Cancel

Figure 6.3. Process flowchart and front panel of the color values extraction plugin for PhenoCam images: (A) Process flowchart outlining the sequence of tasks, and (B) front panel of the developed plugin showing the options available for color values extraction and ROI specification.

images of the folder as a stack. A polygonal ROI (usually a rectangular quadrilateral) on any of the images in the stack can be drawn using the "Polygonal selections" tool. Once drawn, this ROI was applied to all the images in the stack.

Then, running the plugin displayed the front panel of the plugin (fig. 6.3B). The polygonal ROI's vertices coordinates were applied to all the images and they define an area from which the basic color information is extracted and converted into different color spaces. This operation was possible because the PhenoCam has a fixed field of view, hence will not have any distortion/shift among images.

There are three color space options available in the plugin and the user should choose at least one color space (or all three) for the analysis. However, from the ROI options, the user can select only one option — either "Single ROI" or "Multiple ROI." With the multiple ROI option, the user can specify the required number of rows and columns for analysis (fig. 6.3B). The plugin internally divides the polygonal ROI and each sub-ROI was considered separately for color values extraction. By default, the plugin runs for 3 rows and 3 columns when the multiple ROI is chosen. The plugin execution will be started by clicking on the "Run" button.

The plugin automatically opened each image in the stack in sequence, overlaid the ROI on each image, and extracted the color values. All the pixel color values, primarily RGB, were collected and a mean RGB value was calculated. The mean values were used to convert into HSB′ color values (H: hue, S: saturation, and B′: brightness) by `Color.RGBtoHSB()` method and Lab (L: lightness, a: green–red, b: blue–yellow) by `ColorSpaceConverter.RGBtoLAB()` method in Java. The plugin was developed in such a way that the color values were extracted for all the images in the stack, in a single run, and

outputs generated as the comma separated value (CSV) file and stored in the general "Documents" folder. The plugin took a CPU time of $\approx$7.5 min to extract the color values from all images in the folder (1287 images) at nine ROIs, from three color spaces, and store the values in a CSV file. This automated process saved a lot of time in extracting the color values from all three color spaces, otherwise manually performing these tasks would consume more time and cumbersome.

6.3.5. Multiple ROIs used for Color Values Extraction

In this study, a multiple ROI with 3 rows and 3 columns was used for color values extraction. Care was exercised in specifying the polygonal ROI so that it matches with the polygonal ROI specified by the PhenoCam network for Mandan-H5 and Mandan-I2 (fig. 6.4). The ROI was split into nine sub-ROIs (3 × 3 grid) that served as replications in the analysis. To study the effect of time (10:00 to 14:00 at 30 min interval) on the CVIs, three replications were considered by averaging the sub-ROIs along each column (e.g., the average of sub-ROIs 1, 4, and 7 was considered one replication). And, to study the effect of distance from the camera on CVIs, the sub-ROIs within each row were considered as replications. The sub-ROIs 1, 2, and 3 served as replications for testing "far;" similarly, the 4, 5, and 6 for "middle;" and the 7, 8, and 9 for "near" (fig. 6.4).

6.3.6. Color Vegetation Indices Evaluation and Analysis using R

From the data obtained through the developed plugin, the CVIs calculation, statistical analysis, and data visualization codes were developed using the open source statistical analysis software R (R Core Team, 2018). An integrated development environment 'RStudio' (RStudio Team, 2018) was used to run R. The output CSV file, generated by the plugin, was directly imported to RStudio for CVI calculations.
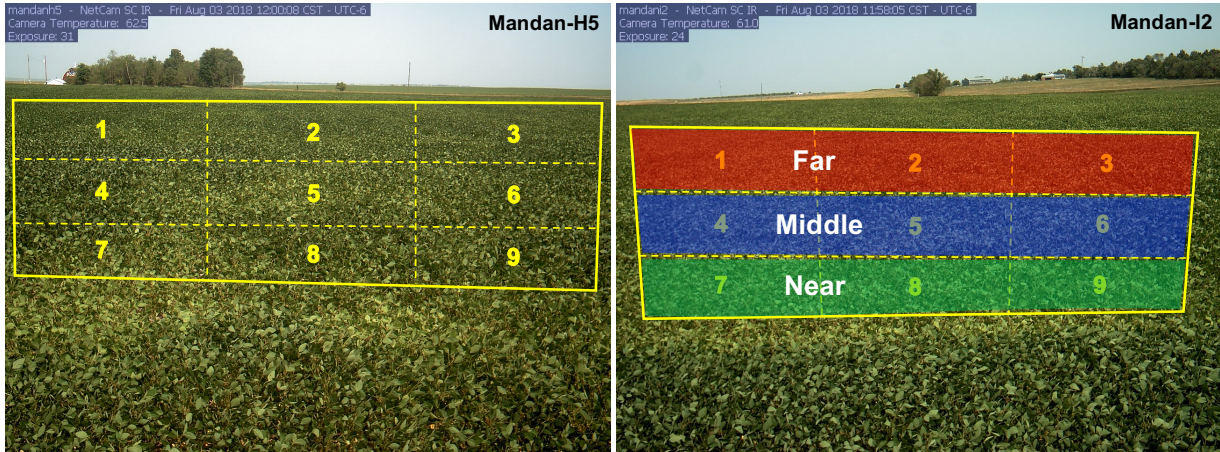
Figure 6.4. An example PhenoCam image of Mandan-H5 and Mandan-I2 soybean fields with ROI provided by the PhenoCam network (yellow solid line) and sub-ROIs (yellow dotted line) considered for analyzing the effect of time of image capture and distance from the camera on various CVIs.

6.3.7. Color Vegetation Indices from RGB Color Space

The CVIs are mathematical expressions of reflectance values from two or more color channels that are designed to highlight a particular property of vegetation (Xue & Su, 2017). The CVIs provide qualitative and quantitative evaluations of vegetation. Using RGB-based CVIs is easy for humans to perceive and interpret (G. Meyer & Neto, 2008; Woebbecke et al., 1995).

In this study, 12 CVIs from RGB color space, which are common in the agricultural application for estimating the vegetation fraction (Torres-Sánchez et al., 2014), identify plant biomass (G. Meyer & Neto, 2008), and weed monitoring (Hague et al., 2006), were considered that were calculated using the following equations:

$$GCC = \frac{G}{R + G + B} \qquad (6.1)$$

$$NGRDI = \frac{G - R}{G + R} \qquad (6.2)$$

$$\text{NGBDI} = \frac{\text{G} - \text{R}}{\text{G} + \text{B}} \tag{6.3}$$

$$\text{MGRVI} = \frac{\text{G}^2 - \text{R}^2}{\text{G}^2 + \text{R}^2} \tag{6.4}$$

$$\text{RGBVI} = \frac{\text{G}^2 - \text{BR}}{\text{G}^2 + \text{BR}} \tag{6.5}$$

$$\text{NDI} = 128 \times \left(\frac{\text{G} - \text{R}}{\text{G} + \text{R}} + 1 \right) \tag{6.6}$$

$$\text{GLI} = \frac{2\text{G} - \text{R} - \text{B}}{2\text{G} + \text{R} + \text{B}} \tag{6.7}$$

$$\text{RGRI} = \frac{\text{R}}{\text{G}} \tag{6.8}$$

$$\text{EXG} = 2g_c - r_c - b_c \tag{6.9}$$

$$\text{CIVE} = 0.441 r_c - 0.881 g_c + 0.385 b_c + 18.787\,45 \tag{6.10}$$

$$\text{VEG} = \frac{g_c}{r_c{}^a b_c{}^{(1-a)}} \tag{6.11}$$

$$\text{EXGR} = \text{EXG} - 1.4 r_c + g_c \tag{6.12}$$

where GCC is green chromatic coordinate, NGRDI is normalized green red difference index, NGBDI is normalized green blue difference index, MGRVI is modified green red vegetation index, RGBVI is red green blue vegetation index, NDI is normalized difference index, GLI is green leaf index, RGRI is red green ratio index, EXG is excess green, g_c is $\text{G}/(\text{R} + \text{G} + \text{B})$, r_c is $\text{R}/(\text{R} + \text{G} + \text{B})$, b_c is $\text{B}/(\text{R} + \text{G} + \text{B})$, CIVE is color index for vegetation, VEG is vegitativen, a is 0.667, and EXGR is excess green minus excess red.

6.3.8. Color Vegetation Indices from Other Color Spaces

Three CVIs from other color spaces, such as HSB′ (hue, saturation, and brightness), and Lab (Lightness, chroma1, chroma2), tested in the study are HSB's hue (HSB$_{\text{hue}}$), Lab's inverted a (Lab$_{a'}$), and dark green color index (DGCI) derived from HSB′. Out of the three

CVIs, dark green color index (DGCI) is a common index used in identifying turfgrass quality (Karcher & Richardson, 2003) and nitrogen status in cotton (Raper, Oosterhuis, Siddons, Purcell, & Mozaffari, 2012). The expressions for these CVIs are:

$$HSB_{hue} = H \tag{6.13}$$

$$Lab_{a'} = 1/a \tag{6.14}$$

$$DGCI = \frac{(H-60)/60 + (1-S) + (1-B')}{3} \tag{6.15}$$

Typically, the Lab "a" channel produced an inverted phenological curve; therefore, to follow the same trend of a typical phenological curve, the "a" channel value was inverted (as "1/a") in the $Lab_{a'}$ and used for comparison.

6.3.8.1. Color vegetation index normalization

For comparisons among different CVIs, while studying the time and distance effects, it is desirable to have the values on a common scale. Therefore, the CVI values were normalized to have a fixed scale between 0 and 1. This operation is common and being followed in some of the reported PhenoCam studies to compare among cameras and sites (Browning et al., 2017; Keenan et al., 2014). If the normalization is performed for each replication (derived from sub-ROIs) as a subset, the variation in the values among replications will be lost. Therefore, in this study, the maximum and minimum values were obtained including the replications and normalized for each CVI that preserves the variation trends among the replications. The maximum and minimum values were obtained using `min()` and `max()` functions in R and used to obtain the normalized values:

$$CVI_{norm} = \frac{CVI_i - CVI_{Gmin}}{CVI_{Gmax} - CVI_{Gmin}} \tag{6.16}$$

where CVI_{norm} in the normalized CVI value ranging between 0 and 1, CVI_i is the value at ith day, CVI_{Gmin} is the minimum CVI value among replications, and CVI_{Gmax} is the maximum CVI value among replications (appropriate sub-ROIs; fig. 6.4).

6.3.9. Color Vegetation Index Variation Comparison with Loess Smoothing

To evaluate the variations in a CVI curve, it is important to have a "reference curve" with which the variation can be compared. A smoothed curve, from the respective PhenoCam data, served as the reference curve in this study. The smoothed curve corresponding to the selected data was obtained using `loess` function (Cleveland, 1979) in R, with a 'span' value fixed as 0.3. This operation determined the smoothed curve parameters and the `predict()` function was used to obtain the values of the smoothed curve.
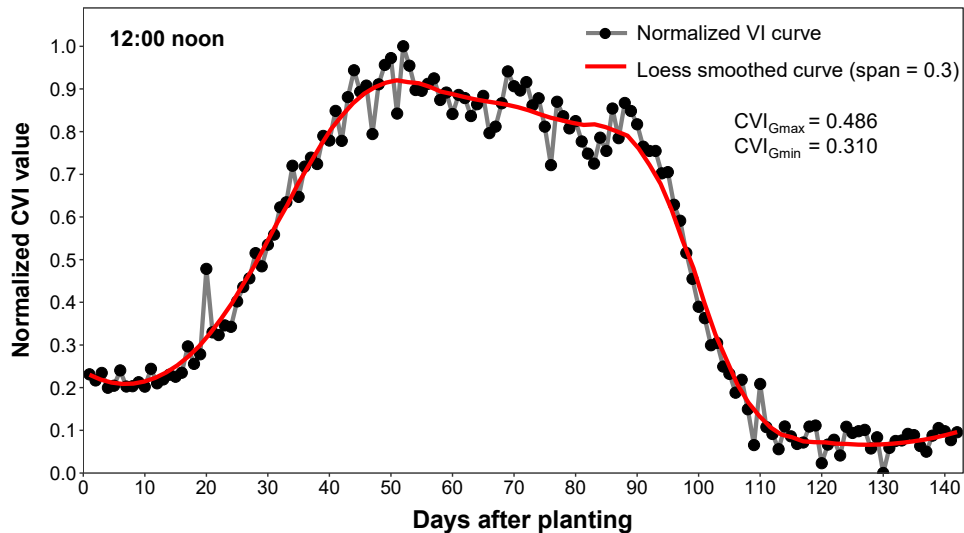
Figure 6.5. An example normalized GCC curve and its respective loess smoothed curve using a span of 0.3. Similar curves can be generated based on different times and selected ROIs and normalized based on equation 6.16.

It can be seen that the curve produced from normalized CVI values (raw points) is very rough (fig. 6.5) when compared to and this variation may be due to the changes in

lighting conditions and not because of changes in the plant canopy among subsequent images. While the smoothed curve (red) had gradual variations. To determine the roughness of the curve, we used the curve length as a measure. With curve length, a rough curve will be lengthier than a smooth curve. The curve length was calculated as:

$$L_{\text{curve}} = \sum_{i=1}^{n} \sqrt{1 + (\text{CVI}_{i+1} - \text{CVI}_i)^2} \qquad (6.17)$$

where L_{curve} is the curve length, i is the day after planting, n is the total number of study days, and CVI_{i+1} and CVI_i are the normalized CVI values at each time (e.g., 12:00 in fig. 6.5) on $i + 1$ and ith day.

Ideally, it is expected to have a gradually varying CVI curve if the effect of scene illumination is minimized. However, it is practically challenging to minimize variations in open atmospheric conditions. The deviation of length between normalized original CVI curve and the reference smooth curve provides a measure of variations caused by scene illuminations and other factors. Therefore, the deviation of curve length ($L_{\text{Deviation}}$) was defined as:

$$L_{\text{Deviation}}(\%) = \frac{(L_{\text{raw}} - L_{\text{smooth}})}{L_{\text{smooth}}} \times 100 \qquad (6.18)$$

where L_{raw} is the length of raw normalized CVI curve and L_{smooth} is the length of the smoothed curve (pixel).

6.3.10. Statistical Analysis and Visualizations

Statistical analysis was performed using R by employing Duncan's multiple range test using the R package `agricolae` (de Mendiburu, 2019). The $L_{\text{Deviation}}$ values (eq. 6.18)

produced by different CVIs for the time span between 10:00 and 14:00 were subjected to the analysis. Mean comparison among time intervals for different CVIs determined a set of best-performing CVIs that produced a minimum $L_{\text{Deviation}}$ among time. Similarly, mean comparison among just the CVI values determined the set of best-performing CVIs.

The effect of distance from the camera was statistically analyzed for three levels far, middle, and near. This analysis will imply whether an ROI can be overlaid at any location in the image. For visualizing the results, the R package ggplot2 (Wickham, 2016) was used and which generated the high-quality vector-based graphics with various customization opportunities.

6.3.11. A Simplified Phenological Curve Analysis With Kymograph Profile

A kymograph is a graphical representation of changes in a particular spatial position over time. It is especially useful in processing time-lapse images. Kymographs can be produced in ImageJ from the basic image stack commands. The time-lapse images should be first loaded in ImageJ and converted as a stack (section 6.3.4). On any image in the stack, a line ROI of any specified length can be drawn on the image as the reference source to build the kymograph. By choosing the inbuilt tool "Reslice [/]..." (Image ▷ Stacks) in ImageJ produced the kymograph. Also, a rectangular ROI will produce a stack of kymographs based on the pixel width of the ROI. The "Rotate 90 degrees" option is usually chosen to have the kymograph with time on the horizontal axis.

For the study, a horizontal line ROI was drawn in the middle along the image width. The resulting image was a composite image with the specified line ROI portion was extracted from each image and stacked side-by-side. The pixel width of this composite image indicates the number of images used for creating the kymograph. The kymograph

gives overall visual information about the changes in each image over time at the specified location. The generated kymograph was further processed with basic inbuilt tools of ImageJ to test whether the same phenological trend can be observed from the original images dataset and the kymograph.

For a simplified analysis, the kymograph was split into individuals channels of RGB, HSB, and Lab color space. In ImageJ, this can be obtained using "RGB Stack," "HSB Stack,", and "Lab Stack" options (Image ▷ Type). The individual channel was subjected to simple profile analysis (Analyze ▷ Plot Profile). This analysis generates a plot that represents the column-wise average of gray pixel values. This plot can be matched with the phenology stages from visual assessment and used for interpretations. If found suitable, this novel approach of using kymographs will considerably reduce the processing time and will be a beneficial approach to study the phenology without any coding.

6.4. Results and Discussion

6.4.1. A Typical GCC Phenological Curve and Comparison with Visual Assessment

The normalized GCC curve obtained from PhenoCam images of two fields, Mandan-H5 and -I2 followed a similar pattern (fig. 6.6). Even though the visual assessment was performed weekly, only the significant phenology stages manually recorded was presented. In both fields, the GCC started to increase rapidly after the VC stage, which is ideally considered as the start of the growing season (D. R. Berglund et al., 2015). The VC stage, in the field, was characterized by the expansion of leaves, and this was displayed in the GCC curve with an increase in the GCC values. It is expected that the soybean reaches the VC stage within 10–15 days after planting (DAP) and this was consistent with visual assessment and GCC curve from PhenoCam images.
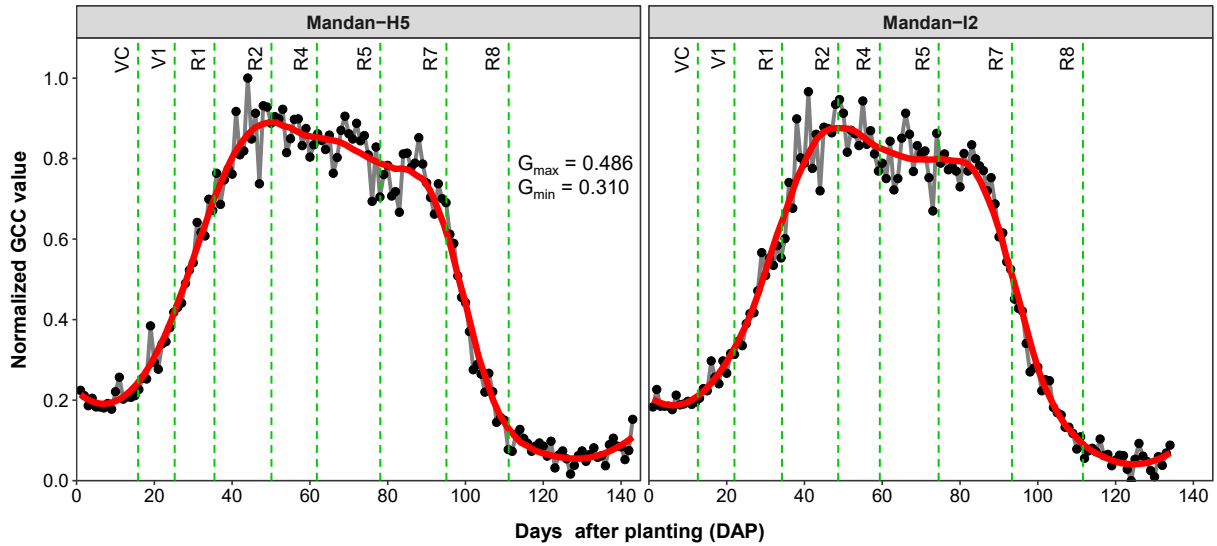
Figure 6.6. An example of a phenological curve from normalized GCC values. Dashed lines indicate different phenology stages observed during manual visual assessment. VC - cotyledon stage, V1 - first trifoliate, R1 - first flowering, R2 - full bloom, R4 - pod development, R5 - seed development, R7 - beginning maturity, and R8 - matured plants.

As the first vegetative stage (V1) was identified on ≈23 DAP, there was no prominent feature observed in the GCC curve except a consistent inclination in the GCC slope. The other vegetative stages (V2 and V3) were short and passed within a week, which was also not captured in the GCC curve. The GCC values further increased and were attaining the maximum as it entered into the reproductive stage (R1). This is the stage when the soybean starts flowering; however, it happens only in the nodes and cannot be seen at the canopy level (above) by the PhenoCam. To capture such subtle phenological stages, a close-up camera setup will be desirable, which is available in other PhenoCam experimental sites in Oregon (Phenocam, 2019).

As the soybean entered the R2 stage, which was identified as "full bloom," the GCC curve attained the maximum peak on 49th and 47th DAP on Mandan-H5 and -I2, respectively. This is a prominent phenological transition and was also characterized by

canopy closure in the PhenoCam images. After the R2, the GCC gradually decreased when the soybean "pod set" started. This decrease in GCC was because the flowers start to convert into pods and it starts from the upper four nodes. Therefore, the canopy greenness reduced which is indicated with a downward slope after the R2.

The gradual decrease continues up to 85–90 DAP, which might be the R6 stage called "full seed set." After the R6, the seeds start to mature (R7) and turns from green to yellow. This phenomenon is observed by a sudden decrease in the GCC value at ≈ 95 DAP. At this stage, the crop is safe from a majority of the stresses (D. R. Berglund et al., 2015). As the crop matures and enters into the R8 stage, it starts to shed leaves and transitions from yellow to brown. This is characterized by a further drop in GCC value to almost the minimum value, which indicates the end of the season. The crop is usually left in the field for 10–15 days after it reached R8 to reduce the moisture. At this stage where harvest and logistics plans were made.

These observations indicate that the GCC curve followed the actual phenology transitions in soybean as observed visually. With such interpretations, the GCC values can be used to obtain the phenology stages even without visiting the fields often. In the future, the PhenoCam images can be potentially used to identify stress that causes discolorations in the leaves, such as the iron deficiency chlorosis in soybean.

6.4.2. Effect of Time on General GCC Phenological Curve

The effect of time on GCC was important as it helps to understand whether the images captured at the same time on different DAP produce similar variation in GCC. Variations with respect to time are illustrated (fig. 6.7) with the GCC value obtained from the whole ROI of each image captured at different times (10:00–14:00; 9 curves).

