5-2023

Tomato Flower Detection and Three-Dimensional Mapping for Precision Pollination

Kaitlyn McKensie Nelms
knelms3@vols.utk.edu

To the Graduate Council:

I am submitting herewith a thesis written by Kaitlyn McKensie Nelms entitled "Tomato Flower Detection and Three-Dimensional Mapping for Precision Pollination." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Biosystems Engineering.

Hao Gan, Major Professor

We have read this thesis and recommend its acceptance:

Lori A. Duncan, Annette L. Wszelaki

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Tomato Flower Detection and Three-Dimensional Mapping for Precision Pollination

A Thesis Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Kaitlyn McKensie Nelms

May 2023

Acknowledgements

I would like to extend my sincerest thanks to my major professor Dr. Hao Gan, who has been the greatest support in this endeavor. His encouragement and aid throughout this process was unprecedented, and he has set a tremendous example of the type of professor and advisor I can only hope to be one day. Additionally, I would like to thank my past and current lab mates for their help on this project.

I would also like to thank Dr. Lori Duncan and Dr. Annette Wszelaki for not only serving on my committee but also being incredible sources of support in both my research and personal life during this project. Their guidance was extremely helpful and incredibly kind.

Finally, I would like to thank my family, friends, and boyfriend for being by my side throughout this process. They have provided me with unwavering love and support, without which I would not have been able to succeed to this level.

Abstract

It is estimated that nearly 75% of major crops have some level of reliance on pollination. Humans are reliant on fruit and vegetable crops for many vital nutrients. With the intensification of agricultural production in response to human demand, native pollinator species are not able to provide sufficient pollination services, and managed bee colonies are in decline due to colony collapse disorder, among other issues. Previous work addresses a few of these issues by designing pollination systems for greenhouse operations or other controlled production systems but fails to address the larger need for development in other agricultural settings with less environmental control. In response to this crisis, this research aims to act as a vital first step towards the development of a more robust autonomous pollination system for agricultural crop production. The main objective of this research is to develop a flower detection and mapping system for a field crop setting. This research presents a method to detect and localize tomato flowers within a three-dimensional (3D) region. Tomato plants were grown in a raised-bed garden where images were collected of the overhead view of the plants. Images were then stitched together using a photogrammetry technique, accomplished by the Pix4Dmapper software, to form an orthomosaic and 3D representation of the raised-bed garden from a high spatial resolution aerial view. Various machine learning architectures were trained to detect tomato flowers from overhead images and were then tested on the orthomosaic images produced by the Pix4D software. The coordinates of the detected flowers in the orthomosaic were then compared to the 3D model representation to find approximate 3D coordinates for each of the flowers relative to a predefined origin. This research serves as a first step in autonomous pollination by presenting a way for machine vision and machine learning to be used to identify the presence and location of flowers on tomato crops. Future work will aim to expand flower detection to other crops varieties in varying field conditions.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Problem Overview

Biotic pollination is a major contributor to both the quality and quantity of yield in global crop production (Rader et al., 2016), but this vital ecosystem service is at risk. Due to colony collapse disorder, among other causes, there has been a notable decline in the availability of both wild and managed bee populations (Potts et al., 2010). With bees acting as one of the major contributors to crop pollination around the world (Iwasaki & Hogendoorn, 2022), their decline threatens overall crop yield and, therefore, the global human food supply. With many of the world's populations heavily reliant on large-scale agricultural production operations for their primary supply of food, it is important that researchers work to develop new methods of pollination to meet this need.

The field of precision agriculture has had success in many areas such as weeding, harvesting, and monitoring (Fountas et al., 2020) due largely in part to advancements in technologies that allow for better integration with natural and agricultural systems. Many of these advancements involve the use of machine learning and vision systems in order to address complex environmental problems. The continued development of new deep learning architectures allows for the creation of more robust algorithms that are able to address complex tasks, like object detection, at a higher accuracy than prior methods. Additionally, advancements in robotic technologies and their associated power requirements encourage the use of many of these novel systems in settings where previous technologies would not be able to venture. By combining these advancements, they can be utilized to create innovative solutions to agricultural problems, such as the need for precision crop pollination.

Previous research on precision pollination has been largely focused on simplified agricultural problems, such as flower detection without localization, or designed for use with only one type of crop. While these studies are novel and provide important progress towards the end goal of autonomous pollination, there still remains a gap in the field for the design of a robust pollination system that can be utilized in a variety of crop production settings. With this in mind, this research aims to investigate and present initial work towards the design of a more robust system for autonomous precision pollination.

Tomato plants were chosen as the initial crop to study because they are a self-pollinating crop, but insect or wind assisted pollination is needed in order to produce fruit (Toni et al., 2021). Tomato plants are commonly grown in controlled environment agricultural (CEA) systems, where pollination is accomplished through flower vibration or hormone treatments (McGregor, 1976). Because tomatoes are one of the most widely grown CEA crops, the development of alternative pollination systems is an important area of research. Additionally, using tomato plants for this study allows for simplification of the flower detection task by eliminated the need for a model to determine the sex of a flower, seen in other types of crops. Furthermore, future testing of the system would only require a flower to be shaken for pollination, excluding the need for pollen to be transferred by a robotic system between flowers. For these reasons, this study focuses on detecting and mapping flowers on tomato plants.

1.2 Crop Pollination Needs

1.2.1 Importance of Animal-Assisted Crop Pollination

Pollination is the method by which plants reproduce, allowing for the production of commercially available fruits, vegetables, and other value-added products. Van der Sluijs and Vaage (2016) define it as "the active and passive transfer of pollen within or between flowers." In agricultural systems, ensuring adequate crop pollination is essential for a successful yield. Many crops grown for food, fuel, and fiber are considered pollinator dependent, meaning that the quality and yield of these crops are either reliant on or improved by animal pollination (Rader et al., 2016).

Approximately 30% of global food production can be attributed in part to animal-based pollination (Khalifa et al., 2021). Animal pollination increases fruit and vegetable production in 20% of crops, with the majority of other crops experiencing potential limitations in production due to a decrease in pollinator contributions (Klein et al., 2007). More specifically, an estimated 75% of crop species benefit from insect pollination (Klatt et al., 2014). While most staple crops are wind-pollinated or self-pollinated, such as wheat, corn, and rice (Ghazoul, 2005), many crops, such as fruit, which contain important micronutrients for humans, are more reliant on flower visitation by animals (Chaplin-Kramer et al., 2014). Ninety-eight percent of available vitamin C is sourced from animal pollinated plants, mainly citrus and other fruits (Eilers et al., 2011). Eilers et al. (2011) also found that around 74% of all lipids produced globally are from oils in plants that rely at least partially on animal pollination. Additionally, these plants are considered to be a primary source of important fat-soluble vitamins. Other plants that produce fruits and nuts, such as almonds, are strongly dependent on pollinators and contain a significant amount of plant derived minerals, including calcium and fluoride. Furthermore, many fiber crops used to feed livestock also have a level of dependence on insect pollination, indicating that animal pollination even affects livestock production (van der Sluijs & Vaage, 2016). Overall, animal-assisted pollination has a significant impact on many of the foods that humans consume globally.

Animal-assisted pollination also plays a role in the economic sector of agriculture as it improves both production quality and quantity of many crops. Pollination contributed to about 9.5% of the economic value of human food production worldwide in 2005, with the leading categories being vegetables and fruits followed by stimulants, nuts, and spices (Khalifa et al., 2021). As part of the global gross domestic product (GDP), pollination services are valued at between one and two percent in the short-term (Lippert et al., 2021). Insect pollination specifically is worth around two-hundred and fifteen billion US Dollars globally (Klatt et al., 2014). With the importance of agriculture to the world economy and global food security, pollination contributes to a significant amount of the value in this sector.

1.2.2 Current Crop Pollination Concerns

The agricultural production of many crops is benefited by insect and animal pollination (Khalifa et al., 2021; Klein et al., 2007), but there is a decline in wild and managed pollinators threating the availability of pollinator services (Potts et al., 2010). When natural pollinators are not abundantly available to visit agricultural fields, the current alternative is to acquire managed honeybee hives to pollinate fields (Klein et al., 2007). Even with this use of commonly managed pollinators, such as honeybees (*Apis mellifera*), a decline can still be seen and largely attributed to colony collapse disorder (CCD) and other environmental factors (Bugin et al., 2022). Because of CCD and other factors, the use of honeybees as the main agricultural pollinators puts the human food supply at risk (Kremen et al., 2002). If managed pollinators are not able to be utilized, reliance on natural pollinators is required but their ability to contribute to the demand for pollination services in agricultural systems is not entirely known. A study by Kremen et al. (2002) found that native bee communities could provide sufficient pollination services to certain farms, but this capability diminished with growing agricultural intensification. Additionally, along with increases in agricultural intensification there was a significant reduction in both native bee abundance and diversity. With the decline in both native and managed bees, a threat to food security due to unmet demands in crop pollination services has arisen (van der Sluijs & Vaage, 2016).

Several contributors to the decline in insect pollinators have emerged, such as insecticides, pathogens and parasites, and a lack of diversity in floral resources (Watson & Stallins, 2016). More specific to CCD, causes of honeybee hive fatality are commonly attributed to neonicotinoids, the emergence of the parasitic varroa mite and viruses, and agricultural monoculture practices. Additionally, these issues have also contributed to the decline of native bee species and other insect pollinators. In a study examining apple pollination by bees, it was found that exposure to neonicotinoids reduced flower visitation rates from colonies, resulting in decreased pollination services overall (Stanley et al., 2015). There was also an observed reduction of thirty-six percent of seeds available in apples pollinated by colonies exposed to the pesticides, indicating a reduced quality in the apple yield. The observed effect of this pesticide in bees actively pollinating apples suggests that the use of this insecticide on other crops may also have effects on pollination services being administered by bees in those areas. It has been recognized that the use of neonicotinoids may negatively affect insect pollinators, but even newer systemic pesticides, with the potential to replace neonicotinoids, have also been shown to reduce foraging behaviors and ultimately pollination services of bees at field levels (Tamburini et al., 2021). Agricultural management practices also have a distinct effect on bee foraging activity (Nicholson et al., 2017). Farms surrounded by more natural area see an increase in native bee abundance and pollination activity attributed to the availability of more diverse floral resources in terms of timing and nutrient contents and availability natural nesting areas for the colonies (Mader et al., 2010). An increase in farming intensity also reflected a negative effect on pollinators, citing an increase in agrochemicals and decrease in surrounding natural pollinator habitats as the primary reasons. It can therefore be concluded that many common agricultural practices contribute to the reduction of both natural and managed pollinator services resulting in a threat to global food security.

1.2.3 Pollination Types and Flower Anatomy

Plants have two different forms of pollination: self-pollination and cross-pollination. Self-pollinating plants are able to fertilize themselves, while cross-pollinating plants require that pollen be taken to another flower of the same species for fertilization to occur (United States Forest Service, 2023b). Some self-pollinating plants include peppers, eggplants, tomatoes, wheat, oats, and barley. Examples of cross-pollinating plants include apples, pears, almonds, watermelons, pumpkins, and many types of squash. There are two ways to categorize the process of pollination: abiotic and biotic (United States Forest Service, 2023a). Abiotic is pollination without organisms, while biotic is pollination with the involvement of organisms. The two primary types of abiotic pollination are wind-mediated and water-mediated. Self-pollinating plants are typically able to be fertilized through abiotic means such as wind since they do not require the pollen to be transferred between plants (Zohary, 2001). This reduces genetic variability, which can be considered both a positive and negative in terms of agricultural production. Because crops pollinate themselves, the gene pool remains homogenous and more easily maintained as compared to cross-pollinating crops which require more maintenance to retain desired traits. Cross-pollination requires that pollen from one plant be transferred to the flower of another plant which can be facilitated through abiotic means, such as wind, or biotic means, such as the use of animal pollinators (Encyclopaedia Britannica, 2020). While cross-pollinating crops may be more difficult in terms of maintaining certain traits, the ability to produce more genetic variability in a species allows for positive adaptations to occur within a crop species, which may not be as prevalent in self-pollinating species.

There are several different anatomies for plant flowers (Woodcock, 2012). Individual flowers can be male, female, or hermaphrodites, meaning they have both male and female parts within one flower. Hermaphrodites are often referred to as complete flowers, while male and female flowers are referred to as incomplete. Plants with incomplete flowers are further classified as *monoecious*, meaning separate male and female flowers on the same plant, or *dioecious*, meaning separate male and female plants. The timing of when certain types of flowers are present on plants can also be varied. Self-pollinating plants with flowers that strictly pollinate themselves are called *autogamous.* Other self-pollinating plants, which are able to fertilize themselves or be cross-pollinated by flowers on the same plant are called *geitonogamous*. Cross-pollinated plants that must be fertilized by a different plant of the same species are called *xenogamous*. There are several ways that plants ensure cross-pollination including the expression of male and female flowers at different times, physical separation of the male and female parts within a flower, and other chemical methods which halt the self-fertilization process. Some crops require cross-pollination with an entirely different cultivar to produce fruit. Examples of such plants include apples, pears, plums, and some additional orchard fruit varieties. It is important to take into account individual plant pollination needs and plant anatomies to create an efficient pollination system for each crop.

1.2.4 Current Pollination Methods and Associated Costs

Traditionally, the majority of biotic pollination services were performed by native insects and animal populations. Several agricultural practices still rely heavily on these ecosystem services, but agricultural intensification has caused disruptions in the ability of native insect populations to serve as the sole source of pollination for some crops (Kremen et al., 2002). In some farm practices, native species may be able to provide sufficient pollination services if the agricultural operations are of a moderate size and land management practices allow for biodiversity in the environment surrounding the farmland. If management of agricultural lands can conserve native bee habitats and provide enough external resources for native bee populations, this pollination service is generally considered free. Having diverse wild bee populations in agricultural areas is shown to increase crop yields, however, the decrease in diversity of floral resources due to agricultural production can decrease their abundance (Iwasaki

& Hogendoorn, 2022). This decrease in population can then ultimately affect whether producers need to bring in other aids to ensure proper pollination and sustain crop yields.

In many cases, native pollinators are not able to meet the pollination requirements of crops at the production level demanded, thus commercially produced and managed bees are used to supplement pollination needs (Velthuis & Doorn, 2006). The most commonly commercially produced pollinator species are honeybees and bumble bees (Iwasaki & Hogendoorn, 2022). The use of honeybees as the primary commercially produced pollinator is due to their ability to be transported in hives and produce honey (Khalifa et al., 2021). They are used to pollinate many different crops including oilseed rape, buckwheat, and strawberries. Bumblebees are also used to pollinate crops but are typically attracted to fields through specific management practices rather than kept and transported in hives. They are also used to pollinate many different crops, especially buzz-pollination crops, such as blueberries and tomatoes. The use of managed bee populations for crop pollination in the US can be traced back to the early 1900s, but it became more of a common practice in the late 1900s (DiDonato & Gareau, 2022). While the use of managed bees can aid in pollination services for agriculture, they can also compete with native insects for floral resources, even further affecting the abundance of native species. Managed bee populations introduced to new agricultural ecosystems may also unknowingly introduce novel pathogens to native species, leading to the endangerment of local pollinator communities. In addition to endangering native bee and insect populations, managed bee colonies are also seeing a decline attributed to colony collapse disorder, among other causes (Bugin et al., 2022). This decline in population threatens the supply of pollination services provided by managed bee producers. Thus, even with the use of managed and native pollinators, some crop producers may still need additional options to meet their pollination demands.

In some agricultural environments, such as greenhouses, mechanized or manual pollination serves as an additional method of pollination (Wu et al., 2022). There are many types of mechanized pollination, where mechanical devices are used to distribute pollen to the stigma of plants in order to pollinate crops. Mechanical methods can often be used to provide pneumatic-assisted pollination, spray pollination, or static electricity-assisted pollination. There are also mechanized methods of pollen collection, such as a vacuuming procedure, but certain crops are more difficult to use this method on and may require manual pollen collection, also known as hand pollination. Depending on the type of crop, hand pollination may only require shaking the plant, while others may require more precise applications of pollen to the flowers. While hand pollination is simple and precise, it is often costly and labor intensive furthering a need for more developments to be made in mechanized methods. In general, the most common way to artificially pollinate crops is by hand, followed by the use of mechanical systems (Wurz et al., 2021). These methods serve as an important solution to the growing need for more efficient and effective pollination in the absence of pollinators.

1.3 Technology Contributing to Robotic Pollination

1.3.1 Advancements in Agricultural Robotics

Advancements in robotics have resulted in more precise and efficient technologies that are slowly being integrated into agricultural production. Agricultural production has long used machines such as tractors, spreaders, sprayers, and harvesters, but advancements in these technologies have allowed for more efficient use of inputs and a reduction in the wasting of fuel, fertilizer, and seed (Rehman et al., 2016). As time goes on, adaptations to agricultural equipment allow for a more sustainable production processes, which aid in conservation of the environment as well as providing economic benefits to producers and consumers of agricultural products. A few examples of agricultural technologies include autopiloting equipment, crop sensors, GPS documentation of fields, and the integration of mobile technology for monitoring of agricultural systems (Khan et al., 2021). Integration of these and other technologies give producers monitoring and sensing capabilities, which were previously unavailable with reliance on more traditional equipment. Even with the availability of new technologies, there are still some challenges with the adoption of certain advancements due to economic, social, and infrastructural factors (Sidibé et al., 2021). With this is mind, it is also important for researchers and developers in the field of agriculture to consider producers' situations and needs when designing new agricultural technologies.