Irrespective of the time of image capture, all GCC curves followed a similar trend. Even though images at 10:00 had slightly lower values and 12:00 had slightly higher GCC values, this difference overall is not apparent but should be subject to statistical analysis (see section 6.4.5).
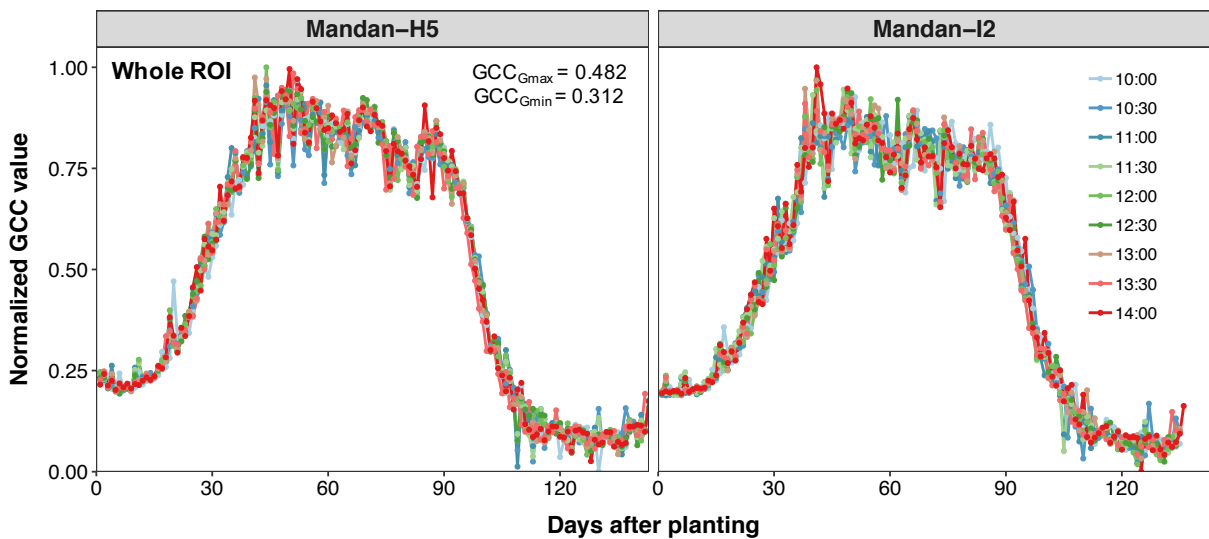
Figure 6.7. Effect of different time of image capture (10:00 to 14:00) on the general GCC trend from the whole ROI.

Until 45–50 DAP, all the curves were overlapped and the trend was steadily increasing among different times. More fluctuations were observed after attaining the reproductive stage (40–85 DAP). After 85 DAP, all the curves merged again and produced less variation among different times. The variations among DAP were random and it is entirely due to ambient conditions. Ideally, each curve is expected to be smooth with minimum fluctuations. But, those fluctuations were unavoidable, which are caused by image color differences due to clouds interference with incoming sun's radiation during image capture.

If desired to minimize these variations, a color calibration procedure with a color calibration chart should be performed on images among the days (Sunoj et al., 2018b). Furthermore, it is also desirable to study the effect of time with other CVIs, so that any particular CVI that produced minimum variation will be useful for phenological analysis. In general, based on the observations with GCC (fig. 6.7), it may be concluded that the time of imaging in the range of 10:00 to 14:00 for the whole ROI (similar to that used in the phenology network) is not affecting the overall phenological trend of soybean. As any time in the range of 10:00 to 14:00 can be suitable to capture the phenological trend, this result has a good implication on data reduction in terms of the number of images required per day to build the phenological trend for simple analysis. It should be noted that within the curve there is a lot of variation across the dates due to variations in lighting conditions.

6.4.3. Effect of Camera Distance on General GCC Phenological Curve

The GCC phenological trend from each ROI of the images captured at 12:00 for Mandan-H5 and -I2 are presented (fig. 6.8). Changing the camera distance can be thought of as selecting the sub-ROI in the vegetation field of view along the vertical direction. To demonstrate the effect of camera distance, nine ROIs were selected (fig. 6.4). A set of three ROIs, which was at the same level vertically, was assumed to be in the same distance from the camera. These sets can be considered as the regions representing far, middle, and near the camera (section 6.3.5).

Up to 45–50 DAP, the nature of trends from sub-ROIs (1–3) that represented the far region from the camera exhibited higher GCC values than the sub-ROIs (7–9) that represented the near region. The GCC trend from the middle sub-ROIs (4–6) was in between the far and the near. Even though the crop growth was uniform all over the field,
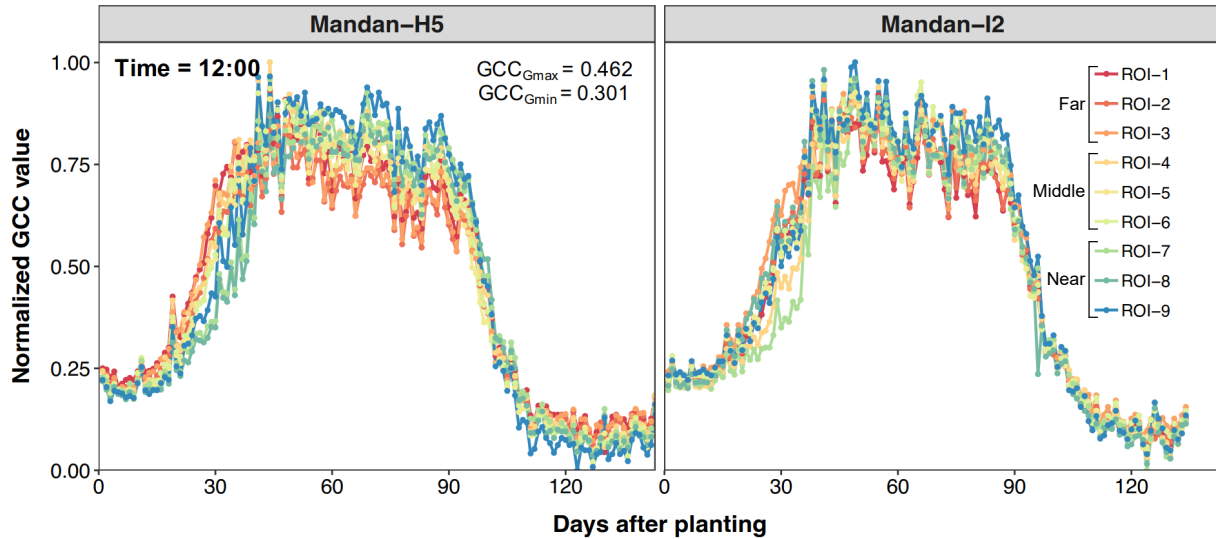
Figure 6.8. Effect of camera distances on the general GCC phenological trend from different ROIs arranged vertically at 12:00.

this phenomenon was observed because of the PhenoCam's oblique view angle, the increasing actual depth in the field with a fixed ROI height as the distance from the camera increases, and the normal fading effect the increased distance creates. The ROIs from the area that was near the camera even captured some portions of soil and residue until 50 DAP, hence lower GCC value was obtained. This pattern was prominently expressed in Mandan-H5, but to a lesser extent in Mandan-I2.

After 50 DAP, the nature of curves reversed. The ROIs that were far from the camera showed lower GCC value than the ones that were near. This was because the regions that were far from the camera were less focused and had poor resolution, which represented only the canopy. While the plants near the camera were well focused and had better resolution such that even some of the leaves were visually noticeable. Also, the reflections on the leaves, caused by the sun, were seen in the near regions compared to far. These reflections increased the color intensity values in the ROI and produced higher GCC values for plants that are near the camera.

204

Based on these general observations (fig. 6.8), it may be concluded that the distance of the camera from the field regions alters the CVIs, unlike the time of image capture, and influences the phenological trends. Therefore, it is necessary to have a consistent distance or averaging the sub-ROIs or a larger ROI may be necessary to address the variations. It can also be concluded that a smaller sub-ROI cannot be used to replace a larger ROI based on the observed variations.

6.4.4. Phenological Curve Pattern of Different CVIs

The phenological curve for all CVIs showed a clear pattern highlighting different phenology stages (fig. 6.9). Among the curves, only RGRI and CIVE showed a flipped pattern because of the nature of their respective mathematical relationships. In the RGRI calculation (eq. 6.8), the G value was considered inversely proportional to RGRI, whereas with CIVE (eq. 6.10), though it was a linear relationship, the increased weight assigned to g (0.881) and the negative sign associated with g produced the flipped pattern. Irrespective of the two different patterns (flipped and non-flipped), the phenology stages matched with the manual visual assessment.

Ideally, the VC stage (fig. 6.6) is expected in the phenological curves at 10–15 DAP. In some of the CVIs, the VC stage was barely noticeable (e.g., NGRDI, NGBDI, RGRI, EXGR, $Lab_{a'}$), which makes it difficult to identify the start of the growing season. While, a few other CVIs (e.g., GCC, MGRVI, RGBVI, GLI, EXG, CIVE, HSB_{hue}) prominently exhibited the VC stage. In PhenoCam studies, different threshold values, such as 10 %, 25 %, and 50 % of amplitude, were reported to identify the start of growing season (Browning et al., 2017; Richardson et al., 2018b). This means the DAP at which the CVI value exceeds the threshold was identified as the VC stage.
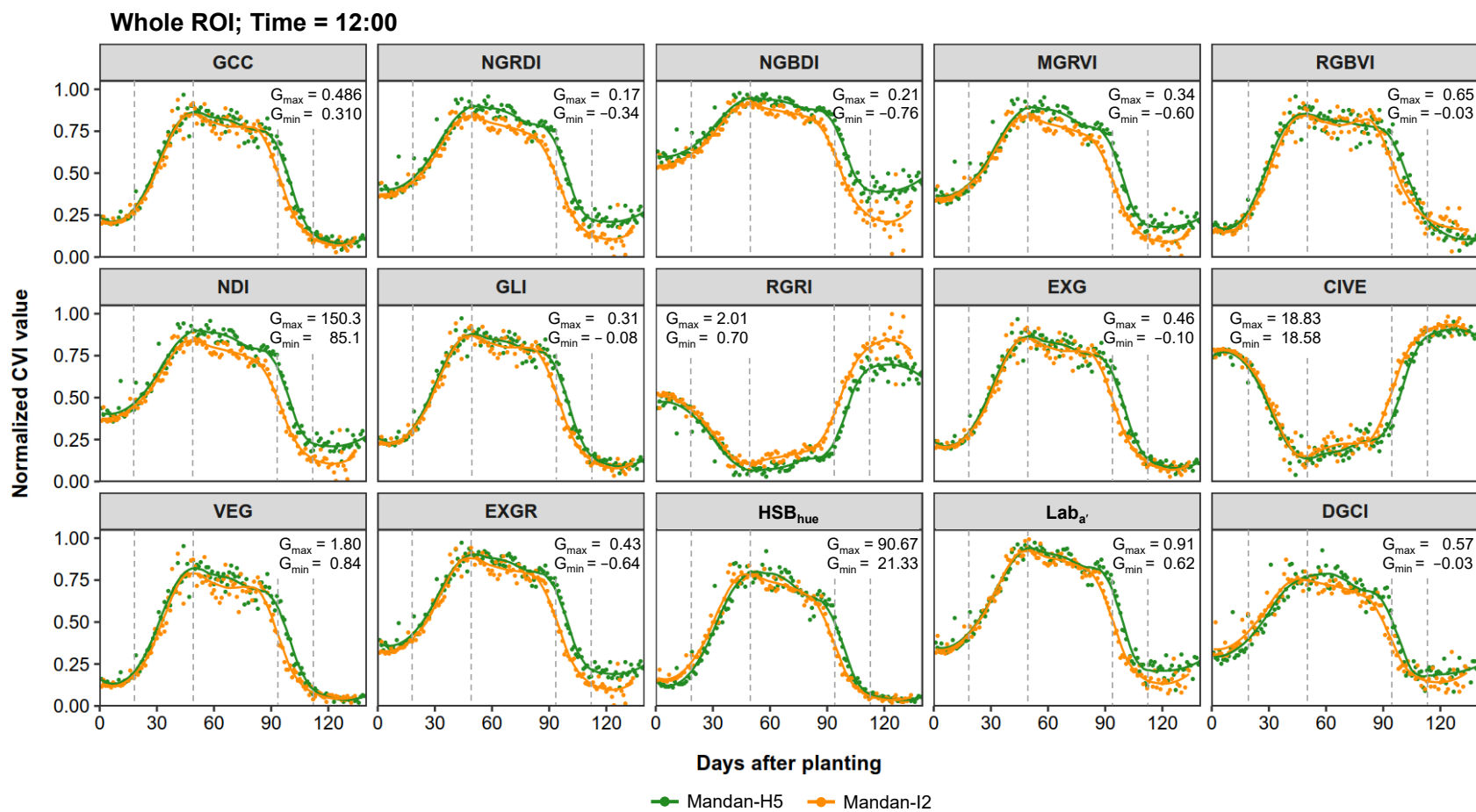
Figure 6.9. Phenological curves of different CVIs for Mandan-H5 and -I2 fields showing the different soybean phenological stages. The G_{max} and G_{min} values used in the normalization can be used to infer the actual CVI values.

In this study, the threshold value can be directly related to the graduations in the *y*-axis (normalized CVI value; fig. 6.8). The threshold value of 10 % and 50 % was found less suitable as they did not correctly represent the VC stage. Based on the observations, a threshold value of 25 % was better, because some of the CVIs (e.g., GCC, RGBVI, EXG, VEG, HSB_{hue}) correctly represented the VC stage.

The peak bloom of vegetation was identified in all the CVIs on 45–50 DAP that was also matched with manual visual assessment. In a phenological curve, this was the point when the CVIs attained the maximum value except for RGRI and CIVE (minimum value). It was interesting to note that the crop growth was uniform in both fields until this point and displayed a minor separation afterward (fig. 6.8). This separation might be due to the actual growth difference between field or some of the agronomic factors such as field history, soil fertility, and nutrition. From the CVI curves, it was clear that the peak bloom (vegetative) was distinctly noticeable from any CVI (45–50 DAP).

The phenology transition into the maturity stage had deviations between both fields — plants in Mandan-I2 matured faster than Mandan-H5 (fig. 6.8). This was observed with the separation in the CVI curve after >85 DAP. However, this phenomenon was observed distinctly only with those CVIs, in which the VC stage was not detectable. Further investigation and comparison with other agronomic factors would reveal striking inferences at this phenology stage.

The end of the growing season was obvious in all CVI curves. This was characterized by the minimum value for all the CVIs, but the maximum value for RGRI and CIVE. The curve almost becomes flat once the plants reached maturity. Some studies identified

the end of the growing season using the same threshold value used to identify the VC stage. However, the threshold value used in this study did not work well for all the CVIs.

6.4.5. Selection of a Set of Best CVIs — Based on Curve Smoothness and Time of Capture

Even though the time of image capture produced similar variation among the GCC values (fig. 6.7), the variations in the GCC values across DAP exist within each curve and an ideal CVI will minimize such variation. Therefore, it is desirable to study the variation caused by all other CVIs across DAP, with respect to a smoothed curve, to determine a set of best CVI that produced minimum variation statistically analyzed across different times of image capture. For this purpose, the variation produced by each CVI was quantified by comparing with their respective smoothed curve and $L_{Deviation}$ (eq. 6.18). The $L_{Deviation}$ produced by different CVIs from the images captured between 10:00 and 14:00 at 30 min interval is presented in table 6.1.

Among different times of image capture, the overall $L_{Deviation}$ for both fields ranged between 0.072 % and 0.412 %, and daily mean ranged between 0.052 % and 0.263 %. It was interesting to observe that none of the CVIs produced statistically the same $L_{Deviation}$ for the entire time span. There were at least two distinct groups observed for each CVI. Although the expressions were different for GCC (eq. 6.1) and EXG (eq. 6.9), the same $L_{Deviation}$ was produced between them. Similarly, the NGRDI (eq. 6.2) and NDI (eq. 6.6) produced the same $L_{Deviation}$.

Table 6.1. Curve length deviations of 15 different color vegetation indices from their respective smoothed curve from the images captured between 10:00 and 14:00 during the 2018 growing season.

Vegetation Index	Curve length deviation from smoothed curve (%)										Rank
	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	Daily mean[†]	
					Mandan-H5						
GLI	0.119±0.021 ab	0.144±0.013 a	0.145±0.015 a	0.102±0.017 b	0.129±0.023 ab	0.125±0.019 ab	0.113±0.015 ab	0.129±0.023 ab	0.148±0.019 a	0.052±0.015 c	1
GCC	0.122±0.022 ab	0.15±0.014 a	0.151±0.017 a	0.109±0.019 b	0.14±0.026 ab	0.132±0.022 ab	0.123±0.017 ab	0.139±0.027 ab	0.16±0.023 a	0.054±0.015 c	2
EXG	0.122±0.022 ab	0.15±0.014 a	0.151±0.017 a	0.109±0.019 b	0.14±0.026 ab	0.132±0.022 ab	0.123±0.017 ab	0.139±0.027 ab	0.16±0.023 a	0.054±0.015 c	3
CIVE	0.115±0.022 ab	0.144±0.013 a	0.142±0.016 a	0.103±0.017 b	0.132±0.024 ab	0.123±0.019 ab	0.113±0.016 ab	0.126±0.024 ab	0.143±0.02 a	0.054±0.014 c	4
VEG	0.14±0.029 b	0.168±0.02 ab	0.171±0.026 ab	0.131±0.028 b	0.185±0.04 ab	0.161±0.035 ab	0.165±0.03 ab	0.182±0.047 ab	0.215±0.043 a	0.058±0.013 c	5
EXGR	0.119±0.023 abc	0.149±0.014 a	0.127±0.018 ab	0.101±0.014 bc	0.124±0.021 abc	0.122±0.014 abc	0.094±0.012 c	0.1±0.015 bc	0.098±0.015 bc	0.07±0.011 d	6
RGBVI	0.202±0.029 b	0.186±0.02 b	0.192±0.026 b	0.159±0.038 b	0.208±0.041 b	0.225±0.048 ab	0.196±0.029 b	0.229±0.046 ab	0.278±0.042 a	0.079±0.019 c	7
MGRVI	0.183±0.025 a	0.172±0.01 a	0.125±0.016 b	0.125±0.01 b	0.172±0.02 a	0.178±0.01 a	0.116±0.009 b	0.117±0.01 b	0.109±0.015 b	0.081±0.014 c	8
NGRDI	0.175±0.025 a	0.177±0.013 a	0.13±0.019 b	0.123±0.012 b	0.162±0.021 a	0.172±0.011 a	0.11±0.009 b	0.113±0.01 b	0.104±0.014 b	0.086±0.013 c	9
NDI	0.175±0.025 a	0.177±0.013 a	0.13±0.019 b	0.123±0.012 b	0.162±0.021 a	0.172±0.011 a	0.11±0.009 b	0.113±0.01 b	0.104±0.014 b	0.086±0.013 c	10
RGRI	0.168±0.027 b	0.211±0.027 a	0.159±0.032 b	0.13±0.022 bcd	0.144±0.027 bc	0.166±0.021 b	0.097±0.015 d	0.113±0.011 cd	0.094±0.013 d	0.106±0.012 cd	11
NGBDI	0.159±0.03 abc	0.21±0.033 a	0.171±0.041 ab	0.132±0.028 bcd	0.135±0.032 bcd	0.168±0.031 ab	0.098±0.019 d	0.11±0.015 cd	0.09±0.012 d	0.118±0.01 c	12
HSB$_{hue}$	0.244±0.028 a	0.096±0.002 d	0.095±0.006 d	0.101±0.006 d	0.172±0.01 b	0.237±0.014 a	0.141±0.008 c	0.107±0.007 d	0.116±0.01 d	0.146±0.01 c	13
Lab$_{a'}$	0.178±0.006 b	0.198±0.014 a	0.165±0.006 bc	0.136±0.009 d	0.16±0.004 c	0.179±0.01 b	0.119±0.007 e	0.158±0.009 c	0.138±0.01 d	0.159±0.008 c	14
DGCI	0.403±0.009 a	0.206±0.019 e	0.199±0.017 ef	0.176±0.015 f	0.277±0.009 c	0.342±0.02 b	0.199±0.012 ef	0.251±0.011 d	0.219±0.006 e	0.252±0.013 d	15
					Mandan-I2						
GLI	0.114±0.016 b	0.133±0.013 b	0.114±0.007 b	0.136±0.007 b	0.14±0.008 b	0.146±0.015 a	0.157±0.02 a	0.13±0.02 b	0.163±0.016 a	0.068±0.087 c	1
GCC	0.124±0.019 b	0.138±0.015 b	0.121±0.008 b	0.144±0.007 b	0.154±0.008 a	0.16±0.018 a	0.171±0.023 a	0.141±0.024 b	0.173±0.019 a	0.072±0.089 c	3
EXG	0.124±0.019 b	0.138±0.015 b	0.121±0.008 b	0.144±0.007 b	0.154±0.008 a	0.16±0.018 a	0.171±0.023 a	0.141±0.024 b	0.173±0.019 a	0.072±0.089 c	4
CIVE	0.119±0.017 b	0.134±0.014 ab	0.115±0.007 b	0.138±0.006 ab	0.143±0.009 a	0.155±0.015 a	0.162±0.021 a	0.136±0.022 ab	0.169±0.017 a	0.071±0.084 c	2
VEG	0.18±0.033 bcd	0.173±0.028 cd	0.164±0.017 d	0.198±0.016 abcd	0.238±0.013 ab	0.237±0.038 ab	0.258±0.043 a	0.204±0.049 abcd	0.234±0.036 abc	0.097±0.084 e	6
EXGR	0.113±0.012 e	0.146±0.007 bc	0.106±0.003 e	0.136±0.009 c	0.116±0.013 de	0.157±0.006 b	0.136±0.011 c	0.132±0.011 cd	0.173±0.01 a	0.08±0.078 f	5
RGBVI	0.209±0.034 b	0.236±0.039 ab	0.233±0.015 ab	0.246±0.024 ab	0.296±0.02 a	0.269±0.042 ab	0.256±0.032 ab	0.224±0.045 b	0.26±0.049 ab	0.149±0.132 c	13
MGRVI	0.125±0.01 de	0.158±0.009 bc	0.099±0.01 f	0.143±0.016 cd	0.102±0.018 f	0.173±0.014 b	0.118±0.01 ef	0.138±0.009 cde	0.197±0.014 a	0.102±0.098 f	7
NGRDI	0.121±0.01 ef	0.166±0.008 bc	0.104±0.011 f	0.145±0.015 cd	0.104±0.018 f	0.172±0.013 ab	0.119±0.01 ef	0.137±0.009 de	0.192±0.014 a	0.107±0.089 f	8
NDI	0.121±0.01 ef	0.166±0.008 bc	0.104±0.011 f	0.145±0.015 cd	0.104±0.018 f	0.172±0.013 ab	0.119±0.01 ef	0.137±0.009 de	0.192±0.014 a	0.107±0.089 f	9
RGRI	0.118±0.012 d	0.228±0.007 a	0.141±0.018 cd	0.17±0.021 bc	0.128±0.024 d	0.166±0.018 bc	0.133±0.012 d	0.146±0.017 cd	0.194±0.022 b	0.133±0.084 d	12
NGBDI	0.133±0.013 e	0.25±0.008 a	0.174±0.018 cd	0.193±0.02 bc	0.159±0.023 de	0.192±0.014 bc	0.149±0.011 de	0.159±0.012 de	0.208±0.019 b	0.161±0.07 de	14
HSB$_{hue}$	0.134±0.009 c	0.08±0.005 ef	0.073±0.005 f	0.127±0.008 cd	0.072±0.012 f	0.178±0.014 b	0.091±0.008 e	0.116±0.003 d	0.224±0.012 a	0.122±0.008 d	11
Lab$_{a'}$	0.111±0.003 cd	0.156±0.012 a	0.103±0.006 d	0.124±0.015 bc	0.103±0.008 d	0.127±0.007 b	0.124±0.003 bc	0.136±0.006 b	0.168±0.008 a	0.128±0.007 b	10
DGCI	0.239±0.011 cde	0.272±0.034 bcd	0.174±0.032 f	0.245±0.032 cde	0.217±0.03 ef	0.307±0.036 b	0.222±0.021 def	0.282±0.024 bc	0.412±0.025 a	0.263±0.027 b	15

Note: [†]Daily mean value of each CVI was calculated from the daily RGB mean and standard deviation values provided by the PhenoCam network for the whole ROI (solid line, fig. 6.4), GLI - green leaf index, GCC - green chromatic coordinate, EXG - excess green, CIVE - color index of vegetation, VEG - vegitativen, EXGR - excess green minus excess red, RGBVI - red green blue vegetation index, MGRVI - modified green red vegetation index, NGRDI - normalized green red difference index, NDI - normalized difference index, RGRI - red green ratio index, NGBDI - normalized green blue difference index, HSB$_{hue}$- hue channel value from HSB color space, Lab$_{a'}$- inverted "a" channel value from Lab color space, and DGCI - dark green color index. Dissimilar letter groups (a–f) denote significant differences in the curve length deviations among time.

The CVIs from HSB and Lab color spaces had more deviation than the ones from RGB color space. Among the three color spaces, the $Lab_{a'}$ produced the minimum deviation, followed by HSB_{hue} and DGCI (table 6.1). The reason for the higher deviation of DGCI might be because the index calculation (eq. 6.15) involved the S and B' components that were influenced due to the color intensity variation. The application of DGCI is primarily on quantifying the variations among fine shades of green and hence it is sensitive to even slight intensity variation (Karcher & Richardson, 2003). If only HSB_{hue} (eq. 6.13) was considered, ignoring S and B' channels, the $L_{Deviation}$ reduced by about half when compared to the DGCI (table 6.1).

The CVIs were ranked based on the $L_{Deviation}$ in Mandan-H5 using daily mean RGB values in ascending order, and it was observed that the GLI produced the minimum and DGCI produced the maximum $L_{Deviation}$ in both fields (table 6.1). The ranking order of the rest of the CVIs was different for Mandan-I2 and did not have a specific order between the fields. This might be because the location of the two fields were away ($\approx 3.2\,km$), one field could have possibly had a cloud cover while others did not. This cloud cover reduced the image color intensity and thus varied the CVI value between fields. Also, the inherent field conditions such as soil type and moisture availability would have created some differences in crop growth. However, the top six CVIs were similar in both fields but a different order. The CIVE, even though the phenological pattern was flipped (fig. 6.8), the $L_{Deviation}$ value was less and comparable with other top-five CVIs.

Out of all the grouping labels from mean separation, the CVI that produced less number of groups was considered ideal in this study, because it denotes significantly less variation among different time. Some of the CVIs produced up to six groups (a–f) among

different times (e.g., DGCI, NGRDI in Mandan-I2). This indicated that any minor changes in ambient lighting conditions among time caused a significant effect on the CVI values. A few CVIs in Mandan-H5, such as MGRVI, NGRDI, NDI, also produced just two groups, but their $L_{\text{Deviation}}$ values were higher. Therefore, the selection of CVI based on a combination of "fewer groups" and "lower $L_{\text{Deviation}}$ value" indicated a better CVI that was less affected by lighting conditions among time.

Based on the above-mentioned criteria, four CVIs from RGB color space selected were: GLI, GCC, EXG, and CIVE. The other CVIs that closely followed were VEG and EXGR. Also, in the interest of exploring other color spaces, all three CVIs in this study, namely, HSB_{hue}, $\text{Lab}_{\text{a}'}$, and DGCI were selected and analyzed. However, it was found that these other color space CVI were ranked the lowest (10–15 out of 15) and have also produced several significant groups among times of image capture (table 6.1).

It can be seen that GCC and EXG were already well established in the PhenoCam network in monitoring phenology (Andresen et al., 2018; Crimmins & Crimmins, 2008). The other two CVIs, such as GLI and CIVE were also comparable with GCC and likely to be suitable for the analysis. The GLI produced the least $L_{\text{Deviation}}$ among the CVIs. Therefore, along with GCC and EXG, the GLI and CIVE can also be possibly considered in future PhenoCam related measurement and analysis. It will be also worthwhile to validate these CVIs with different crops and ecosystems (forest and agricultural) in future studies.

6.4.6. Phenological Pattern on Selected CVIs

The phenological curve for selected CVIs at different times of image capture showed differences among CVIs (fig. 6.10). To consider the selected CVIs in future studies, it is important to see whether the CVIs followed the phenological pattern. The variation in the

phenological pattern among CVIs at a specific time can be visually observed by following through the panels in each column (fig. 6.10). It should be noted that each column corresponds to the same set of images (143 images for Mandan-H5 and 139 images for Mandan-I2) at a specified time and thus the color values extracted are also the same throughout each panel in the columns. The differences in pattern among CVI at a specified time are solely due to the mathematical relationships used for deriving these CVIs.

The GLI closely followed a typical phenological curve and there was no much difference between Mandan-H5 and -I2. Irrespective of the time of image capture, GLI clearly showed some of the significant phenology stages, such as VC, R2, and R8 at 10–15 DAP, 40–45 DAP, and 110–120 DAP, respectively. Though there were some distortions observed in the pattern, it is possibly due to the cloud cover during image capture. Although the $L_{\text{Deviations}}$ produced by GCC and EXG were the same (table 6.1), the phenological pattern produced looks different. On the basis of following the phenological pattern among time, the CIVE was better than EXG, even though it was flipped. The difference in the curve between the two fields was also less in CIVE. Specifically, considering the image at 12:00 (midday), except for $Lab_{a'}$, all other CVIs produced an irregular pattern.

In other color spaces, the simple HSB_{hue} and $Lab_{a'}$ were moderately suitable for this application of monitoring phenological trend; however, it can be seen that DGCI was less suitable because of the irregular fluctuations throughout the growing season. But, it may be useful in identifying crop stress such as nitrogen (Raper et al., 2012). Therefore, this can be further explored in PhenoCam fields with stress trials in the field.
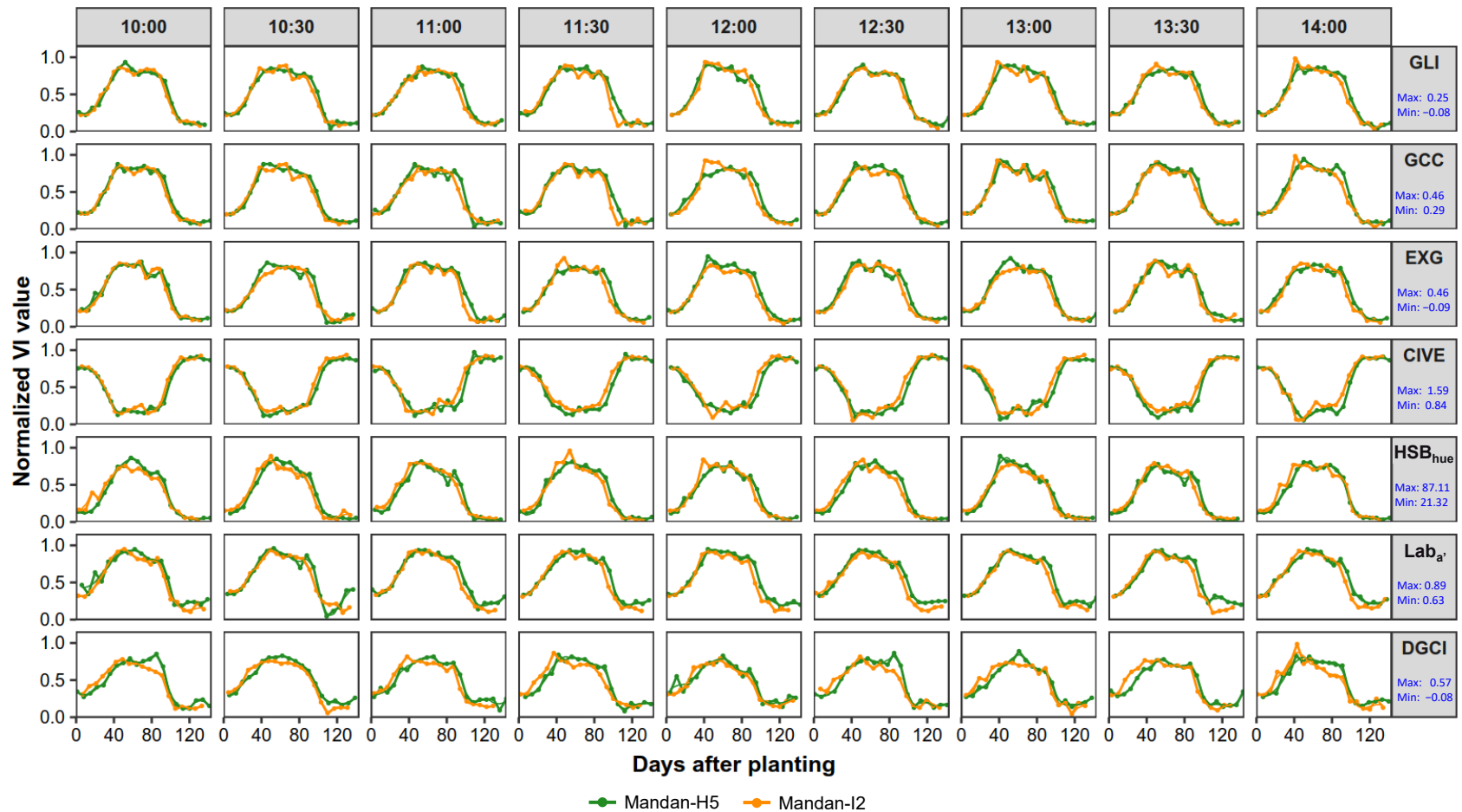
Figure 6.10. Phenological trend variations in the selected CVIs between 10:00 and 14:00 for Mandan-H5 and Mandan-I2. Note: GLI - green leaf index, GCC - green chromatic coordinates, EXG - excess green, CIVE - color index of vegetation, HSB$_{hue}$- hue channel value from HSB color space, Lab$_{a'}$- inverted "a" channel value from Lab color space, and DGCI - dark green color index. Note: The Max and Min values provided in the row panel title can be used to obtain the actual CVI value from the normalized CVI value.

6.4.7. Comparison Among Different CVIs

Statistical comparisons among different CVIs help understand the similarity among them and lead to the selection of a set of similar CVIs. The curve length deviations from the daily mean values of each CVI was used to perform the analysis (fig. 6.11). It was found that the selected CVIs from RGB color space such as GLI, GCC, EXG, and CIVE were statistically similar in both fields. For Mandan-H5, the CVIs such as RGBVI, MGRVI, NGRDI, and NDI were statistically similar, while not for Mandan-I2.
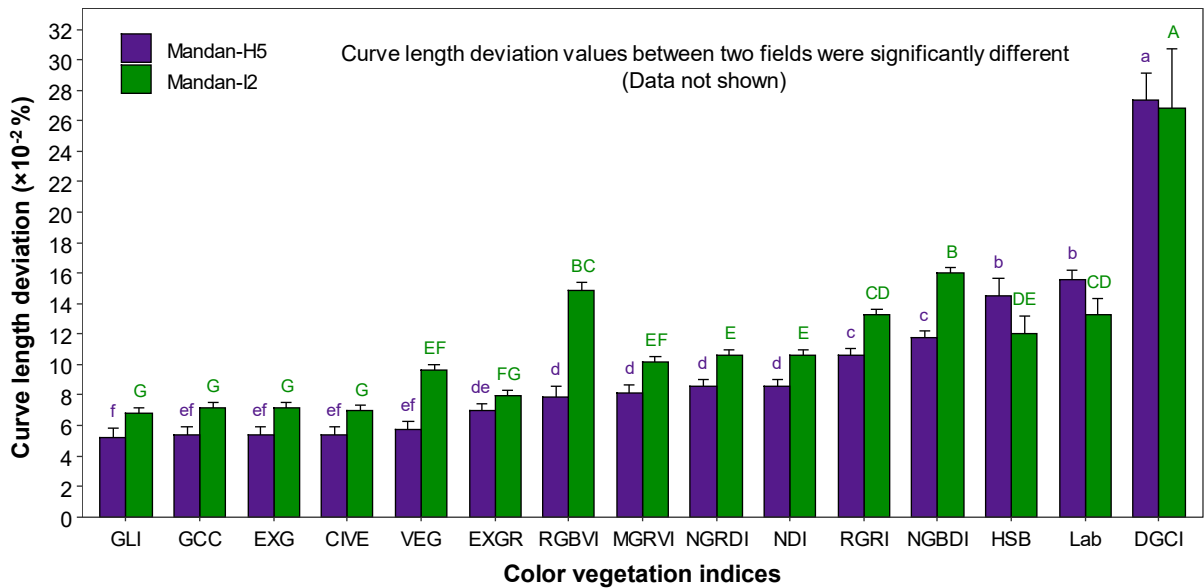
Figure 6.11. Curve length deviations for different CVIs with daily RGB mean values. Note: Dissimilar letter groups (a-g) of the same color denote significant differences among CVIs of the same experimental plot.

For all the CVIs from RGB color space, Mandan-I2 had consistently produced a higher curve length deviation than Mandan-H5, while in HSB and Lab color spaces, Mandan-I2 produced lower curve length deviation. This is an interesting observation and gives an insight that the mathematical relationships with RGB color space and other color spaces are different. Also, it can be seen that DGCI produced the minimum deviation

between the two fields, but the phenological pattern (fig. 6.9) was not suitable. Therefore, slightly modifying the DGCI equation or deriving a new index similar to DGCI would possibly suit the phenological applications and which will be a future scope of the study.

6.4.8. Effect of Camera Distance on Selected CVIs

Only the selected CVIs that had the minimum deviations and had two mean groups were used for testing the effect of camera distance. The camera distances — represented by far, middle, and near ROIs were assumed at the same horizontal level (fig. 6.4), with the component sub-ROIs at each level served as the replications. The results indicated that camera distance had a significant effect on some CVIs but not on all (fig. 6.12).
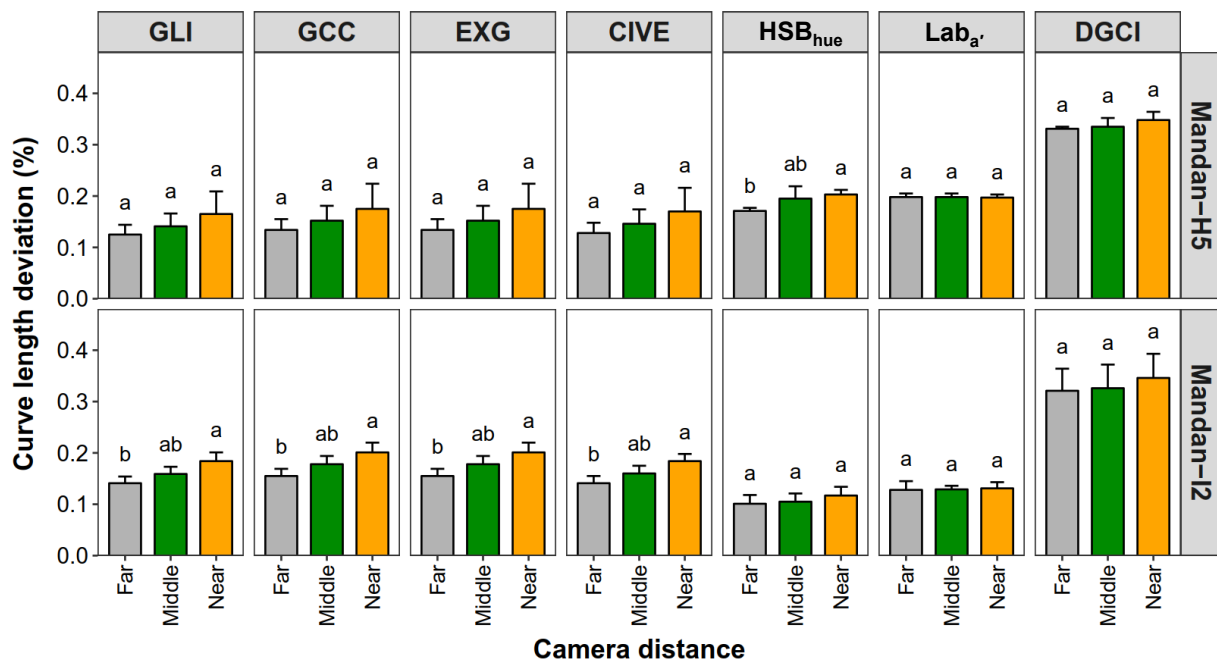
Figure 6.12. Effect of camera distance on selected CVIs for Mandan-H5 and -I2. Dissimilar letter groups (a and b) denote that the selected CVI is significantly different from camera distance.

A general trend of the CVIs variation with the distance of near producing higher deviation than middle and far was observed. This was because the regions near the camera

215

captured more features with a better resolution and focus. For example, even the sun's reflections on the leaves were noticeable in the near portions, which caused more fluctuations in the CVI values and were captured by curve length deviation. While the regions that were away from the camera barely captured the leaves (low-resolution and leaves defined by fewer pixels) and also had the fading effect. A combination of these effects reduced the variation in the CVI curves and produced smaller values of curve length deviation for the near camera distance.

In both the fields, the DGCI produced the maximum deviation. In Mandan-H5, except HSB$_{hue}$ all other CVIs did not have a significant effect on distance. However, in Mandan-I2 all the CVIs from RGB color space produced significant differences among camera distance (fig. 6.11). In some of the PhenoCam studies, the ROI is selected randomly without considering the effect of distance (Yan, Scott, Moore, Biederman, & Smith, 2019). The ROIs that were far from the camera is not a proper representative to monitor the crop growth. Therefore, it can be concluded that the selection of the ROI location in the PhenoCam image is crucial and should be chosen appropriately so that it provides a considerable and representative region for phenology monitoring and other analysis.

6.4.9. Results of Simplified Phenological Curve Analysis by Kymograph

The downloaded images from Mandan-H5 and -I2 were used to generate the kymograph in ImageJ, a composite image made from one pixel ROI from each time-lapse image that provides a glimpse of the whole growing season (fig. 6.13A). Simple operations such as separating the color channels into HSB and Lab stack and analyzing the profile enabled to observe the entire phenological trend of the crop from this simple image (fig. 6.13B–E).
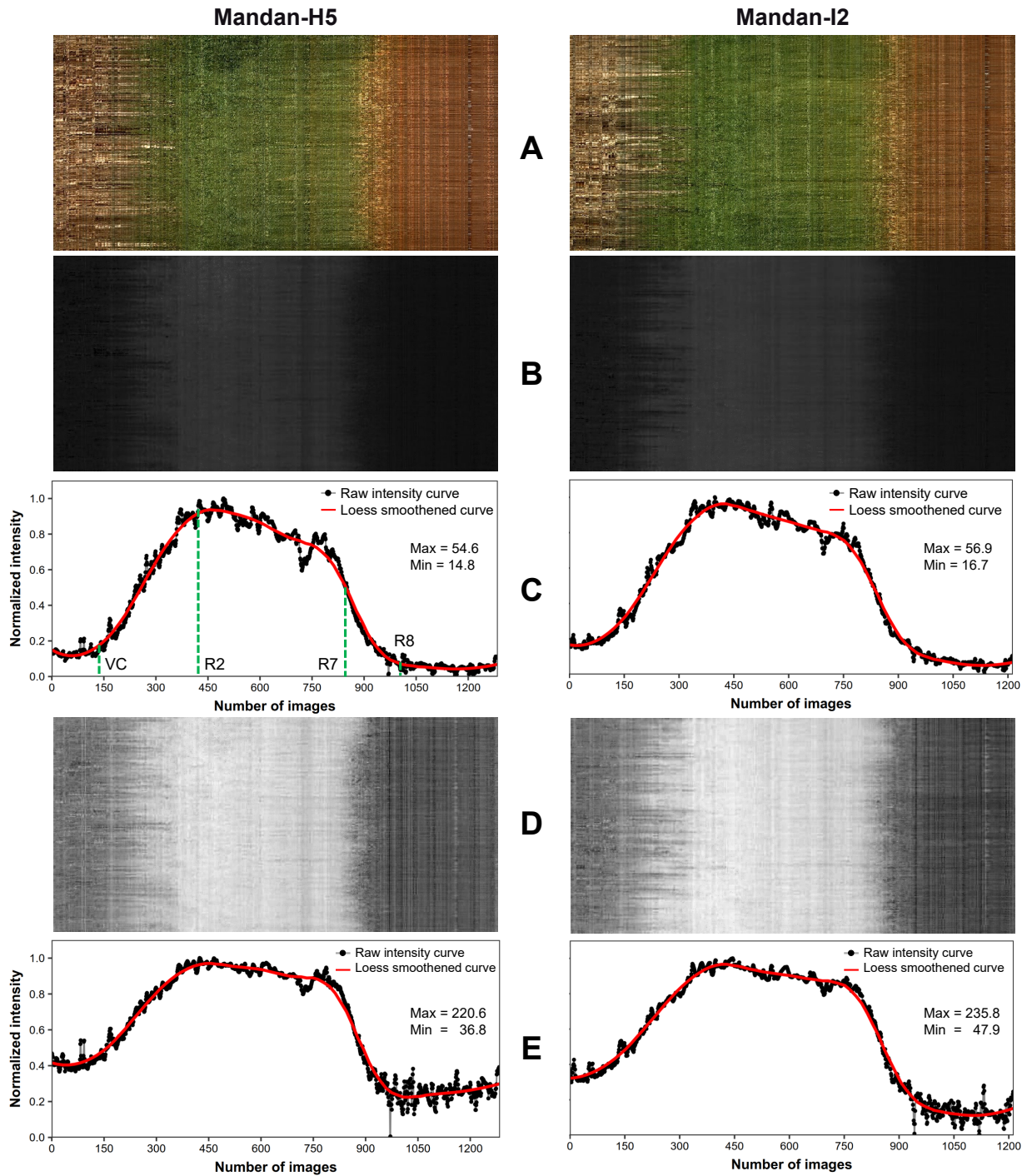
Figure 6.13. Simplified phenological curve analysis from PhenoCam images: (A) kymographs generated using ImageJ for Mandan-H5 and Mandan-I2, (B) hue channel image from HSB color space, (C) hue channel profile for the whole kymograph, (D) inverted "a" channel image from Lab color space, and (E) inverted "a" channel profile for the whole kymograph.