Many robotic systems designed for crop production attempt to be autonomous, mobile, and have decision-making capabilities (Lowenberg-DeBoer et al., 2020). Agricultural field crop systems are particularly difficult to automate due to the unstructured environmental conditions that result in constant changes to the system (Fountas et al., 2020). Many robotic field technologies focus on automating tasks such as weeding, seeding, plant/crop monitoring, spraying (i.e., herbicide application, fertilizer application, etc.), and harvesting. Many of these techniques, such as weeding, seeding, and harvesting, are more successful overall since the tasks are typically more defined, such as distinguishing a red apple from the green foliage of a tree for harvesting. Tasks such as pest and disease detection and plant management are more complex as the solution is not able to be as neatly defined. Many advancements in field robots remain heavily reliant on human operation, meaning they struggle to be fully autonomous for certain tasks (Lowenberg-DeBoer et al., 2020).

Field robotics can be split into three main categories: large terrestrial, small terrestrial, and unmanned aerial vehicles. Large terrestrial robotics typically resemble traditional farm equipment, such as tractors, and are designed to be semi-autonomous meaning that they can complete certain tasks without requiring a driver (Spykman et al., 2021). Small terrestrial robots are a broader category of robotics, which typically describes newer robotic technologies designed to complete field tasks such as harvesting, weeding, or pollinating. Many of these smaller robotic designs are not based on traditional farm equipment. In agriculture, unmanned aerial vehicles (UAVs) are typically used for remote sensing, soil and crop monitoring, yield estimation, and spraying (del Cerro et al., 2021). Together, these three categories provide unique and important advancements to the field of agriculture.

In general, protected cultivation and greenhouse crop production have seen more growth in robotic systems compared to field crop production as they are more controlled environments than field crop operations (Lowenberg-DeBoer et al., 2020). Robotic applications within these systems are different than field robotics as they can often be designed to be stationary or with limited mobility. They may also have more reliable access to power supplies than robotics in agricultural fields. Substantial research has been conducted on greenhouse mechanization and found that many tasks can be almost fully automated (van Henten et al., 2013). Some examples of tasks automated in greenhouses include seeding, harvesting, transporting, and sorting. These tasks are well-defined and require limited reasoning or decision making. Many greenhouse tasks also do not require a high degree of mobility from the robotic system. While some tasks are more easily accomplished in the controlled environment of a greenhouse, there are still many tasks that require complex solutions and need further research.

1.3.2 Advancements in Deep Learning and Computer Vision

Over the years, the field of machine learning has experienced a tremendous amount of growth with applications seen in almost every field, including agriculture (Liakos et al., 2018). Machine learning succeeds by using large amounts of data to learn tasks without being precisely programmed to do so. Machine learning can be categorized into two main classes, supervised and unsupervised, based on how the model learns. Supervised learning utilizes a set of training data, which includes the input and outputs, so that the model is able to learn some set of rules in order to complete or predict the output given the input data. Unsupervised learning does not have a set of training data, but rather attempts to find patterns among unlabeled data. Some examples of supervised learning techniques include regression, Bayesian, decision trees, support vector machines (SVMs), and artificial neural networks (ANN). One of the most common examples of unsupervised learning is clustering.

Machine vision is another technique growing in application alongside machine learning. Traditionally, machine vision is the process of using a device to capture an image and then the processing of that image data through various techniques in order to extract different characteristics or meaning from the data (Gomes & Leta, 2012). Most machine vision applications involve several steps including data acquisition, pre-processing, and processing. Data acquisition is typically accomplished by obtaining an image in some form, which can be captured using many different types of devices and contains numerical representations of information from the ‘real life’ image which can be computationally processed. Pre-processing aims at improving the quality of the data collected and highlighting features of interest within image data. Finally, the processing step involves the final extraction of what is being recognized and interpreted from the image. Common tasks typically accomplished using machine vision are classification, object detection, and object segmentation. Applications of this technique in agriculture and food production include, but are not limited to, object detection tasks (i.e., weeding and harvesting), quality detection (i.e., disease and deficiency detection), and vision-based guidance systems (Mavridou et al., 2019).

Many of the complex agricultural tasks listed above are accomplished with the use of deep learning. Deep learning is a technique which uses complex multilayer ANNs, one of the machine learning techniques mentioned previously, to accomplish a goal. Deep learning is often implemented in machine vision applications using convolutional neural networks (CNNs). CNNs are neural networks that organize data differently than traditional networks and take into account spatial aspects of the image data. This is an important feature as it allows models to relate locations of pixels enabling them to be able to perform tasks such as object detection. While deep learning is currently considered the state-of-the-art technique in the field of machine learning, it relies heavily on the presence of large amounts of data and significant computational power, which limits its ability to be used in certain applications (Mavridou et al., 2019).

1.3.3 Other Technological Advancements in Agriculture

In addition to advancements in robotics and computational techniques, the integration of Internet of Things (IoT) into agricultural production has brought many positive changes to the field (Xu et al., 2022). IoT involves the connection of physical objects to the internet via the use of technology, such as sensors, to exchange and communicate information. Agricultural IoT specifically monitors agricultural processes such as the tracking of environmental conditions or actions of livestock as well as the transmission of this data. This allows for intelligent management of various components of agricultural systems. The integration of IoT into agriculture also allows for the collection of a large amount of data, which lends itself well to potential machine learning applications. As previously mentioned, many state-of-the-art computing techniques require large amounts of data for training, and the increase in agricultural data collection opens up the possible use of deep learning on a more significant scale. Additionally, advancements in communication technology and availability also increase potential for integration of more computationally intensive techniques as data collection and processing can be done at different locations and transmitted between devices (Garg & Alam, 2020).

Overall, development potential of more advanced agricultural technologies is greatly improved by the integration of IoT into the field.

1.4 Existing Research on Flower Detection and Precision Pollination

Using a combination of crop pollination knowledge and advancements in agricultural robotics and machine learning techniques, many researchers are taking on the challenging task of developing precision pollination systems. While still at its infancy, this research aims to provide producers with additional options for pollinating their crops. Many of the current research studies in this field focus on either flower detection systems or fully designed mechanisms for crop pollination.

1.4.1 Crop Flower Detection Research

Many previous research studies on crop flower detection focus on the detection of flowers from up-close images or the segmentation of flowers from the background. One study focused on detecting the flowers of various apple varieties tested several deep learning models and found that YOLOv4 produced an accuracy of 97.3%, but when looking at image data used, the study only implemented up-close images of the flower blooms, shown in Figure 1.1 (Wu et al., 2020). This study achieved a high accuracy but excluded the challenge of detecting the bloom from the complex background of apple trees at farther distances. Another study focused on the detection and segmentation of flowers on pear, peach, and apple trees by training a DeepLab-ResNet network to detect flowers and then implemented a delineation method to segment flowers from the rest of the image (Sun et al., 2021). The study presented an average F1 score, an accuracy metric which will be further explained in Chapter 2, of 80.9% across all datasets. The images (Fig. 1.2) used were taken farther away from the apple trees rather than up-close like the previous research. Another study focused on detection of apple king flowers in field conditions, used a Mask R-CNN network with instance segmentation to improve upon other apple flower detection models, which only aimed to detect flower clusters (Mu et al., 2023).

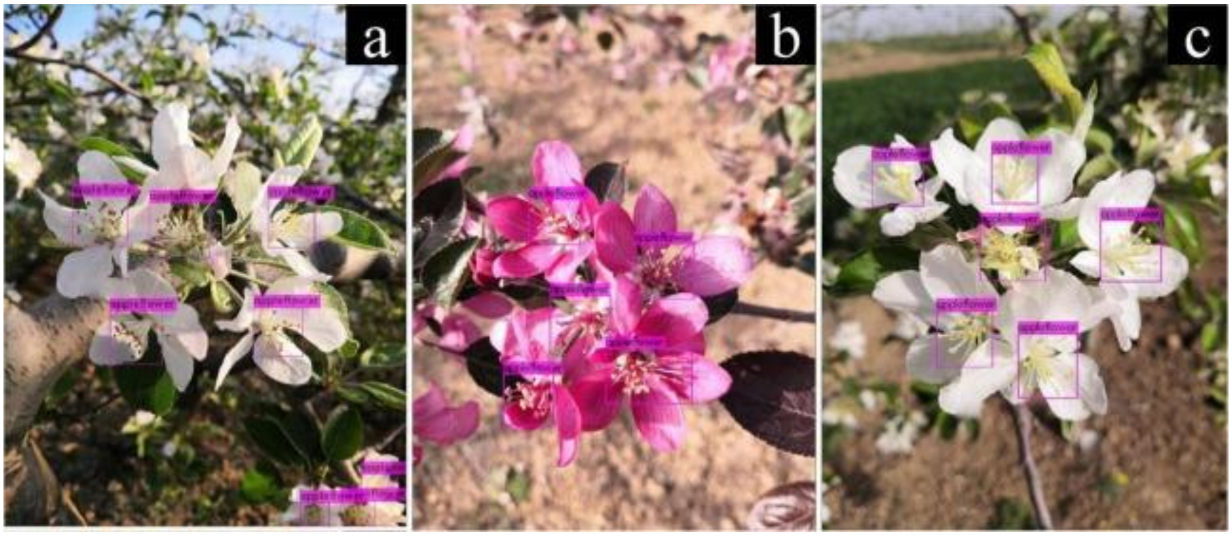

Figure 1.1 Images of flower detection by YOLOv4 algorithm (Wu et al., 2020)

Figure 1.2 Images of flower detection by DeepLab-ResNet and segmentation (Sun et al., 2021)

Apple king flower detection, shown in Figure 1.3, is important as the pollination of the king flower is key for increased yield production in apples. This model reported a range of accuracy from 65.5% to 98.7% based on floral bloom stage. While detection accuracy generally decreases with each additional step of added complexity, it demonstrates the need for continued research into this field as pollination requires these additional complexities and more. Another example of added complexity to the task of flower detection for pollination is shape classification of crop flowers. The goal of shape classification is to better predict the readiness of a flower for pollination. One study looking at tomato flowers accomplished this utilizing a CNN to classify whether or not a flower was ready for pollination, shown in Figure 1.4, and presented an 87.3% accuracy (Hiraguri et al., 2023). Overall this further depicts the complexity of flower detection when the intended use of this method is pollination.

1.4.2 Precision Pollination Systems

In addition to previously mentioned examples of flower detection research for pollination purposes, researchers are also actively working to design full end-to-end precision pollination systems for other specific crops. Some examples of crops currently being researched include tomatoes (Yuan et al., 2016), bramble plants (Strader et al., 2019), vanilla (Shaneyfelt et al., 2013), and kiwifruit (Barnett et al., 2017). One study centered on designing a robotic pollination system for tomato plants grown on a wire system in greenhouses (Yuan et al., 2016). This system utilized a ground-based robot that went alongside the plants and detected flowers by thresholding hue, saturation, and intensity values (HSI), filtering noise, and applying mathematical equations based on epipolar geometries to predict three-dimensional locations. Reported success rates for flower detection ranged from 50.0-87.5% based on the number of flowers in a cluster, with more flowers leading to a higher accuracy and single flowers having poor recognition accuracy. Another study designed a pollination robot named ‘BrambleBee,’ which pollinates bramble plants in a greenhouse setting. The robot is ground-based and maneuvers around bramble plants in order to detect flowers (Strader et al., 2019). The portion of the research focused on detection of flowers utilizes the deep learning architecture Inception-v3 with transfer learning on their own dataset of flowers. The position of the flower is acquired using a plant reconstruction created

Figure 1.3 Apple king flower detection denoted by blue outline (Mu et al., 2023)

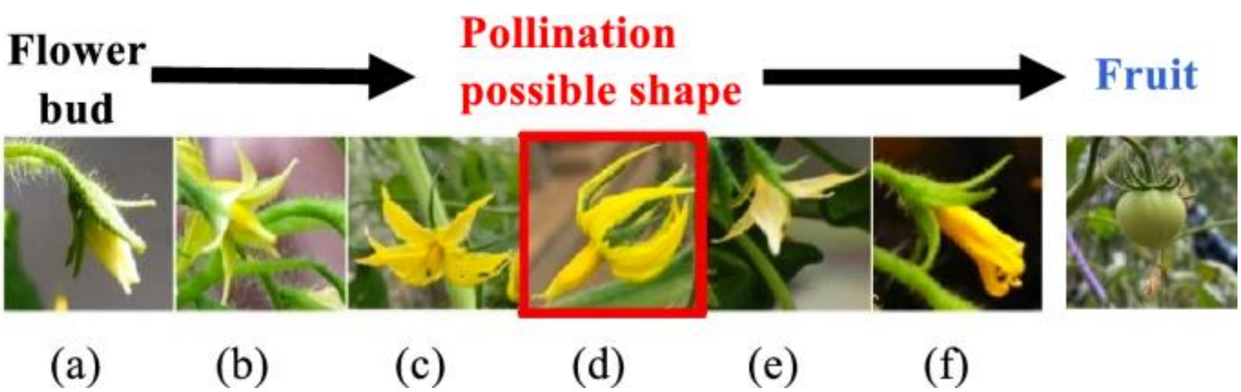

Figure 1.4 Tomato flower shapes for pollination readiness prediction (Hiraguri et al., 2023)

through a simultaneous localization and mapping (SLAM) algorithm. This study reports an average precision of 78.6% and recall of 90.0%. An additional study designed a stationary pollination system for vanilla which included a pole that vanilla plants would grow up and robots extending from the top of the pole would reach downward to pollinate flowers (Shaneyfelt et al., 2013). The method for flower detection included several steps of segmentation that appear to be based on color and 'high energy' area detection, but no accuracy metrics or results for flower detection were reported in the study. Another novel study created a robotic system for pollinating kiwi flowers in an orchard (Barnett et al., 2017). The robot traveled underneath the kiwi tree canopy and captured images of flowers in both day- and night-time conditions. Flower detection was accomplished through the use of an unspecified CNN, and flower localization was accomplished through stereo matching. They reported a flower detection accuracy of 89.3% with a localization accuracy of 71.9%. While all of these studies present novel designs for crop pollination, they are increasingly specific to one crop and generally lack robustness in their ability to be adapted to other crop production scenarios. Further examples and research will be explored in the coming chapters.

1.5 Research Objectives

The main objective of this research is to detect and map the three-dimensional locations of tomato flowers relative to a predefined origin. The ability for crop flower locations to be mapped serves as an initial step towards future research and development of a more robust autonomous pollination system for agricultural crop production. This objective has been broken down into two parts, the first being two-dimensional detection of flowers using deep learning as an accurate method of detection, and the second being three-dimensional mapping of tomato plants and subsequent flowers in order to identify the third axis coordinate position for each flower. In an effort to reduce complexity and increase adaptability of this research to other crops, an overhead view was used to collect images of the tomato flowers. The results of this research will investigate the accuracy of methods used for both flower detection and three-dimensional mapping and will include additional discussion on future research based on resulting analyses of methods explored.

Chapter 2: Tomato Flower Detection

2.1 Introduction

2.1.1 Overview

A vital step towards the development of a robust autonomous pollination system is the establishment of a reliable and accurate method for crop flower detection. The complexity of detecting flowers varies by crop variety. Factors that can affect accuracy of detection include size, color, number, and visibility/occlusion. A proper detection system should take into account all of these factors and be designed accordingly in order to maximize detection accuracy and minimize complexity when possible. In a controlled setting, reducing complexity may not be as important, but as this research is extrapolated to real world ecosystems its importance drastically increases. In a field setting, it is difficult to access to proper power requirements needed for processing intensive computational tasks, which is therefore a critical aspect to consider when developing these designs for agricultural systems.

Natural pollinator species, such as bees, identify flowers through both visual and olfactory cues (Orbán & Plowright, 2014). Based on current technological advancements, visual detection of flowers is the most realistic method for identification in this application. Machine vision is the field of research focused on analyzing and interpreting the content of image data. In machine vision, the task of detecting a flower could be considered object detection or object segmentation. Object detection focuses on finding an object in an image and creating a bounding box around the object. Conversely, segmentation aims to detect an object and separate it completely from the rest of the background by classifying each pixel in an image as either the object or not the object. For the purpose of this research, object detection provides enough accuracy without adding the increased complexity of segmenting the flowers from the rest of the image. Techniques commonly used for these tasks include color filtering, detection of feature descriptors, and deep learning-based models. Many deep learning models are considered state-of-the art as they consistently outperform other methods, but they also typically require significant amounts of training data and computational power compared to other methods (Mavridou et al., 2019).

2.1.2 Object Detection Methods

Object segmentation by color is a commonly used technique that thresholds pixels in an image using different numerical color values. Colors are typically represented as a combination of three different values. Humans perceive colors as a combination of red, green, and blue (RGB), and these are usually considered the three primary colors that make up a colored image (Cheng et al., 2001). Based off the RGB color space, other color spaces can be derived. A few of the more commonly used color spaces include hue, saturation, value (HSV), hue, saturation, intensity (HSI), and CIELUV, a color space adopted by the International Commission on Illumination, commonly used in computer graphics. Within each color space, threshold values can be applied to each channel and only the pixels that fall within those threshold values will be segmented from the rest of the image. These resulting pixels represent the segmented object regions. An example of color segmentation can be seen in Figure 2.1, where the foliage is separated from the background sky, and the citrus fruits contain a mix of both red and yellow values. The selection of the best color space to use for object segmentation in an image is a complex decision and can be different for each application based on individual research data and objectives. In some contexts where there is a large contrast of color between the target object and the background of the image, color thresholding can be an accurate and valuable technique. In more complex settings where the color of the target object is similar to background objects or the lighting drastically changes within the image data, color segmentation may not be the best choice.

In addition to color segmentation, other feature descriptors, such as edges and key points can be used to detect objects. Traditionally object detection was accomplished in three different stages: information region selection, feature extraction, and classification (Zhao et al., 2019). Information region selection entails scanning the whole image with different scaled windows in order to find objects that may appear in any region of the image and at many different scales. Feature extraction is the process of extracting features in an effort to identify objects and classification is the process of classifying those objects. In general, this traditional pipeline is both unnecessarily computationally expensive and lacks robustness.

Figure 2.1 Color space thresholding of citrus fruit (Shamir, 2006)

The current state-of-the-art method for object detection is using deep neural networks (DNN). More specifically, the use of convolutional neural networks (CNN) has drastically improved object detection capabilities. Traditional DNNs contain several fully connected layers, which can be seen in Figure 2.2. CNNs (Fig. 2.3) are DNNs designed specifically to work with multi-dimensional data. CNNs have several types of layers including convolutional, pooling, and fully connected. Each layer acts as a feature map, performing different mathematical operations to extract features without being directly told what to look for by humans. Essentially, this allows the CNN model to learn the features that aid in object detection directly from the input data itself. For tasks where it is difficult to describe and program a model to detect what makes an object different from the background, CNNs are incredibly useful.

2.1.3 Literature Review

Traditionally, object detection was accomplished by determining a specific feature(s) that differentiates an object, and then developing algorithms to look for those feature characteristics in order to detect an object within an image. A common example of this is color thresholding, which was previously described in section 2.1.2. Several studies have utilized this method in various forms in an effort to detect crop flowers. One such study presented a method for yield prediction in apple orchards by looking at flower density detected through traditional image processing techniques (Aggelopoulou et al., 2011). Their detection process first converted the RGB color image to grayscale and then set an image threshold with a specific value to turn it into a binary image where only white or black pixels remain. The white pixels then corresponded with apple flowers and were used to make predictions of yield based on flower densities. They report an average error of around 18.1%, but this is based on very specific lighting conditions. Additionally, a black screen was placed behind each apple tree to reduce complexity of the image background which contributes to a decrease in error compared to field condition testing. Another study converted images to a hue, saturation, luminance (HSL) color space and applied threshold values based on those three channels in an effort to detect apple flowers in an orchard (Hočevar et al., 2014). While this research is closer to field conditions than the previously mentioned study, it still had drastically different results based on lighting conditions and times of

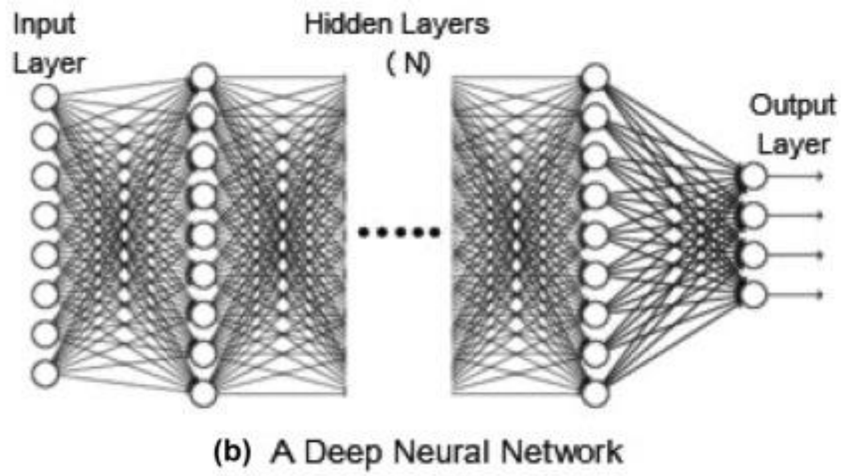

Figure 2.2 Example of a deep neural network architecture (Sarker, 2021)

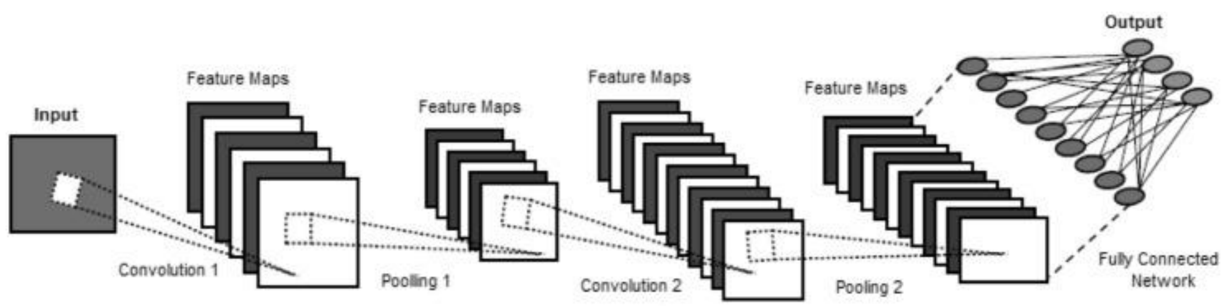

Figure 2.3 Example of a convolutional neural network architecture (Sarker, 2021)

day, making it an overall unreliable system for field settings. One study aimed at detecting tomato flowers attempted to use color thresholding as a detection method, but with the added step of illumination estimation in an effort to correct any issues caused by inconsistencies in lighting (Ting et al., 2012). This research used a reference color region to detect lighting conditions and correct colors in the image based on those conditions. They adjusted brightness, white balance, and gains to compensate for changes in color and intensity caused by fluctuations in lighting conditions. While the study did not provide numerical results, they did present before and after comparisons indicating an improvement in color segmentation after applying this additional technique. Overall, while color segmentation using threshold values is a widely researched technique with several adjustments made to improve results, its accuracy is too variable to be considered a robust and reliable method for flower detection.

Deep learning is quickly becoming the state-of-the-art technique for machine vision tasks such as object detection due to its ability to address many different problems that may be difficult to directly program. Several studies have investigated the use of deep learning models for crop flower detection. Dias et al. (2018) used a combined CNN and support vector machine (SVM) approach to detect apple flowers in real-world field conditions. They broke their process into three steps: region proposals, feature extraction, and classification. The CNN was used for feature extraction, and then they applied a principal component analysis (PCA) to reduce feature dimensionality before using the SVM to classify superpixels identified by the previous steps. The CNN architecture employed in this research was the Clarifai network, and their method resulted in around 92.0% recall and 92.7% precision. This result shows a substantial increase in accuracy compared to the color segmentation methods. Another study presented a method of tomato flower detection using a CNN without the addition of an SVM for classification. This study focuses on optimizing a CNN called Faster Region-based Convolutional Network (R-CNN) on tomato flowers by using a pre-training architecture and applying transfer learning before applying additional fine-tuning methods to increase its sensitivity to tomato flowers (Rahim & Mineno, 2020). They report a recall of 96.0% and a precision of 93.1%. These results are comparable to the other approach presented, but it is also important to note the images from this study were taken in a greenhouse that may allow for some increases in accuracy as compared to more uncontrolled field conditions.

As mentioned in previous sections, deep CNNs are computationally expensive and have intensive power requirements that may be difficult to achieve in some agricultural settings. For this reason, research has been conducted on the implementation of lightweight CNNs that provide similar processing to traditional CNNs architectures, but with fewer layers than many of the popular deep CNNs. One study presented around a 90.3% accuracy on validation data but only a 69.2% accuracy for detecting flower locations in their testing data (Ärje et al., 2019). This result is unsatisfactory overall when compared to other deep CNNs, even if the lightweight CNN is less computationally expensive.

Several other frameworks for deep CNNs exist, with each having different strengths and weakness. Currently, three of the major deep CNNs used for object detection tasks are Single Shot Detection (SSD), Faster Region based Convolutional Neural Networks (Faster R-CNN) and You Only Look Once (YOLO). There are several improvements upon each of these models, but in general they have certain attributes based on the general architecture of each that make them useful. YOLO is currently considered one of the most efficient models, as each improvement on the original has addressed previous issues making the YOLO models fast with an overall good performance, but the decision on which model to use for research should be based on many factors (Srivastava et al., 2021). For instance, if speed is not a concern and the dataset is relatively small Faster R-CNN might be a better choice.

In research regarding flower detection for pollination, a few of the YOLO models have been explored. One study looked at using YOLOv4 for real-time apple flower detection (Wu et al., 2020). The results of this method showed around 98.2% recall and 89.4% accuracy with a detection speed of 72 frames per second. They also compared their results with results from Faster R-CNN, YOLOv2, SSD 300, YOLOv3, and EfficientDet-D5 and found that YOLOv4 outperformed these models overall in terms of both recall and accuracy. An additional study presented an apple flower detection method based on applying a generative method to YOLOv5 (Zhang et al., 2021). Before applying a generative method, models were tested for accuracy and YOLOv5 was found to have a recall of 92.8%, a precision of 87.1%, and a mean average precision (mAP) of 91.8%, making it one of the most accurate models compared to the several other models tested. After the addition of the generative model there was an improvement of detection on smaller objects. Overall, YOLO models, while slightly behind the accuracy given by some Faster R-CNN models, have shown to be both accurate and fast as compared to most other models in a review of current literature on the topic.

2.2 Methods

2.2.1 Data Collection and System Design

Based on literature review, the complexity of an image and its background plays a significant role in the ability for any method to accurately detect flowers. Several studies chose to use a background screen to separate plants of interest or chose to take images at night using artificial illumination. Because this research aims to serve as a first step towards a robust pollination system, it was important to collect images in a way that is more realistic for applications in agricultural production settings. For this reason, an overhead view of the plants was selected in order to reduce background complexity and mimic the way images could be captured realistically in a field setting. The overhead view allows for the soil to act as a solid background. It also minimizes confusion on whether a detected flower is present on one specific plant or another plant in the background of the image. Based on observations, a majority of flowers on tomato plants can be seen from an overhead view, and if any flowers are occluded in one image, they can often be seen in another overhead image slightly offset from the initial position of the previous image. For this reason, the images collected contain a high degree of overlap at around 99%, which is helpful for detecting as many flowers as possible, even with occlusion, and will allow for three-dimensional mapping of the plants, which will be discussed in Chapter 3.

In order to collect overhead images with this high degree of overlap, a raised-bed garden with a robotic gantry system mounted on top was used for data collection. The raised-bed garden measured approximately 9-ft 10-in by 3-ft 10-in. Fourteen plants were placed in the raised-bed garden, which was placed outdoors in ambient weather conditions. Three of the plants were Big Boy tomatoes, and the remaining were Roma tomatoes. The selection of tomato types was based on flower size and plant availability later in the growing season. The raised-bed garden layout can be seen in Figure 2.4. The gantry system used for data collection was the Farmbot Genesis v1.5 (Cruz et al., 2014), which was mounted on the raised- bed garden (Fig. 2.5). A Logitech StreamCam was mounted on the gantry system approximately 3-ft 10-in above soil level and was used to collect the image data. The Logitech camera captures red, green, blue (RGB) images, has a 1080p/60fps maximum video resolution, and a 78-degree field of view. The system was controlled by a Raspberry Pi 3 Model B, which communicated with the Farmduino microcontroller on the FarmBot system, to control movements. Image collection was accomplished using Python scripts to move along a set path (Fig. 2.6) and collect several images with 99% overlap at a rate of 10 frames per second.

In addition to the set-up described above, several adaptations were made to the data collection process before finalizing the design. The initial system was set-up indoors with grow lights to reduce environmental effects on data collection, but this resulted in tomato plants not producing flowers. Since flower production is essential to this research, the raised-bed garden was moved outdoors during the next growing season, where flowering and plant growth drastically increased to expected levels. Once outside, the camera began having issues in bright sunlight conditions which caused the flowers and leaves to become over-exposed in the images. Even with changes to camera settings, the sunlight was too intense and therefore a canopy was placed over the system for data collection during bright times of the day. The canopy was not used for data collection during overcast or evening times. Additionally, initial images were collected at a lower resolution in an effort to reduce computational load later in the process, but testing determined that a higher resolution was necessary for accurate results, so the resolution was increased to 1080p for the remainder of the data collection. The resulting image data used in this research was collected September 2022 through November 2022.

2.2.2 Data Pre-processing

A subset of images was selected to incorporate a representative sample from the dataset including images with only a single flower and images with a large number of flowers. Special attention was paid to the selection of images from several vantage points and from different dates throughout the image collection timeline, so as to ensure the data was representative of several

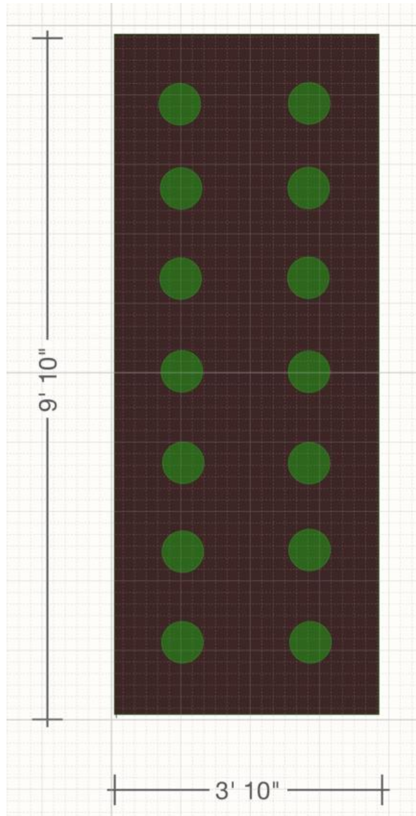

Figure 2.4 Layout of tomato plants in raised-bed garden

Figure 2.5 Web camera mounted on the Farmbot gantry system

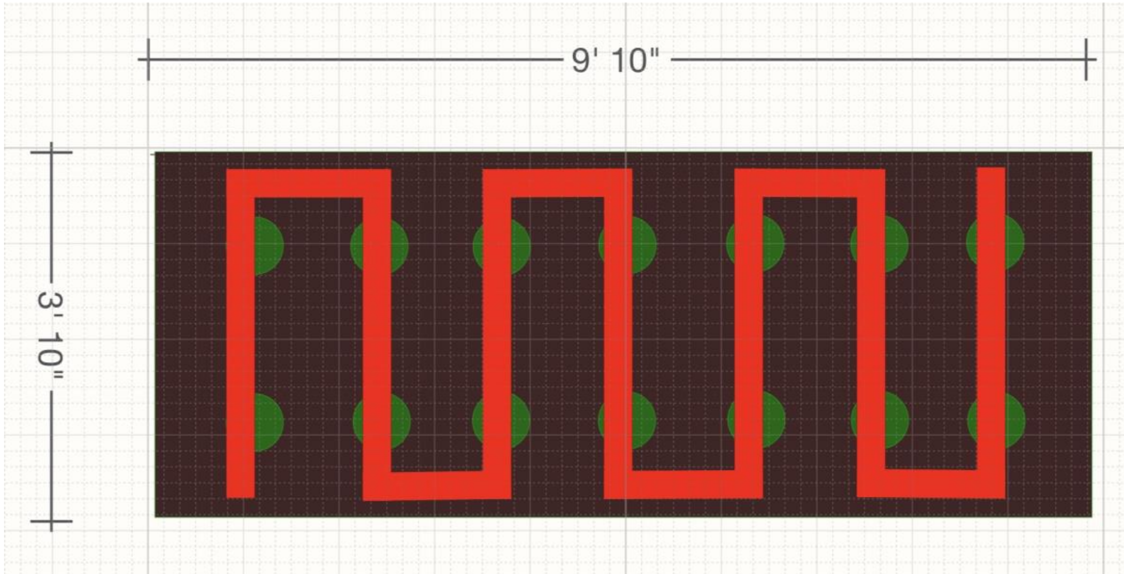

Figure 2.6 Image collection path over raised-bed garden shown in red

different flowering stages of multiple plants. As previously mentioned, collected images were 1080 by 1920 pixels in size. A large portion of the collected image data was saved for three-dimensional models, which will be discussed in Chapter 3. There was a total of 273 images used for the flower detection task. For deep learning models, the data were separated into 60% training and 40% testing. For YOLO models, the testing data were further split into 50% testing and 50% validation. Each flower in the images was hand-labeled using the LabelImg (Tzutalin, 2015) annotating software to draw bounding boxes, which were then used in the deep learning model training. No additional data augmentation techniques were used in this research outside of augmentation built into the CNN models.