The kymograph, as a single image, itself served as a good reference to observe the phenology transitions. It can be observed that the soil or residue on the left side of the kymograph slowly disappeared and the vegetation started to increase. Once the peak vegetation bloom/canopy closure was attained, the green vegetation portion remained until the plants started to mature. The transition from green to "yellowish" can be seen in the kymograph. Once the plant was fully matured, the portion towards the right of the kymograph was in brown. The color difference between soil (left of kymograph) and matured plant (right of kymograph) was interesting to observe and this color difference can be used to discern between matured plant and soil.

The hue channel produced from the kymograph (fig. 6.13B) showed a slight variation in a gray color that represented the vegetation. The beginning of maturity (yellowness) was not observed in this channel image through simple observation. However, the profile (fig. 6.13C) obtained from the hue image was useful and exhibited the phenological trend. The x-axis of the profile showed the number of images used in the kymograph. The corresponding DAP values can be obtained by dividing the x-axis by nine since nine images per day (10:00–14:00) were used in this study. All four significant phenology stages were identified in the profile similar to the typical GCC curve (fig. 6.6).

The "a" channel image from Lab color space was not directly useful; hence, the image was inverted and used for analysis (fig. 6.13D). The vegetation portion was seen in lighter shades of gray, while the matured portion was seen in dark gray. Visually the phenology transitions were noticeable from the image. The corresponding profile plot (fig. 6.13E) was flat compared to the hue channel profile (fig. 6.13C). The VC stage was

not seen in Mandan-I2. The variations produced in the profile after the plant reached maturity did not convey any meaningful observation from the kymograph.

This simplified analysis using kymograph was straightforward and does not require any plugin coding (fig. 6.13). The generation of kymographs from 1287 images took a CPU time of ≈ 2 min including opening the image as a stack. This is quite understandable as the kymograph used only a single line ROI to build the composite image, and thus took less time in processing but the major portion of time in reading the images and the stack creation. The reduction in processing time compared to the plugin-based method, which used all the images in the raw form, was about $\approx 73.3\%$.

The results showed that this simplified approach has a potential scope to be used in the PhenoCam network as it considerably reduces the processing time and storage (once kymographs were made). This simplified approach should also be extended to other CVIs and compared with the regular method and statistically analyzed. A similar concept of obtaining a kymograph with a rectangular ROI window will be a potential for further investigation and extension to this simplified approach; however, it requires user-coded plugin development.

6.5. Conclusions

This study successfully determined the phenological trends of soybean (two PhenoCam sites: Mandan-H5 and Mandan-I2) through various color vegetation indices (CVIs; 15 CVIs RGB, HSB, and Lab color spaces), evaluated the effect of time of image capture (10:00–14:00) and distance from the PhenoCam (far, middle, and near) on the CVIs, and selected a set of best CVIs that will be suitable for PhenoCam application and analysis in agricultural environment. The developed user-coded ImageJ plugin was found

efficient in extracting the color values from a specified region of interest (ROI) from the PhenoCam images and generated values in the different color spaces for analysis. It was observed that the phenology stages from manual visual assessment matched well with the features in the typical green chromatic coordinates (GCC) phenological curve.

The time of image capture had a significant effect on all CVIs with at least two mean groups and the best CVIs were selected based on minimum deviation from its respective smoothed curve and least number of statistical groups. The best CVIs selected were green leaf index (GLI), green chromatic coordinates (GCC), excess green (EXG), and a color index of vegetation (CIVE). Out of these four CVIs that were not significantly different, the GLI was the best and it closely followed the phenological trend. The CVIs from other color spaces such as HSB and Lab were not found suitable. The regions near the PhenoCam produced more deviation than the middle followed by far because of better resolution and focus when tested with the selected CVIs. There was a significant difference among camera distances for all RGB-based CVIs with Mandan-I2 site, but not in Mandan-H5 site. A simplified no-code approach to identify the phenology using the kymograph has produced promising results. The hue channel from the kymograph displayed distinct phenology stages. The kymograph method was simple and was $\approx 73.3\%$ faster in analysis than the plugin-based method dealing with multiple PhenoCam images.

7. GENERAL CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

7.1. General Conclusions

There are several digital image processing applications available in agriculture; however, the literature on the field applications is limited (<35 %). Working with agricultural field images is quite challenging due to the variations in lighting, improper focus, and the presence of artifacts. Moreover, several of the applications are developed using commercial software and it was reported that it requires high technical knowledge to operate in general and the processing software packages is inaccessible to users/farmers. Therefore, this study focused on developing four agricultural field applications of digital image processing using Fiji, an open source no-cost image processing platform. Fiji is a distribution of ImageJ and enables easier plugins development experience using Java language. All the plugins were developed on a personal computer with the following configuration of Windows 10, Intel Core i5, and 8 GB RAM. The selection of four objectives was based on field requirements in modern agriculture and problems already existing that can be addressed through image processing applications. The findings from each objective are presented here:

- Objective 1: An ImageJ plugin to measure the dimensions of the field sunflower floral components was developed and tested successfully. The plugin performed multiple radial measurements using the "pixel-march" method and produced the accurate

221

dimensions. The recommended number of measurements that best represented sunflower head and ray florets was 32, and for the disc was 8. Dimensions of the sunflower head and disc obtained from the pixel-march method were statistically similar to the equivalent diameter as well as minor axis methods of ImageJ. However, accurate dimensions of ray florets were possible only through this plugin as there were no standard methods available in ImageJ or any other image processing software. The measurement took ≈11 s to perform 180 measurements on all sunflower components.

- Objective 2: A color calibration ImageJ plugin termed "ColorCal" to standardize the image intensity variations was developed and tested with laboratory and field images. The user-developed plugin produced calibrated images by deriving a color calibration matrix (CCM) by comparing the standard 24 patches color chart in the image with its respective standard values. The developed calibration performance index (CPI) was useful in observing the effect of the number of color patches and deciding the minimum number of color patches. The calibration using only the 3 patches namely red, green, and blue (R, G, and B) produced visually homogenous intensity images, while the commonly followed neutral colors namely white and gray patches produced differences in image intensity. The plugin took ≈7 s/image to produce a calibrated image and performed well equally in laboratory and field conditions.

- Objective 3: A user-coded open source ImageJ plugin, developed from scratch using a simple profile and geometrical approach to perform stand counting and spatial distribution from UAS images was developed successfully. The rows were accurately

identified up to a row length of ≈100 m based on profile analysis, and is expected to work even >100 m, as long as the rows are straight. A novel segmentation approach to resolve the plant cluster based on sideways profile and object geometry showed promising results. The plugin achieved an accuracy of 95.9 % and 84.9 % on the stand count tested with two sets of images (different sites, resolution, and growth stages). For the spatial distribution analysis, a new index termed "ideal spacing uniformity index" (ISUI) was developed, utilizing the ideal seed spacing, and found to be the best to evaluate plant spatial distribution. The plant spacing distribution was compared with ideal spacing and accuracy of >76.5 % was found. The graphical outputs such as the spatial distribution map and the histogram helped in visualizing the "location" and "count" of ideal, skip, and multiples. The CPU time taken for the analysis was 31.7 s and 4.2 s for two sets of images, respectively. The CPU time was affected by image resolution and expected to increase with a higher resolution for the same field area.

- Objective 4: Soybean growth monitoring using various color vegetation indices derived from PhenoCam images was evaluated. The developed user-coded ImageJ plugin efficiently extracted the color values from a specified region of interest (ROI) from the PhenoCam images and generated values in the different color spaces (RGB, HSB, and Lab) for analysis. The phenology stages from manual visual assessment matched well with the features in the typical phenology curve. The time of image capture (10:00–14:00) had a significant effect on all CVIs and the best CVIs were selected based on minimum deviation value and the least number of statistical groups. The best CVIs selected were green leaf index (GLI), green chromatic

223

coordinates (GCC), excess green (EXG), and a color index of vegetation (CIVE). These four CVIs were not significantly different from each other, but the GLI was the best as it closely followed the observed phenology trend. The CVIs from other color spaces such as HSB and Lab were not found suitable. There was a significant difference among PhenoCam distances (far, middle, near) for all RGB-based CVIs in one field but not on the other field. The regions near the phenocam produced more deviation than the middle followed by far because of better resolution and focus for the near regions. A simplified no-code approach to identify the phenology using the kymograph displayed distinct phenology stages from the hue channel. The kymograph method was simple and significantly reduced the analysis time by $\approx$73.3 % compared to the plugin-based method dealing with multiple phenocam images.

7.2. 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 dissertation.

1. Develop an image processing approach to count the number of seeds in the sunflower disc and correlate it with the seed yield estimates.

2. Develop a higher order color calibration matrix by including the square terms and interaction terms.

3. Develop an efficient row identification method for curvilinear rows and perform stand count on irregular field geometry.

4. Study the phenology trend for different crops and evaluate the variation in color vegetation indices as well as apply time-series analysis for phenology trend prediction.

REFERENCES

Ahrends, H. E., Brügger, R., Stöckli, R., Schenk, J., Michna, P., Jeanneret, F., ... Eugster, W. (2008). Quantitative phenological observations of a mixed beech forest in northern Switzerland with digital photography. *J. Geophys. Res. Biogeosci.*, *113*, G04004 (1–11). (`https://doi.org/10.1029/2007jg000650`)

Akkaynak, D., Chan, E., Allen, J. J., & Hanlon, R. T. (2011). Using spectrometry and photography to study color underwater. In *OCEANS'11* (pp. 1–8).

Akkaynak, D., Treibitz, T., Xiao, B., Gürkan, U. A., Allen, J. J., Demirci, U., & Hanlon, R. T. (2014). Use of commercial off-the-shelf digital cameras for scientific data acquisition and scene-specific color calibration. *J. Opt. Soc. Am. A*, *31*(2), 312–321.

Ali, A., Streibig, J. C., Christensen, S., & Andreasen, C. (2011). Estimation of weeds leaf cover using image analysis and its relationship with fresh biomass yield of maize under field conditions. In *Proceedings of the International Conference on Information and Communication Technologies* (pp. 41–49).

Alsam, A., & Finlayson, G. (2008). Integer programming for optimal reduction of calibration targets. *Color Res. Appl.*, *33*(3), 212–220.

Amatya, S., Karkee, M., Zhang, Q., & Whiting, M. D. (2017). Automated detection of branch shaking locations for robotic cherry harvesting using machine vision. *Robotics*, *6*(4), 31 (1–16).

Andresen, C. G., Tweedie, C. E., & Lougheed, V. L. (2018). Climate and nutrient effects on

    arctic wetland plant phenology observed from phenocams. *Remote Sens. Environ.*,

    205, 46–55. (`https://doi.org/10.1016/j.rse.2017.11.013`)

ASF. (2018). *Apache Commons Math.* Apache Commons™. The Apache Software

    Foundation (ASF), Wakefield, MA, USA. Available at:

    <`http://commons.apache.org/proper/commons-math/download_math.cgi`>.

    (Accessed October 11, 2019)

Atlagić, J., Joksimović, J., Sakač, Z., Miklič, V., & Dušanić, N. (2003). Mode of inheritance

    and heritability of disc flower corolla length and nectar content in sunflower.

    Genetika, *35*(1), 59–65.

Ault, A., Krogmeier, J., & Buckmaster, D. (2018). Ag's future belongs to open source.

    Resource - ASABE Magazine, *25*(2), 7–9.

Bailer, W. (2006). Writing ImageJ plugins — A tutorial [Computer software manual].

    Retrieved from

    `http://www.gm.fh-koeln.de/~konen/WPF-BV/tutorial-ImageJ_V1.71.pdf`

    (Based on ImageJ release 1.36)

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.

Barbedo, J. G. A. (2013). Digital image processing techniques for detecting, quantifying

    and classifying plant diseases. *SpringerPlus*, *2*(1), 660.

Barbedo, J. G. A. (2014). Using digital image processing for counting whiteflies on

    soybean leaves. *J. Asia-Pac. Entomol.*, *17*(4), 685–694.

Barbedo, J. G. A. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. *Biosyst. Eng., 144,* 52–60.

Baret, F., & Guyot, G. (1991). Potentials and limits of vegetation indices for LAI and APAR assessment. *Remote Sens. Environ., 35*(2-3), 161–173.

Bechar, A., & Vigneault, C. (2016). Agricultural robots for field operations: Concepts and components. *Biosyst. Eng., 149*, 94–111.

Beeri, O., & Peled, A. (2009). Geographical model for precise agriculture monitoring with real-time remote sensing. *ISPRS J. Photogramm. Remote Sens., 64*(1), 47–54.

Bendig, J., Yu, K., Aasen, H., Bolten, A., Bennertz, S., Broscheit, J., . . . Bareth, G. (2015). Combining UAV-based plant height from crop surface models, visible, and near infrared vegetation indices for biomass monitoring in barley. *Int. J. Appl. Earth Obs. Geoinf., 39*, 79–87.

Berglund, D. (2007). *Sunflower production.* Available at: <https://www.ag.ndsu.edu/extensionentomology/recent-publications-main/ publications/A-1331-sunflower-production-field-guide>. (Accessed October 11, 2019)

Berglund, D. R., McWilliams, D. A., & Endres, G. J. (2015). *Soybean Growth and Management Quick Guide.* Available at: <https://www.ag.ndsu.edu/pubs/plantsci/rowcrops/a1174.pdf>. (Accessed October 11, 2019)

Blue River Technology. (2018). *Make every plant count.* Available at: <http://www.bluerivertechnology.com/>. (Accessed October 11, 2019)

Bock, C. H., Poole, G. H., Parker, P. E., & Gottwald, T. R. (2010). Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging. *Crit. Rev. Plant Sci.*, *29*(2), 59–107.

Borger, C. P. D., Hashem, A., & Pathan, S. (2010). Manipulating crop row orientation to suppress weeds and increase crop yield. *Weed Sci.*, *58*(2), 174–178.

Borianne, P., Subsol, G., Fallavier, F., Dardou, A., & Audebert, A. (2018). GT-RootS: An integrated software for automated root system measurement from high-throughput phenotyping platform images. *Comput. Electron. Agric.*, *150*, 328–342.

Brown, L., Datta, A., & Pankanti, S. (2012). Exploiting color strength to improve color correction. In *IEEE International Symposium on Multimedia* (pp. 179–182).

Brown, M., Majumder, A., & Yang, R. (2005). Camera-based calibration techniques for seamless multiprojector displays. *IEEE Trans. Visual Comput. Graphics*, *11*(2), 193–206.

Brown, T. B., Hultine, K. R., Steltzer, H., Denny, E. G., Denslow, M. W., Granados, J., . . . others (2016). Using phenocams to monitor our changing earth: Toward a global phenocam network. *Front. Ecol. Environ.*, *14*(2), 84–93.

Browning, D., Karl, J., Morin, D., Richardson, A., & Tweedie, C. (2017). Phenocams bridge the gap between field and satellite observations in an arid grassland ecosystem. *Remote Sens.*, *9*(10), 1071 (1–21).

Bruin, J. D., & Pedersen, P. (2004). *Early season scouting.* Available at: <https://crops.extension.iastate.edu/early-season-scouting>. (Accessed October 11, 2019)

Burger, W., & Burge, M. J. (2016a). *Digital Image Processing: An Algorithmic Introduction Using Java*. Springer.

Burger, W., & Burge, M. J. (2016b). Geometric operations. In D. Gries & F. B. Schneider (Eds.), *Digital Image Processing: An Algorithmic Introduction Using Java* (pp. 513–534). Springer.

Camargo, A., & Smith, J. S. (2009). An image-processing based algorithm to automatically identify plant disease visual symptoms. *Biosyst. Eng.*, *102*(1), 9–21.

Camargo, N. J. (2003). A combined statistical-soft computing approach for classification and mapping weed species in minimum-tillage systems. In *ETD Collection for University of Nebraska - Lincoln* (pp. 1–170).

Castle, M. H. (2016). *Has the usage of precision agriculture technologies actually led to increased profits for Nebraska producers?* [MS thesis.] Lincoln, Nebraska: University of Nebraska, Lincoln, Agricultural Economics Department.

Chen, B., Tojo, S., & Watanabe, K. (2003). Machine vision based guidance system for automatic rice transplanters. *Appl. Eng. Agric.*, *19*(1), 91.

Chen, Y., Yu, Q., & Xu, C.-B. (2017). A convenient method for quantifying collagen fibers in atherosclerotic lesions by ImageJ software. *Int. J. Clin. Exp. Med.*, *10*(10), 14904–14910.

Cheng, H.-D., Jiang, X. H., Sun, Y., & Wang, J. (2001). Color image segmentation: advances and prospects. *Pattern Recognit.*, *34*(12), 2259–2281. (https://doi.org/10.1016/s0031-3203(00)00149-7)

Christiansen, J. (1942). Irrigation by sprinkling. *California Agricultural Experiment Station Bulletin*, *670*, 94–96.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *J. Am. Stat. Assoc.*, *74*(368), 829–836.

Conner, J. K., & Rush, S. (1996). Effects of flower size and number on pollinator visitation to wild radish, *Raphanus raphanistrum*. *Oecologia*, *105*(4), 509–516.

Crimmins, M. A., & Crimmins, T. M. (2008). Monitoring plant phenology using digital repeat photography. *Environ. Manage.*, *41*(6), 949–958.

Crnojević, V., Lugonja, P., Brkljač, B. N., & Brunet, B. (2014). Classification of small agricultural fields using combined Landsat-8 and RapidEye imagery: Case study of northern Serbia. *J. Appl. Remote Sens.*, *8*(1), 083512.

Dafni, A., & Kevan, P. G. (1997). Flower size and shape: Implications in pollination. *Israel J. Plant Sci.*, *45*(2-3), 201–211.

de Mendiburu, F. (2019). *Agricolae: Statistical Procedures for Agricultural Research (Ver. 1.3-1)*. Available at: <http://CRAN.R-project.org/package=agricolae>. (Accessed October 11, 2019)

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.

Deery, D., Jimenez-Berni, J., Jones, H., Sirault, X., & Furbank, R. (2014). Proximal remote sensing buggies and potential applications for field-based phenotyping. *Agronomy*, *4*(3), 349–379.

Demirbas, A. (2007). Biodiesel from sunflower oil in supercritical methanol with calcium oxide. *Energy Convers. Manage.*, *48*, 937–941.

de Souza, C. H. W., Lamparelli, R. A. C., Rocha, J. V., & Magalhães, P. S. G. (2017). Mapping skips in sugarcane fields using object-based analysis of unmanned aerial vehicle (UAV) images. *Comput. Electron. Agric.*, *143*, 49–56.

dos Santos Ferreira, A., Freitas, D. M., da Silva, G. G., Pistori, H., & Folhes, M. T. (2017). Weed detection in soybean crops using ConvNets. *Comput. Electron. Agric., 143*, 314–324.

Dudek, T. A., & Fernandez, R. T. (2015). Conducting a water application uniformity evaluation for an overhead sprinkler irrigation system in the nursery. *Michigan State University Extension. Consultado, 15*, 1–2. (`http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.717.296&rep=rep1&type=pdf`)

DuPont, P. (2015). *Corn stand evaluation.* Available at: <`https://www.pioneer.com/home/site/us/agronomy/library/corn-stand-evaluation/`>. (Accessed October 11, 2019)

Elghandour, M. M. Y., Vallejo, L. H., Salem, A. Z. M., Salem, M. Z. M., Camacho, L. M., Odongo, N. E., et al. (2017). Effects of schizochytrium microalgae and sunflower oil as sources of unsaturated fatty acids for the sustainable mitigation of ruminal biogases methane and carbon dioxide. *J. Cleaner Prod., 168*, 1389–1397.

FAO. (2017). *FAO-Food and Agriculture Organization of the United Nations Food Outlook: Oilseeds.* Available at: <`http://www.fao.org/fileadmin/templates/est/COMM_MARKETS_MONITORING/Oilcrops/Documents/Food_outlook_oilseeds/FO_July_2017.pdf`>. (Accessed October 11, 2019)

Flores, P., Schatz, B., & Aberle, E. (2017). Sunflower stand counts from imagery collected with a small off-the-shelf UAS. *NDSU Carrington Research Extension Center Annual Report, 58*, 32–35. (`https://www.ag.ndsu.edu/carringtonrec/documents/annual-reports/2017-annual-report/view`)

Ford, A., & Roberts, A. (1998). Colour space conversions. *Westminster University, London, 1998*, 1–31.

FreePNGimg. (2018). *Green leaf PNG*. Available at: <`http://www.freepngimg.com/png/2293-green-leaf-png`>. (Accessed October 11, 2019)

Gamon, J., & Surfus, J. (1999). Assessing leaf pigment content and activity with a reflectometer. *New Phytol., 143*(1), 105–117.

Garcìa-Santillán, I. D., Montalvo, M., Guerrero, J. M., & Pajares, G. (2017). Automatic detection of curved and straight crop rows from images in maize fields. *Biosyst. Eng., 156*, 61–79.

Giant Gardening. (2018). *Measuring sunflower heads*. Available at: <`http://giantgardening.com/artcl_measuringsunflowers.php`>. (Accessed October 11, 2019)

Gitelson, A. A., Kaufman, Y. J., Stark, R., & Rundquist, D. (2002). Novel algorithms for remote estimation of vegetation fraction. *Remote Sens. Environ., 80*(1), 76–87.

Golbach, F., Kootstra, G., Damjanovic, S., Otten, G., & van de Zedde, R. (2016). Validation of plant part measurements using a 3D reconstruction method suitable for high-throughput seedling phenotyping. *Mach. Vision Appl., 27*(5), 663–680.

Goldstein, B. (2011). *Visualizing with ImageJ.* Available at:
<https://imagej.nih.gov/ij/docs/ImageJPrimerMakeVol27.pdf>. (Accessed
October 11, 2019)

Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing (4th edition).* Prentice
Hall, Inc. Upper Saddle River, NJ, USA.

Greenleaf, S. S., & Kremen, C. (2006). Wild bees enhance honey bees' pollination of hybrid
sunflower. *Proc. Natl. Acad. Sci. USA., 103*(37), 13890–13895.

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.

Grisso, R. D., Jasa, P. J., & Rolofson, D. E. (2002). Analysis of traffic patterns and yield
monitor data for field efficiency determination. *Appl. Eng. Agric., 18*(2), 171–178.

Grisso, R. D., Kocher, M. F., Adamchuk, V. I., Jasa, P. J., & Schroeder, M. A. (2004). Field
efficiency determination using traffic pattern indices. *Appl. Eng. Agric., 20*(5),
563–572.

Guerrero, J. M., Ruz, J. J., & Pajares, G. (2017). Crop rows and weeds detection in maize
fields applying a computer vision system based on geometry. *Comput. Electron. Agric.,
142,* 461–472.

Gunturk, B. K., Glotzbach, J., Altunbasak, Y., Schafer, R. W., & Mersereau, R. M. (2005).
Demosaicking: Color filter array interpolation. *IEEE Signal Process Mag., 22*(1),
44–54.