2.2.3 Models

For the purposes of this research, multiple object detection techniques were tested on tomato flower images. The first technique tested was color segmentation. Several images of tomato flowers were processed using different threshold values for yellow in the HSV color space in order to separate them from the background and detect their locations. This was achieved using a python script to find all pixels considered yellow by the using a lower threshold value of (20, 100, 100) and an upper threshold value of (30, 255, 255). Then a red contour was drawn around the yellow values to identify them as flowers. Sections of yellow pixels were filtered using size constraints in order to avoid bounding extremely small or extremely large sections that were unlikely to be flowers. Additional image processing techniques such as image blurring and erosion were also tested in an effort to eliminate noise and improve accuracy.

Due to the desire to limit unnecessary computational complexity for the purpose of agricultural field use, a light-weight neural network was explored before attempting other more complex networks. The light-weight CNN chosen was MobileNetv2 (Sandler et al., 2018). MobileNetv2 is a deep CNN specifically designed to run in mobile and resource constrained environments. For this research, MobileNetv2 is used as a feature extractor with SSDLite, a variant of regular SSD, in order to achieve object detection. In previous studies, MobileNetv2 with SSDLite outperforms YOLOv2 and other SSD variants on the COCO dataset object detection task. COCO stands for Common Objects in Context and is a dataset commonly used as a metric to compare different computer vision tasks. MobileNetv2 was chosen due to its comparatively high accuracy over other light-weight CNNs.

In addition to a light-weight CNN model, this study also compares the accuracy of several YOLO models, which many consider to be state-of-the-art for object detection. It is worth noting that many in the computer vision community only consider YOLO versions one through four and version seven, as “official” versions. YOLOv5 and YOLOv6 were created by separate creators but were heavily based upon the previous YOLO architectures. For the purpose of this paper, all of the YOLO models will be classified together. All of the YOLO models are considered to be single-stage object detectors. The first YOLO model explored is YOLOv5 (Jocher et al., 2020). No peer-reviewed study has been published about YOLOv5, so all advantages and disadvantages of the model are derived from researchers comparing YOLOv5 performance metrics to other models. YOLOv5 improves on previous models most notably by solving the “small object problem,” where traditionally smaller objects are detected at a lower accuracy than larger objects (Jiang et al., 2022). For this reason, YOLOv5 was chosen as the first YOLO model to explore in this research. YOLOv5 comes in several different sizes, four of which were tested on the tomato flower data. The other two YOLO models explored include YOLOv6 (Li et al., 2022) and YOLOv7 (Wang et al., 2022). YOLOv6 reports a higher performance on the COCO dataset than previous models in both mAP and in speed. Several size options are also available for YOLOv6 and four were explored in this research. Finally, YOLOv7 claims to be the fastest of all previous models while retaining accuracy. While YOLOv7 offers a few different model versions including a tiny version and one optimized for cloud computing, only the basic model was explored in this research with three different batch sizes.

2.2.4 Metrics Used for Evaluation

Several metrics are used to quantify the performance of different models on the tomato flower detection task. Mean Average Precison (mAP) is used to determine how well the models bound the flowers compared to labeled bounding boxes. The calculation for mAP is shown in Equation 2.1 Precision is the number of actual flowers out of all predicted flowers and is calculated by Equation 2.2. Recall is the number of actual flowers found out of the total number of flowers and is calculated by Equation 2.3. The F1 score is a harmonic mean between the precision and recall values, and the calculation is shown in Equation 2.4. The F1 scores provides an estimate of the overall model performance, considering both the precision and recall.

$$mAP = \frac{1}{n}\sum_{k=1}^{k=n} APk \tag{2.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{2.3}$$

$$F1\ Score = 2\ * \frac{Precision * Recall}{Precision + Recall} \tag{2.4}$$

where,

n = number of bounding boxes

AP = average precision

TP = true positive

FP = false positive

FN = false negative

2.3 Results

2.3.1 HSV Color Segmentation

Several images were tested with different threshold HSV values for yellow, and the results varied drastically based on the lighting and image contents. Two examples of yellow segmentation with the background masked are shown in Figures 2.7 and 2.8. Figure 2.7 shows significant noise, attributed to other parts of the plant being a similar yellow color to the flower.

Figure 2.7 HSV segmentation of yellow color in tomato plant image

Figure 2.8 Erosion on segmented pixels from tomato plant image

This image was then further processed, using an erosion image processing technique to reduce noise. The results of this process can be seen in Figure 2.8. It can also be noted that while it had a significant noise reduction effect, there are still non-flower segments of the image being identified by the yellow threshold. Further processing of the image attempted to draw a red contour around the flower (Fig. 2.9), but the contour does not cleanly segment the flower, and other non-flower objects in the image are also being bound by the red contour. For this reason, further experimentation with this technique was halted as it was not considered a robust enough method to be utilized in this research.

2.3.2 MobileNetv2

MobileNetv2 was trained using TensorFlow (Abadi et al., 2016) and results were evaluated during the testing process. MobileNetv2 was trained on a computer with an Intel Core i7-10750H central processing unit (CPU) with 6 cores and 12 threads, 16GB RAM, and Intel UHD graphical processing unit (GPU). The metrics from MobileNetv2 evaluated on testing data can be seen in Table 2.1. Overall, while the mAP was moderate, the precision and recall for this model are too low for this to be a potential option for tomato flower detection in this research. While the light-weight architecture is appealing for this application, the accuracy was not high enough to move forward with further research using this model.

2.3.3 YOLOv5

Several different sizes of the YOLOv5 model were trained and evaluated. The sizes tested were nano, small, medium, and large. The standard structure of the YOLOv5 model stays the same amongst sizes, but they differ in the number of tunable parameters, which means they also differ in training time and inference time. Typically, the larger model size with more parameters offers an increase in accuracy at the expense of training and inference time. The selection of size depends on the goal of the application. If the application is on a mobile device with less computational resources the nano size might be the best option, but if the model is run on a computer with adequate resources the large size may be the best option. Each model was

Figure 2.9 Red contours shown around HSV segmented pixels

Table 2.1 MobileNetv2 Results

	mAP 0.5	**Precision**	**Recall**	**F1**
Standard Model	0.822	0.399	0.543	0.460

trained using a batch size of 16 for 600 epochs max, but some models stopped training sooner if improvements in performance became negligible. YOLOv5 was trained on a computer with an AMD Ryzen threadripper pro 3975wx CPU with 32-cores and 64 threads, 264GB RAM, and NVIDIA RTX A6000 GPU with 48GB VRAM. Results for all of the sizes can be seen in Table 2.2. Overall, the results from the YOLOv5 models showed drastic improvement over the MobileNetv2 model.

2.3.4 YOLOv6

Similar to YOLOv5, the YOLOv6 model comes in several sizes. The sizes tested were nano, small, medium, and large. Again, the general structure of the YOLOv6 model remains the same amongst sizes, but the sizes differ in tunable parameters meaning they also have differences in training time and inference time. Each model was trained with a batch size of 16 for 600 epochs max, but some models stopped sooner based on negligible improvements in performance over time. YOLOv6 was trained on a computer with an AMD Ryzen threadripper pro 3975wx CPU with 32-cores and 64 threads, 264GB RAM, and NVIDIA RTX A6000 GPU with 48GB VRAM. Results for all sizes can be seen in Table 2.3. YOLOv6 performed comparable to YOLOv5 with a slight decrease in performance, but still a large improvement over the MobileNetv2 model.

2.3.5 YOLOv7

YOLOv7 has three different models: tiny, basic, and one optimized for cloud-based computing. For this research only the basic size was tested as it is a relatively new YOLO model release, having been officially published in July 2022 (Wang et al., 2022). While there are ways to alter the size and create even more variety to the YOLOv7 models, for the purpose of this research, the basic model gave enough information on the overall performance on this YOLO version compared to previous versions. The YOLOv7 model was trained on a computer with an AMD Ryzen threadripper pro 3975wx CPU with 32-cores and 64 threads, 264GB RAM, and

Table 2.2 YOLOv5 Results

Model Size	mAP 0.5	Precision	Recall	F1	Inference Time (ms)
Nano	0.907	0.948	0.847	0.895	2.8
Small	0.939	0.942	0.888	0.914	4
Medium	0.948	0.97	0.918	0.943	4.4
Large	0.937	0.959	0.904	0.931	4.8

Table 2.3 YOLOv6 Results

Model Size	mAP 0.5	Precision	Recall	F1	Inference Time (ms)
Nano	0.831	0.798	0.75	0.773	1.87
Small	0.842	0.84	0.881	0.841	2.64
Medium	0.874	0.86	0.914	0.867	4.92
Large	0.882	0.87	0.912	0.876	7.07

NVIDIA RTX A6000 GPU with 48GB VRAM. The model was trained for at least 600 epochs, but some stopped earlier if performance improvements became negligible. Results for 3 different batch sizes can be seen in Table 2.4. In general, YOLOv7 still performed well overall compared to MobileNetv2, but had slightly decreased performance compared to other YOLO models tested.

2.4 Discussion

Overall, the use of a deep CNN for object detection proved to be the most accurate method, which is consistent amongst literature. Color segmentation, while the least computationally expensive method, was not robust enough to changes in environmental conditions to be used as a flower detector for real-world applications. Due to limitations in power requirements and computational resources in agricultural settings, the light-weight neural network, MobileNetv2, was researched for this application. MobileNetv2 had a relatively poor performance in terms of flower detection (recall) at around 54.3% and true positive detection (precision) at around 39.9%. It did have a moderate mAP, meaning the model is relatively stable, and while it could be trained further, it likely would not drastically improve the accuracy. YOLOv5 did well across all of the metrics for all four of the model sizes tested. The medium version of YOLOv5 had the best performance amongst all of the models tested in this study. A visual example of the medium weight YOLOv5 model's performance can be seen in Figure 2.10, where the original image of the tomato plants can be seen on the left, and the flower detection with bounding boxes can be seen on the right. The high F1 score of 94.3%, which balances both recall and precision, in combination with a mAP of 94.8% insinuates that this model is both accurate and well trained for the task of flower detection. Additionally, it had a fast average inference speed at 4.4ms compared to the other YOLO models of its same size. YOLOv6 had a similar performance to YOLOv5, but with slightly lower metrics overall. It also outperforms color segmentation and MobileNetv2 by significant margins. YOLOv7 also outperformed both the color segmentation and MobileNetv2 models but had lower accuracy metrics than the other YOLO models tested. Interestingly, the mAP scores are slightly higher for YOLOv7 compared

Table 2.4 YOLOv7 Results

Batch Size	mAP 0.5	Precision	Recall	F1	Inference Time (ms)
8	0.942	0.863	0.892	0.877	5.3
16	0.922	0.898	0.84	0.868	6.2
32	0.859	0.881	0.775	0.825	6.6

Figure 2.10 Original image (left) vs YOLOv5 flower detection results (right)

to YOLOv6. YOLOv7 also has the slowest average inference times compared to the other two models in the medium size range. This is an interesting result because in literature one of its stated improvements over previous YOLO models was its speed. The results from this research did not support that claim. In general, all of the YOLO models performed relatively well, but the medium weight YOLOv5 model had the best performance in tomato flower detection and localization overall.

Chapter 3: Tomato Flower Three-Dimensional Mapping

3.1 Introduction

3.1.1 Overview

One key aspect when designing a robust pollination system is obtaining the locations of the flowers that the system intends to pollinate. There are generally two different methods for locating flowers, the first is real-time detection and pollination of flowers and the second is the mapping of flower locations relative to a predefined origin for visitation and pollination at a later time. Doing real-time detection and pollination typically has high computational and power requirements associated with the task, limiting current applications. While this is ultimately the end goal, the process of mapping and revisiting flowers is more realistic for real-world field applications given the current technology. Mapping and revisiting flowers would require data collection to happen in the field, but the heavier processing such as flower detection and mapping could occur offsite. Given the state of agricultural technology reviewed in section 1.3, image collection could be handled by a UAV or other robotic system, and once the data is collected it would then be sent to another location for processing using wireless communication technologies. The secondary processing location would ideally be placed somewhere with adequate access to power enabling the complex computing that is needed for the flower mapping process. Once the process is complete, the 3D locations of the flowers can then be sent to another robotic system, which will visit and pollinate the flowers.

With the goal of flower mapping and revisitation in mind, the objective of this research is to present a method for predicting 3D locations of tomato flowers relative to a predefined origin. Based on research explored in Chapter 2, the CNN YOLOv5 will be used to detect flowers and localize them in two of the three dimensions. The third dimension needed will be the vertical position along the z-axis. This research will explore photogrammetric methods in order to predict the z-coordinates of tomato flowers.

3.1.2 Methods for 3D Coordinate Predictions and Associated Applications in 3D Mapping

Photogrammetry is defined by the American Society for Photogrammetry and Remote Sensing (2023) as “the art, science and technology of obtaining reliable information about physical objects and the environment through processes of recording measuring and interpreting images and patterns of electromagnetic radiant energy and other phenomena.” There are two general types of photogrammetry: aerial and terrestrial (Aber et al., 2010). Aerial photogrammetry requires that the camera is in the air while terrestrial constitutes the camera being handheld or on a tripod. Small format aerial photography combines both by using an aerial vantage point while also maintaining high image detail using a small format camera. Stereophotogrammetry is a subset of photogrammetry specifically focused on deriving 3D coordinates of points on an object visible in two or more photos from different positions. In general, stereophotogrammetry works by using images from different vantage points in order to derive information from them using projection geometry, stereo matching, and additional mathematical and computer vision techniques (Do & Nguyen, 2019). In order for this method to be accurate, the photos need to have a large degree of overlap. These photos can be captured from several cameras specifically set at certain angles or one camera that moves to different positions over time. In addition to using a camera for passive sensing of depth (photogrammetry), active sensors, such as laser scanners and radars, can be used to directly gather 3D information (Remondino, 2011). Many applications use a combination of these 3D imaging techniques, but the choice of which methods to use heavily depends on dimension and surface structures of imaged objects, desired accuracy, location constraints, and more.

One common application of 3D imaging is aerial photogrammetry and mapping. In this application, images are taken from an aerial viewpoint and stitched together to create a 3D map representation of the surveyed area (Nex & Remondino, 2014). The system holding the camera should follow a set-flight path, which ensures proper overlap of images, and the images should then be rectified using ground control points (GCPs) in order to ensure proper orientation, alignment, and scale. Once the images are properly oriented, the creation of the 3D reconstruction can begin, which produces a densified point cloud representing surfaces and geometries found in the images. Once a dense point cloud has been created, an orthomosaic, which can be defined as an orthorectified image mosaicked from the input images, can be created of the scene. There are several commercial software programs available that produce this workflow given the proper image inputs. In general, aerial photogrammetry has been used for 3D mapping of urban, agricultural, and environmental landscapes, but its applications have continued to grow along with UAV technology.

3.1.3 Literature Review

The process of 3D mapping and modeling has been studied with applications in many different fields, including agriculture. Initially, remote sensing techniques for 3D mapping and modeling had a heavy reliance on expensive and complex equipment, but the advancements in UAV technologies have enabled simpler, low-cost options to become more accessible (Jurado et al., 2022). Examples of its use in agriculture include 3D reconstructions of terrain, plant morphologies, and more. Common data types for 3D mapping of terrain include thermal, multispectral, and hyperspectral images, but RGB images and other technology, such as ranging sensors, also have broad applications in certain fields. While the use of larger scale 3D mapping of terrain, from an aerial viewpoint, has seen growth in both research and real-life implementations, the localization of up-close agricultural and environmental systems is still considered a difficult problem that has yet to be fully addressed.

A common algorithm used for localization and mapping (SLAM), briefly mentioned in section 1.4.2, typically has a certain level of reliance on detecting surrounding structures, which are not as readily available in agricultural settings. One study modified SLAM for an agricultural setting using 3D light detecting and ranging (LIDAR) data to help map a field and localize a robot's location (Aguiar et al., 2022). Another study presented a similar method for using 3D LIDAR data to map and localize a robot in an agricultural field, but instead based their research on using the LIDAR Odometry and Mapping (LOAM) algorithm to build a 3D map and then localized the robot by comparing 3D scans of an area to the pre-built 3D map (Le et al., 2019). These studies are a step towards the development of a more detail mapping process for agricultural systems, but they only addressed the field as a larger 3D map for which robotic systems can travel through (Fig. 3.1), without including a method for the precise mapping of the plants in the field.