Haeghen, Y. V., Naeyaert, J. M. A. D., Lemahieu, I., & Philips, W. (2000). An imaging system with calibrated color image acquisition for use in dermatology. *IEEE Trans. Med. Imaging, 19*(7), 722–730.

Hague, T., Tillett, N. D., & Wheeler, H. (2006). Automated crop and weed monitoring in widely spaced cereals. *Precis. Agric., 7*(1), 21–32.

Hall, D., Dayoub, F., Perez, T., & McCool, C. (2018). A rapidly deployable classification system using visual data for the application of precision weed management. *Comput. Electron. Agric., 148*, 107–120.

Hirt, C., Filmer, M. S., & Featherstone, W. E. (2010). Comparison and validation of the recent freely available ASTER-GDEM ver1, SRTM ver4. 1 and GEODATA DEM-9S ver3 digital elevation models over Australia. *Aust. J. Earth Sci., 57*(3), 337–347.

Hough, P. V. C. (1962). *Method and means for recognizing complex patterns.* Google Patents. (US Patent 3,069,654)

Huete, A. R. (1988). A soil-adjusted vegetation index (SAVI). *Remote Sens. Environ., 25*(3), 295–309.

Hufkens, K., Friedl, M., Sonnentag, O., Braswell, B. H., Milliman, T., & Richardson, A. D. (2012). Linking near-surface and satellite remote sensing measurements of deciduous broadleaf forest phenology. *Remote Sens. Environ., 117*, 307–321.

Hufkens, K., Keenan, T. F., Flanagan, L. B., Scott, R. L., Bernacchi, C. J., Joo, E., . . . Richardson, A. D. (2016). Productivity of North American grasslands is increased under future climate scenarios despite rising aridity. *Nat. Clim. Change, 6*(7), 710–716.

Hunt, E. R., Cavigelli, M., Daughtry, C. S. T., Mcmurtrey, J. E., & Walthall, C. L. (2005). Evaluation of digital photography from model aircraft for remote sensing of crop biomass and nitrogen status. *Precis. Agric., 6*(4), 359–378.

Hunt, E. R., Daughtry, C. S. T., Eitel, J. U. H., & Long, D. S. (2011). Remote sensing leaf chlorophyll content using a visible band index. *Agron. J., 103*(4), 1090–1099.

Igathinathane, C. (1997). *Studies on drying and tempering of parboiled paddy and development of mathematical models for the processes*. [PhD dissertation] Kharagpur, India: Indian Institue of Technology, Kharagpur, Agricultural and Food Engineering Department.

Igathinathane, C., Melin, S., Sokhansanj, S., Bi, X., Lim, C. J., Pordesimo, L. O., & Columbus, E. P. (2009a). Machine vision based particle size and size distribution determination of airborne dust particles of wood and bark pellets. *Powder Technol., 196*(2), 202–212.

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

Igathinathane, C., Pordesimo, L. O., Columbus, E. P, Batchelor, W. D., & Sokhansanj, S. (2009c). Sieveless particle size distribution analysis of particulate materials through computer vision. *Comput. Electron. Agric., 66*(2), 147–158.

Igathinathane, C., & Ulusoy, U. (2016). Machine vision methods based particle size distribution of ball- and gyro-milled lignite and hard coal. *Powder Technol., 297*, 71–80.

Ilie, A., & Welch, G. (2005). Ensuring color consistency across multiple cameras. In *10th IEEE International Conference on Computer Vision (ICCV'05)* (Vol. 1, pp. 1268–1275).

Jenicka, S., & Suruliandi, A. (2014). A textural approach for land cover classification of remotely sensed image. *CSI Trans. ICT*, *2*(1), 1–9.

Jenicka, S., & Suruliandi, A. (2016). Distributed texture-based land cover classification algorithm using hidden markov model for multispectral data. *Surv. Rev.*, *48*(351), 430–437.

Ji, R., & Qi, L. (2011). Crop-row detection algorithm based on random Hough transformation. *Math. Comput. Modell.*, *54*(3-4), 1016–1020.

Jiang, G., Wang, Z., & Liu, H. (2015). Automatic detection of crop rows based on multi-ROIs. *Expert Syst. Appl.*, *42*(5), 2429–2441.

Jiang, Y., Li, C., Robertson, J. S., Sun, S., Xu, R., & Paterson, A. H. (2018). GPhenoVision: A ground mobile system with multi-modal imaging for field-based high throughput phenotyping of cotton. *Sci. Rep.*, *8*, 1-15.

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. (`https://doi.org/10.1016/j.rse.2017.06.007`)

Jordan, C. F. (1969). Derivation of leaf-area index from quality of light on the forest floor. *Ecology*, *50*(4), 663–666.

Joshi, N., Wilburn, B., Vaish, V., Levoy, M. L., & Horowitz, M. (2005). *Automatic color calibration for large camera arrays*. [Department of Computer Science and Engineering], University of California, San Diego.

Kachman, S. D., & Smith, J. A. (1995). Alternative measures of accuracy in plant spacing for planters using single seed metering. *Trans. ASAE*, *38*(2), 379–387.

Kagarlitsky, S., Moses, Y., & Hel-Or, Y. (2009). Piecewise-consistent color mappings of images acquired under various conditions. In *IEEE 12th International Conference on Computer Vision* (pp. 2311–2318).

Kandel, H. (2010). *Soybean production.* Available at: `<https://www.ag.ndsu.edu/extensionentomology/field-crops-insect-pests/Documents/soybean/a-1172-soybean-production-field-guide>`. (Accessed October 11, 2019)

Kandel, H. (2018). *Estimating sunflower yield, North Dakota State University.* Available at: `<https://www.ag.ndsu.edu/cpr/plant-science/estimating-sunflower-yield-8-16-12>`. (Accessed October 11, 2019)

Karcher, D. E., & Richardson, M. D. (2003). Quantifying turfgrass color using digital image analysis. *Crop Sci.*, *43*(3), 943–951.

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* (Vol. 2, pp. b1079–b1083). `(10.1109/AIM.2003.1225492)`

Kaufman, Y. J., & Tanre, D. (1992). Atmospherically resistant vegetation index (ARVI) for EOS-MODIS. *IEEE Trans. Geosci. Remote Sens.*, *30*(2), 261–270.

Keenan, T. F., Gray, J., Friedl, M. A., Toomey, M., Bohrer, G., Hollinger, D. Y., ... others (2014). Net carbon uptake has increased through warming-induced changes in temperate forest phenology. *Nat. Clim. Change*, *4*(7), 598–604.

Khan, M. F. J., Nag, T. C., Igathinathane, C., Osuagwu, U. L., & Rubini, M. (2015). A new method of detecting changes in corneal health in response to toxic insults. *Micron*, *78*, 45–53.

Khanal, S., Fulton, J., & Shearer, S. (2017). An overview of current and potential applications of thermal remote sensing in precision agriculture. *Comput. Electron. Agric.*, *139*, 22–32.

Kriegler, F. J., Malila, W. A., Nalepka, R. F., & Richardson, W. (1969). Preprocessing transformations and their effects on multispectral recognition. In *Remote Sensing of Environment, VI* (p. 97).

Kross, A., McNairn, H., Lapen, D., Sunohara, M., & Champagne, C. (2015). Assessment of RapidEye vegetation indices for estimation of leaf area index and biomass in corn and soybean crops. *Int. J. Appl. Earth Obs. Geoinf.*, *34*, 235–248.

Kulbaba, M. W., & Worley, A. C. (2012). Selection on floral design in *Polemonium brandegeei* (Polemoniaceae): female and male fitness under hawkmoth pollination. *Evolution*, *66*(5), 1344–1359. (https://doi.org/10.1111/j.1558-5646.2011.01536.x)

Lam, E. Y. (2005). Combining gray world and retinex theory for automatic white balance in digital photography. In *Proceedings of the Ninth International Symposium on Consumer Electronics* (pp. 134–139).

Lamb, A. (2016). *Drone-based plant count increases accountability*. Available at: <https://blog.dronedeploy.com/ drone-based-plant-count-increases-accountability-d48ed60033b6>. (Accessed October 11, 2019)

Landset, S., Khoshgoftaar, T. M., Richter, A. N., & Hasanin, T. (2015). A survey of open

    source tools for machine learning with big data in the hadoop ecosystem. *J. Big Data*,

    2(1), 24.

Lauer, J. G., & Rankin, M. (2004). Corn response to within row plant spacing variation.

    Agron. J., *96*(5), 1464–1468.

Lee, S.-Y., & Ortega, A. (2001). A novel approach of image compression in digital cameras

    with a Bayer color filter array. In *Proceedings of 2001 International Conference on*

    Image Processing (Vol. 3, pp. 482–485).

    (https://doi.org/10.1109/icip.2001.958156)

Levin, R. L., & Levin, G. V. (2004). Solving the color calibration problem of martian lander

    images. In *Instruments, Methods, and Missions for Astrobiology VII* (Vol. 5163, pp.

    158–171).

Li, P., Luo, Y., Bernhardt, P., Kou, Y., & Perner, H. (2008). Pollination of *Cypripedium*

    plectrochilum (Orchidaceae) by *Lasioglossum* spp. (Halictidae): The roles of generalist

    attractants versus restrictive floral architecture. *Plant Biol.*, *10*(2), 220–230.

    (https://doi.org/10.1111/j.1438-8677.2007.00020.x)

Li, Y., Chen, D., Walker, C., & Angus, J. (2010). Estimating the nitrogen status of crops

    using a digital camera. *Field Crops Res.*, *118*(3), 221–227.

Liang, W., Kirk, K. R., & Greene, J. K. (2018). Estimation of soybean leaf area, edge, and

    defoliation using color image analysis. *Comput. Electron. Agric.*, *150*, 41–51.

    (https://doi.org/10.1016/j.compag.2018.03.021)

Liu, W., Tollenaar, M., Stewart, G., & Deen, W. (2004). Impact of planter type, planting speed, and tillage on stand uniformity and yield of corn. *Agronomy J.*, *96*(6), 1668–1672.

Maharlooei, M., Sivarajan, S., Bajwa, S. G., Harmon, J. P., & Nowatzki, J. (2017). Detection of soybean aphids in a greenhouse using an image processing technique. *Comput. Electron. Agric.*, *132*, 63–70.

Mallinger, R. E., & Prasifka, J. R. (2017a). Bee visitation rates to cultivated sunflowers increase with the amount and accessibility of nectar sugars. *J. Appl. Entomol.*, *141*, 561–573.

Mallinger, R. E., & Prasifka, J. R. (2017b). Benefits of insect pollination to confection sunflowers differ across plant genotypes. *Crop Sci.*, *57*(6), 3264–3272.

McCamy, C. S., Marcus, H., & Davidson, J. G. (1976). A color-rendition chart. *J. App. Photog. Eng*, *2*(3), 95–99.

McCarthy, C. L., Hancock, N. H., & Raine, S. R. (2010). Applied machine vision of plants: A review with implications for field deployment in automated farming operations. *Intell. Serv. Rob.*, *3*(4), 209–217. (`https://doi.org/10.1007/s11370-010-0075-2`)

Melaas, E. K., Sulla-Menashe, D., Gray, J. M., Black, T. A., Morin, T. H., Richardson, A. D., & Friedl, M. A. (2016). Multisite analysis of land surface phenology in North American temperate and boreal deciduous forests from Landsat. *Remote Sens. Environ.*, *186*, 452–464. (`https://doi.org/10.1016/j.rse.2016.09.014`)

Meyer, G., & Neto, J. C. (2008). Verification of color vegetation indices for automated crop imaging applications. *Comput. Electron. Agric.*, *63*(2), 282–293.

Meyer, G. E. (2011). Machine vision identification of plants. In *Recent Trends for Enhancing the Diversity and Quality of Soybean Products* (pp. 401–419). IntechOpen.

Mirzabe, A. H., Chegini, G. R., & Khazaei, J. (2018). Sunflower petals: Some physical properties and modeling distribution of their number, dimensions, and mass. *Inf. Process. Agric.*, *5*(2), 185–198.

Mitchell, R., Karron, J., Holmquist, K., & Bell, J. (2004). The influence of mimulus ringens floral display size on pollinator visitation patterns. *Funct. Ecol.*, *18*(1), 116–124.

Mounir, L., Michael, M. B., & Douglas, E. J. (2001). Spatially located platform and aerial photography for documentation of grazing impacts on wheat. *Geocarto Int.*, *16*(1), 65–70.

Moyers, B. T., Owens, G. L., Baute, G. J., & Rieseberg, L. H. (2017). The genetic architecture of UV floral patterning in sunflower. *Ann. Bot.*, *120*, 39–50.

Mulla, D. J. (2013). Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. *Biosyst. Eng.*, *114*(4), 358–371.

Mutterer, J., & Rasband, W. (2015). *ImageJ macro language programmer's reference guide v1. 46d.* Available at: `<https://imagej.nih.gov/ij/docs/macro_reference_guide.pdf>`. (AccessedOctober 11, 2019)

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

Nielsen, R. (2004). *Effect of plant spacing variability on corn grain yield.* Available at: `<https://www.agry.purdue.edu/ext/corn/research/psv/update2004.html>`. (Accessed October 11, 2019)

Nieuwenhuizen, A. T., Tang, L., Hofstee, J. W., Müller, J., & Van Henten, E. J. (2007). Colour based detection of volunteer potatoes as weeds in sugar beet fields using machine vision. *Precis. Agric.*, *8*(6), 267–278.

O'Connell, J., Bradter, U., & Benton, T. G. (2015). Wide-area mapping of small-scale features in agricultural landscapes using airborne remote sensing. *ISPRS J. Photogramm. Remote Sens.*, *109*, 165–177. (https://doi.org/10.1016/j.isprsjprs.2015.09.007)

Okalebo, J., Das Choudhury, S., Awada, T., Suyker, A., LeBauer, D., Newcomb, M., & Ward, R. (2017). Application of near-surface remote sensing and computer algorithms in evaluating impacts of agroecosystem management on *Zea mays* (corn) phenological development in the platte river-high plains aquifer long term agroecosystem research network field sites. In *Agu Fall Meeting Abstracts.*

Panning, J. W., Kocher, M. F., Smith, J. A., & Kachman, S. D. (2000). Laboratory and field testing of seed spacing uniformity for sugarbeet planters. *Appl. Eng. Agric.*, *16*(1), 7–13. (https://doi.org/10.13031/2013.4985)

Park, C., Huang, J. Z., Ji, J. X., & Ding, Y. (2013). Segmentation, inference and classification of partially overlapping nanoparticles. *IEEE Trans. Pattern Anal. Mach. Intell.*, *35*(3), 669–681. (https://doi.org/10.1109/tpami.2012.163)

Parker, F. D. (1981). Sunflower pollination: Abundance, diversity and seasonality of bees and their effect on seed yields. *J. Apic. Res.*, *20*(1), 49–61. (https://doi.org/10.1080/00218839.1981.11100473)

Pereira, L. F. S., Barbon, S., Valous, N. A., & Barbin, D. F. (2018). Predicting the ripening of papaya fruit with digital imaging and random forests. *Comput. Electron. Agric.*, *145*, 76–82.

Pham-Delegue, M., Etievant, P., Guichard, E., Marilleau, R., Douault, P., Chauffaille, J., & Masson, C. (1990). Chemicals involved in honeybee-sunflower relationship. *J. Chem. Ecol.*, *16*(11), 3053–3065.

Phenocam. (2019). *Phenocam close-up site–willamettepoplar.* Available at: <https://phenocam.sr.unh.edu/webcam/sites/willamettepoplar/>. (Accessed October 11, 2019)

Philipp, I., & Rath, T. (2002). Improving plant discrimination in image processing by use of different colour space transformations. *Comput. Electron. Agric.*, *35*(1), 1–15.

Pointer, M. R., & Attridge, G. G. (1998). The number of discernable colours. *Color Res. Appl.*, *23*, 52–54.

Poncet, A., Fulton, J., Port, K., McDonald, T., & Pate, G. (2016). *Optimizing field traffic patters to improve machinery efficiency: Path planning using guidance lines.* Available at: <https://ohioline.osu.edu/factsheet/fabe-5531>. (Accessed October 11, 2019)

Porikli, F. (2003). Inter-camera color calibration by correlation model function. In *2003 International Conference on Image Processing* (Vol. 2, p. 133).

Portlas, Z. M., Tetlie, J. R., Prischmann-Voldseth, D., Hulke, B. S., & Prasifka, J. R. (2018). Variation in floret size explains differences in wild bee visitation to cultivated sunflowers. *Plant Genet. Resour.*, 1–6.

Posey, A., Katayama, R., & Burleigh, J. (1986). The abundance and daily visitation patterns of bees (hymenoptera: Apoidea) on oilseed sunflower, *Helianthus annuus* L., in southeastern Arkansas. *J. Kansas Entomol. Soc.*, 494–499.

Pothula, A. K., Igathinathane, C., & Kronberg, S. (2015). Profile based image analysis for identification of chopped biomass stem nodes and internodes. *Ind. Crops Prod.*, *70*, 374–382.

Pothula, A. K., Igathinathane, C., Kronberg, S., & Hendrickson, J. (2014). Digital image processing based identification of nodes and internodes of chopped biomass stems. *Comput. Electron. Agric.*, *105*, 54–65.

Prasad, S., Peddoju, S. K., & Ghosh, D. (2016). Multi-resolution mobile vision system for plant leaf disease diagnosis. *Signal Image Video Process.*, *10*(2), 379–388.

Prasad, S., & Singh, P. P. (2017). Vision system for medicinal plant leaf acquisition and analysis. In *Applications of Cognitive Computing Systems and IBM Watson* (pp. 37–45). Springer.

Prasifka, J. R., & Hulke, B. S. (2016). Relative susceptibility of sunflower maintainer lines and resistance sources to natural infestations of the banded sunflower moth (Lepidoptera: Tortricidae). *Can. Entomol.*, *148*(6), 736–741.

R Core Team. (2018). *R: A language and environment for statistical computing.* Available at: <http://www.R-project.org/>. (Accessed October 11, 2019)

Raper, T. B., Oosterhuis, D. M., Siddons, U., Purcell, L. C., & Mozaffari, M. (2012). Effectiveness of the dark green color index in determining cotton nitrogen status from multiple camera angles. *Int. J. Appl. Sci. Technol.*, *2*(1).

Rasband, W. (2018). *ImageJ, US National Institutes of Health, Bethesda, MD, USA.* Available

at: <https://imagej.nih.gov/ij/>. (Accessed October 11, 2019)

Rele, A. S., & Mohile, R. B. (2003). Effect of mineral oil, sunflower oil, and coconut oil on

prevention of hair damage. *J. Cosmet. Sci.*, *54*, 175–92.

Richardson, A. D., Bailey, A. S., Denny, E. G., Martin, C. W., & O'Keefe, J. (2006). Phenology

of a northern hardwood forest canopy. *Global Change Biol.*, *12*(7), 1174–1188.

Richardson, A. D., Braswell, B. H., Hollinger, D. Y., Jenkins, J. P., & Ollinger, S. V. (2009).

Near-surface remote sensing of spatial and temporal variation in canopy phenology.

Ecol. Appl., *19*(6), 1417–1428.

Richardson, A. D., Hufkens, K., Milliman, T., Aubrecht, D. M., Chen, M., Gray, J. M., . . .

others (2017a). Phenocam dataset V1.0: Vegetation Phenology from Digital Camera

Imagery, 2000–2015. *ORNL DAAC: Oak Ridge, TN, USA.*

Richardson, A. D., Hufkens, K., Milliman, T., Aubrecht, D. M., Chen, M., Gray, J. M., . . .

others (2018b). Tracking vegetation phenology across diverse North American

biomes using PhenoCam imagery. *Sci. Data*, *5*, 180028 (1–24).

(https://doi.org/10.1038/sdata.2018.28)

Richardson, A. D., Hufkens, K., Milliman, T., Aubrecht, D. M., Furze, M. E., Seyednasrollah,

B., . . . Hanson, P. J. (2018c). *SPRUCE Vegetation Phenology in Experimental Plots

from Phenocam Imagery, 2015–2016* (Tech. Rep.). Oak Ridge National Labs

Terrestrial Ecosystem Science Scientific Focus Area), Oak Ridge, TN, USA.

Richardson, A. D., Hufkens, K., Milliman, T., & Frolking, S. (2018d). Intercomparison of

phenological transition dates derived from the PhenoCam Dataset V1.0 and MODIS

satellite remote sensing. *Sci. Rep.*, *8*(1), 5679.

Richardson, A. D., Jenkins, J. P., Braswell, B. H., Hollinger, D. Y., Ollinger, S. V., & Smith, M. L. (2007). Use of digital webcam images to track spring green-up in a deciduous broadleaf forest. *Oecologia*, *152*(2), 323–334.

Richardson, A. D., Klosterman, S., & Toomey, M. (2013b). Near-surface sensor-derived phenology. In *Phenology: An Integrative Environmental Science* (pp. 413–430). Springer.

Richardson, A. D., Seyednasrollah, B., Hufkens, K., & Milliman, T. (2017b). *Phenocam installation instructions (revised Dec. 2017)*. Available at: <https://phenocam.sr.unh.edu/pdf/PhenoCam_Install_Instructions.pdf>. (Accessed October 11, 2019)

Richardson(AJ), A. J., & Wiegand, C. L. (1977). Distinguishing vegetation from soil background information. *Photogramm. Eng. Remote Sens.*, *43*(12), 1541–1552.

RStudio Team. (2018). RStudio: Integrated development environment for R. Ver. 1.2.1335. [Computer software manual]. Boston, MA. Retrieved from http://www.rstudio.com/

Sakamoto, T., Wardlow, B. D., Gitelson, A. A., Verma, S. B., Suyker, A. E., & Arkebauer, T. J. (2010). A two-step filtering approach for detecting maize and soybean phenology with time-series MODIS data. *Remote Sens. Environ.*, *114*(10), 2146–2159.

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.

Sankaran, S., Wang, M., & Vandemark, G. J. (2016). Image-based rapid phenotyping of chickpeas seed size. *Eng. Agric. Environ. Food*, *9*(1), 50–55.

Scharf, P. C., & Lory, J. A. (2002). Calibrating corn color from aerial photographs to predict

    sidedress nitrogen need. *Agron. J.*, *94*(3), 397–404.

Scharf, P. C., Shannon, D. K., Palm, H. L., Sudduth, K. A., Drummond, S. T., Kitchen, N. R.,

    ... Oliveira, L. F. (2011). Sensor-based nitrogen applications out-performed

    producer-chosen rates for corn in on-farm demonstrations. *Agron. J.*, *103*(6),

    1683–1691.

Schimmelpfennig, D. (2016). *Farm profits and adoption of precision agriculture* (Tech.

    Rep.). ERR-217, US Department of Agriculture, Economic Research Service.

Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., ... others

    (2012). Fiji: An open-source platform for biological-image analysis. *Nat. Methods*,

    9(7), 676–682. (https://doi.org/10.1038/nmeth.2019)

Schirrmann, M., Hamdorf, A., Garz, A., Ustyuzhanin, A., & Dammer, K. H. (2016).