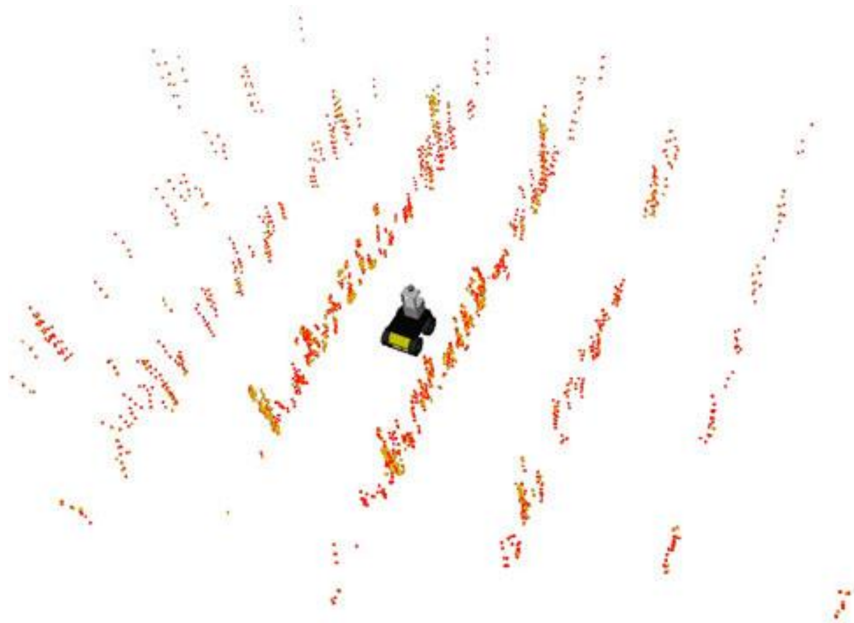

Figure 3.1 Example of Generic Row Crop Detection in a 3D Agricultural Mapping System (Aguiar et al., 2022)

In order to accomplish more concise agricultural tasks, such as harvesting or pollination, a method for 3D mapping crops and their features is needed. In an attempt to address this challenge, a study used LIDAR to map individual maize plants and create 3D point cloud representations (Fig. 3.2) of each (Weiss & Biber, 2011). While this is a useful technique for certain tasks, the LIDAR sensors cannot easily detect specific features on a crop, like flowers or fruits, which is a desired capability for many precision agriculture tasks. For this reason, another study developed a machine vision method, based off of the SLAM algorithm, to map sorghum crops (Qadri & Kantor, 2021). They designed their algorithm to work by using prior knowledge of the structure of agricultural fields, as well as using sorghum seeds as landmarks in the environment. In order to use the seeds as landmarks, they were initially segmented using a CNN, and then feature matched between stereo images in order to determine their relative positions. One final study created 3D models of almond trees in three different orchards by creating photogrammetric points clouds and applying object-based image analysis (OBIA) on them to extract information about the trees from the model (Torres-Sanchez et al., 2018). By using aerial images, the study was able to construct a 3D map of the trees in the orchard including the canopy of the trees and the ground between them. The OBIA techniques were used to characterize features such as individual tree height, width, and volume. The use of photogrammetry in the last two studies allows for better feature extraction than previously mentioned methods and provides a starting point for further development of precision agriculture tasks, such as automated pollination.

3.2 Methods

3.2.1 Data Collection

Based on methods explored in the literature review, the 3D modeling of plants using photogrammetry of an up-close aerial view was chosen as the method to be explored in this research. The set-up for the 3D model data collection system is the same as previously mentioned in Chapter 2. A layout of the tomato plants in the raised-bed garden can be seen in Figure 2.4, and the path followed by the camera for image collection can be seen in Figure 2.6.

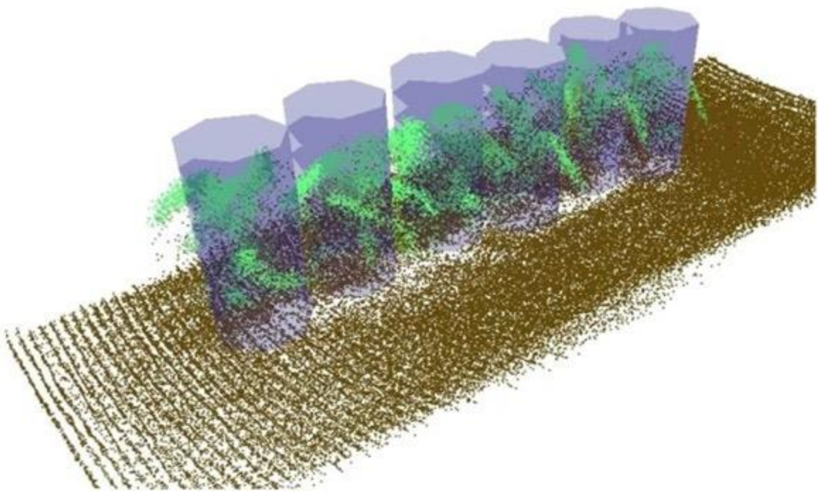

Figure 3.2 Point Cloud Models of Maize using LIDAR sensors (Weiss & Biber, 2011)

As previously mentioned, the camera used was a Logitech StreamCam mounted approximately 3-ft 10-in above soil level, and the images collected had a resolution of 1080 by 1920 pixels. Images were collected along the entire path with 99% overlap at a rate of 10 frames per second. The purpose of this data collection design was to mimic the up-close image collection process that could be carried out by a UAV or similar device. The Farmbot system, described in more detail in Chapter 2, allowed for precise movements and data collection in a grid like pattern in order to produce images oriented in a manner conducive to 3D mapping of the small area, similar to larger-scale 3D mapping applications mentioned in section 3.1.3. Five replications, containing images along the entire path for the raised-bed garden, were fully processed using the workflow described in the following sections in order to predict 3D flower coordinates for the entire raised-bed garden area. The images for the 5 models were taken throughout the data collection timeline on the dates September 21, October 28, November 1, November 6, and November 7 of 2022. For the remainder of this paper, these replications will be referred to as Model 1, Model 2, Model 3, Model 4, and Model 5, respectively.

3.2.2 Pix4Dmapper Modeling

There are several photogrammetry software programs available for 3D mapping. Some of the software packages available remotely connect to UAV devices to help with flight planning and image collection, but because the images in this research were collected using a gantry system to mimic an aerial set-up and not a UAV or other similar device, a software was chosen that focused purely on 3D mapping and modeling from input images. Additionally, the images collected of the tomato plants in the raised-bed garden are much closer to the camera than a traditional UAV flight, and they are not geolocated, meaning GCPs will need to be manually added to help scale the model. Given these requirements, Pix4Dmapper (Pix4D, 2022) was the chosen software as it can turn images into precise 3D maps, even without georeferenced images. Pix4mapper produces several useful outputs, such as densified 3D point clouds, digital surface and terrain models (DSM), orthomosaic images, and many more, which will be used in this research for predicting the 3D locations of tomato flowers.

For each model, approximately 1,000 images were uploaded to the Pix4Dmapper software. The software stitched the images together along the same path they were taken, which can be seen in Figure 3.3. Pix4Dmapper offers several model options, including 3D mapping and 3D modeling. The main difference between the two is that mapping expects images to be from a camera pointed vertically down at the scene, while modeling can use images from different orientations to model an object. For this research, the 3D mapping option was used to recreate the raised-bed garden system. The computer used had an 11th Gen Intel Core i7-11700K CPU, 32GB RAM, and an NVIDIA Quadro RTX 4000 GPU with 8GB VRAM. After the model finished initially processing the images, the software produced a ray cloud, GCPs were manually added, and it was subsequently reprocessed to scale it accordingly. The GPCs used can be seen in Figure 3.4, where three of the corners are used and the two points in the middle of the raised-bed are at the top and bottom of a pole in order to provide GCPs with different z-values. After processing, error values were computed based on the GCPs. Several positions in the models were also compared to measurements of the actual raised-bed in order to determine error of the model produced in all three axes. The six validation points were placed around the raised-bed to create a representative sample of points along various locations on each axis and can be seen in Figure 3.5. After the model was scaled, the software created a densified point cloud (Fig. 3.6), orthomosaic (Fig. 3.7), and DSM (Fig. 3.8). This process was repeated for all five model replications.

3.2.3 YOLOv5 Orthomosaic Flower Detection

After the Pix4Dmapper software produced all outputs, the orthomosaic was then cropped around only the soil area in the raised-bed garden. The cropped orthomosaic was then processed by the YOLOv5 object detection network in order to detect and localize the tomato flowers in the image. Details on training and accuracy of the YOLOv5 were previously discussed in Chapter 2. Flower bounding box coordinates are output by the YOLOv5 model in terms of pixel locations in the cropped orthomosaic image.