    Estimating wheat biomass by combining image clustering with crop height. *Comput.*

    Electron. Agric., *121*, 374–384.

    (https://doi.org/10.1016/j.compag.2016.01.007)

Schmid-Hempel, P., & Speiser, B. (1988). Effects of inflorescence size on pollination in

    Epilobium angustifolium. *Oikos*, 98–104.

Shang, J., Liu, J., Huffman, T. C., Qian, B., Pattey, E., Wang, J., ... others (2014).

    Estimating plant area index for monitoring crop growth dynamics using Landsat-8

    and RapidEye images. *J. Appl. Remote Sens.*, *8*(1), 085196 (1–12).

    (https://doi.org/10.1117/1.jrs.8.085196)

Shi, Y., Thomasson, J. A., Murray, S. C., Pugh, N. A., Rooney, W. L., Shafian, S., ... others

    (2016). Unmanned aerial vehicles for high-throughput phenotyping and agronomic

research. *PloS One, 11*(7), e0159781 (1–26).
(`http://datadryad.org/review?doi=doi:10.5061/dryad.65m87`)

Shrestha, A., & Fidelibus, M. (2005). Grapevine row orientation affects light environment, growth, and development of black nightshade (*Solanum nigrum*). *Weed Sci., 53*(6), 802–812.

Shrestha, D. S., & Steward, B. L. (2003). Automatic corn plant population measurement using machine vision. *Trans. ASAE, 46*(2), 559–565.
(`https://doi.org/10.13031/2013.12945`)

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.

Sidhu, H., Wiesenborn, D., Johnson, B., Monono, E., & Eriksmoen, E. (2019). Coating of hulled seeds improved field plantability and grain yield of extra-large confectionary sunflower achenes. *Crop Sci., 59*, 1–9.

Simard, M., Pinto, N., Fisher, J. B., & Baccini, A. (2011). Mapping forest canopy height globally with spaceborne lidar. *J. Geophys. Res. Biogeosci., 116*(G4), 1–12.

Singh, V., & Misra, A. K. (2017). Detection of plant leaf diseases using image segmentation and soft computing techniques. *Comput. Electron. Agric., 4*, 41–49.

Škorić, D. (1992). Achievements and future directions of sunflower breeding. *Field Crops Res., 30*(3-4), 231–270.

Søgaard, H. T., & Olsen, H. J. (2003). Determination of crop rows by image analysis without segmentation. *Comput. Electron. Agric., 38*(2), 141–158.

Sonnentag, O., Hufkens, K., Teshera-Sterne, C., Young, A., Friedl, M., Braswell, B. H., . . . Richardson, A. D. (2012). Digital repeat photography for phenological research in forest ecosystems. *Agric. For. Meteorol.*, *152*, 159–177.

Stowe, K. D. (2018). *Evaluating plant stands.* Available at: `<http://ncsoy.org/article/evaluating-plant-stands/>`. (Accessed October 11, 2019)

Subhashree, S. N., Sunoj, S., Xue, J., & Bora, G. C. (2017). Quantification of browning in apples using colour and textural features by image analysis. *Food Qual. Saf.*, *1*(3), 221–226.

Sunoj, S., Igathinathane, C., & Hendrickson, J. (2016). Monitoring plant phenology using phenocam: A review. In *ASABE Paper No: 162461829* (pp. 1–9).

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., 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.

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

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

Tillett, R. (1991). Image analysis for agricultural processes: A review of potential opportunities. *J. Agric. Eng. Res.*, *50*, 247–258.

Torres-Sánchez, J., Peña, J. M., de Castro, A. I., & López-Granados, F. (2014). Multi-temporal mapping of the vegetation fraction in early-season wheat fields using images from UAV. *Comput. Electron. Agric.*, *103*, 104–113.

Tucker, C. J. (1979). Red and photographic infrared linear combinations for monitoring vegetation. *Remote Sens. Environ.*, *8*(2), 127–150.

van Iersel, W., Straatsma, M., Addink, E., & Middelkoop, H. (2018). Monitoring height and greenness of non-woody floodplain vegetation with UAV time series. *ISPRS J. Photogramm. Remote Sens.*, *141*, 112–123. (https://doi.org/10.1016/j.isprsjprs.2018.04.011)

Varela, S., Dhodda, P. R., Hsu, W. H., Prasad, P., Assefa, Y., Peralta, N. R., ... 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 (1–14).

Vega, F. A., Ramírez, F. C., Saiz, M. P., & Rosúa, F. O. (2015). Multi-temporal imaging using an unmanned aerial vehicle for monitoring a sunflower crop. *Biosyst. Eng.*, *132*, 19–27.

Verger, A., Vigneau, N., Chéron, C., Gilliot, J.-M., Comar, A., & Baret, F. (2014). Green area index from an unmanned aerial system over wheat and rapeseed crops. *Remote Sens. Environ.*, *152*, 654–664.

Verrelst, J., Schaepman, M. E., Koetz, B., & Kneubühler, M. (2008). Angular sensitivity analysis of vegetation indices derived from CHRIS/PROBA data. *Remote Sens. Environ.*, *112*(5), 2341–2353.

Vollmann, J., Walter, H., Sato, T., & Schweiger, P. (2011). Digital image analysis and chlorophyll metering for phenotyping the effects of nodulation in soybean. *Comput. Electron. Agric.*, *75*(1), 190–195.

Wan, P., Toudeshki, A., Tan, H., & Ehsani, R. (2018). A methodology for fresh tomato maturity detection using computer vision. *Comput. Electron. Agric.*, *146*, 43–50.

White, M. A., de Beurs, K. M., Didan, K., Inouye, D. W., Richardson, A. D., Jensen, O. P., . . . others (2009). Intercomparison, interpretation, and assessment of spring phenology in North America estimated from remote sensing for 1982–2006. *Global Change Biol.*, *15*(10), 2335–2359.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. Retrieved from `https://ggplot2.tidyverse.org`

Wikipedia. (2018). *Image processing software.* Available at: <`https://en.wikipedia.org/wiki/Category:Image_processing_software`>. (Accessed October 11, 2019)

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

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* (Vol. 1836, pp. 208–220).

X-Rite. (2018). *Colorimetric values for ColorChecker family of targets.* Available at: <

    `http://xritephoto.com/`

    `ph_product_overview.aspx?ID=824&Action=Support&SupportID=5159` >.

    (Accessed October 11, 2019)

Xie, Y., Sha, Z., & Yu, M. (2008). Remote sensing imagery in vegetation mapping: A review.

    J. Plant Ecolog., 1(1), 9–23.

Xu, Q., Hou, Z., & Tokola, T. (2012). Relative radiometric correction of multi-temporal

    ALOS AVNIR-2 data for the estimation of forest attributes. *ISPRS J. Photogramm.*

    Remote Sens., 68, 69–78. (`https://doi.org/10.1016/j.isprsjprs.2011.12.008`)

Xue, J., & Su, B. (2017). Significant remote sensing vegetation indices: A review of

    developments and applications. *J. Sens.*, 1353691 (1–17).

    (`https://doi.org/10.1155/2017/1353691`)

Yan, D., Scott, R. L., Moore, D. J. P., Biederman, J. A., & Smith, W. K. (2019).

    Understanding the relationship between vegetation greenness and productivity across

    dryland ecosystems through the integration of PhenoCam, satellite, and eddy

    covariance data. *Remote Sens. Environ., 223*, 50–62.

Yuntang, L. W. H. (2012). Image segmentation method based on YUV color space for

    tomato harvesting robot. *Trans. Chin. Soc. Agric. Mach., 12*, 176–180.

Zafari, S., Eerola, T., Sampo, J., Kälviäinen, H., & Haario, H. (2015). Segmentation of overlapping elliptical objects in silhouette images. *IEEE Trans. Image Process.*, *24*(12), 5942–5952.

Zeng, L., Wardlow, B. D., Wang, R., Shan, J., Tadesse, T., Hayes, M. J., & Li, D. (2016). A hybrid approach for detecting corn and soybean phenology with time-series MODIS data. *Remote Sens. Environ.*, *181*, 237–250.

Zhang, S., Wu, X., You, Z., & Zhang, L. (2017). Leaf image based cucumber disease recognition using sparse representation classification. *Comput. Electron. Agric.*, *134*, 135–141.

Zhang, X., Friedl, M. A., & Schaaf, C. B. (2006). Global vegetation phenology from Moderate Resolution Imaging Spectroradiometer (MODIS): Evaluation of global patterns and comparison with in situ measurements. *J. Geophys. Res., 111, G04017*(G4), 1–14.

Zhao, B., Zhang, J., Yang, C., Zhou, G., Ding, Y., Shi, Y., . . . 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 (1-17). (https://doi.org/10.3389/fpls.2018.01362)

Zhao, Y., Gong, L., Huang, Y., & Liu, C. (2016). A review of key techniques of vision-based control for harvesting robot. *Comput. Electron. Agric.*, *127*, 311–323.

Zhong, L., Hu, L., Yu, L., Gong, P., & Biging, G. S. (2016). Automated mapping of soybean and corn using phenology. *ISPRS J. Photogramm. Remote Sens.*, *119*, 151–164.

APPENDIX A. PSEUDOCODE OF THE DEVELOPED

SUNFLOWER DIMENSION MEASUREMENT PLUGIN

This appendix provides the pseudocode of the sunflower floral dimensions measurement plugin with appropriate sections, task specific Java methods, package imports, and comments are presented subsequently for following the overall algorithm and better understanding of the processes.

```java
/******************************************************************************
*              Sunflower floral dimensions measurement plugin
*          Developed by S. Sunoj & C. Igathinathane; ABEN, NDSU
******************************************************************************/

// Other standard ImageJ plugin imports
import java.awt.geom.Point2D;
import imagingbook.lib.interpolation.InterpolationMethod;
import imagingbook.pub.geometry.mappings.Mapping;
import imagingbook.pub.geometry.mappings.linear.ProjectiveMapping;
/**************************************************************************/
public class Sunflower_dimensions_plugin implements PlugIn {

  public void run(String arg) {
   Read_user_inputs_and_image; // Generic dialog box - front panel of plugin
   int[][] cmarkers = Find_circular_markers (Input_image);
   ImagePlus cor_image = Projective_transform(Input_image, cmarkers);
   ImagePlus bw_image = binarize_image (cor_image);
   ImagePlus wp_image = draw_wrapping_polygon (bw_image);
   double direct_mean_dim = pixel_march(bw_image, n_measurements); // direct method
   double wp_mean_dim = pixel_march(wp_image, n_measurements); // wrapping polygon
   float[] shapes_bw = get_shape_param (bw_image); // e.g., area, circularity...
   float[] shapes_wp = get_shape_param (wp_image);
}

/****************** Developed Java methods *******************/
/***************** Identifying circular markers *****************/
  public int[][] Find_circular_markers (Input_image){
     IJ.run(Input_image, "8-bit", "");
```

```
        IJ.run(Input_image, "Minimum", ""); // minimum thresholding scheme
        IJ.run("Set Measurements...", "area centroid redirect=None decimal=3");
        IJ.run(imp, "Analyze Particles...", "size=0-2000 circularity = 0.85-1.00
            show=mask"); // identifies only markers

        rt = Analyzer.getResultsTable();
        float[] x_cent = rt.getColumn(ResultsTable.X_CENTROID);
        float[] y_cent = rt.getColumn(ResultsTable.Y_CENTROID);

        // convert float[] to int[4][2] using a ''for'' loop
    return Cent_XY;
    }


/********************* Distortion correction *********************/
    public ImagePlus Projective_transform(Input_image, Cent_XY){
        Point2D p0 = new Point(Cent_XY[0][0], Cent_XY[0][1]);
        Point2D p1 = new Point(Cent_XY[1][0], Cent_XY[1][1]);
        Point2D p2 = new Point(Cent_XY[2][0], Cent_XY[2][1]);
        Point2D p3 = new Point(Cent_XY[3][0], Cent_XY[3][1]);

        // Target was fixed as 3000 by 3000 pixels with 100 pixel margin
        Point2D q0 = new Point(100, 100);
        Point2D q1 = new Point(3100, 100);
        Point2D q2 = new Point(3100, 3100);
        Point2D q3 = new Point(100, 3100);

          // Methods from ''imagingbook'' imports
        Mapping map = new ProjectiveMapping(p0, p1, p2, p3, q0, q1, q2, q3);
        map.applyTo(ip2, InterpolationMethod.Bicubic);

        ImagePlus cor_image = new ImagePlus ("Projective transformation", ip2);

    return cor_image;
    }


/********************* Image binarization *********************/
    public ImagePlus binarize_image (cor_image){
        // Sunflower head (SH) --- ''default'' thresholding scheme
        // Sunflower disc (SD) --- ''shanbhag'' thresholding scheme

        // Obtaining ray florets image
        ImageCalculator ic = new ImageCalculator();
        ImagePlus RF = ic.run("Ray florets image", SH, SD); // ray florets

    return RF;
    }
```

```
/******************** Drawing wrapping polygon ********************/
  public ImagePlus draw_wrapping_polygon (bw_image){
      int[][] obj_cent = get_object_centroids (bw_image);
      int[] obj_outline = get_obj_outline (bw_image); // use ''Wand()''
      double[] rad_dist = get_radial_distance (obj_cent, obj_outline);

      if (rad_dist[i-1] < rad_dist[i] && rad_dist[i] > rad_dist[i+1]){
         // Select as a rayfloret tip --- (xpts, ypts)
      }

      // Draw a polygonal ROI with selected tip points and fill with black
      PolygonRoi proi = new PolygonRoi(xpts, ypts, npts, Roi.POLYGON);
      WP_img.setRoi(proi);
      IJ.setColor(Color.BLACK);
      IJ.run(WP_img, "fill...", "");

  return WP_img;
  }

/******************** Pixel-march dimensions ********************/
  public double pixel_march (bw_image, n_measurements){
      ang_step = 180/n_measurements;

      double ang = 0.0;
      for (int i = 0; i < n_measurements; i++){
         if (ang < 45 || ang > 135){
            // x_increment
         } else {
            // y_increment
         }
      // March outwards from centroid and find object edge point
      // Calculate distance and store it in a double[] "dists".
      ang = ang + ang_step;
      }

      DescriptiveStatistics ds = new DescriptiveStatistics(dists);
      mean_dim = ds.getMean();

  return mean_dim;
  }

/******************** Obtain object shape parameters ********************/
  public float[] get_shape_param (bw_image){
      IJ.run("Set Measurements...", "area centroid perimeter bounding fit shape
         feret's redirect=None decimal=3");
```

```
    IJ.run(imp, "Analyze Particles...", "size=2000-Infinity show=mask"); //
        Records all listed shape parameters

    rt = Analyzer.getResultsTable();
    float[] area = rt.getColumn(ResultsTable.AREA);
    float[] perim = rt.getColumn(ResultsTable.PERIMETER);
    float[] circularity = rt.getColumn(ResultsTable.CIRCULARITY);
    float[] soliditiy = rt.getColumn(ResultsTable.SOLIDITY);

  return shape_values;
  }

} // End class
```

APPENDIX B. DATA AND PSEUDOCODE OF THE

DEVELOPED COLOR CALIBRATION PLUGIN

This appendix provides (i) some result data obtained by running the plugin, and (ii) pseudocode of the ColorCal plugin. The data provided were used to generate the modified total error plot (fig. B1), calibration performance index plot (fig. B2), and intensity variations plot (fig. B3). A common spreadsheet program was used to generate the plots. The corresponding figures are also reproduced for ready reference. The pseudocode is also provided in Java that briefly describes the major components of the ColorCal plugin.

B1. Data used for Modified Total Error in Calibrated Images by Different Color Schemes in Dark and Light Conditions (Fig. 4.7 - Reproduced as Fig. B1)

The equation shown in the figure:

$$e_{\mathrm{m}} = \frac{\sum_{i=1}^{N=24} \sqrt{(R_S - R_i)^2 + (G_S - G_i)^2 + (B_S - B_i)^2}}{N \times (MD_R + MD_G + MD_B)}$$
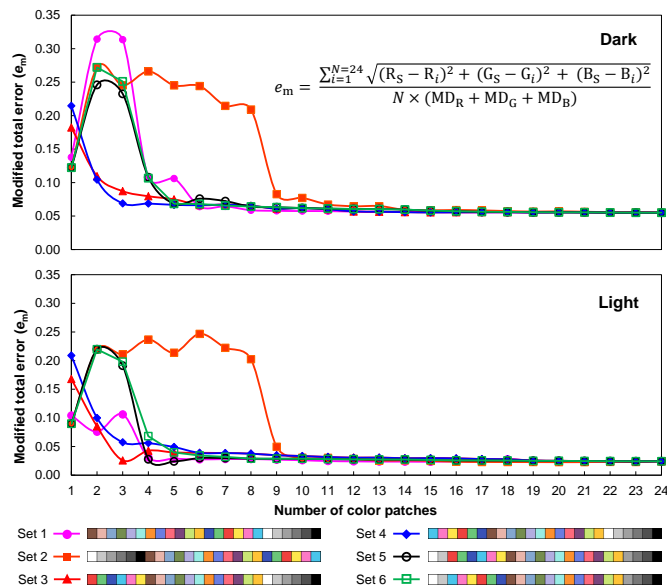
Figure B1. **Variations in modified total error in calibrated images by different color schemes in dark and light conditions.**

Table B1. Data used for variations in modified total error in calibrated images by different color schemes in dark and light conditions.

| Lighting condition	Number of patches	Modified total error (e_m)					
		Set 1	Set 2	Set 3s	Set 4	Set 5	Set6
Dark	1	0.14	0.12	0.18	0.21	0.12	0.12
	2	0.31	0.27	0.11	0.10	0.25	0.27
	3	0.31	0.25	0.09	0.07	0.23	0.25
	4	0.11	0.27	0.08	0.07	0.11	0.11
	5	0.11	0.25	0.08	0.07	0.07	0.07
	6	0.06	0.24	0.07	0.07	0.08	0.07
	7	0.06	0.21	0.07	0.07	0.07	0.07
	8	0.06	0.21	0.07	0.07	0.06	0.06
	9	0.06	0.08	0.06	0.06	0.06	0.06
	10	0.06	0.08	0.06	0.06	0.06	0.06
	11	0.06	0.07	0.06	0.06	0.06	0.06
	12	0.06	0.06	0.06	0.06	0.06	0.06
	13	0.06	0.06	0.06	0.06	0.06	0.06
	14	0.06	0.06	0.06	0.06	0.06	0.06
	15	0.06	0.06	0.06	0.06	0.06	0.06
	16	0.06	0.06	0.06	0.06	0.06	0.06
	17	0.06	0.06	0.06	0.06	0.06	0.06
	18	0.06	0.06	0.06	0.06	0.06	0.06
	19	0.06	0.06	0.06	0.06	0.06	0.06
	20	0.06	0.06	0.06	0.06	0.06	0.06
	21	0.06	0.06	0.06	0.06	0.06	0.06
	22	0.06	0.06	0.06	0.06	0.06	0.06
	23	0.06	0.06	0.06	0.06	0.06	0.06
	24	0.06	0.06	0.06	0.06	0.06	0.06
Light	1	0.10	0.09	0.17	0.21	0.09	0.09
	2	0.08	0.22	0.09	0.10	0.22	0.22
	3	0.11	0.21	0.03	0.06	0.19	0.20
	4	0.03	0.24	0.04	0.06	0.03	0.07
	5	0.03	0.21	0.04	0.05	0.02	0.04
	6	0.03	0.25	0.04	0.04	0.03	0.03
	7	0.03	0.22	0.04	0.04	0.03	0.03
	8	0.03	0.20	0.04	0.04	0.03	0.03
	9	0.03	0.05	0.03	0.03	0.03	0.03
	10	0.03	0.03	0.03	0.03	0.03	0.03
	11	0.02	0.03	0.03	0.03	0.03	0.03
	12	0.02	0.03	0.03	0.03	0.03	0.03
	13	0.02	0.03	0.03	0.03	0.03	0.03
	14	0.02	0.03	0.03	0.03	0.03	0.03
	15	0.02	0.02	0.03	0.03	0.03	0.03
	16	0.02	0.02	0.03	0.03	0.03	0.03
	17	0.02	0.02	0.03	0.03	0.03	0.03
	18	0.03	0.02	0.03	0.03	0.03	0.03
	19	0.03	0.02	0.03	0.03	0.03	0.03
	20	0.03	0.02	0.03	0.03	0.03	0.03
	21	0.02	0.02	0.02	0.02	0.02	0.02
	22	0.02	0.02	0.02	0.02	0.02	0.02
	23	0.02	0.02	0.02	0.02	0.02	0.02
	24	0.02	0.02	0.02	0.02	0.02	0.02
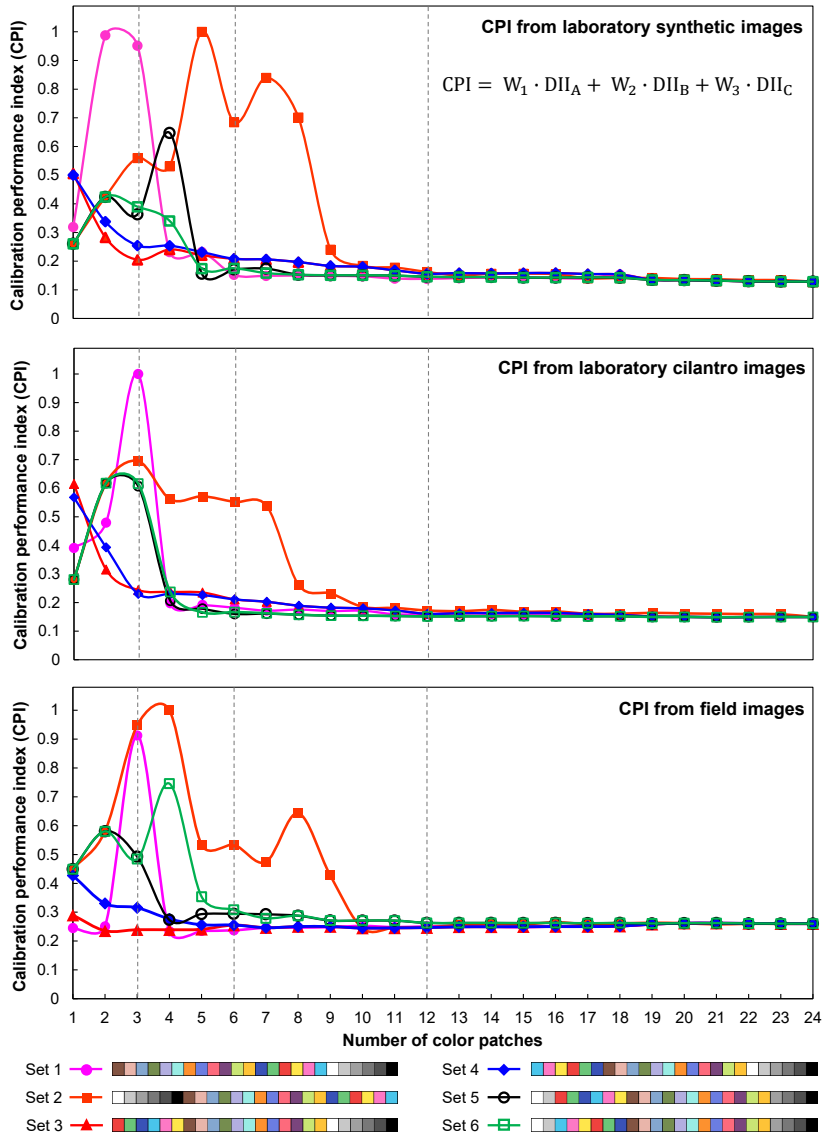
Figure B2. Variations in calibration performance index (CPI) in calibrated images by different color schemes in dark and light conditions.