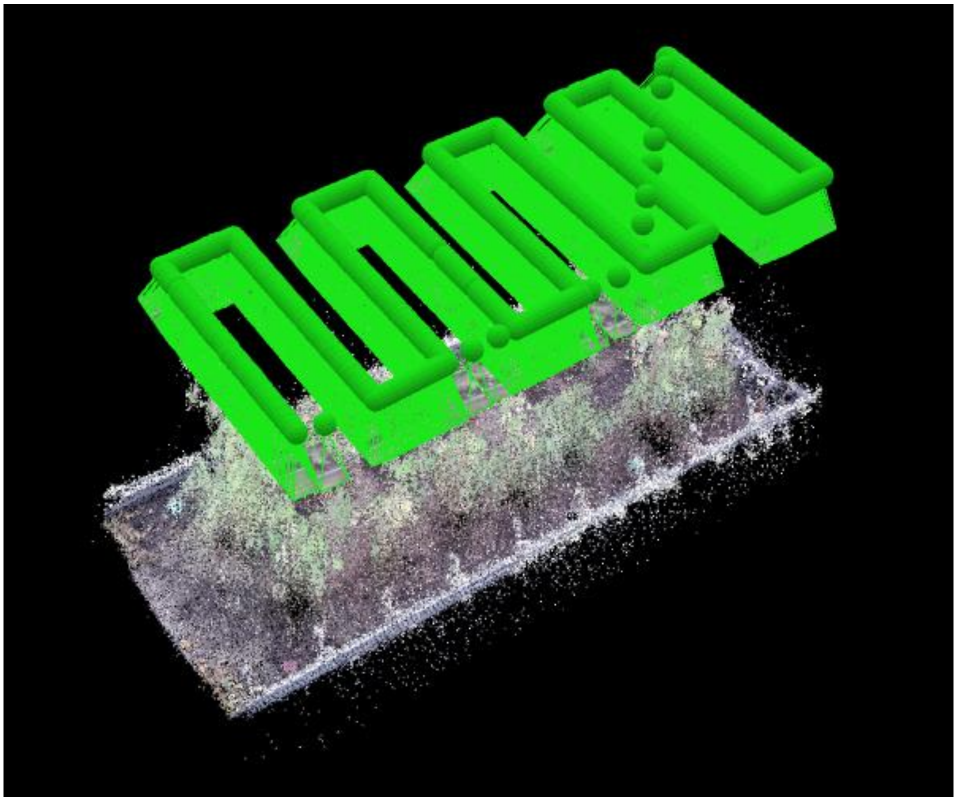

Figure 3.3 Image Placement Path on 3D Map

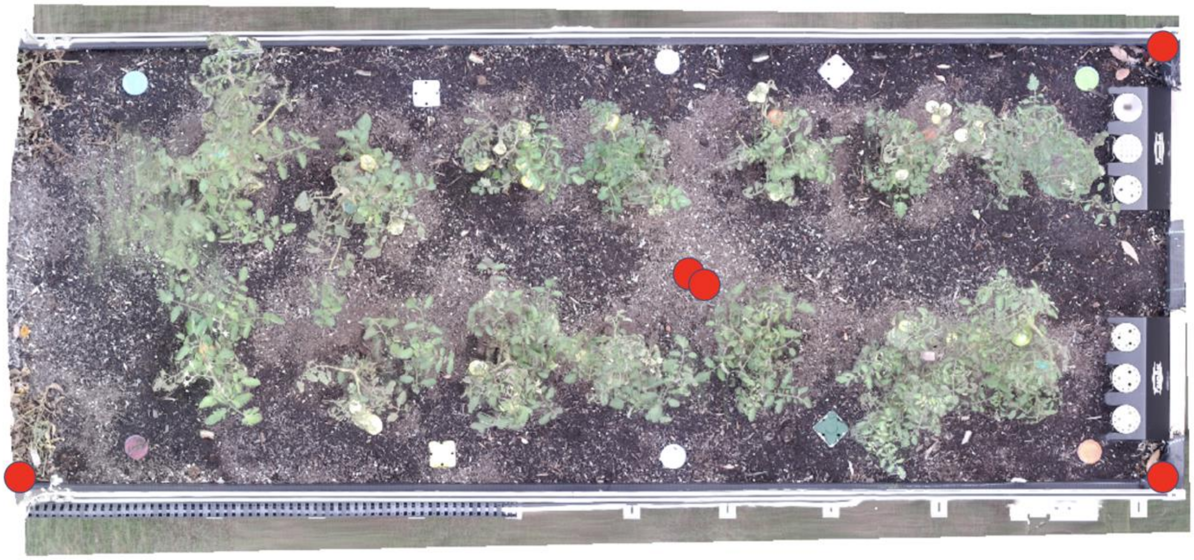

Figure 3.4 Ground Control Point Locations Shown in Red

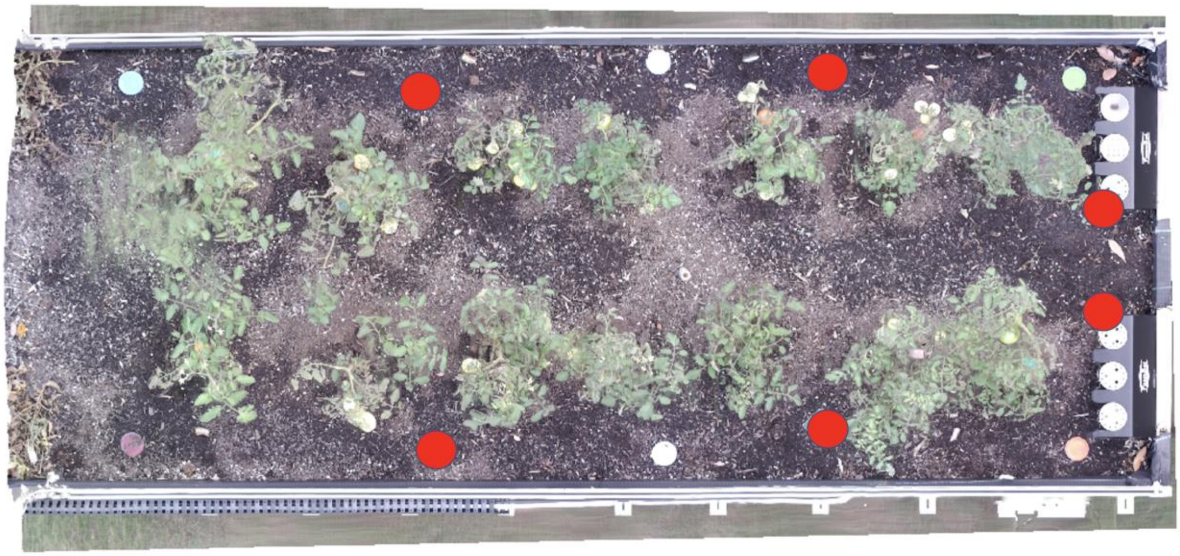

Figure 3.5 Validation Point Locations Shown in Red

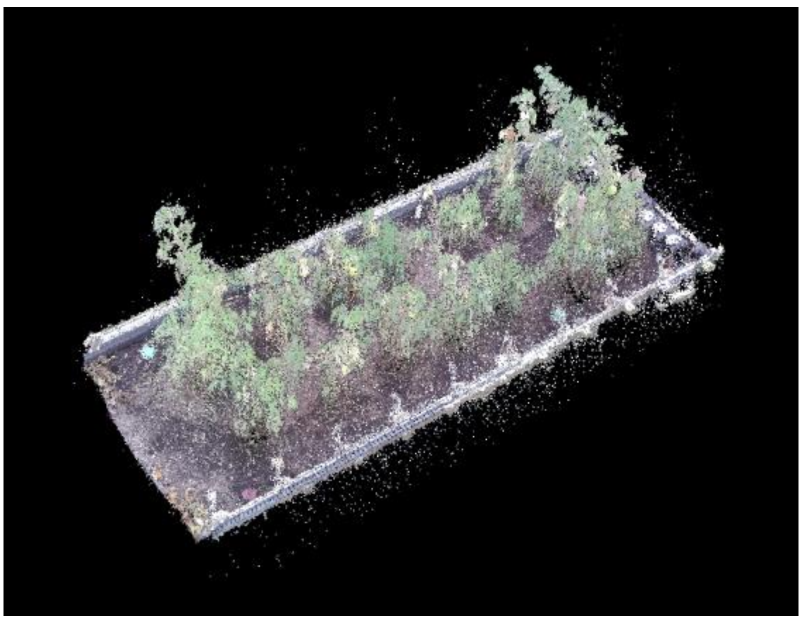

Figure 3.6 Densified Point Cloud

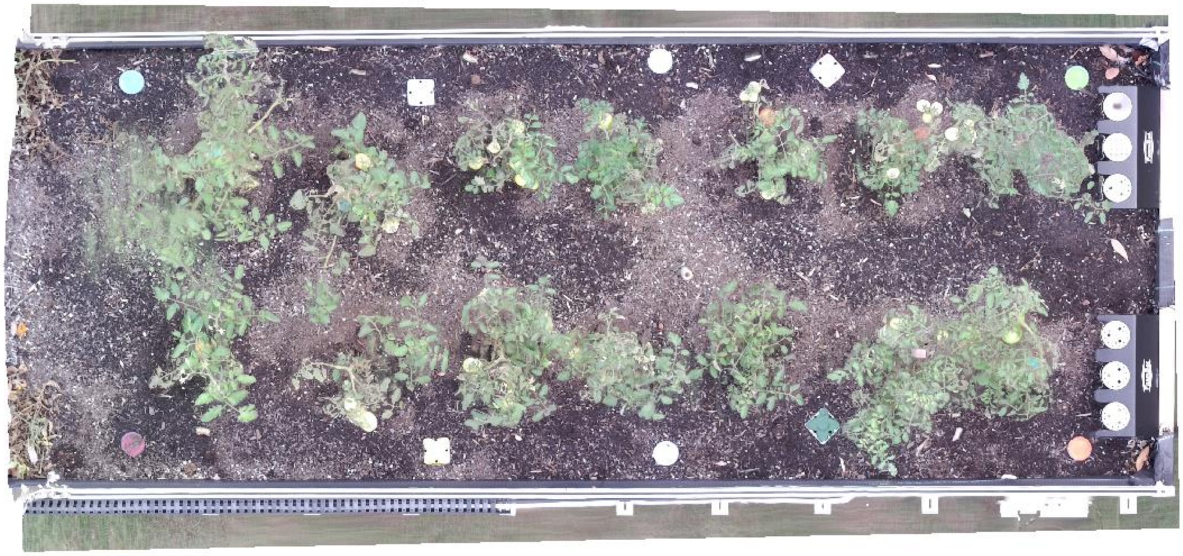

Figure 3.7 Orthomosaic Image

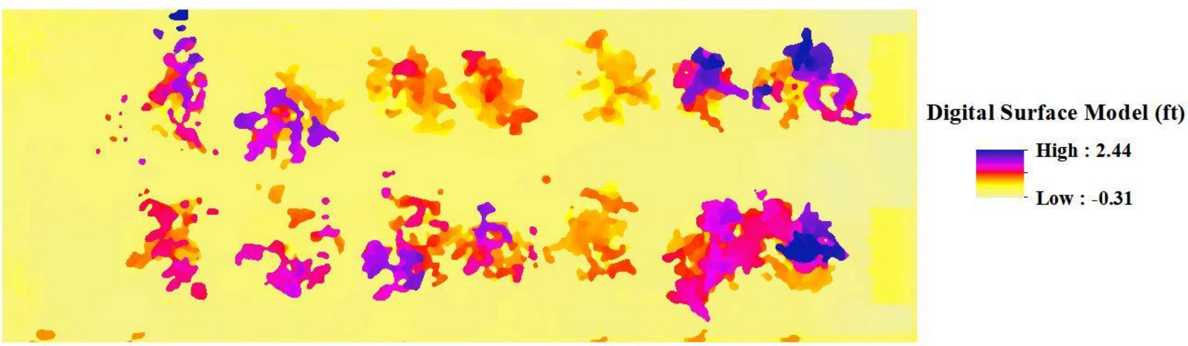

Figure 3.8 Digital Surface Model with Scale

3.2.4 3D Coordinate Predictions

Once bounding box coordinates for each flower are found using the YOLOv5 model, they are then converted to coordinate values in feet. This is accomplished by interpolating between the pixel dimensions of the raised-bed garden in the cropped orthomosaic and the actual dimensions of the raised-bed garden in feet. Once the bounding box coordinates are converted, the midpoint of each box is then calculated and used as the approximate x- and y- coordinates for each flower. These coordinates are then found in the DSM in Pix4Dmapper in order to estimate the z-coordinate using the model. This estimation is based on the assumption that the flower is visible to the camera and is therefore going to have a z-position that is predicted by the DSM. Once the 3D coordinate predictions have been gathered for each flower, these 3D points will then be marked on the densified point cloud as manual tie points in order to assess the accuracy of flower detection.

3.2.5 Complete Workflow of 3D Flower Mapping

The processes for mapping 3D coordinates of the tomato flowers in the raised-bed garden consisted of several steps including data collection, 3D modeling, flower detection with the orthomosaic (x-and y- coordinate prediction), and z-coordinate prediction from the DSM. A visual of the total workflow for this process can be seen in Figure 3.9.

3.3 Results

3.3.1 3D Model Creation Run Times

Due to the complexity of processing 1,000 images using photogrammetry, the time it takes to process each model is an important factor to consider. Pix4Dmapper splits the processing time between the creation of the densified point cloud and the creation of the orthomosaic and DSM models. Table 3.1 shows the run time for each of these processes. In addition to the run time, each model is also scaled using five GCPs that have been manually

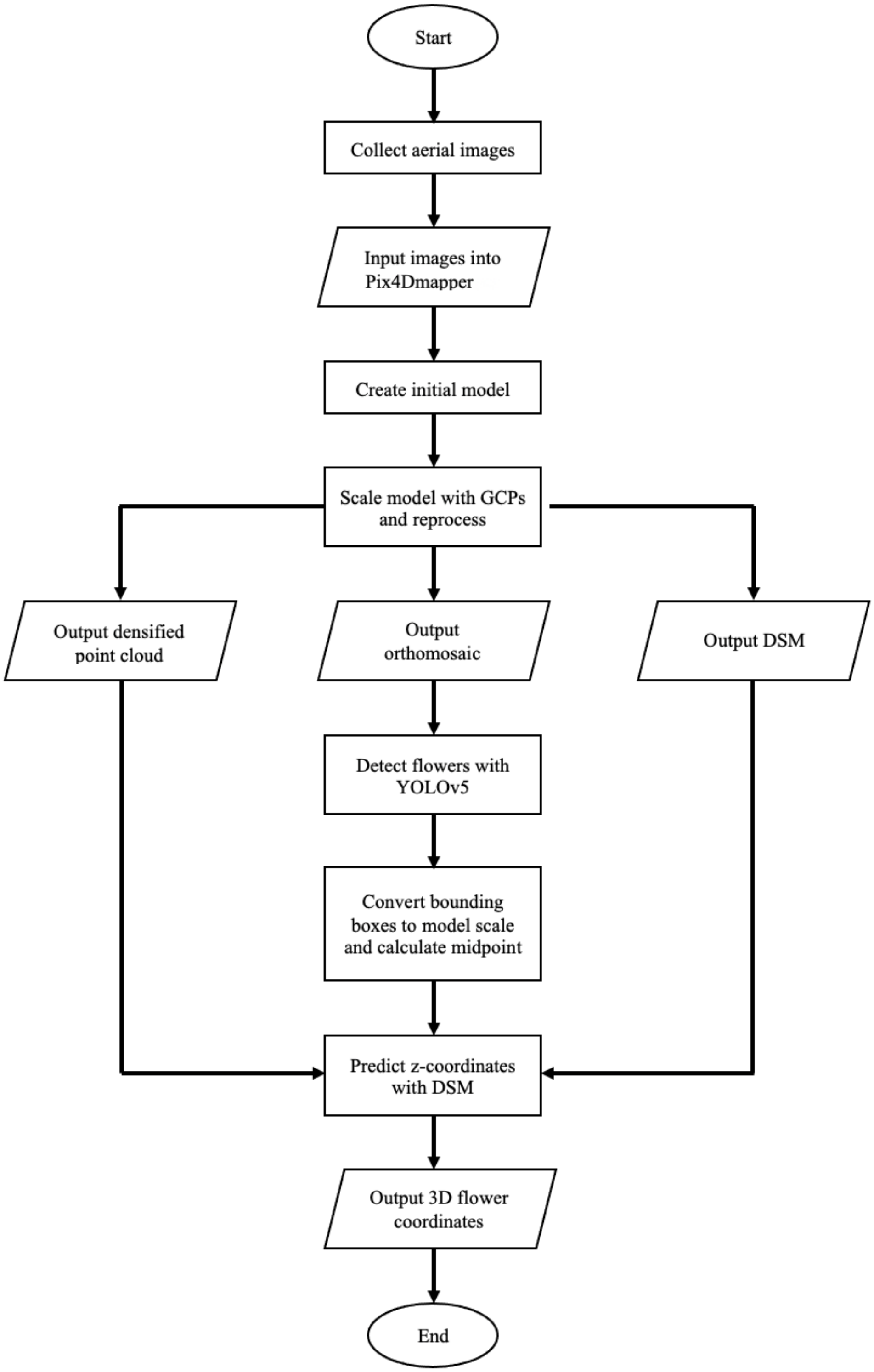

Figure 3.9 Complete 3D Flower Prediction Workflow

Table 3.1 Run Times for Model Construction [min]

Process	Model 1	Model 2	Model 3	Model 4	Model 5
Point Cloud	49.15	22.62	19.83	23.73	23.3
Orthomosaic and DSM	28.38	28.03	19.30	21.97	20.58

labeled on several images since they are not georeferenced. The time for manually marking GCPs in several images and doing the image reprocessing ranged from 2 hrs to around 4 hrs depending on the model reprocessing time and if the GCPs had to be re-marked based on human error when initially labeling the GCPs on the images. Overall, the creation of the orthomosaic and DSM had similar run times, but the creation of the densified point clouds had variations among the models.

3.3.2 3D Model Error

After scaling the models, the error in the model can be estimated in several different ways. The measurements for error in GCPs are mean error (Table 3.2), standard deviation of the error (Table 3.3), and root mean square (RMS) error (Table 3.4). Additionally, six measurements on the actual raised-bed were used to test overall accuracy of measurements in the model compared to the real measurements. These measures included mean error (Table 3.5), standard deviation of the error (Table 3.6), and RMS error (Table 3.7). Compared to the GCP error, the error between the model and the actual measurements is much larger, with the largest error being along the z-axis.

3.3.3 Orthomosaic Flower Visibility

In the orthomosaic image, the foliage of the tomato plants was relatively blurry, but some flowers were still visible. In general, the orthomosaic images do not show some of the flowers, which are visible in the individual images themselves. Table 3.8 shows a comparison of flowers visible in the orthomosaic images for each model compared to the actual number of flowers present in the raised-bed garden. The visible flowers in the orthomosaic images were counted manually by visually observing each of the images.

Table 3.2 GCPs Mean Error [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5	Average
X	0.00105	0.00011	0.00071	0.00001	0.00075	0.00053
Y	0.00200	0.00005	0.00025	0.00012	0.00056	0.00060
Z	0.00399	0.00050	0.00006	0.00082	0.00388	0.00185
Average	0.00235	0.00022	0.00034	0.00032	0.00173	

Table 3.3 GCPs Standard Deviation of the Error [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5	Average
X	0.01314	0.02662	0.01422	0.01054	0.03568	0.02004
Y	0.01404	0.02188	0.01586	0.01226	0.03463	0.01973
Z	0.01723	0.01933	0.02351	0.02198	0.01291	0.01899
Average	0.01480	0.02261	0.01786	0.01493	0.02774	

Table 3.4 GCPs Root Mean Square Error [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5	Average
X	0.01319	0.02662	0.01424	0.01054	0.03568	0.02005
Y	0.01418	0.02188	0.01587	0.01226	0.03464	0.01901
Z	0.01768	0.01934	0.02351	0.02199	0.01348	0.01920
Average	0.01502	0.02261	0.01787	0.01493	0.02793	

Table 3.5 Actual Mean Error of Model [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5	Average
X	0.02376	0.02959	0.03376	0.02709	0.03626	0.03010
Y	0.03604	0.02937	0.03771	0.03771	0.03437	0.03504
Z	0.04167	0.06000	0.05333	0.05667	0.06167	0.05668
Average	0.03382	0.03965	0.04160	0.04049	0.04410	

Table 3.6 Actual Standard Deviation of the Error of Model [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5	Average
X	0.02261	0.02519	0.02716	0.02418	0.02492	0.02481
Y	0.03272	0.02592	0.02851	0.02851	0.03024	0.02918
Z	0.03656	0.05367	0.01633	0.01862	0.04579	0.03419
Average	0.03063	0.03493	0.02400	0.02377	0.03365	

Table 3.7 Actual RMS Error of Model [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5	Average
X	0.03147	0.03747	0.04188	0.03494	0.04280	0.03771
Y	0.04680	0.03772	0.04581	0.04581	0.04408	0.04404
Z	0.05339	0.07749	0.05538	0.05916	0.07449	0.06398
Average	0.04389	0.05089	0.04769	0.04664	0.05379	

Table 3.8 Number of Visible Flowers vs. Number of Actual Flowers

Flowers	Model 1	Model 2	Model 3	Model 4	Model 5
Visible	2	16	20	18	25
Actual	5	28	33	35	39

3.3.4 YOLOv5 Flower Detection

The YOLOv5 model was run on all five orthomosaic images in order to detect and localize the tomato flowers. The inference time for each model is shown in Table 3.9. The visual representations of the YOLOv5 model performance for Model 1 (Fig. 3.10), Model 2 (Fig. 3.11), Model 3 (Fig. 3.12), Model 4 (Fig. 3.13), and Model 5 (Fig. 3.14) show the detected flowers bounded in red, along with corresponding confidence values. A confusion matrix for each model is shown in Tables 3.10 – 3.14, which provides more detailed information on the YOLOv5 model's performance for each orthomosaic image. When constructing the confusion matrix, the total number of flowers present in the orthomosaic was considered to be the number of flowers visually observed rather than the total amount present since this error was already addressed in section 3.3.3.

3.3.5 3D Coordinate Predictions

The midpoint for each flower, calculated from the bounding box predictions, was then manually identified on the Pix4Dmapper DSM, and a z-coordinate for each flower was predicted. The predicted 3D coordinates were then compared to the densified point cloud to determine the accuracy of the 3D predictions compared to the 3D model locations of the flowers. Only points that were considered true flower detections were included in error calculations. The mean, standard deviation, and RMS error for each axis on each of the five models can be seen in Tables 3.15-3.17. Two outlier points were left out of the error calculations (one in Model 3 and one in Model 4) in order to prevent skewing of the error results. In both cases, the z-predication was completely incorrect by a large margin, likely caused by errors in previous parts of the modeling process.

Table 3.9 Inference Time [ms] for Orthomosaic Images

	Model 1	Model 2	Model 3	Model 4	Model 5
Inference Time	24.3	34	30.8	31.3	39.4

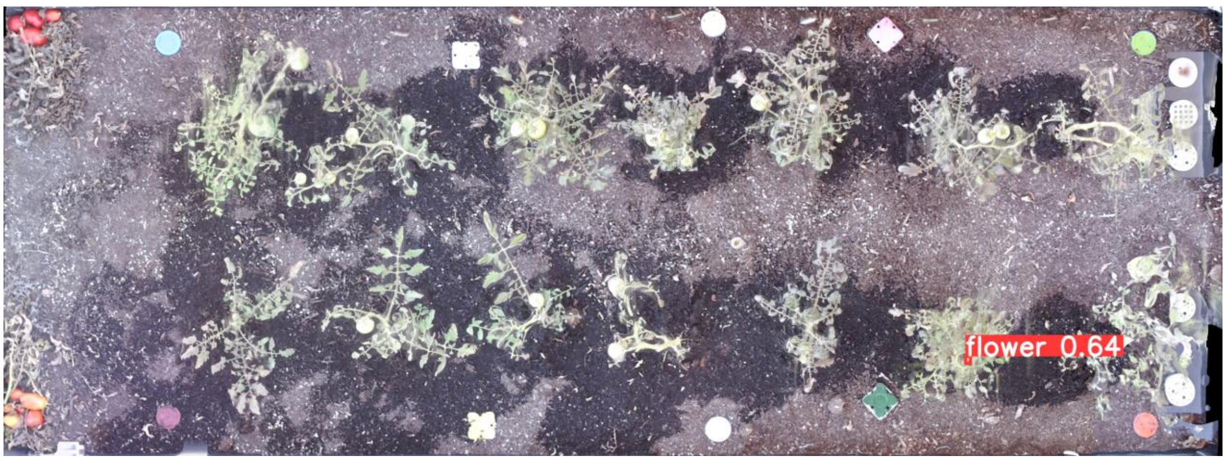

Figure 3.10 YOLOv5 Flower Detection Results for Model 1

Figure 3.11 YOLOv5 Flower Detection Results for Model 2

Figure 3.12 YOLOv5 Flower Detection Results for Model 3

Figure 3.13 YOLOv5 Flower Detection Results for Model 4

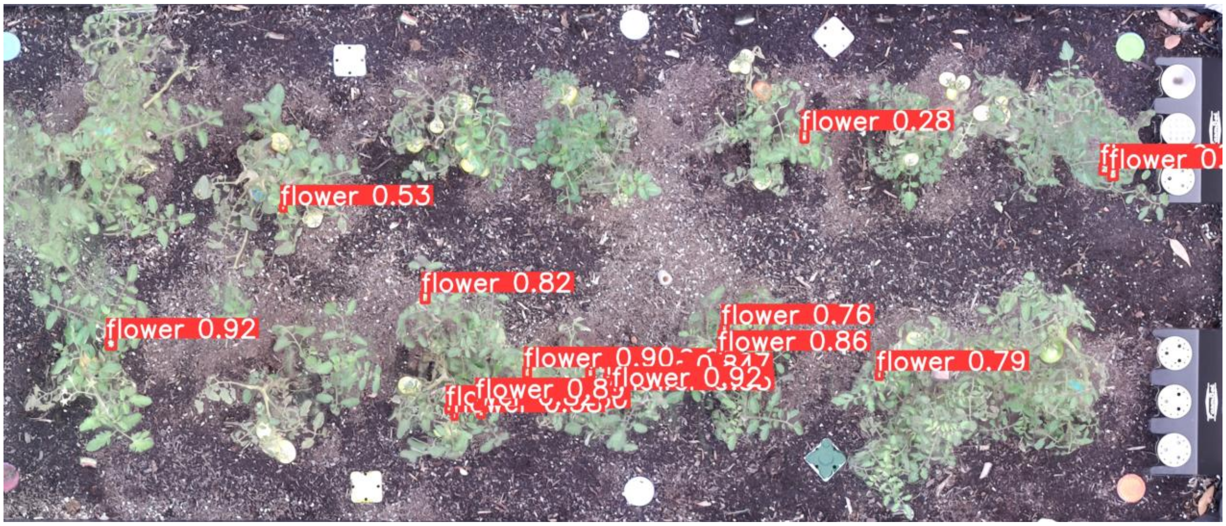

Figure 3.14 YOLOv5 Flower Detection Results for Model 5

Table 3.10 Model 1 Confusion Matrix

	Actual Flower	**Actual No Flower**
Predicted Flower	1	0
Predicted No Flower	1	0

Table 3.11 Model 2 Confusion Matrix

	Actual Flower	**Actual No Flower**
Predicted Flower	8	1
Predicted No Flower	8	0

Table 3.12 Model 3 Confusion Matrix

	Actual Flower	**Actual No Flower**
Predicted Flower	8	2
Predicted No Flower	12	0

Table 3.13 Model 4 Confusion Matrix

	Actual Flower	**Actual No Flower**
Predicted Flower	9	1
Predicted No Flower	9	0

Table 3.14 Model 5 Confusion Matrix

	Actual Flower	**Actual No Flower**
Predicted Flower	18	2
Predicted No Flower	7	0

Table 3.15 3D Coordinate Mean Error [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5
X	0.00000	0.00000	0.00360	0.00125	0.00200
Y	0.00000	0.00305	0.00428	0.00500	0.00000
Z	0.00000	0.01393	0.00571	0.00500	0.07278
Average	0.00000	0.01393	0.00453	0.00375	0.02493

Table 3.16 3D Coordinate Standard Deviation of Error [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5
X	0.00000	0.00000	0.00951	0.00354	0.00522
Y	0.00000	0.00862	0.01132	0.01414	0.00000
Z	0.00000	0.08642	0.00787	0.01414	0.20134
Average	0.00000	0.03168	0.00956	0.01061	0.06886

Table 3.17 3D Coordinate RMS Error [ft]

Axis	Model 1	Model 2	Model 3	Model 4	Model 5
X	0.00000	0.00000	0.00951	0.00354	0.00546
Y	0.00000	0.00863	0.01132	0.01414	0.00000
Z	0.00000	0.08965	0.00926	0.01414	0.20877
Average	0.00000	0.03276	0.01003	0.01061	0.07141

3.4 Discussion

3.4.1 Pix4Dmapper Model Creation

Among all models, the process of producing the point cloud, orthomosaic, and DSM took around an hour. Initial processing of data and scaling the model could take between 2 hrs – 5 hrs. This would estimate the total processing time for the 3D model to be around 3 hrs – 6 hrs total, which is too long for field applications considering the size of the raised-bed garden compared to a field used for agricultural production. Some of this time can be attributed to the manual marking of the GCPs but automating this process in future research would reduce some of the total time required to create the 3D model.

After adding GCPs to accurately orient and scale the model, the resulting error between the GCPs marked positions and the positions computed by Pix4Dmapper can be used to assess the accuracy of the model relative to the GCPs. In general, the mean error was relatively low with the maximum average amongst the models being less than 3/100s of an inch. The standard deviation was slightly higher with an average max of around 1/3 of an inch, meaning there is likely a large amount of variability among the error in each axis depending on the ground control point. This is similarly reflected in the RMS values. This is likely attributed to human errors in marking GCPs on the model, specifically in the ability to accurately reflect the z-coordinate values considering the soil level is highly varied across the garden area. Furthermore, the locations of the GCPs used could contribute to the error, especially in the z-axis where the majority of points were at the soil level. The addition of GCPs that are more varied in the z-axis could improve accuracy overall.

Additionally, measurements on the model were compared to actual measurements taken from the raised-bed garden at six different points in order to determine the error. Overall, this error was higher than the error seen with the GCPs. The highest average mean error was around 1/2 of an inch with the standard deviation close to 1/3 of an inch and the RMS error close to 1/2 of an inch. Again, a significant source of error seen between the model and the real-world is introduced in the z-axis as it is difficult to accurately measure the z-position of points in the soil. The addition of GCPs that are more representative of variations in the z-axis could help. In future work, a reliable method for determining z-coordinates will need to be developed in order to increase the relative accuracy of the 3D models.

3.4.2 Tomato Flower Detection and Localization

Looking at the results for the orthomosaic image, there seems to be a large discrepancy between the actual number of flowers present and the ones that are visible in the orthomosaic. When looking at the input images, the majority of flowers on the tomato plants are visible, but this same result was not seen in the orthomosaic image. This is likely due to how the orthomosaic is constructed using all of the input images, which causes blur around the foliage and likely covers many of the flowers. In future research, flower detection will likely need to occur in the individual images before the 3D model creation, or a better method for developing a clearer orthomosaic image will need to be investigated.

The analysis of the results of YOLOv5 on the orthomosaic image are based upon the number of visible flowers in the orthomosaic. In general, the YOLOv5 model produced a low number of false positives, meaning the majority of flowers detected were truly flowers. This indicates that the YOLOv5 model had a relatively high precision across the orthomosaic images. Unfortunately, only about 50% of visible flowers were detected by the YOLOv5 model on a majority of the orthomosaic images with the exception of Model 5, which has a slightly higher detection rate. This means the YOLOv5 model had a lower recall value. The reason for this can likely be attributed to distortions in the flower caused by the creation of the orthomosaic image. The YOLOv5 model was trained on the original tomato flower images, not the tomato flowers present in the orthomosaic images, which means if the distorted flower in the orthomosaic images were different enough from the original images they may not be detected. In order to address this issue and increase flower detection accuracy, the YOLOv5 model would either need to be run on the images before the 3D model creation, or it would need to be trained on flowers present in orthomosaic images.

Finally, the accuracy of the 3D coordinate predictions from the flowers detected with the YOLOv5 model and located on the DSM were relatively accurate. There were two outliers where the DSM predicted the z-coordinate to be close to zero for the flowers, but for the most part, the 3D coordinates, which were predicted for each flower were on average under a 1/4 inch in error from the actual flowers seen in the point cloud representation. Some flowers had a larger error in the z-direction, specifically in Model 5, which is likely due to errors in the 3D model produced earlier in the process. Overall, considering the tomato plants are not completely stationary and have slight movements that may alter their positions, the resulting predictions were relatively accurate based on the 3D models.

Chapter 4: Conclusions

4.1 Introduction

The main goal of this research was to investigate methods for crop flower detection and 3D mapping as an initial step towards the development of a robust automated pollination system. The first objective was to determine an accurate method for flower detection through the exploration of machine vision and deep learning techniques. The second objective was to predict the 3D coordinates of flowers, relative to a predefined origin, through the exploration of photogrammetry as a method for 3D reconstruction of the crops. A summary of the findings of this research, its contributions to the field, its limitations, and future work will be discussed further in this chapter.

4.2 Research Findings

4.2.1 Flower Detection Methods

Several methods of flower detection were explored including color segmentation and object detection using four different deep CNN architectures. The most accurate methods were the YOLO-based neural networks, with YOLOv5 having the highest accuracy on the validation data with an F1 score of 94.3%. YOLOv5 was therefore used to detect flowers in the orthomosaic images produced by the Pix4Dmapper software. The creation of the orthomosaic images seemed to cause areas of deep foliage to have a significant blur resulting in the loss of visibility of a large percentage of flowers present in the raised-bed garden. Additionally, several of the flowers that were still visible in the orthomosaic images had significant distortions introduced by the creation of the orthomosaic, which affected the accuracy of the YOLOv5 model. Even with these complications, the model performed moderately well and detected about 50% of the visible flowers in the orthomosaic images.

4.2.2 3D Flower Mapping

The 3D model recreation had a relatively low mean error in terms of GCP locations compared to computed locations in the model, but there was a high degree of variability, which can likely be attributed to human error in collecting accurate measurements along the z-axis. Additionally, the error calculated between the model's predicted measurements and the actual measured values was larger at around half of an inch. Again, this is likely due to human error in the collection of measurements used to scale the 3D model. The processing time for each model was between 2 hrs – 5 hrs. Overall, even with the error values, the software did well in the overall 3D reconstruction of the plants in the raised-bed garden. Once flowers were detected using the YOLOv5 model, the DSM was used to predict z-coordinates. Compared to the 3D locations of the flowers on the densified point cloud, the flower locations predicted had a relatively lower error overall, with the exception being in Model 5 where a larger error in the z-axis was seen. If scaling of the model can be improved and the visibility of the flowers in the orthomosaic addressed, this technique may be applicable for the mapping of 3D flower coordinates of crops.

4.2.3 Limitations in this Work

There were several complications in the collection of data for this research such as issues with growing tomato plants in the raised-bed garden and with the collection of clear images of the flowers. For this reason, this work was based on a smaller subset of data than initially planned, which limits the ability for its results to be fully extrapolated to the real world without further research. Ideally, images collected of tomato plants would be spread throughout the growing season and replicated for multiple seasons. This would add robustness to the dataset creating a more realistic representation of the tomato plants overall. Additionally, without access to a greenhouse, the growth of tomato plants was constrained to the warmer months, which limited available time for data collection throughout this study.

4.3 Contributions to the Field of Precision Agriculture

Based on a current review of literature, deep learning has been widely researched for use in precision agriculture as a method for object detection. This research extrapolates that work to the detection of tomato flowers from an overhead view. Additionally, while there are a small number of studies looking at the development of autonomous pollination systems, few attempt to map the 3D locations of flowers. Furthermore, the use of photogrammetry as a 3D recreation technique for detailed mapping of plant features has been rarely studied in literature. This research presents a novel approach to the detection and 3D mapping of crop flowers by integrating deep learning methods for flower detection and photogrammetry for 3D reconstruction of the plants. While there is room for improvement, this method acts as a building block for future work on the detection and 3D mapping of detailed crop features in the field of precision agriculture.

4.4 Future Work

4.4.1 Improvements in Flower Detection

The first recommendation for future work would be the collection of more data in order to build a more robust dataset for training and testing of the methods described in this work. Additionally, due to the reduced visibility of flowers in the orthomosaic images, research should be done on detecting the flowers in the input images before creating an orthomosaic. If flowers are detected, located, and marked as tie-points in the input images, this will hopefully improve the accuracy of flower detection overall when mapping the entire growing area. This may also reduce the number of flowers that are missed by the YOLOv5 model since the flowers in the images will not yet be distorted by the orthomosaic. Additionally, moving the camera farther away from the plants could reduce distortions in the orthomosaic.

4.4.2 Improvements in 3D Mapping

In order to improve the reconstruction of the 3D model, an accurate method of measuring locations of GCPs in all three axes should be determined. The higher error along the z-axis was likely due to the inability to accurately set a datum position that was consistent on the uneven soil. Because the testing area was smaller, compared to typical sizes of aerial mapping projects, even slight deviations in z-axis measurements cause large errors in the model. Future work should aim to find an accurate and precise method to define and measure the origin, datum, and GCPs in the modeled system.

4.5 Summary

This research serves as an initial step towards the design of a robust autonomous pollination system. The objectives of researching flower detection of 3D mapping methods were addressed by combining the use of both deep learning and photogrammetry. Several methods for flower detection were explored, such as color segmentation, light-weight CNNs, and deep CNNs. YOLOv5 was determined to be the most accurate of these methods. The software Pix4Dmapper was researched to apply the technique of photogrammetry to up-close aerial images in order to reconstruct a 3D model of tomato plants. The orthomosaic image created by the software was processed by YOLOv5 to detect visible flowers, and the z-coordinate for those flowers was predicted by the DSM. While this approach had a few errors and limitations, it adequately addressed the research objectives. This research contributes to the field of precision agriculture by presenting a novel approach for crop flower detection and mapping, which has not previously been explored in literature.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., & Isard, M. (2016). Tensorflow: a system for large-scale machine learning. Osdi,

Aber, J. S., Marzolff, I., & Ries, J. B. (2010). Introduction to Small-Format Aerial Photography. In J. S. Aber, I. Marzolff, & J. B. Ries (Eds.), *Small-Format Aerial Photography* (pp. 1-13). Elsevier. https://doi.org/10.1016/b978-0-444-53260-2.10001-8

Aggelopoulou, A. D., Bochtis, D., Fountas, S., Swain, K. C., Gemtos, T. A., & Nanos, G. D. (2011). Yield prediction in apple orchards based on image processing. *Precision Agriculture*, *12*(3), 448-456. https://doi.org/10.1007/s11119-010-9187-0

Aguiar, A. S., Neves dos Santos, F., Sobreira, H., Boaventura-Cunha, J., & Sousa, A. J. (2022). Localization and Mapping on Agriculture Based on Point-Feature Extraction and Semiplanes Segmentation From 3D LiDAR Data [Original Research]. *Frontiers in Robotics and AI*, *9*. https://doi.org/10.3389/frobt.2022.832165

American Society for Photogrammetry and Remote Sensing. (2023). *What is ASPRS?* Retrieved March 15 from https://www.asprs.org/organization/what-is-asprs.html

Ärje, J., Milioris, D., Tran, D. T., Jepsen, J. U., Raitoharju, J., Gabbouj, M., Iosifidis, A., & Høye, T. T. (2019). Automatic Flower Detection and Classification System Using a Light-Weight Convolutional Neural Network.

Barnett, J., Seabright, M., Williams, H., Nejati, M., Scarfe, A., Bell, J., Jones, M., Martinsen, P., Schaare, P., & Duke, M. (2017). Robotic Pollination-Targeting kiwifruit flowers for commercial application.