Table B2. Data used for variations in calibration performance index in calibrated images by different color schemes in dark and light conditions.

Lighting condition	Number of patches	Calibration performance index (CPI)					
		Set 1	Set 2	Set 3s	Set 4	Set 5	Set6
Laboratory synthetic images	1	0.319	0.261	0.506	0.501	0.261	0.261
	2	0.988	0.424	0.283	0.338	0.424	0.424
	3	0.952	0.560	0.205	0.254	0.363	0.390
	4	0.232	0.531	0.241	0.254	0.647	0.340
	5	0.232	1.000	0.221	0.232	0.156	0.174
	6	0.152	0.685	0.208	0.208	0.171	0.175
	7	0.149	0.839	0.206	0.206	0.174	0.158
	8	0.150	0.702	0.197	0.197	0.153	0.153
	9	0.148	0.240	0.183	0.183	0.151	0.151
	10	0.147	0.184	0.180	0.180	0.151	0.151
	11	0.139	0.178	0.168	0.168	0.150	0.150
	12	0.138	0.162	0.156	0.156	0.146	0.146
	13	0.141	0.150	0.158	0.158	0.145	0.145
	14	0.143	0.156	0.157	0.157	0.145	0.145
	15	0.143	0.156	0.158	0.158	0.143	0.143
	16	0.140	0.154	0.158	0.158	0.143	0.143
	17	0.140	0.140	0.155	0.155	0.143	0.143
	18	0.153	0.141	0.153	0.153	0.142	0.142
	19	0.135	0.142	0.135	0.135	0.135	0.135
	20	0.134	0.138	0.134	0.134	0.134	0.134
	21	0.132	0.137	0.132	0.132	0.132	0.132
	22	0.130	0.134	0.130	0.130	0.130	0.130
	23	0.129	0.134	0.129	0.129	0.129	0.129
	24	0.128	0.128	0.128	0.128	0.128	0.128
Laboratory cilantro images	1	0.391	0.281	0.616	0.568	0.281	0.281
	2	0.480	0.618	0.316	0.393	0.618	0.618
	3	1.000	0.695	0.244	0.230	0.608	0.617
	4	0.198	0.561	0.238	0.231	0.206	0.237
	5	0.192	0.572	0.235	0.226	0.179	0.166
	6	0.182	0.552	0.211	0.211	0.160	0.167
	7	0.172	0.538	0.203	0.203	0.161	0.163
	8	0.175	0.260	0.189	0.189	0.158	0.158
	9	0.171	0.232	0.182	0.182	0.154	0.154
	10	0.171	0.184	0.180	0.180	0.154	0.154
	11	0.159	0.181	0.172	0.172	0.153	0.153
	12	0.159	0.172	0.161	0.161	0.152	0.152
	13	0.164	0.170	0.163	0.163	0.152	0.152
	14	0.160	0.174	0.164	0.164	0.152	0.152
	15	0.159	0.168	0.164	0.164	0.153	0.153
	16	0.158	0.169	0.164	0.164	0.152	0.152
	17	0.158	0.161	0.160	0.160	0.152	0.152
	18	0.157	0.161	0.157	0.157	0.152	0.152
	19	0.151	0.164	0.151	0.151	0.150	0.150
	20	0.150	0.162	0.150	0.150	0.150	0.150

Table B2. Data used for variations in calibration performance index in calibrated images by different color schemes in dark and light conditions (continued).
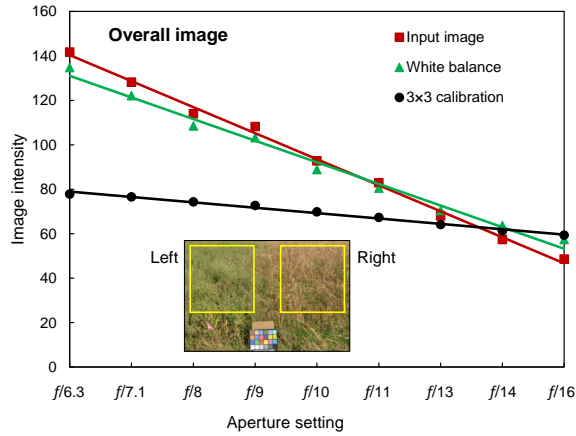
Lighting condition	Number of patches	Calibration performance index (CPI)					
		Set 1	Set 2	Set 3s	Set 4	Set 5	Set6
	21	0.149	0.161	0.149	0.149	0.149	0.149
	22	0.149	0.160	0.149	0.149	0.149	0.149
	23	0.150	0.159	0.150	0.150	0.150	0.150
	24	0.150	0.150	0.150	0.150	0.150	0.150
Field images	1	0.246	0.449	0.290	0.427	0.449	0.449
	2	0.251	0.580	0.236	0.330	0.580	0.580
	3	0.913	0.948	0.239	0.315	0.492	0.485
	4	0.234	1.000	0.239	0.277	0.273	0.746
	5	0.236	0.534	0.241	0.256	0.293	0.354
	6	0.237	0.534	0.255	0.255	0.295	0.308
	7	0.246	0.474	0.246	0.246	0.293	0.279
	8	0.247	0.643	0.250	0.250	0.288	0.288
	9	0.248	0.427	0.250	0.250	0.271	0.271
	10	0.253	0.244	0.245	0.245	0.271	0.271
	11	0.249	0.246	0.246	0.246	0.271	0.271
	12	0.251	0.247	0.247	0.247	0.263	0.263
	13	0.252	0.256	0.249	0.249	0.263	0.263
	14	0.250	0.259	0.249	0.249	0.263	0.263
	15	0.248	0.258	0.250	0.250	0.262	0.262
	16	0.251	0.265	0.250	0.250	0.263	0.263
	17	0.252	0.259	0.250	0.250	0.262	0.262
	18	0.252	0.261	0.252	0.252	0.263	0.263
	19	0.258	0.263	0.258	0.258	0.261	0.261
	20	0.262	0.262	0.262	0.262	0.262	0.262
	21	0.262	0.258	0.262	0.262	0.262	0.262
	22	0.261	0.260	0.261	0.261	0.261	0.261
	23	0.260	0.259	0.260	0.260	0.260	0.260
	24	0.260	0.260	0.260	0.260	0.260	0.260

B3. Data used for Intensity Variation of Input Field Images at Different Aperture

Settings and the Fitted Linear Models (Fig. 4.11 - Reproduced as Fig. B3)

Images processed	Alfalfa (green, left)	Spring wheat (brown, right)	Overall image (both)
Input	$y = -12.04x + 154.28$ ($R^2 = 0.996$)	$y = -11.96x + 157.74$ ($R^2 = 0.997$)	$y = -11.67x + 152.02$ ($R^2 = 0.996$)
White balanced	$y = -10.14x + 143.91$ ($R^2 = 0.991$)	$y = -9.88x + 145.98$ ($R^2 = 0.989$)	$y = -9.72x + 140.8$ ($R^2 = 0.989$)
Calibrated	$y = -2.69x + 83.55$ ($R^2 = 0.976$)	$y = -2.31x + 84.21$ ($R^2 = 0.996$)	$y = -2.42x + 81.38$ ($R^2 = 0.990$)

Figure B3. Intensity variations of input, white balanced, and the calibrated images in field (alfalfa and spring wheat) at different aperture settings. The calibrated image was obtained using the three R, G, and B color patches and the fitted linear models.

Table B3. Data used for intensity variation of input field images of alfalfa and spring wheat at different aperture settings.

Aperture setting	Input image	White balance	3×3 Calibration
$f/6.3$	141.7	134.72	77.83
$f/7.1$	128.19	122.13	76.58
$f/8$	113.96	108.53	74.31
$f/9$	108.18	103.24	72.74
$f/10$	92.8	88.87	69.82
$f/11$	82.92	80.55	67.35
$f/13$	68.04	70.4	64.13
$f/14$	57.36	63.65	61.39
$f/16$	48.61	57.44	59.35

B4. Pseudocode of the ColorCal Plugin Developed in ImageJ using Java Language

The pseudocode of the plugin with appropriate sections and comments is presented subsequently for following the overall algorithm and better understanding of the processes.

```java
/*********************************************************************************
*              ColorCal - Image color calibration and comparison
*          Developed by S. Sunoj & C. Igathinathane; ABEN, NDSU
*********************************************************************************/

// Other standard ImageJ plugin imports
import org.apache.commons.math3.linear.Array2DRowRealMatrix;
import org.apache.commons.math3.linear.RealMatrix;
import org.apache.commons.math3.linear.SingularValueDecomposition;
/*******************************************************************************/
public class ColorCal_plugin implements PlugIn {

  double[][] ColorChecker_values = {{115, 82, 69},..., {52, 52, 52}};
          // X-Rite color chart standard values imported

  public void run(String arg) {
    Read_user_inputs_and_image; // Generic dialog box - front panel of plugin
    Get_ROI_average(Input_image); // Semi-automatic color patch selection
    Calculate_CCM (ROI_average, ColorChecker_values, Selected_patches);
    Apply_CCM_to_image(Input_image, CCM); // Calibrated image generated
    Evaluate_performance_measures (Input_image, Cal_image); // em and DII calculated
    Calibration_performance_index(Input_image1, Input_image2); // CPI calculated
  }

/***************** Calculating color calibration matrix *****************/
  public Calculate_CCM (ROI_avg, CC_std_values, sel_patches){

    RealMatrix I_RGB = new Array2DRowRealMatrix(SelectedROI); // 3 x N
    RealMatrix S_RGB = new Array2DRowRealMatrix(SelectedStdCC); // 3 x N
    RealMatrix I_RGBT = I_RGB.transpose(); // N x 3 matrix

    RealMatrix RHSa = S_RGB.multiply(I_RGBT); // Produces 3 x 3 matrix
    RealMatrix RHSb = I_RGB.multiply(I_RGBT);
    RealMatrix RHSbinverse = new SingularValueDecomposition(RHSb)
    RHSbinverse.getSolver();
    RHSbinverse.getInverse();
    RealMatrix CCM = RHSa.multiply(RHSbinverse); // Color calibration matrix

    return CCM;
  }
```

```
/************************ Applying CCM to image ************************/
  public Apply_CCM_to_image (Input image, CCM){
    RealMatrix ImagePix = new Array2DRowRealMatrix(AllPixels); // 3 x WH
    RealMatrix Cal_image = CCM.multiply(ImagePix); //(3 x 3).(3 x WH) = (3 x WH)

    return Cal_image;
  }


/******************* Evaluating performance measures *******************/
  public Evaluate_performance_measures (Input_image, Cal_image){

//********** Total error **********
// std[][] contains ColorChecker_values from X-Rite -> 24 x 3 matrix
// CC[][] contains ROI_average from image -> 24 x 3 matrix

  for (int i = 0; i<std.length ; i++){
    T_error += (Math.sqrt(Math.pow((std[i][0]- CC[i][0]),2) + ...
              Math.pow((std[i][1]-CC[i][1]),2) + ...
              Math.pow((std[i][2]-CC[i][2]), 2)));
  }
  e_m = ((T_error/(double)CC.length))/(247+203+224);

//********** Difference index **********
  if (NoImages != 2){
    IJ.error("Two images required for comparison!");

    return 0.0;
  }

  ImageCalculator ic = new ImageCalculator();
  ImagePlus imp3 = ic.run("Difference create", imp1, imp2);
  imp3.show(); // Display difference index

  for (int v = 0; v < h ; v++){
    for (int u=0; u<w; u++){
      Pix = imp3.getPixel(u,v,Pix);
      DI_R = Pix[0]; DI_G = Pix[1]; DI_B = Pix[2];
      sumR += DI_R; sumG += DI_G; sumB += DI_B;
    }
  }

  long denom = (w*h*3*255);
  long numer = (sumR + sumG + sumB);
  double DII = numer/denom; // DII calculated here
  perf_meas[0] = e_m; // Modified total error
```

```
    perf_meas[1] = DII; // Difference image index

    return perf_meas;
  }

//************ Calculating calibration performance index (CPI) **************
  public Calibration_performance_index(Input_image1, Input_image2){
    W1 = 0.35; W2 = 0.35; W3 = 0.30; // Weight factors

    dark_ref = DII (dark_calimg, dark_refimg);
    light_ref = DII (light_calimg, light_refimg);
    dark_light = DII (dark_calimg, light_calimg);
    CPI = W1*dark_ref + W2*light_ref + W3*dark_light;

    return CPI;
  }

} // End class
```

*N*ote: The lead authors are willing to extend their help and collaboration to the interested

readers in calibrating the images. If the reader provides the images with the ColorChecker

calibration chart within the image, we can make the calibrated images and send them back

to the readers for educational and non-commercial use at no-cost for a reasonable number

of requests.

APPENDIX C. DATA AND PSEUDOCODE OF THE

DEVELOPED PLANT STAND COUNT, SPATIAL

DISTRIBUTION, AND VIGOR ANALYSIS PLUGIN

This appendix provides (i) profile ratio values for planting pattern determination, (ii) pseudocode of the developed plant stand count plugin, and (iii) plant stand count labeled output for a UAS stitched image of field area $\approx 995\,\mathrm{m}^2$.

C1. Data used for Profile Ratio Cutoff Selection in Planting Pattern Determination

A few sample images of lengthwise and widthwise planting pattern were rotated between 0° and 160° and the data obtained is presented (table C1).

Table C1. Variation in developed profile ratio at field angles between 0° and 160° for planting pattern determination.

Field angle, θ_f	Profile ratio	
(degrees)	Lengthwise	Widthwise
0	0.191±0.053	0.886±0.049
20	0.123±0.053	0.873±0.074
40	0.173±0.086	0.843±0.086
60	0.146±0.077	0.815±0.062
80	0.183±0.085	0.824±0.077
100	0.221±0.081	0.797±0.091
120	0.233±0.098	0.849±0.073
140	0.159±0.057	0.837±0.093
160	0.172±0.085	0.846±0.088
Mean	0.177±0.075	0.841±0.077
SD	0.033±0.077	0.026±0.077

C2. Pseudocode of the Developed Plant Stand Count Plugin in ImageJ using Java Language

Pseudocode of the plant stand count, spatial distribution, and vigor analysis plugin with important package imports, sequence of operations, appropriate sections, task specific Java methods, and comments are presented for better understanding of the processes.

```java
/*********************************************************************************
*          Plant stand counting, spatial distribution, and plant vigor
*                       analysis plugin for UAS images
*              Developed by S. Sunoj & C. Igathinathane; ABEN, NDSU
**********************************************************************************/

// Other ImageJ imports
import org.apache.commons.math3.stat.descriptive.DescriptiveStatistics;
import org.apache.commons.math3.analysis.interpolation.LoessInterpolator;
/**********************************************************************************/
public class PlantSCSDPV_plugin implements PlugIn {

  int n_iter = 3; // Iterations for angle decimal search
  double clsprt = 87.0; // Cluster selection cutoff in percentile
  double ARcut = 0.04; // Area ratio cutoff
  double plant_spacing = 22; // 8.5" converted to cm
  double row_spacing = 76; // 30" converted to cm

  public void run(String arg) {
    Read_image_and_user_inputs; // Generic dialog box - front panel of plugin
    Preprocess_image (input_image, U_thresh, area_filt); // RGB to binary image
    Align_crop_rows_vertically (binary_image); // Row orientation determination
    Identify_crop_rows (vertical_rows_img); // Crop row identification
    Segment_plant_clusters_by_profile(vertical_rows_img); // Segment by profile
    Segment_plant_clusters_by_geometry(prof_seg_img); // Segment by geometry
    Count_plants_rowwise(final_seg_img, row_pts); // Counting row-wise
    Spatial_distribution_map(final_seg_img, row_pts); // Spatial distribution map
    Plant_vigor_map(final_seg_img, row_pts); // Color-coded plant vigor map
  }

/************************* Preprocessing input image *************************/
  public Preprocess_image(input_image, U_thresh, area_filt){

    // Image binarization
    pixcol = ip.getPixel(x,y); get pixel value at x and y
```

```
  Y = 0.299*R + 0.587*G + 0.114*B; // RGB to YUV conversion
  U = -0.147*R - 0.289*G + 0.436*B;
  V = 0.615*R - 0.515*G + 0.100*B;

 if (U < U_thresh){ // U_thresh is 127 for Set I, 118 for Set II
  bwip.putPixel(x, y, 255);
 }

 // Area filtering and morphological operations
 pa = new ParticleAnalyzer( , , , 0.0, Inf.); // area lower limit as 0.0
 float[] area = rt.getColumn(rt.AREA); // area of all objects
 avg_area = getMean(area); // getMean from DescriptiveStatitstics class
 area_cut = area_filt * avg_area; // Area cutoff found in pixels

 pa = new ParticleAnalyzer( , , , area_cut, Inf.); // area limit as area_cut
 // Mask image from particle analysis is used further
 return binary_img;
 }

/*********************** Aligning crop rows vertically ************************/
 public Align_crop_rows_vertically (binary_image) {

  theta = get_field_angle (binary_image); //pass field ROI to IJ.measure()

  if ((theta >= 90 && planting_pattern == Lengthwise) ||
      (theta < 90 && planting_pattern == Widthwise)){
      perform_CW_rotation;

  } else if ((theta <= 90 && planting_pattern == Lengthwise) ||
             (theta > 90 && planting_pattern == Widthwise)) {
             perform_CCW_rotation;
  }

  double sel_ang = 0.0; // Initializing variables for iteration
  double ang_step = 10.0; // First iteration step angle
  double st_pt = -10.0, e_pt = 100.0; // Start and end point

  for(int k = 0 ; k < n_iter; k++){ // Iteration start here
    n_ang = ((e_pt - st_pt)/ang_step) + 1; // No. of test angles

    for (int i = 0; i < n_ang; i++){
        test_angle[i] = st_pt + (i * ang_step);
        IJ.run(binary_image, "Rotate...", test_angle[i]); // Rotate by test angle
        binary_image.setRoi(0, 0, wd, ht); // SetRoi to get profile
        double[] prof_vals = IJ.getProfile(); // Profile obtained
```

```
        DescriptiveStatistics ds = new DescriptiveStatistics(prof_vals);
        // Mean, median, Pmax, SD, skewness, kurtosis, and SD*maximum

        if (SD*Pmax == maximum){
            sel_angle = test_angle[i]; // Optimum rotation angle selected
            st_pt = test_angle[i-1]; // One step before optimum
            e_pt = test_angle[i+1]; // One step after optimum
        }
    }
    ang_step = ang_step/10; // reduced by 1/10th after each iteration
} // end angle iterations

IJ.run(binary_image, "Rotate...", "sel_angle"); // Rotate image by sel_angle
return vertical_rows_img;
}


/*********************** Identifying vertical crop rows ***********************/
    public Identify_crop_rows (vertical_rows_img){

    vertical_rows_img.setRoi(0,0, wd, ht);
    double[] prof = pplot.getProfile();
    double[] pvals = ls.smooth(xvals, prof); // Method from LoessInterpolator

    double min_sd = 10000.0; // Initializing with a high value
    for (int i = 0; i < pvals.length; i++){
  pcut = i;  // Profile cut point

    // Finding left edge of each peak
    for (int j = 0; j < (pvals.length)-1; j++){
       if (pvals[j] < pcut < pvals[j+1]){
         left_edge.add(); // Stored using arraylist
       }
    }
    int nrows = (left_edge.length+1); // No. rows identified at each cutoff

    // Calculate distance between two consecutive cut points
    for (int k = 0; k < (left_edge.size())-1; k++){
      del_x[k] = left_edge[k+1] - left_edge[k]; // Difference in x points
    }

    mean_del_x = getMean(del_x); // Using DescriptiveStatistics() class
    SD_del_x = getgetStandardDeviation(del_x);

        if (SD_del_x < min_sd){ // Finding the minimum SD
           sel_ycutoff = pcut;  // Selected cutoff value
           min_sd = SD_del_x;   // Updating min_sd for next check
```

```
              sel_mean = mean_del_x; // Selected mean for overlaying row ROI
              sel_rows = nrows;     // No. rows at selected cutoff
           }
        }

        // Find peak points as maximum between two consecutive left_edge points
        row_pts[0][i] = peak_pt[i] - 0.2*sel_mean; // 20\% on either sides of peak
        row_pts[1][i] = peak_pt[i] + 0.2*sel_mean;
        return row_pts;
     }

/******************** Segmenting plant clusters by profile ********************/
    public Segment_plant_clusters_by_profile (vertical_rows_img){

       pa = new ParticleAnalyzer(); // Get object properties for the binary image
       BBH = rt.getColumn(ResultsTable.ROI_HEIGHT);
       BBHcut = getPercentile(clsprt); // Calculating BBHcut value

       if (BBH > BBHcut){
          cluster = imp.crop(); // Crop only selected cluster portion
          cluster.rotateLeft(); // Rotate CCW
          double[] pvals = pplot.getProfile(); // Sideways profile obtained
          valley_area = getValleyArea(pvals); // By Heron's formula
          profile_area = getProfileArea(pvals); // By method of trapezoid
          AR = valley_area/profile_area;

          if (AR > ARcut) { // Potential segmentation location
             ip.drawLine(x1, y1, x2, y2); // Digital segmentation
             ip.insert(x, y, cluster); // Insert back in binary image
          }
       }
       return prof_seg_img;
    }

/******************** Segmenting plant clusters by geometry ********************/
    public Segment_plant_clusters_by_geometry (prof_seg_img){

       pa = new ParticleAnalyzer();
       A = rt.getColumn(ResultsTable.AREA);
       BBH = rt.getColumn(ResultsTable.ROI_HEIGHT);
       Acut = getPercentile(clsprt); // Area cutoff calculated

       if (BBH > BBHcut && A < Acut){
          cut_by_BBH(); // Cut horizontally
       }
       else if (BBH > BBHcut && A > Acut){
```

```java
        cut_by_Area(); // Cut horizontally
    }
    else if (BBH < BBHcut && A > Acut){
        cut_by_Area(); // Cut vertically
    }
    return final_seg_img;
}
```

/*************************** Counting plants row-wise ***************************/

```java
    public ImagePlus Count_plants_rowwise(final_seg_img, row_pts){

        imp.setRoi(row_pts[0][i], 0, roi_wd, roi_ht);
        pa = new ParticleAnalyzer();
        p = rt.getCounter();
        centx = rt.getColumn(ResultsTable.X_CENTROID); // Centroid points for marker
        centy = rt.getColumn(ResultsTable.Y_CENTROID);

        rgb_ip.fillOval(centx, centy, marker_size, marker_size);
        int xloc = xcen+35; // providing a shift for label
        int yloc = ycen-35;
        label_no = plant_no + p; // Running counter - inside for loop
        String plabel = String.valueOf(label_no);
        tr = new TextRoi(xloc, yloc, plabel); // Labeling near centroid
        plant_count += label_no; // Total plant stand count - updated for each row
        return labeled_image;
    }
```

/*************************** Spatial distribution map ***************************/

```java
    public ImagePlus spatial_distribution_map(final_seg_img, row_pts){

        ImagePlus spatial_dist_map = NewImage.createRGBimage(W, H);
        ImageProcessor distribution_map_ip = distribution_map.getProcessor();

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

        pl_dist = Math.sqrt(((y2-y1)*(y2-y1)) + ((x2-x1)*(x2-x1)));
        plants_dist = pl_dist*sf; // Plants dist in cm

        if (plants_dist <= R1){
            distribution_map_ip.setColor(Color.BLUE);
            multple++;
        } if (plants_dist > R1 && plants_dist <= R2){
            distribution_map_ip.setColor(Color.GREEN);
```

```
            idl++;
        } if (plants_dist > R2 && plants_dist <= R3){
            distribution_map_ip.setColor(Color.YELLOW);
            sskp++;
        } if (plants_dist > R3 && plants_dist <= R4){
            distribution_map_ip.setColor(Color.ORANGE);
            dskp++;
        } if (plants_dist > R4){
            distribution_map_ip.setColor(Color.RED);
            tskp++;
        }
    distribution_map_ip.fillRect(rect_start_x, rect_start_y, rect_wd, rect_ht);
    spatial_dist_map.setTitle("Spatial distribution map");

    return spatial_dist_map;
    }

/******************************* Plant vigor map *******************************/
    public ImagePlus plant_vigor_map(final_seg_img, row_pts){

        ImagePlus vigor_map = NewImage.createRGBimage(W, H, #);
        // # Further arguments required – e.g., image title, background fill color

        pa = new ParticleAnalyzer(); // Get object properties for the binary image
        pl_area = rt.getColumn(ResultsTable.AREA);
        nplants = rt.getCounter(); area_max = getMax(pl_area);
        area_min = getMin(pl_area);

        int R = 255, G = 255, B = 255;
        for (int i = 0; i < nplants; i++){

            IJ.doWand(cent_x[i], cent_y[i]);
            norm_area = (pl_area[i] - area_min)/(area_max-area_min);
            int fR = norm_area * R;
            int fG = norm_area * G;
            int fB = norm_area * B;
            IJ.setForegroundColor(fR, fG, fB);
            IJ.run(vigor_map, "Fill", "slice");
        }

    vigor_map.killRoi();
    vigor_map.setTitle("Plant vigor map");
    return vigor_map;
    }

} // End class
```
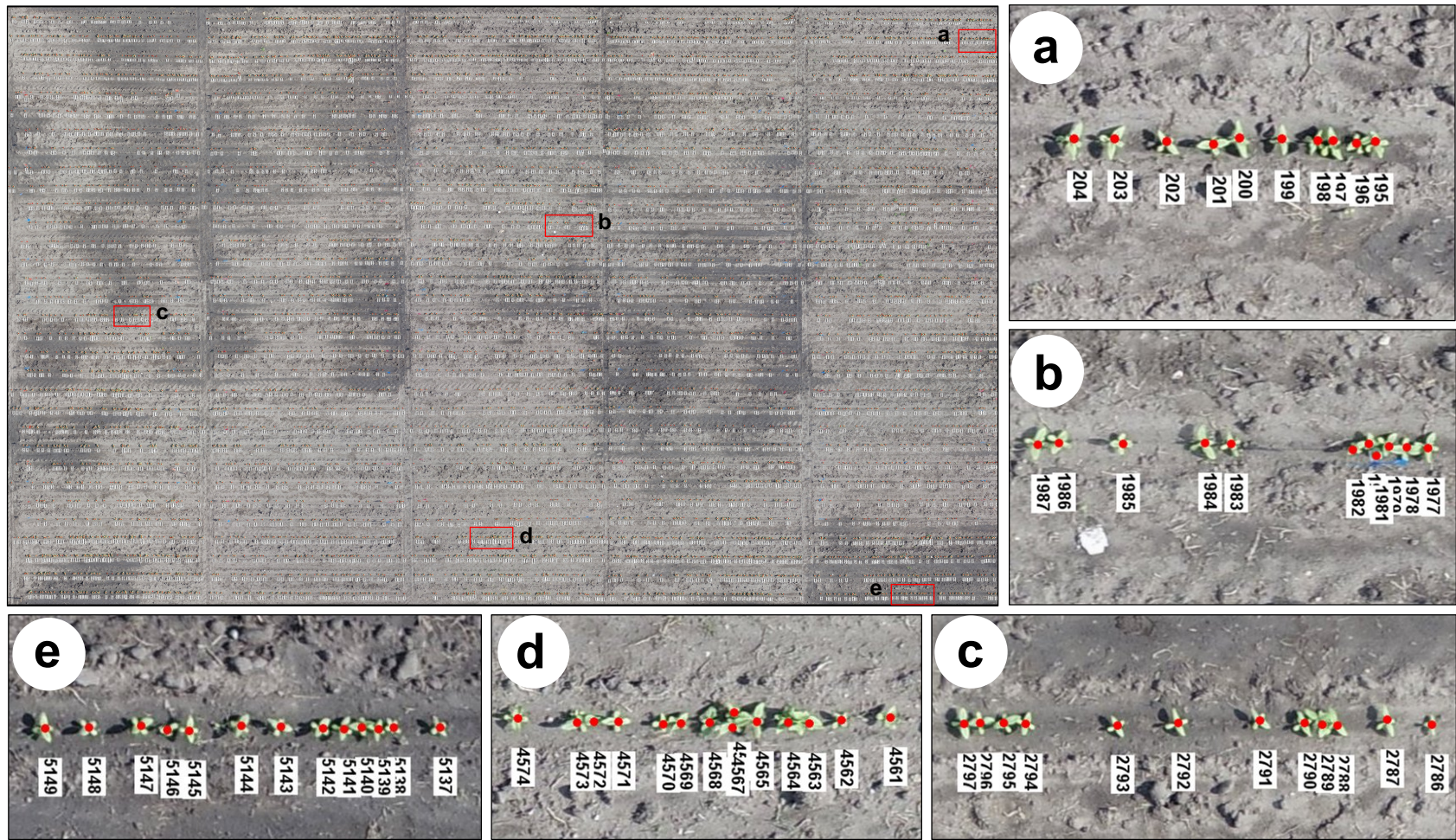
Figure C1. **Plant stand count plugin performance on a stitched image from CREC. UAS - DJI Phantom 4 Pro, flight altitude - 40 ft, field area - 995 m^2 ($\approx$0.24 ac.). Plugin counted plants - 5347, CPU time taken - 916.28 $\pm$ 12.06 s ($\approx$15 min). Five selected portions of the labeled output image are provided (a–e) to indicate the segmented clusters and sequentially counted plant stands.**

APPENDIX D. PHENOCAM APPLICATIONS AND IMAGE

DOWNLOADING PROCEDURE

This appendix provides more information about the applications of PhenoCam and the detailed procedure to download the PhenoCam images for specific period and time span.

D1. Application of Phenocam and Phenocam Network

The application of PhenoCam can be broadly grouped into four classes: (i) close-up observation sites to capture subtle phenological events (e.g., "willamettewheat" and "willamettepoplar" sites at Oregon); (ii) long-term observation of phenology at forest landscapes (Bartlett forest, Richardson et al. (2007)); (iii) regional and continental scale networks to observe phenology changes across different climatic zones and vegetation types (Hufkens et al., 2016; Melaas et al., 2016); (d) controlled environment experiments, where factors such as temperature, precipitation, and CO_2 were manually controlled in a chamber to observe the effects on phenology (Richardson et al., 2018c).

A collective near-surface remote sensing method termed the "PhenoCam Network," was established in 2008 to address the spatial and temporal resolution issues from manual- and satellite-based methods and to continuously monitor the vegetation phenology in forested ecosystem (Richardson et al., 2018b). This was earlier a collaborative, regional-scale camera network in the northeastern US, but currently, PhenoCam refers to any digital camera that captures time-lapse images to study phenology (T. B. Brown et al.,

2016). As of April 2019, there are about 592 sites installed with these camera spanning from Alaska to Texas, and from Hawaii to Maine. The images from these PhenoCams are accessible to anyone through the PhenoCam website (`https://phenocam.sr.unh.edu/webcam/gallery/`) and also provides a permanent, continuous visual record of the ecosystem for many years (Richardson et al., 2017a), and are available for further analyses, such as image processing, time-series modeling, to mention a few. These images also help to bridge the gap between on-the-ground manual observations and satellite-based monitoring (Browning et al., 2017; Richardson, Hufkens, Milliman, & Frolking, 2018d) as the image acquisition is rather continuous, unlike the aforementioned methods.

D2. Downloading Information from Phenocam Network

Users can download the PhenoCam images of any desired PhenoCam site through the PhenoCam website (`https://phenocam.sr.unh.edu/webcam/`). The images can be downloaded for one site at a time and only 2 GB allowed per download. For >2 GB, multiple download sessions have to be performed. The step by step procedure to download the PhenoCam images are given below:

Step 1: Open any browser and enter `https://phenocam.sr.unh.edu/webcam/`. This opens the home page of the PhenoCam network (fig. D1). Multiple page links are available for the user to know about the network, educational materials, list of site locations, etc.
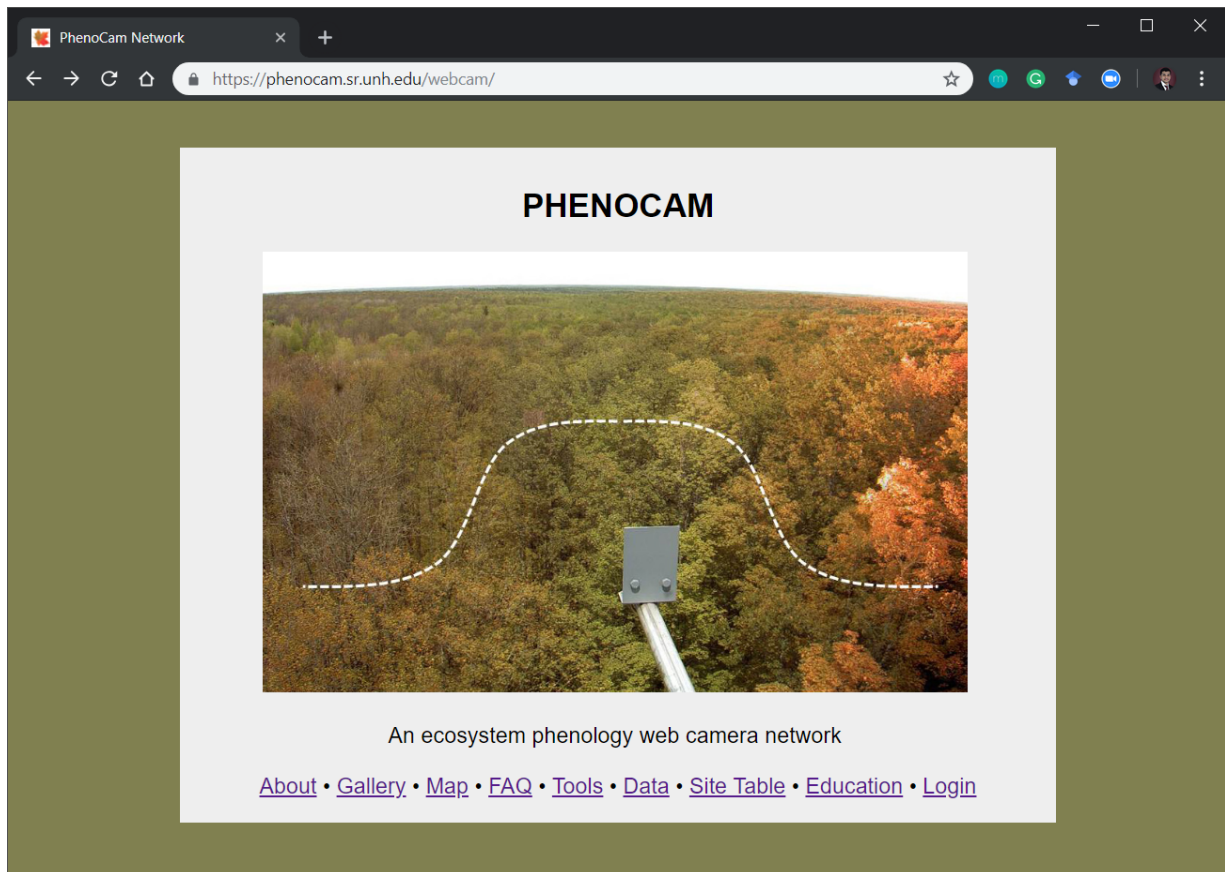
Figure D1. Home page of the PhenoCam website accessed through `https://phenocam.sr.unh.edu/webcam/`.

Step 2: Click on "Login" in the homepage and the PhenoCam - Login page (fig. D2) appears.
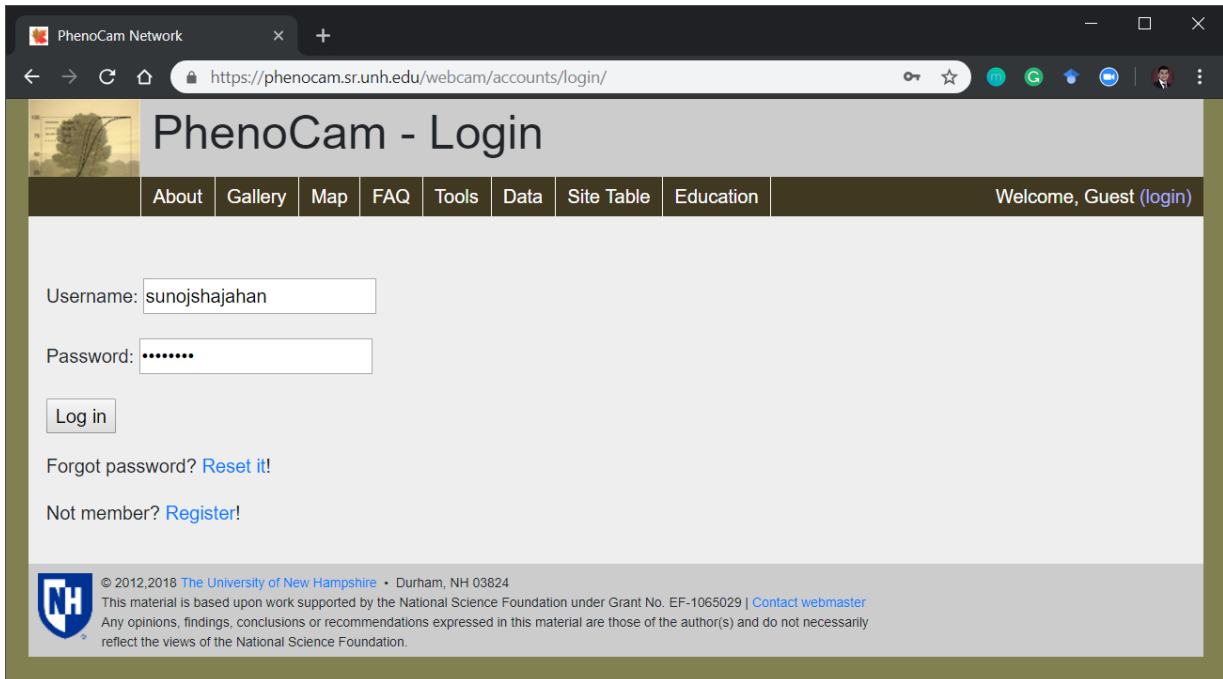
Figure D2. Login page for getting access to download images. New members should register first and then login.

The user should be registered as a member to download the images. To register, click on the "Register!" link. It is a free registration process and basic details such as preferred username and password should be provided there. The user should also have a valid email address to receive the confirmation email from the PhenoCam network.

Step 3: Enter the credentials (Username and password) in the login page and click on "Log in". It automatically directs you to the PhenoCam Gallery page (fig. D3). User can find their username appear on the right top of the page.
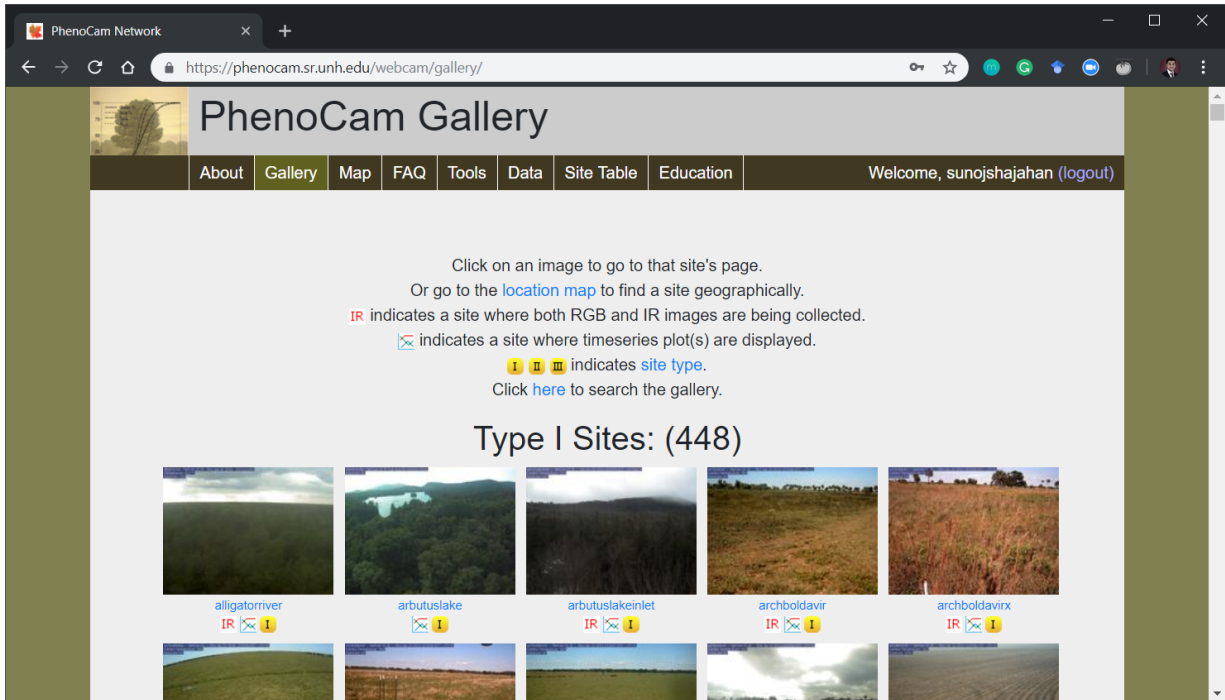
Figure D3. **Gallery page of the PhenoCam website showing the thumbnails of all the active sites. The thumbnails provided in this page are the latest image that was uploaded to the server.**

Step 4: To go to the download page, the user should click on the "Data" tab that is provided on the top panel. The page directs to the PhenoCam - Image Data Download Form.

Figure D4. **PhenoCam image data download form with instructions for entering in the respective fields.**

The user should select a site from the drop down list provided. All the active sites are listed alphabetically in the drop down menu. Enter the "Start Date" and "End Date" as the desired period in YYYY-MM-DD format. Similarly, the "Start Time" and "End Time" should be entered in the 24 h format. It is always better to have a 10 min additional margin for the desired time range. This was because sometimes there will be some delay of 2–3 min in the image capture. This clock is set by the PhenoCam network and any technical issues will slightly alter the time of image capture.

Step 5: Once all the fields are entered click on "Submit" button (fig. D4). The website will take ≈2 min to gather all the images at the specified date and time and automatically directs to the PhenoCam - Data Request page (fig. D5).
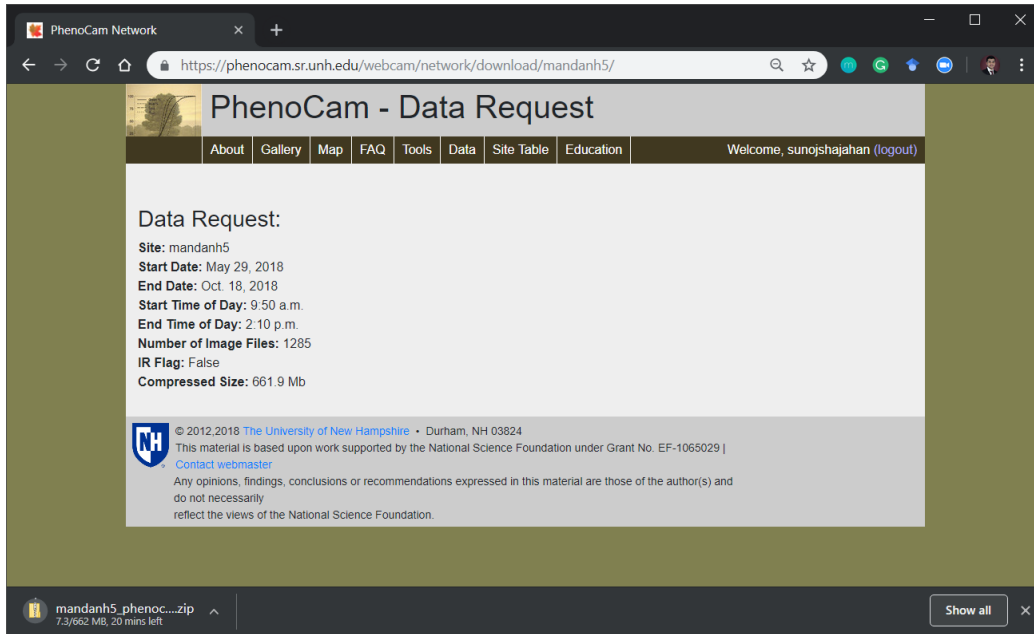

Figure D5. Data request summary page showing entered details, number of images in the file, and download folder size.

This page (fig. D5) provides a summary of the request such as details entered by the user, the number of images gathered based on the entries, and the total size of the download folder. As seen the the summary, there are 1285 images that falls in our requirement and size of compressed folder is 661.9 MB. The download automatically starts and the folder is stored in the "Downloads" folder of the user's computer. All the images will be sequentially sorted based on date and time of the images. This folder can be directly loaded to ImageJ, which opens the image as stack. The developed plugin runs directly on the stack images.