Bugin, G., Lenzi, L., Ranzani, G., Barisan, L., Porrini, C., Zanella, A., & Bolzonella, C. (2022). Agriculture and Pollinating Insects, No Longer a Choice but a Need: EU Agriculture’s Dependence on Pollinators in the 2007–2019 Period. *Sustainability*, *14*(6), 3644. https://www.mdpi.com/2071-1050/14/6/3644

Chaplin-Kramer, R., Dombeck, E., Gerber, J., Knuth, K. A., Mueller, N. D., Mueller, M., Ziv, G., & Klein, A.-M. (2014). Global malnutrition overlaps with pollinator-dependent micronutrient production. *Proceedings of the Royal Society B: Biological Sciences*, *281*(1794), 20141799. https://doi.org/10.1098/rspb.2014.1799

Cheng, H. D., Jiang, X. H., Sun, Y., & Wang, J. (2001). Color image segmentation: advances and prospects. *Pattern Recognition*, *34*(12), 2259-2281. https://doi.org/https://doi.org/10.1016/S0031-3203(00)00149-7

Cruz, J., Herrington, S., & Rodriguez, B. (2014). Farmbot.

del Cerro, J., Cruz Ulloa, C., Barrientos, A., & de León Rivas, J. (2021). Unmanned Aerial Vehicles in Agriculture: A Survey. *Agronomy*, *11*(2), 203. https://www.mdpi.com/2073-4395/11/2/203

DiDonato, S., & Gareau, B. J. (2022). Be(e)coming pollinators: Beekeeping and perceptions of environmentalism in Massachusetts. *PLoS One*, *17*(3), e0263281. https://doi.org/10.1371/journal.pone.0263281

Do, P. N. B., & Nguyen, Q. C. (2019). A review of stereo-photogrammetry method for 3-D reconstruction in computer vision. 2019 19th International Symposium on Communications and Information Technologies (ISCIT),

The Editors of Encyclopaedia Britannica. (2020). cross-pollination. *Encyclopaedia Britannica*. https://www.britannica.com/science/cross-pollination

Eilers, E. J., Kremen, C., Smith Greenleaf, S., Garber, A. K., & Klein, A. M. (2011). Contribution of pollinator-mediated crops to nutrients in the human food supply. *PLoS One*, *6*(6), e21363. https://doi.org/10.1371/journal.pone.0021363

Fountas, S., Mylonas, N., Malounas, I., Rodias, E., Hellmann Santos, C., & Pekkeriet, E. (2020). Agricultural Robotics for Field Operations. *Sensors (Basel)*, *20*(9). https://doi.org/10.3390/s20092672

Garg, D., & Alam, M. (2020). Deep Learning and IoT for Agricultural Applications. In (pp. 273-284). https://doi.org/10.1007/978-3-030-37468-6_14

Ghazoul, J. (2005). Buzziness as usual? Questioning the global pollination crisis. *Trends in Ecology & Evolution*, *20*(7), 367-373. https://doi.org/https://doi.org/10.1016/j.tree.2005.04.026

Gomes, J. F. S., & Leta, F. R. (2012). Applications of computer vision techniques in the agriculture and food industry: a review. *European Food Research and Technology*, *235*(6), 989-1000. https://doi.org/10.1007/s00217-012-1844-2

Hiraguri, T., Kimura, T., Endo, K., Ohya, T., Takanashi, T., & Shimizu, H. (2023). Shape classification technology of pollinated tomato flowers for robotic implementation. *Scientific Reports*, *13*(1), 2159. https://doi.org/10.1038/s41598-023-27971-z

Hočevar, M., Širok, B., Godeša, T., & Stopar, M. (2014). Flowering estimation in apple orchards by image analysis. *Precision Agriculture*, *15*(4), 466-478. https://doi.org/10.1007/s11119-013-9341-6

Iwasaki, J. M., & Hogendoorn, K. (2022). Mounting evidence that managed and introduced bees have negative impacts on wild bees: an updated review. *Curr Res Insect Sci*, *2*, 100043. https://doi.org/10.1016/j.cris.2022.100043

Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A Review of Yolo Algorithm Developments. *Procedia Computer Science*, *199*, 1066-1073. https://doi.org/https://doi.org/10.1016/j.procs.2022.01.135

Jocher, G., Nishimura, K., Mineeva, T., & Vilariño, R. (2020). yolov5. *Code repository https://github. com/ultralytics/yolov5*, 9.

Jurado, J. M., López, A., Pádua, L., & Sousa, J. J. (2022). Remote sensing image fusion on 3D scenarios: A review of applications for agriculture and forestry. *International Journal of Applied Earth Observation and Geoinformation*, *112*, 102856. https://doi.org/https://doi.org/10.1016/j.jag.2022.102856

Khalifa, S. A. M., Elshafiey, E. H., Shetaia, A. A., El-Wahed, A. A. A., Algethami, A. F., Musharraf, S. G., AlAjmi, M. F., Zhao, C., Masry, S. H. D., Abdel-Daim, M. M., Halabi, M. F., Kai, G., Al Naggar, Y., Bishr, M., Diab, M. A. M., & El-Seedi, H. R. (2021). Overview of Bee Pollination and Its Economic Value for Crop Production. *Insects*, *12*(8). https://doi.org/10.3390/insects12080688

Khan, N., Ray, R. L., Sargani, G. R., Ihtisham, M., Khayyam, M., & Ismail, S. (2021). Current Progress and Future Prospects of Agriculture Technology: Gateway to Sustainable Agriculture. *Sustainability*, *13*(9), 4883. https://www.mdpi.com/2071-1050/13/9/4883

Klatt, B. K., Holzschuh, A., Westphal, C., Clough, Y., Smit, I., Pawelzik, E., & Tscharntke, T. (2014). Bee pollination improves crop quality, shelf life and commercial value. *Proc Biol Sci*, *281*(1775), 20132440. https://doi.org/10.1098/rspb.2013.2440

Klein, A., Vaissière, B., Cane, J., Steffan-Dewenter, I., Cunningham, S., Kremen, C., & Tscharntke, T. (2007). Importance of pollinators in changing landscapes for world crops. *Proceedings. Biological sciences / The Royal Society*, *274*, 303-313. https://doi.org/10.1098/rspb.2006.3721

Kremen, C., Williams, N. M., & Thorp, R. W. (2002). Crop pollination from native bees at risk from agricultural intensification. *Proceedings of the National Academy of Sciences*, *99*(26), 16812-16816. https://doi.org/10.1073/pnas.262413599

Le, T., Omholt Gjevestad, J. G., & From, P. J. (2019). Online 3D Mapping and Localization System for Agricultural Robots. *IFAC-PapersOnLine*, *52*(30), 167-172. https://doi.org/https://doi.org/10.1016/j.ifacol.2019.12.516

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., & Nie, W. (2022). YOLOv6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*.

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine Learning in Agriculture: A Review. *Sensors*, *18*(8), 2674. https://www.mdpi.com/1424-8220/18/8/2674

Lippert, C., Feuerbacher, A., & Narjes, M. (2021). Revisiting the economic valuation of agricultural losses due to large-scale changes in pollinator populations. *Ecological Economics*, *180*, 106860. https://doi.org/https://doi.org/10.1016/j.ecolecon.2020.106860

Lowenberg-DeBoer, J., Huang, I. Y., Grigoriadis, V., & Blackmore, S. (2020). Economics of robots and automation in field crop production. *Precision Agriculture*, *21*(2), 278-299. https://doi.org/10.1007/s11119-019-09667-5

Mader, E. O., Spivak, M., & Evans, E. (2010). Managing Alternative Pollinators: A Handbook for Beekeepers, Growers, and Conservationists, SARE Handbook 11, NRAES-186.

Mavridou, E., Vrochidou, E., Papakostas, G. A., Pachidis, T., & Kaburlasos, V. G. (2019). Machine Vision Systems in Precision Agriculture for Crop Farming. *Journal of Imaging*, *5*(12), 89. https://www.mdpi.com/2313-433X/5/12/89

McGregor, S. E. (1976). *Insect pollination of cultivated crop plants*. Agricultural Research Service, US Department of Agriculture.

Mu, X., He, L., Heinemann, P., Schupp, J., & Karkee, M. (2023). Mask R-CNN based apple flower detection and king flower identification for precision pollination. *Smart Agricultural Technology*, *4*, 100151. https://doi.org/https://doi.org/10.1016/j.atech.2022.100151

Nex, F., & Remondino, F. (2014). UAV for 3D mapping applications: a review. *Applied Geomatics*, *6*(1), 1-15. https://doi.org/10.1007/s12518-013-0120-x

Nicholson, C. C., Koh, I., Richardson, L. L., Beauchemin, A., & Ricketts, T. H. (2017). Farm and landscape factors interact to affect the supply of pollination services. *Agriculture, Ecosystems & Environment*, *250*, 113-122. https://doi.org/https://doi.org/10.1016/j.agee.2017.08.030

Orbán, L. L., & Plowright, C. M. (2014). Getting to the start line: how bumblebees and honeybees are visually guided towards their first floral contact. *Insectes Soc*, *61*(4), 325-336. https://doi.org/10.1007/s00040-014-0366-2

Pix4D. (2022). *Pix4Dmapper*. Retrieved March 15 from https://www.pix4d.com/product/pix4dmapper-photogrammetry-software/

Potts, S. G., Biesmeijer, J. C., Kremen, C., Neumann, P., Schweiger, O., & Kunin, W. E. (2010). Global pollinator declines: trends, impacts and drivers. *Trends in Ecology & Evolution*, *25*(6), 345-353. https://doi.org/https://doi.org/10.1016/j.tree.2010.01.007

Qadri, M., & Kantor, G. (2021). *Semantic Feature Matching for Robust Mapping in Agriculture*.

Rader, R., Bartomeus, I., Garibaldi, L. A., Garratt, M. P., Howlett, B. G., Winfree, R., Cunningham, S. A., Mayfield, M. M., Arthur, A. D., Andersson, G. K., Bommarco, R., Brittain, C., Carvalheiro, L. G., Chacoff, N. P., Entling, M. H., Foully, B., Freitas, B. M., Gemmill-Herren, B., Ghazoul, J., . . . Woyciechowski, M. (2016). Non-bee insects are important contributors to global crop pollination. *Proc Natl Acad Sci U S A*, *113*(1), 146-151. https://doi.org/10.1073/pnas.1517092112

Rahim, U. F., & Mineno, H. (2020). Tomato flower detection and counting in greenhouses using faster region-based convolutional neural network. *Journal of Image and Graphics*, *8*(4), 107-113.

Rehman, A., Jingdong, L., Khatoon, R., Hussain, I., & Iqbal, M. S. (2016). Modern agricultural technology adoption its importance, role and usage for the improvement of agriculture. *Life Science Journal*, *14*(2), 70-74.

Remondino, F. (2011). Heritage Recording and 3D Modeling with Photogrammetry and 3D Scanning. *Remote Sensing*, *3*(6), 1104-1138. https://www.mdpi.com/2072-4292/3/6/1104

Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4510-4520.

Shaneyfelt, T., Jamshidi, M., & Agaian, S. (2013). A vision feedback robotic docking crane system with application to vanilla pollination. *Int. J. of Automation and Control*, *7*, 62-82. https://doi.org/10.1504/IJAAC.2013.055096

Sidibé, A., Olabisi, L. S., Doumbia, H., Touré, K., & Niamba, C. A. (2021). Barriers and enablers of the use of digital technologies for sustainable agricultural development and food security: Learning from cases in Mali. *Elementa: Science of the Anthropocene*, *9*(1). https://doi.org/10.1525/elementa.2020.00106

Spykman, O., Gabriel, A., Ptacek, M., & Gandorfer, M. (2021). Farmers' perspectives on field crop robots – Evidence from Bavaria, Germany. *Computers and Electronics in Agriculture*, *186*, 106176. https://doi.org/https://doi.org/10.1016/j.compag.2021.106176

Srivastava, S., Divekar, A. V., Anilkumar, C., Naik, I., Kulkarni, V., & Pattabiraman, V. (2021). Comparative analysis of deep learning image detection algorithms. *Journal of Big Data*, *8*(1), 66. https://doi.org/10.1186/s40537-021-00434-w

Stanley, D. A., Garratt, M. P. D., Wickens, J. B., Wickens, V. J., Potts, S. G., & Raine, N. E. (2015). Neonicotinoid pesticide exposure impairs crop pollination services provided by bumblebees. *Nature*, *528*(7583), 548-550. https://doi.org/10.1038/nature16167

Strader, J., Nguyen, J., Tatsch, C., Du, Y., Lassak, K., Buzzo, B., Watson, R., Cerbone, H., Ohi, N., Yang, C., & Gu, Y. (2019, 3-8 Nov. 2019). Flower Interaction Subsystem for a Precision Pollination Robot. 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),

Sun, K., Wang, X., Liu, S., & Liu, C. (2021). Apple, peach, and pear flower detection using semantic segmentation network and shape constraint level set. *Computers and Electronics in Agriculture*, *185*, 106150. https://doi.org/https://doi.org/10.1016/j.compag.2021.106150

Tamburini, G., Pereira-Peixoto, M.-H., Borth, J., Lotz, S., Wintermantel, D., Allan, M. J., Dean, R., Schwarz, J. M., Knauer, A., Albrecht, M., & Klein, A. M. (2021). Fungicide and insecticide exposure adversely impacts bumblebees and pollination services under semi-field conditions. *Environment International*, *157*, 106813. https://doi.org/https://doi.org/10.1016/j.envint.2021.106813

Ting, Y., Kondo, N., & Wei, L. (2012). Sunlight Fluctuation Compensation for Tomato Flower Detection Using Web Camera. *Procedia Engineering*, *29*, 4343-4347. https://doi.org/https://doi.org/10.1016/j.proeng.2012.01.668

Toni, H. C., Djossa, B. A., Ayenan, M. A. T., & Teka, O. (2021). Tomato (Solanum lycopersicum) pollinators and their effect on fruit set and quality. *The Journal of Horticultural Science and Biotechnology*, *96*(1), 1-13. https://doi.org/10.1080/14620316.2020.1773937

Torres-Sanchez, J., de Castro, A. I., Pena, J. M., Jimenez-Brenes, F. M., Arquero, O., Lovera, M., & Lopez-Granados, F. (2018). Mapping the 3D structure of almond trees using UAV acquired photogrammetric point clouds and object-based image analysis. *Biosystems engineering*, *176*, 172-184.

Tzutalin. (2015). LabelImg. In.

United States Forest Service. (2023a). *Plant Pollination Strategies*. United States Forest Service. Retrieved January 10 from https://www.fs.usda.gov/wildflowers/pollinators/Plant_Strategies/index.shtml

United States Forest Service. (2023b). What is Pollination? https://www.fs.usda.gov/managing-land/wildflowers/pollinators/what-is-pollination

van der Sluijs, J. P., & Vaage, N. S. (2016). Pollinators and Global Food Security: the Need for Holistic Global Stewardship. *Food Ethics*, *1*(1), 75-91. https://doi.org/10.1007/s41055-016-0003-z

van Henten, E. J., Bac, C. W., Hemming, J., & Edan, Y. (2013). Robotics in protected cultivation. *IFAC Proceedings Volumes*, *46*(18), 170-177. https://doi.org/https://doi.org/10.3182/20130828-2-SF-3019.00070

Velthuis, H. H. W., & Doorn, A. v. (2006). A century of advances in bumblebee domestication and the economic and environmental aspects of its commercialization for pollination. *Apidologie*, *37*(4), 421-451. https://doi.org/10.1051/apido:2006019

Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*.

Watson, K., & Stallins, J. A. (2016). Honey Bees and Colony Collapse Disorder: A Pluralistic Reframing. *Geography Compass*, *10*(5), 222-236. https://doi.org/https://doi.org/10.1111/gec3.12266

Weiss, U., & Biber, P. (2011). Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. *Robotics and Autonomous Systems*, *59*(5), 265-273. https://doi.org/https://doi.org/10.1016/j.robot.2011.02.011

Woodcock, T. S. (2012). *Pollination in the Agricultural Landscape: Best Management Practies for Crop Pollination*. https://studylib.net/doc/8933720/pollination-in-the-agricultural-landscape

Wu, D., Lv, S., Jiang, M., & Song, H. (2020). Using channel pruning-based YOLO v4 deep learning algorithm for the real-time and accurate detection of apple flowers in natural environments. *Computers and Electronics in Agriculture*, *178*, 105742. https://doi.org/https://doi.org/10.1016/j.compag.2020.105742

Wu, S., Liu, J., Lei, X., Zhao, S., Lu, J., Jiang, Y., Xie, B., & Wang, M. (2022). Research Progress on Efficient Pollination Technology of Crops. *Agronomy*, *12*(11), 2872. https://www.mdpi.com/2073-4395/12/11/2872

Wurz, A., Grass, I., & Tscharntke, T. (2021). Hand pollination of global crops – A systematic review. *Basic and applied ecology*, *56*, 299-321.

Xu, J., Gu, B., & Tian, G. (2022). Review of agricultural IoT technology. *Artificial Intelligence in Agriculture*, *6*, 10-22. https://doi.org/https://doi.org/10.1016/j.aiia.2022.01.001

Yuan, T., Zhang, S., Sheng, X., Wang, D., Gong, Y., & Li, W. (2016, 19-21 Nov. 2016). An autonomous pollination robot for hormone treatment of tomato flower in greenhouse. 2016 3rd International Conference on Systems and Informatics (ICSAI),

Zhang, Y., He, S., Wa, S., Zong, Z., & Liu, Y. (2021). Using Generative Module and Pruning Inference for the Fast and Accurate Detection of Apple Flower in Natural Environments. *Information*, *12*(12), 495. https://www.mdpi.com/2078-2489/12/12/495

Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object Detection With Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(11), 3212-3232. https://doi.org/10.1109/TNNLS.2018.2876865

Zohary, D. (2001). Domestication of Crop Plants. In S. A. Levin (Ed.), *Encyclopedia of Biodiversity (Second Edition)* (pp. 657-664). Academic Press. https://doi.org/https://doi.org/10.1016/B978-0-12-384719-5.00199-4

Vita

McKensie was born and raised in Knoxville, TN. She received her Bachelor of Science degree in Biosystems Engineering from the University of Tennessee in December of 2020. In January of 2021, she continued her studies at the University of Tennessee to pursue a Master of Science degree in Biosystems Engineering, which she completed in May 2023. Her research is focused on developing crop flower detection and three-dimensional mapping techniques as initial steps towards the design of a robust autonomous pollination system.